EFFICIENTLY ACQUIRING REFLECTANCE FIELDS USING PATTERNED ILLUMINATION

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF
ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES OF
STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Gaurav Garg

September 2006

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Marc Levoy) Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Mark Horowitz)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

(Hendrik P. A. Lensch)

Approved for the University Committee on Graduate Studies.

Abstract

The use of the reflectance fields of real world objects to render realistic looking images is rapidly increasing. The reflectance field describes the transport of light between the light incident on an object and the light exitant from it. This has numerous applications in areas that include entertainment, cultural heritage, digital libraries and space exploration. The central problem with this approach is the lack of fast methods to acquire the reflectance field data. This dissertation addresses this problem and describes a system for acquiring the reflectance field of real world objects that performs many orders of magnitude faster than the previous approaches.

The system models the 8D reflectance field as a transport matrix between the 4D incident light field and the 4D exitant light field. It is a challenging task to measure this matrix because of its large size. However, in some cases the matrix is sparse, e.g. in scenes with little or no inter-reflections. To measure such matrices, this thesis describes a hierarchical technique called *dual photography* which exploits this sparseness to parallelize the acquisition process. This technique, however, performs poorly for scenes with significant diffuse inter-reflections because in such cases the matrix is dense. Fortunately, in these cases the matrix is often data-sparse. Data-sparseness refers to the fact that sub-blocks of the matrix can be well approximated using low-rank representations. Additionally, the transport matrix is symmetric. Symmetry enables simultaneous measurements from both sides, rows and columns, of the transport matrix. These measurements are used to develop a hierarchical acquisition algorithm that can exploit the data-sparseness by a local rank-1 approximation. This technique, called *symmetric photography*, parallelizes the acquisition for dense but data-sparse transport matrices.

In the process, this thesis introduces the use of hierarchical tensors as the underlying data structure to represent data-sparse matrices, specifically through local rank-1 factorizations of the transport matrix. Besides providing an efficient representation for storage, it enables fast acquisition of the approximated transport matrix and fast rendering of the images from the captured matrix. The prototype acquisition system consists of an array of mirrors and a pair of coaxial projector and camera controlled by a computer. The effectiveness of the system is demonstrated with scenes rendered from reflectance fields that were captured by this system. In these renderings one can change the viewpoint as well as relight objects using arbitrary incident light fields.

Acknowledgements

Five years at Stanford have been an exciting journey; a great learning experience and personally extremely rewarding. This dissertation is a culmination of those years and it would not have been possible without the encouragement and support of many individuals. I would like to acknowledge them here.

First and foremost, I would like to thank my adviser Marc Levoy who gave me the opportunity to work with him during my stay at Stanford. Marc's enthusiasm for computer graphics and research in general, is highly contagious. He leads by example, and has encouraged me to achieve excellence in the process. His research style gave me a lot of freedom to explore new ideas and led to two other successful projects before I settled on my dissertation topic. I am grateful to him for his guidance.

I also had the good fortune of working with two other members of my reading committee, Mark Horowitz and Hendrik Lensch. Mark Horowitz has this innate ability to get to the heart of the matter in any situation. Many a times he has provided excellent insights for solving critical problems. Hendrik Lensch has been like a mentor and friend for the last two years. In terms of blood and sweat that has gone into this dissertation, his is at least as much as mine. His experience with building acquisition systems was very useful while working on this project. I thank both, Mark and Hendrik for their guidance and support.

I would like to thank Pradeep Sen, Eino-Ville Talvala, Billy Chen and Steve Marschner for collaborating with me during different parts of this dissertation. Steve's initial experiments laid the foundation of this research. I would also like to thank Augusto Román with whom I worked on the Cityblock Project and Vaibhav Vaish, Emilio Antunez and Bennett Wilburn with whom I worked on the Camera Array Project. These are the people I spent countless nights with, working under tight deadlines. The ability of these individuals to maintain a cordial atmosphere in tough situations, and their grit and determination are remarkable. I thank all of them for getting the projects through completion while still keeping the work environment fun. I would also like to acknowledge the lab staff John Gerth, Heather Gentner and Ada Glucksman for keeping the lab running like a well-oiled machine.

I would also like to thank my friends who have been my support system outside work during my life at Stanford. In particular, I would like to thank my roommate for last five years, Abhishek Bapna whose ability to look at the positive side of things even in the toughest of circumstances has guided me out of many difficult situations. We have grown together as individuals over these years and I hope our friendship will continue to grow. The list of my friends who have left me with unforgettable memories is long so I will just mention a few here: Gaurav Agarwal, Mohit Agnihotri, Neha Kumar, Aditi Nabar, Akanksha Bapna, Niloy Mitra, Chirag Panjikar, Anish Jain, Chirag Sethi, Kedar Kulkarni, Kunal Girotra, Dinesh Patil, Priyanka Jain, Amit Jain, Piya Sorcar, Ashwini Doshi, Madhuri Kudaravalli, Nishant Garg, Bhavna Hariharan, Munish Singla, Pramod Sharma, Gaurav Chandra, Mausam Mathur, Nandita Segan and many others. Also, thanks to Akanksha for proof-reading draft of this dissertation.

Finally, I would like to thank my family, in particular my parents and my sister Sakshi, for their unwavering love and support. Without their confidence in me, this work would have not been possible.

Reed-Hodgson Stanford Graduate Fellowship, Max Planck Center for Visual Computing and Communication, Google Inc., NSF and DARPA provided the funding for this research.

To Sidhant

Contents

List of Tables

List of Figures

Chapter 1

Introduction: Reflectance Fields in Computer Graphics

Synthesizing realistic looking images is one of the central problems in computer graphics. In order to achieve photorealism, the traditional graphics pipeline works by trying to simulate real-world physics accurately. The typical input to such a pipeline is a scene consisting of object shapes, material properties and a set of lights. Although it is possible to provide fairly accurate descriptions of object shapes, material properties are harder to model, as in the case of translucent materials. This limitation affects the degree of realism in images rendered from synthetic scenes. Moreover, there exists a trade-off between the degree of realism and the computational resources required for simulating physical effects such as diffuse inter-reflections, shadows, caustics, sub-surface scattering etc.

An alternative to modeling the shape and the material properties of objects is capturing the appearance of the objects/scene directly. The appearance of a scene can be described by an 8D function called the reflectance field which was introduced in the graphics community by Debevec et al. [DHT$^+$00]. The reflectance field describes the transport of light between the light incident on an object and the light exitant from it. Once available, this representation can be used to render realistic images of the scene from any viewpoint under arbitrary lighting. The resulting images capture all global illumination effects such as diffuse inter-reflections, shadows, caustics and sub-surface scattering, without the need for an explicit physical simulation. This dissertation focuses on techniques for acquiring reflectance fields of real-world objects.

In addition to being used as a primitive for computer graphics, reflectance fields have a wide variety of applications:

- **Entertainment:** Convincingly inserting real actors inside a virtually generated set is a major issue in current movie production. For this purpose, a reflectance field is the primary dataset required to render images of people under arbitrary changes in lighting and viewing direction. Techniques for capturing and rendering from reflectance fields have already been used in motion pictures such as *Spiderman 2*, *King Kong* and *Superman Returns*.

- **Cultural heritage:** Digital documentation of cultural artifacts is very important. Besides providing an online catalogue of the artifacts it provides an easy way to study the working techniques and design choices of artists who created the artifacts. Reflectance fields provide a complete representation for digitally storing such artifacts as compared to traditional approach of acquiring just a geometric model as has been explored in [HCD01].

- **Space exploration:** Photographs of rocks found on other planets provide scientists with a window through which they can explore the outer space. Various missions (e.g. Mars Rover Spirit) have been used for this purpose before. If instead of capturing just a photograph, the whole reflectance field of the rock were acquired, then it can be used to visualize the structure of the rock in far greater detail.

It is important to note that none of the applications described above have used a complete reflectance field. This is because with current hardware and algorithms it is not possible to measure a complete reflectance field in reasonable time.

1.1 Problem Statement

The 8D reflectance field of a scene or object is defined as a transport matrix that describes the transfer of energy between a light field [LH96] of incoming rays (the illumination) and a light field of outgoing rays (the view), each of which are 4D. The rows of this matrix correspond to the view rays and the columns correspond to the illumination rays. Even for small light field representations, say only 3×3 angular and 100×100 spatial, the transport matrix would contain about 10^{10} entries. If constructed by measuring one entry per video frame, it could take several days to capture even at video rate, making this approach intractable.

Therefore, we must (1) devise algorithms that speed up the acquisition process; and (2) devise infrastructure that can support the algorithms. In this dissertation we describe techniques which accomplishes the above mentioned goals.

1.2 Previous Work

The idea of capturing the reflectance field of a scene is recent, but researchers have been trying to measure the reflectance properties of surfaces, the bidirectional reflectance functions, for much longer. Since measuring the bidirectional reflectance function requires similar machinery as that required for measuring reflectance fields, we will start by reviewing bidirectional reflectance functions first. The survey presented in Sections 1.2.1 and 1.2.2 has been adapted from [LHG$^+$05].

1.2.1 Measurement of Bidirectional Reflectance Functions

The optical properties of an opaque homogeneous surface can be characterized by reflectance as a function of incident light direction (two angles) and reflected light direction (two angles). The resulting 4D function is called the bidirectional reflectance distribution function (BRDF) [NRH$^+$77]. If the surface is textured rather than homogeneous, then its optical properties depend on position on the surface (two spatial coordinates) as well as direction, leading to a 6D function called the spatially varying bidirectional reflectance distribution function (SBRDF). Dana et al. [DNGK97] also call it the bidirectional texture function (BTF). Finally, if the object exhibits subsurface scattering, as does marble or human skin, then its reflectance properties depend on the outgoing as well as incoming position, adding two more spatial coordinates. The resulting function is 8D, and is called the bidirectional surface scattering distribution function (BSSRDF).

Characterizing the 4D, 6D, and 8D reflectance (or transmittance) functions of surfaces is an active research area in applied physics and computer graphics. To realistically render some materials, such as plastics and metals, analytic descriptions of these functions are known and tractable to compute. In other cases, analytic models are known, but these include free parameters than can be measured from the physical samples of the material. Recent examples of this are marble [JMLH01], human hair [MJC$^+$03], wood [MWAM05], and smoke [HED05b]. However, some materials are hard to describe using analytical models, for example human skin. Others like woven cloth, are combinations of various subspecies. In order to render such materials, the bulk optical properties of a physical sample of the material can be measured under a sufficiently wide range of viewing and illumination conditions to create a high-dimensional lookup table. This table can then be indexed to make a new rendering of the object from an arbitrary viewpoint and under arbitrary illumination, without the need for an analytical model. This is the idea behind capturing reflectance fields, which we explore in this dissertation.

Researchers have proposed numerous devices for performing the measurements outlined above. Devices for measuring 4D BRDF are called scatterometers or gonioreflectometers. In applied physics, the emphasis has been on accuracy, and as a result most of these devices employ lasers, cooled sensors, and precision mechanical movements. In computer graphics, the emphasis has been on speed rather than accuracy, spurring a trend towards optoelectronic solutions (having fewer moving parts). For example, Ward [War92] captures a BRDF using a movable light source, half-silvered hemisphere, and wide-angle camera. Marschner et al. [MWLT00] constructed another significant BRDF measurement system. This system, although limited to only isotropic BRDF measurements, was both faster and robust than Ward's. In particular, the system took unique advantage of reciprocity, bilateral symmetry, and multiple simultaneous measurements to achieve unprecedented leverage from each reflection measurement. The system used two cameras and a movable light source for this purpose. Dana et al. [Dan01] measure SBRDF by using curved mirrors which remove the need for hemispherical positioning of the camera and illumination source. Instead, simple planar translations of optical components are used to vary the illumination direction and to scan the surface. Recently, Han and Perlin [HP03] have captured SBRDF using a stationary camera, a video projector and a kaleidoscope to redirect the incoming and outgoing light. This system has no moving parts, but it produces datasets of low directional resolution. Because of its daunting size, few research groups have undertaken to measure a BSSRDF. The Stanford Spherical Gantry has been used to capture subsets of this function [JMLH01, MJC$^+$03], but measurement of an entire BSSRDF has never been tried.

1.2.2 Measurement of Reflectance Fields

Bidirectional reflectance functions specify the optical properties of a particular surface, but if the scene is composed of many objects then the optical properties of the whole scene can be specified using the 4D, 6D, or 8D reflectance field depending on how much variation is permitted in the illumination and viewer position. The 8D reflectance field can also be described as a transport matrix describing the transfer of energy between the illumination rays and the view rays. 4D and 6D reflectance fields are projection of this matrix into lower dimensions. For obvious reasons, the work on measuring reflectance fields has focused on capturing various lower dimensional projections.

If the illumination is fixed and the viewer allowed to move, the appearance of the scene as a function of outgoing ray position and direction is a 4D slice of the reflectance field. The light field [LH96] and the lumigraph [GGSC96] effectively describe this exitant reflectance field. By extracting appropriate 2D slices of the light field, one can virtually fly around a scene but the illumination cannot be changed. Light fields can be captured by a single moving camera [LH96] or an array of cameras [WJV$^+$05]. If the viewpoint is fixed and the illumination is provided by a set of point light sources, one obtains another 4D slice of the 8D reflectance field. Various researchers [ZWCS99, CZH$^+$00, DHT$^+$00, MGW01, SNB03, HED05a] have acquired such data sets where a weighted sum of the captured images can be combined to obtain relit images from a fixed viewpoint only. However, since point light sources radiate light in all directions, it is impossible to cast sharp shadows onto the scene with this technique.

If the illumination is provided by an array of video projectors, and the scene is captured as illuminated by each pixel of each projector, but still as seen from a single viewpoint, then one obtains a 6D slice of an 8D reflectance field. By extracting appropriate slices from this function, one can render views under lighting that includes local light source and shadows. Masselus et al. [MPDW03] capture such data sets using a single moving projector. At each position of the projector, the scene is captured under a set of illumination basis functions rather than one pixel at a time, to increase speed. Nevertheless, the necessity to move the projector limits the resolution of data that can be captured using such a system. Goesele et al. [GLL$^+$04] use a scanning laser, a turntable and a moving camera to capture a reflectance field for the case of translucent objects under a diffuse sub-surface scattering assumption. Although one can view the object from any position and relight it with arbitrary light fields, the captured data set is still essentially 4D because of their assumption.

In this dissertation, we extend this previous work by describing a system that requires at least one projector/camera pair, an array of mirrors and a beam-splitter for measuring 8D reflectance fields. Our system has no moving parts. Having produced such a system, we next look at prior attempts to efficiently storing this data.

1.2.3 Use of Hierarchical Data Structures for Reflectance Field

Hierarchical data structures have been previously used for representing reflectance fields. These representations provide greater efficiency both in terms of storage and capture time. A typical setup for capturing reflectance fields consists of a scene under controlled illumination, as imaged by one or more cameras. Peers and Dutré [PD03] illuminate a scene with wavelet patterns in order to capture environment mattes (another 4D slice of the reflectance field). A feedback loop determines the next pattern to use based on knowledge of previously recorded photographs. The stopping criteria is based on the error of the current approximation. Although their scheme adapts to the scene content, it does not try to parallelize the capture process. Matusik et al. [MLP04] use a kd-tree based subdivision structure to represent environment mattes. They express environment matte extraction as an optimization problem. Their algorithm progressively refines the approximation of the environment matte with an increasing number of training images taken under various illumination conditions. However, the choice of their patterns is independent of the scene content.

In this dissertation, we also use a hierarchical data structure to represent the reflectance field. It is a subdivision structure with a low-rank representation for nodes. Also, our subdivision is adaptive and is based on the scene content.

1.2.4 Helmholtz Reciprocity

Helmholtz reciprocity is the idea that the flow of light can be effectively reversed without altering its transport properties. There has been some previous work in the computer vision community that takes advantage of Helmholtz reciprocity. Zickler et al. used reciprocity to reconstruct the geometry of surfaces with arbitrary BRDFs in what they call Helmholtz stereopsis [ZBK02]. The authors observed that by interchanging light source and camera during acquisition, they can use Helmholtz reciprocity to guarantee that points on the surface would have exactly the same transfer characteristic in both configurations. This simplifies stereo matching, even for surfaces with complex BRDFs. A similar approach was taken by Tu et al. [TMRM03], who utilized reciprocity for the task of 3D to 2D registration. Finally, reciprocity has been used in a manner analogous to the duality explained in this dissertation by Zotkin et al. [ZDGG04] in order to model sound transfer through the human head. In this work, the authors noted that an array of microphones and a single speaker can be used to replace the more common setup of a single microphone and multiple speakers because of the duality of the sound transport between the microphone and the speaker. In this dissertation, we take advantage of helmholtz reciprocity to speed-up our acquisition algorithm.

1.3 Contributions

This dissertation makes two contributions to the area of acquisition of reflectance fields:

- On the **theoretical** side, it observes that the reflectance field, when expressed as a transport matrix, is data-sparse and symmetric. This means that it can be captured efficiently by separately measuring the row sums and the column sums, i.e. by co-locating a camera and a projector.

- On the **system** side, we present a prototype system for acquiring reflectance fields. Our system uses the properties of reflectance field matrix to achieve faster acquisition using a hierarchical adaptive scheme. The heart of this system is a novel representation of the matrix using local low rank approximation.

Part of the work presented in this dissertation has been described in previous publications. The dual photography work was presented at SIGGRAPH 2005 [SCG$^+$05] and the symmetric photography work was presented at the Eurographics Symposium on Rendering (EGSR) 2006 [GTLL06].

1.4 Outline of the Dissertation

This dissertation is organized as follows. Chapter 2 defines the reflectance field and describes how to abstract it as a matrix, called the light transport matrix, for the purpose of this thesis. Three important properties of the transport matrix - data-sparseness, symmetry and duality are also explained. Chapter 3 describes dual photography, our first technique to measure the transport matrix. This method exploits sparseness in the matrix. Chapter 4 introduces a hierarchical tensor structure for representing data-sparse matrices/tensors. Chapter 5 describes a second technique to measure the transport matrix - symmetric photography - which exploits data-sparseness in the matrix. Finally, Chapter 6 presents the conclusions about our acquisition system, and suggestions for future work.

Chapter 2

Reflectance Field as the Light Transport Matrix: Theory and Properties

In this chapter, we formally define the reflectance field and the light transport matrix. We also describe three important properties of the transport matrix; data-sparseness, symmetry and duality. These properties will be used to develop our acquisition algorithm in subsequent chapters.

2.1 Definitions: Reflectance Field and Light Transport Matrix

The light field [LH96], plenoptic function [AB91], and lumigraph [GGSC96] all describe the flow of light within space. These ideas were first introduced in the classic work of Gershun on the vector irradiance field [Ger36] and Moon's paper on the scalar irradiance field, which he called the photic field [MS81]. Light fields are used to describe the radiance at each point $\mathbf{x}$ and in each direction ω in a scene. Ignoring wavelength and fixing time, this is a 5D function which we denote by $\widetilde{L}(\mathbf{x},\omega)$. Thus, $\widetilde{L}(\mathbf{x},\omega)$ represents the radiance leaving a point $\mathbf{x}$ in direction ω. (A complete table of the mathematical terms used in this chapter is provided in Table 2.1.)

Levoy and Hanrahan [LH96], and Gortler et al. [GGSC96] observed that if the viewer

V	3D space of all points in a volume, domain of functions
Ω	2D space of all directions at a point, domain of functions
$\mathbf{x}$, $\mathbf{x}'$	two points in domain V, e.g., $(x,y,z),(x',y',z')$
ω, ω'	two points in domain Ω, e.g., $(\theta,\phi),(\theta',\phi')$
$d\mathbf{x}'$	a differential volume at $\mathbf{x}'$, i.e., $dx'dy'dz'$
$d\omega'$	a differential direction at ω', i.e., $d\theta'd\phi'$
$\widetilde{L}(\mathbf{x},\omega)$	5D light field function on domains V and S, radiance at $(\mathbf{x},\omega)$
$\widetilde{L}_i(\mathbf{x},\omega)$	5D incoming light field
$\widetilde{L}_o(\mathbf{x},\omega)$	5D outgoing light field
$K(\mathbf{x},\omega;\mathbf{x}',\omega')$	the direct light transport from $(\mathbf{x}',\omega')$ to $(\mathbf{x},\omega)$
$\widetilde{\mathbf{L}}_i$	5D discrete incoming light field
$\widetilde{\mathbf{L}}_o$	5D discrete outgoing light field
$\mathbf{K}$	matrix of the direct light transport coefficients
$\widetilde{\mathbf{T}}$	matrix of the light transport coefficients for 5D representation
Ψ	4D space of all incoming directions on all points of a sphere, domain of functions
ψ	a point in domain Ψ, e.g., (u,v,θ,ϕ)
ψ_i,ψ_o	two points in domain Ψ, e.g., $(u_i,v_i,\theta_i,\phi_i)$, $(u_o,v_o,\theta_o,\phi_o)$
$L(\psi)$	4D light field function on domain Ψ, radiance at a point ψ
$L_i(\psi_i)$	4D incoming light field
$L_o(\psi_o)$	4D outgoing light field
$R(\psi_i,\psi_o)$	8D reflectance field mapping $L_i(\psi_i)$ to $L_o(\psi_o)$
$\mathbf{L}_i$	4D discrete incoming light field
$\mathbf{L}_o$	4D discrete outgoing light field
$\mathbf{t}_i$	impulse response to unit illumination along a ray i
$\mathbf{T}$	matrix of the light transport coefficients for 4D representation, symmetric
$\widehat{\mathbf{T}}$	a sub-block of $\mathbf{T}$, not necessarily symmetric

Table 2.1: Table of terms and variables

is moving within the unoccluded space, then the 5D representation of the light field can be reduced to 4D. We can characterize this function as $L(\psi)$ where ψ specifies a point and an incoming direction on a sphere [DHT$^+$00]. A 4D light field can be used to generate an image from any viewing position and direction, but it will always show the scene under the same lighting. In general, each field of incident illumination on a scene will induce a different field of exitant illumination from the scene. Debevec et al. [DHT$^+$00] showed that the exitant light field from the scene under every possible incident field of illumination can be represented as an 8D function called the *reflectance field*:

$$R(L_i(\psi_i); L_o(\psi_o)) = R(\psi_i; \psi_o) \tag{2.1}$$

Here, $L_i(\psi_i)$ represents the incident light field on the scene, and $L_o(\psi_o)$ represents the exitant light field reflected off the scene.

We have so far described the light field concepts using continuous functions, but for actual measurements, we work with discrete forms of these functions. In order to do so, let us parameterize the domain Ψ of all incoming directions by an array indexed by i. The outgoing direction corresponding to an incoming direction is also parameterized by the same index, i. Now, consider emitting unit radiance along ray i towards the scene (e.g., using a laser beam or a projector). The resulting light field, which we denote by vector $\mathbf{t}_i$, captures the full transport of light in response to this impulse illumination. This is called the impulse response [GLL$^+$04] or the impulse scatter function [SMK05]. We can concatenate all the impulse responses into a matrix $\mathbf{T}$ which we call the *light transport matrix*:

$$\mathbf{T} = [\mathbf{t}_1 \mathbf{t}_2 \ldots \mathbf{t}_n] \tag{2.2}$$

Since light transport is linear, any outgoing light field represented by a vector $\mathbf{L}_o$ can be described as linear combination of the impulse responses, $\mathbf{t}_i$. Thus, for an incoming illumination described by vector $\mathbf{L}_i$, the outgoing light field can be expressed as:

$$\mathbf{L}_o = \mathbf{T}\mathbf{L}_i \tag{2.3}$$

The light transport matrix $\mathbf{T}$, is thus the discrete analog of the reflectance field $R(\psi_i; \psi_o)$.

Seitz et al. [SMK05], in work contemporaneous to our own, also use transport matrices to model the light transport. Their work provides a theory for decomposing the transport matrix into individual bounce light transport matrices, while our work describes efficient techniques to measure it.

2.2 Data-Sparseness

2.2.1 Sparseness, Smoothness and Data-sparseness

To efficiently store large matrices, sparseness and smoothness are two ideas that are typically exploited. A sparse matrix has only a small number of non-zero elements in it and hence can be represented compactly. A data-sparse matrix on the other hand may have many non-zero elements, but the actual information content in the matrix is small enough that it can still be expressed compactly. A simple example will help convey this concept. Consider taking the cross product of two vectors, each of length n. Although the resulting matrix (which is rank-1 by construction) could be non-sparse, we only need two vectors $(O(n))$ to represent the contents of the entire $(O(n^2))$ matrix. Such matrices are data-sparse. More generally, any matrix in which a significant number of sub-blocks can have a low-rank representation is data-sparse. Note that a low-rank sub-block of a matrix need not be smooth and may contain high frequencies. A frequency or wavelet-based technique would be ineffective in compressing this block. Therefore, the concept of data-sparseness is more general and powerful than sparseness or smoothness.

Ramamoorthi and Hanrahan [RH01] analyze the smoothness in BRDFs and use it for efficient rendering and compression. A complete frequency space analysis of light transport has been presented by Durand et al. [DHS$^+$05]. The idea of exploiting data-sparseness for factorizing high dimensional datasets into global low-rank approximations has also been investigated, in the context of BRDFs [KM99, MAA01, LK02] and also for light fields and reflectance fields [VT04, WWS$^+$05]. In contrast to these global approaches, we have developed a technique called symmetric photography that uses local low-rank factorizations to exploit the data-sparseness of the transport matrices. We tie in the factorization with a hierarchical subdivision scheme (see Chapter 5). This hierarchical subdivision allows us to exploit the data-sparseness locally. We also present another method for acquiring reflectance fields called dual photography. This method only exploits the sparseness in transport matrices (see Chapter 3). The exposition of the techniques should make it clear that symmetric photography, which exploits data-sparseness, is more powerful and general than dual photography, which exploits only sparseness.

2.2.2 Data-Sparseness of the Transport Matrix

In order to understand the data-sparseness of transport matrices, we will work with the 5D representation of the light fields. Under the light field's paradigm, the appearance of a scene can be completely described by an outgoing radiance distribution function, $\widetilde{L}_o(\mathbf{x},\omega)$. Similarly, the illumination incident on the scene can be described by an incoming radiance distribution function, $\widetilde{L}_i(\mathbf{x},\omega)$.

The relationship between $\widetilde{L}_i(\mathbf{x},\omega)$ and $\widetilde{L}_o(\mathbf{x},\omega)$ can be expressed by an integral equation, the well known rendering equation [Kaj86]:

$$\widetilde{L}_o(\mathbf{x},\omega) = \widetilde{L}_i(\mathbf{x},\omega) + \int_V \int_\Omega K(\mathbf{x},\omega;\mathbf{x}',\omega')\widetilde{L}_o(\mathbf{x}',\omega')d\mathbf{x}'d\omega' \tag{2.4}$$

The function $K(\mathbf{x},\omega;\mathbf{x}',\omega')$ defines the proportion of flux from $(\mathbf{x}',\omega')$ that gets transported as radiance to $(\mathbf{x},\omega)$. It is a function of the BSSRDF, the relative visibility of $(\mathbf{x}',\omega')$ and $(\mathbf{x},\omega)$ and foreshortening and light attenuation effects. When $\mathbf{x}=\mathbf{x}'$, $K(\mathbf{x},\omega;\mathbf{x}',\omega')=0$.

If we assume that the scene is composed of a collection of small planar facets, and if we discretize the space of rays, eq. (2.4) can be expressed in discrete form as:

$$\widetilde{\mathbf{L}}_o[i] = \widetilde{\mathbf{L}}_i[i] + \sum_j \mathbf{K}[i,j]\widetilde{\mathbf{L}}_o[j] \tag{2.5}$$

where $\widetilde{\mathbf{L}}_o$ and $\widetilde{\mathbf{L}}_i$ are discrete representations of outgoing and incoming light fields respectively, and $\mathbf{K}[i,i]=0$. We can rewrite eq. (2.5) as a matrix equation:

$$\widetilde{\mathbf{L}}_o = \widetilde{\mathbf{L}}_i + \mathbf{K}\widetilde{\mathbf{L}}_o \tag{2.6}$$

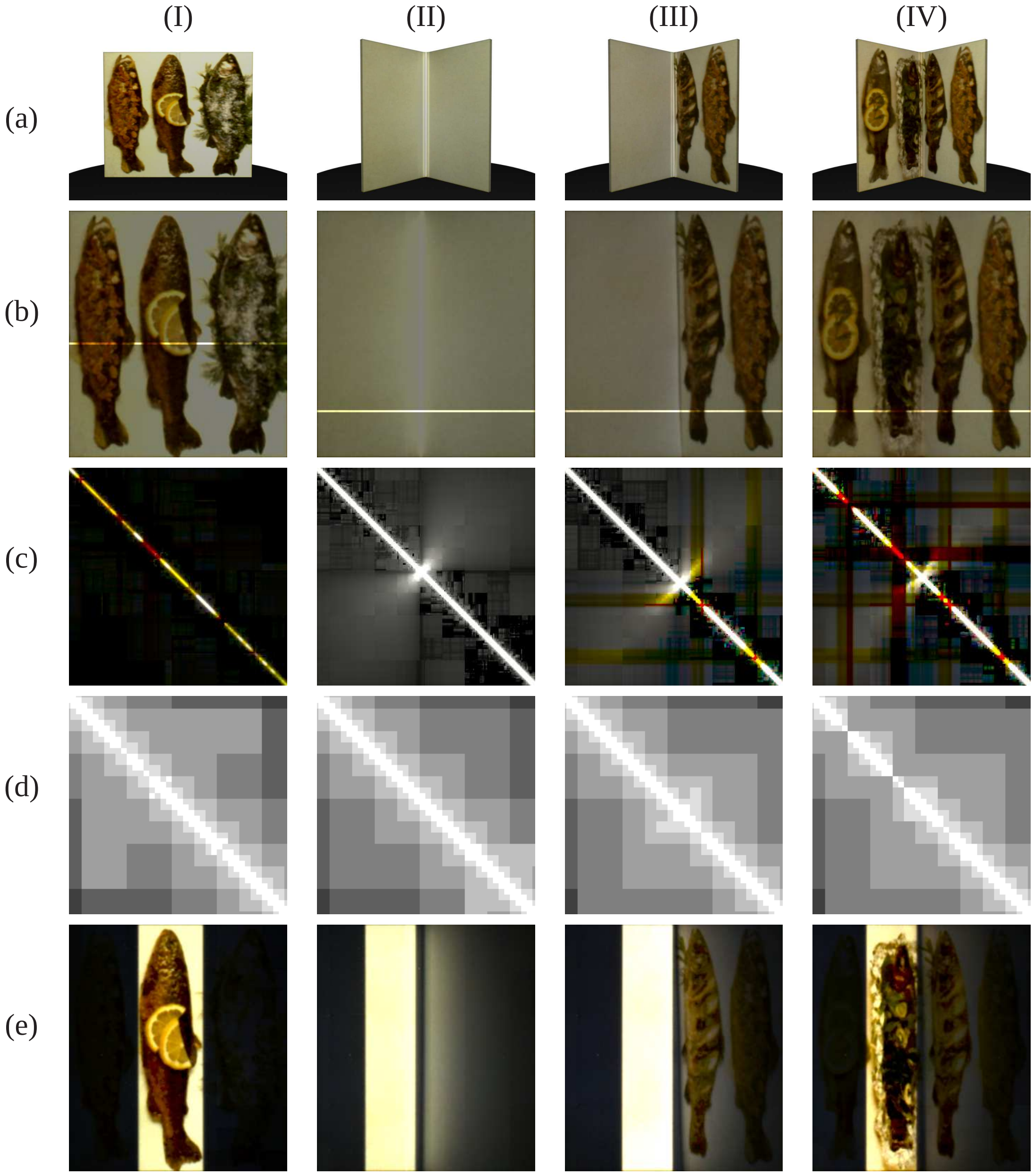

(I)
(II)
(III)
(IV)
(a)
(b)
(c)
(d)
(e)

Figure 2.1: **Understanding the transport matrix.** To explain the intrinsic structure of reflectance fields, we capture the transport matrix for 4 real scenes shown in row (a) with a coaxial projector/camera pair. The scenes in different columns are: (I) a diffuse textured plane, (II) two diffuse white planes facing each other at an angle, (III) a diffuse white plane facing a diffuse textured plane at an angle, and (IV) two diffuse textured planes facing each other at an angle. Row (b) shows the images rendered from the captured transport matrices under floodlit illumination. A 2D slice of the transport matrix for each configuration is shown in row (c). This slice describes the light transport between every pair of rays that hits the brightened line in row (b). Note that the transport matrix is symmetric in all 4 cases. Since (I) is a flat diffuse plane, there are no secondary bounces and the matrix is diagonal. In (II), (III) and (IV) the diagonal corresponds to the first bounce light and is therefore much brighter than the rest of the matrix. The top-right and bottom-left sub-blocks describe the diffuse-diffuse light transport from pixels on one plane to the other. Note that this is smoothly varying for (II). In case of (III) and (IV), the textured surface introduces high frequencies but these sub-blocks are still data-sparse and can be represented using rank-1 factors. The top-left and bottom-right sub-blocks correspond to the energy from 3rd-order bounces in our scenes. Because this energy is around the noise threshold in our measurements we get noisy readings for these sub-blocks. Row (d) is a visualization of the level in the hierarchy when a block is classified as rank-1. White blocks are leaf nodes, while darker shades of gray progressively represent lower levels in the hierarchy. Finally, row (e) shows the result of relighting the transport matrix with a vertical bar. Note the result of indirect illumination on the right plane in (II), (III) and (IV). Since the left plane is textured in (IV) the indirect illumination is dimmer than in (III). Note that the matrix for a line crossing diagonally through the scene would look similar.

Eq. (2.6) can be directly solved [Kaj86] to yield:

$$\widetilde{\mathbf{L}}_o = (\mathbf{I} - \mathbf{K})^{-1}\widetilde{\mathbf{L}}_i \tag{2.7}$$

The matrix $\widetilde{\mathbf{T}} = (\mathbf{I} - \mathbf{K})^{-1}$ describes the complete light transport between the 5D incoming and outgoing light fields as a linear operator[1]. Heckbert [Hec91] uses a similar matrix in the context of radiosity problems and shows that such matrices are not sparse. This is also observed by Börm et al. [BGH03] in the context of linear operators arising from an integral equation such as eq. (2.4). They show that even though the kernel $\mathbf{K}$ might be sparse, the resulting matrix $(\mathbf{I} - \mathbf{K})^{-1}$ is not. However, it is typically data-sparse. In particular, the kernel is sparse because of occlusions, but due to multiple scattering events one typically observes light transport between any pair of points in the scene, resulting in a dense $\widetilde{\mathbf{T}}$. On the other hand, we observe that a diffuse bounce off a point on the scene contributes the same energy to large regions of the scene in a similar fashion. Therefore, large portions of the transport matrix, e.g. those resulting from inter-reflections of diffuse and glossy surfaces, are *data-sparse*. One can exploit this data-sparseness by using local low-rank approximations for sub-blocks of $\widetilde{\mathbf{T}}$. We choose a rank-1 approximation.

Figure 2.1 illustrates this data-sparseness for a few examples of transport matrices that we have measured, and also demonstrates the local rank-1 approximation. To gain some intuition, let us look at the light transport between two homogeneous untextured planar patches. The light transport between the two is smooth and can be easily factorized. It can be seen in the top-right and bottom-left sub-blocks of the transport matrix for scene (II). Even if the surfaces are textured, it only results in appropriate scaling of either the columns or rows of the transport matrix as shown in (III) and (IV). This will not change the factorization. If a blocker is present between the two patches, it will introduce additional diagonal elements in the matrix sub-blocks. This can only be handled by subdividing the blocks and factorizing at a finer level, as explained in Chapter 5.

[1] Note that our derivation is similar to that of Seitz et al. [SMK05]. They derive the formula for light transport between the first bounce 4D light field and outgoing 4D light field, whereas our derivation is for complete 5D radiance transfer.

2.3 Symmetry of the Transport Matrix

Helmholtz reciprocity, the idea that the flow of light can be effectively reversed without altering its transport properties, was proposed by von Helmholtz in his original treatise in 1856 [vH56]. He proposed the following reciprocity principle for beams traveling through an optical system (i.e., collections of mirrors, lenses, prisms, etc.):

Suppose that a beam of light **A** *undergoes any number of reflections or refractions, eventually giving rise (among others) to a beam* **B** *whose power is a fraction* **f** *of beam* **A***. Then on reversing the path of the light, an incident ray* $\mathbf{B}'$ *will give rise to a beam* $\mathbf{A}'$ *whose power is the same fraction* **f** *of beam* $\mathbf{B}'$*.*[2]

In other words, the path of a light beam is always reversible, and furthermore the relative power loss is the same for the propagation in both directions. Although Helmholtz only made this claim for specular interactions, Rayleigh later extended the reciprocity to include non-specular interactions [Ray00]. Helmholtz reciprocity has been exploited in many graphics applications to reduce computational complexity, for example, in ray-tracing systems [Whi80]. For a more complete discussion on reciprocity, interested readers are referred to Veach [Vea97].

For the purpose of this work, this reciprocity can be used to derive an equation describing the symmetry of the radiance transfer between incoming and outgoing directions ψ_i and ψ_o:

$$R(\psi_i;\psi_o) = R(\psi_o;\psi_i) \tag{2.8}$$

where R is the reflectance field (see Appendix A for a detailed proof). For the light transport matrix defined in Section 2.1, this implies that the transport of light between a ray i and and a ray j is equal in both directions, i.e.

$$\begin{aligned} \mathbf{T}[i,j] &= \mathbf{T}[j,i] \quad (2.9) \\ \Rightarrow \mathbf{T} &= \mathbf{T}^T \quad (2.10) \end{aligned}$$

Therefore, **T** is a symmetric matrix. This is also clear from Figure 2.1 row (c). Also, note that since we are looking at a subset of rays (4D from 5D), **T** is just a sub-block of $\widetilde{\mathbf{T}}$.

[2] Paraphrased from Chandrashekhar [Cha60]

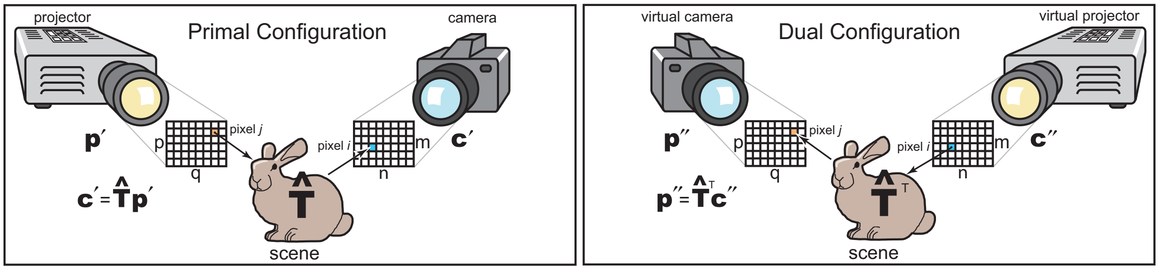


Figure 2.2: **Duality of the transport matrix.** The left diagram shows the primal configuration, with light being emitted by a real projector and captured by a real camera. Matrix $\widehat{\mathbf{T}}$ describes the light transport between the projector and the camera (element $\widehat{\mathbf{T}}[i,j]$ is the transport coefficient from projector pixel j to camera pixel i). The right diagram shows the dual configuration, with the positions of the projector and camera reversed. Suppose $\widehat{\mathbf{T}}''$ is the transport matrix in this dual configuration, so that $\widehat{\mathbf{T}}''[j,i]$ is the transport between pixel i of the virtual projector and pixel j of the virtual camera. As shown in Appendix A, Helmholtz reciprocity specifies that the pixel-to-pixel transport is equal in both directions, i. e. $\widehat{\mathbf{T}}''[j,i] = \widehat{\mathbf{T}}[i,j]$, which means $\widehat{\mathbf{T}}'' = \widehat{\mathbf{T}}^T$. As explained in the text, given $\widehat{\mathbf{T}}$, we can use $\widehat{\mathbf{T}}^T$ to synthesize the images that would be acquired in the dual configuration.

Therefore, **T** is also data-sparse.

2.4 Duality of the Transport Matrix

The symmetry property of the transport matrix holds when our acquisition system can measure the outgoing ray corresponding to each incoming ray. If the acquisition setup is such that the source of radiation (e.g. a projector) and the sensor (e.g. a camera) sample different subsets of incoming and outgoing rays respectively, then the transport matrix which describes this light transport is not symmetric. It turns out that the transport matrix exhibits an interesting duality property in this case.

We explain this with reference to Figure 2.2. We have a projector of resolution $p \times q$ shining light onto a scene and a camera of resolution $m \times n$ capturing the reflected light. Since the light transport is linear, we can express the light transport from the projector through the scene and into the camera with the following simple equation:

$$\mathbf{c}' = \widehat{\mathbf{T}}\mathbf{p}' \tag{2.11}$$

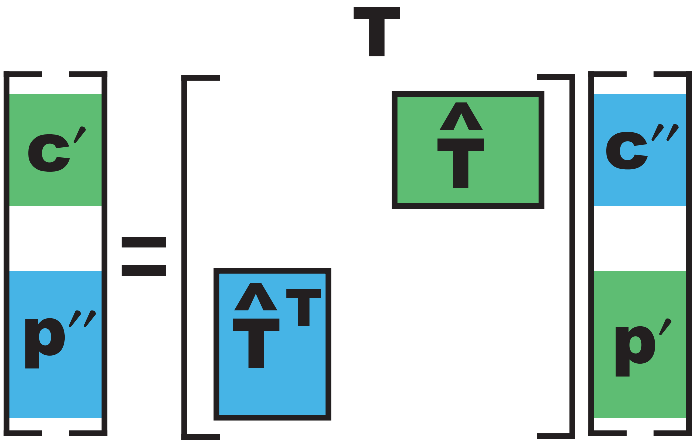


Figure 2.3: **Duality as a corollary of symmetry.** The duality of the light transport can be seen as a corollary of the symmetry of the transport matrix. Because $\mathbf{T} = \mathbf{T}^T$, therefore, if $\widehat{\mathbf{T}}$ describes the light transport from $\mathbf{p}'$ to $\mathbf{c}'$, then $\widehat{\mathbf{T}}^T$ describes the light transport from $\mathbf{c}''$ to $\mathbf{p}''$.

The column vector $\mathbf{p}'$ is the projected pattern (size $pq \times 1$), and $\mathbf{c}'$ (size $mn \times 1$) represents the image captured by the camera. Matrix $\widehat{\mathbf{T}}$ (size $mn \times pq$) is the transport matrix that describes how light from each pixel of $\mathbf{p}'$ arrives at each pixel of $\mathbf{c}'$.

We use the prime subscript ($'$) to indicate that we are working in the primal space to distinguish it from its dual counterpart, which we will introduce in a moment. Then, by using the principle of Helmholtz reciprocity as described in the previous section, we can represent the dual of eq. 2.11 as follows:

$$\mathbf{p}'' = \widehat{\mathbf{T}}^T \mathbf{c}'' \tag{2.12}$$

In this equation, the transport matrix $\widehat{\mathbf{T}}$ of the scene is the same as before except that we have now transposed it to represent the light going from the camera to the projector. We shall refer to eq. 2.11 as the “primal” equation and eq. 2.12 as the “dual” equation. In the dual space, $\mathbf{p}''$ represents the virtual image that would be visible at the projector if the camera were “projecting” pattern $\mathbf{c}''$. We can also derive this duality as a corollary to the symmetry of full transport matrix $\mathbf{T}$ of which $\widehat{\mathbf{T}}$ is just a sub-block. This is described in Figure 2.3

Thus, because of the duality, the $\widehat{\mathbf{T}}$ matrix can be acquired in either space and then transposed to represent the light transport in the other space. It is important to note that the two equations are not mathematical inverses of each other (i.e. $\widehat{\mathbf{T}}\widehat{\mathbf{T}}^T \neq \mathbf{I}$). This is because energy is lost in any real system through absorption or scattering. Therefore, if we measure $\mathbf{c}'$ after applying $\mathbf{p}'$, we cannot put this back in $\mathbf{c}''$ and expect the resulting $\mathbf{p}''$ to equal the original $\mathbf{p}'$.

Chapter 3

Dual Photography: Exploiting Sparseness

The full transport matrix between a single camera and projector pair is extremely large (on the order of $10^6 \times 10^6$ for conventional projector/camera resolutions). Thus, as described in the introduction (Chapter 1), a brute-force scan, in which each projector pixel is illuminated individually, would take days to acquire it. In this chapter, we develop a technique called *dual photography* which exploits the sparseness in the transport matrix to speed-up the acquisition process. The basic idea of this scheme is that instead of sequentially illuminating the projector pixels, we would multiplex the illuminated pixels spatially over the projector pattern, i.e. illuminate multiple pixels at the same time. In order to understand when we can do this multiplexing, let us analyze the following instance of the transport matrix:

$$\widehat{\mathbf{T}} = \begin{bmatrix} \mathbf{U1} & \mathbf{0} \\ \mathbf{0} & \mathbf{U2} \end{bmatrix} \tag{3.1}$$

where **U1** and **U2** are unknown. As per our abstraction of the transport matrix, the row indices correspond to the camera pixels while the column indices correspond to the projector pixels. For the above matrix, we observe that the blocks **U1** and **U2** are radiometrically isolated, i.e. the projector pixels corresponding to **U1** do not affect the camera pixels corresponding to **U2**; and vice versa. Thus, we can illuminate the projector pixels corresponding to **U1** in parallel with projector pixels corresponding to **U2** without mixing the contributions due to both in the camera image. The demultiplexing would simply require identifying the pixels corresponding to **U1** and **U2** in the camera image. The dual photography technique is based on this idea and it tries to identify such radiometrically isolated regions in the matrix. A sparse transport matrix will have many such regions and hence can be acquired efficiently by this technique. We describe the technique in subsequent sections.

3.1 Adaptive Multiplexed Illumination

Our algorithm tries to acquire the transport matrix with as few patterns as possible while ensuring that projector pixels affecting the same camera pixel are never illuminated simultaneously. We avoid such conflicts by subdividing the projector space adaptively; starting with the floodlit projector image, we subdivide it into four blocks, which are refined recursively until we reach the pixel level. Whenever we subdivide a block, we illuminate the four children in sequence. A walk-through example of the algorithm is given in Figure 3.1.

Two blocks can be investigated/subdivided in parallel if no camera pixel received a contribution from both blocks. At each level of the subdivision, we determine for each camera pixel k the blocks $\mathbf{B}_k = \{B_0, \ldots, B_n\}$ which illuminate pixel k either indirectly or directly. For all possible pairs of blocks contributing to the same camera pixel, we generate a conflict set $\mathbf{C}_k = \{(B_i, B_j) : B_i, B_j \in \mathbf{B}_k\}$. In the next iteration, the only blocks that need to be subdivided are in $\mathbf{B} = \bigcup_k \mathbf{B}_k$, i.e. only the blocks that contributed to any camera pixel. In this manner, blocks that do not contribute to the final image in any way are immediately culled away.

Given the set of all generated conflicts across all camera pixels $\mathbf{C} = \bigcup_k \mathbf{C}_k$, we define a graph $(\mathbf{B}, \mathbf{C})$. A graph coloring scheme is used to determine conflict-free subsets of $\mathbf{B}$ which can be investigated in parallel. While there might be a conflict for two blocks in one iteration, further subdivision might allow sub-blocks to be parallelized.

At the final subdivision level, each block is the size of a pixel. However, we can guarantee that no two projector pixels have a conflict in the camera image because these pixels would not have been scheduled in the same pattern otherwise. Because we know the history of the subdivisions for that pixel, we can determine the exact correspondence between projector and camera pixels. Using this fact, we can fill in the entries of the $\widehat{\mathbf{T}}$ matrix with the

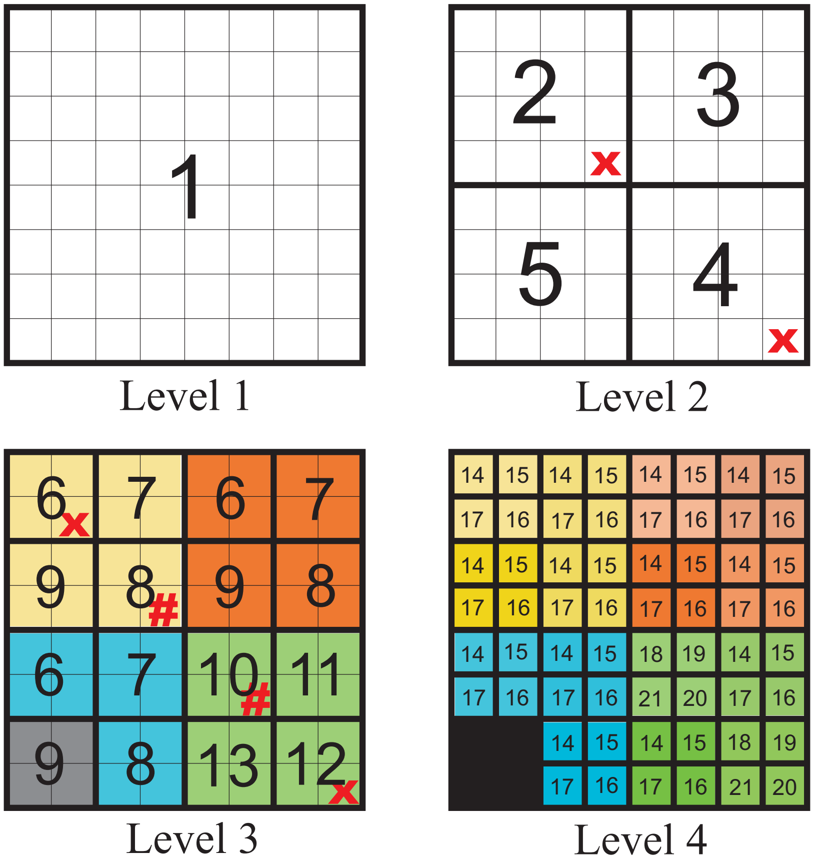


Figure 3.1: **Example of adaptive algorithm.** This example shows patterns that would be displayed on an 8 × 8 pixel projector for each level of subdivision. The numbers in each block indicate the frame when it is lit. In the first frame (level 1), all pixels are on. We subdivide it in level 2 into four children, which are acquired sequentially in clockwise order (frames 2-5). In this example, we assume some camera pixels respond to both blocks 2 and 4, e.g. due to inter-reflection within the scene. We denote the conflict between these blocks with red X's. In level 3 (frames 6-13), we ensure that the children of these two blocks are not scheduled for acquisition during the same frame. While acquiring level 3, we discover two additional conflicts: (6 and 12) and (8 and 10). Suppose also that block 9 in the lower left measured nothing, so it is culled. We now schedule level 4, avoiding scheduling the children of conflicting blocks together, which brings us to frame 21. Thus, we can acquire the transport matrix in this example with only 21 frames when 64 would be required with the brute-force scan. Had there been no conflicts, the number of frames would be the number of subdivision levels times four (4 children at each level) plus one for the floodlit image. This gives us $4 \times \log_4(pq) + 1 = 13$ for this example.

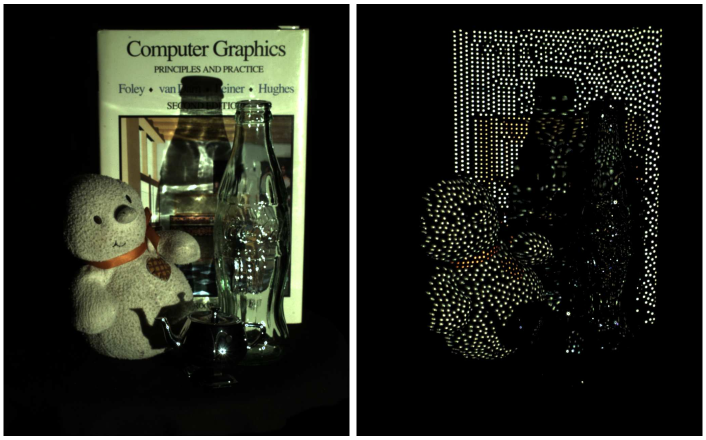


Figure 3.2: **Adaptively parallelized patterns.** This figure demonstrates that how our algorithm adapts to the scene content. Because of the complex light transport within the bottle visible in the left image, only a few pixels can be investigated in parallel in this region. Thus, the bottle remains relatively dark when we project an adaptive pattern as shown on the right.

values measured at the camera. Figure 3.2 shows the projection of one of the patterns onto the scene. The large number of white projector pixels highlights the efficient parallelization of the acquisition.

Although the adaptive parallelization algorithm just described works on most scenes, it may perform poorly in scenes where diffuse inter-reflections or subsurface scattering dominates the appearance. These scenes are particularly challenging because the energy emitted by a single projector pixel might be spread over large areas in the scene. In an extreme case, this overlap might cause the algorithm to schedule every pixel of the projector in a separate frame, thereby degenerating into the brute-force scan algorithm.

The adaptive scheme just presented can also fail to capture all of the energy measured by the projector. In certain cases, a point in the scene might reflect only a small fraction of its energy towards the camera. If this contribution is below the noise threshold of the camera, some blocks may be erroneously culled and their energy lost. This causes the technique to fail to capture diffuse-diffuse inter-reflections, as shown in Figure 3.3. We show in the next section that by modifying the adaptive algorithm to store the energy in a hierarchical fashion, we can avoid this problem.

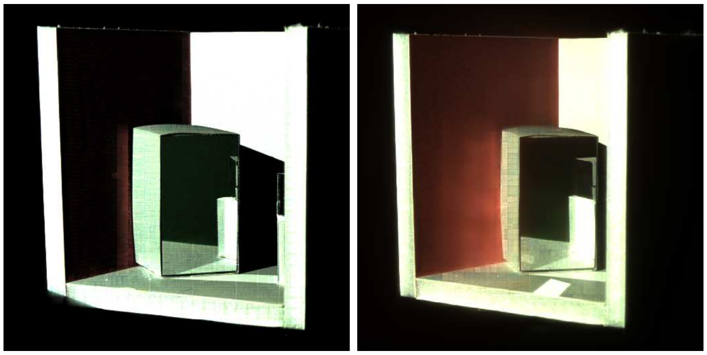
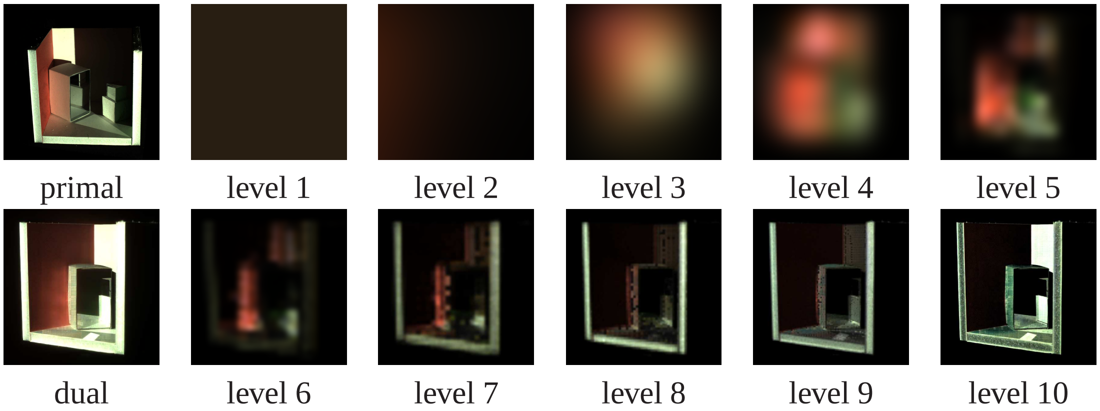


Figure 3.3: **The problem of capturing diffuse inter-reflections.** Applying the adaptive parallelization algorithm described in Section 3.1 and subdividing down to the pixel level, we produce the dual image on the left. The contribution of the diffuse inflections between the box and the red wall are nearly lost in the camera noise. Using the hierarchical assembly of the transport matrix described in Section 3.2, we preserve the energy from higher levels in the subdivision, leading to the improved dual image on the right.

3.2 Hierarchical Assembly of the Transport Matrix

To address the problem of signal loss for scenes with significant non-localized light transport, we employ a hierarchical representation of the transport matrix. This method is related to the wavelet environment matting technique by Peers and Dutré [PD03]. It is also similar to the hierarchical technique of Matusik et al. [MLP04], with adaptation added. Specifically, we follow the subdivision scheme of the previous section, but build a finer and finer representation of $\widehat{\mathbf{T}}$ at every subdivision level. At level 1, our approximation of $\widehat{\mathbf{T}}$ is simply a column vector of length mn representing the image $\mathbf{c}'$ captured while illuminating a floodlit image $\mathbf{p}'$. We call this approximation $\widehat{\mathbf{T}}_1$. Intuitively, $\widehat{\mathbf{T}}_1$ represents the light transport between the camera and a one-pixel projector. At the next level, our approximation $\widehat{\mathbf{T}}_2$ contains four columns, one for each of the four subdivided regions. This continues down to the pixel level where the matrix $\widehat{\mathbf{T}}_k$ with 4^{k-1} columns matches the resolution of the original $\widehat{\mathbf{T}}$. The energy for each element of the matrix is stored at only one level (at the highest possible resolution that still returns a measured value) since we do not want to double-count the energy. For a complete overview of the algorithm, readers are referred to the pseudocode in Appendix C.1.

The benefit of this hierarchical representation is that the energy is stored at the last



primal level 1 level 2 level 3 level 4 level 5

dual level 6 level 7 level 8 level 9 level 10

Figure 3.4: **Construction of the dual image with a hierarchical representation.** The primal and dual image show diffuse to diffuse inter-reflections which could only be captured by use of the hierarchical acquisition. Energy that might have been lost when further subdividing a block is deposited at a coarse level of the $\widehat{\mathbf{T}}$ matrix. To synthesize the dual image, the levels are individually reconstructed by applying the appropriate basis functions, then added together to obtain the image on the left. In this figure the intensity of the images for level 1 to 9 has been increased to visualize their contribution.

level where it can still be accurately measured. A threshold is used to decide whether to subdivide a specific block or to store its contribution at the current level of the hierarchical structure. If the contribution is stored, we terminate subdivision of that block. The threshold is set empirically and depends on the characteristics of the measurement system.

In order to synthesize an image from the acquired transport data, the contribution of the different levels of the hierarchy need to be added together for each pixel in the final image. This reconstruction can be expressed in the following mathematical form:

$$\mathbf{c}' = \sum_k f(\widehat{\mathbf{T}}_k \mathbf{p}') \tag{3.2}$$

$$\mathbf{p}'' = \sum_k f(\widehat{\mathbf{T}}_k^T \mathbf{c}'') \tag{3.3}$$

where eq. (3.2) is the rendering equation for primal configuration and eq. (3.3) is the rendering equation for the dual configuration.

We will explain the elements of these equations as we describe our rendering algorithm for the dual case, eq. (3.3) (note that the primal case is analogous). First, the desired light pattern $\mathbf{c}''$ is applied to the $\widehat{\mathbf{T}}$ approximation at each level to illuminate the scene

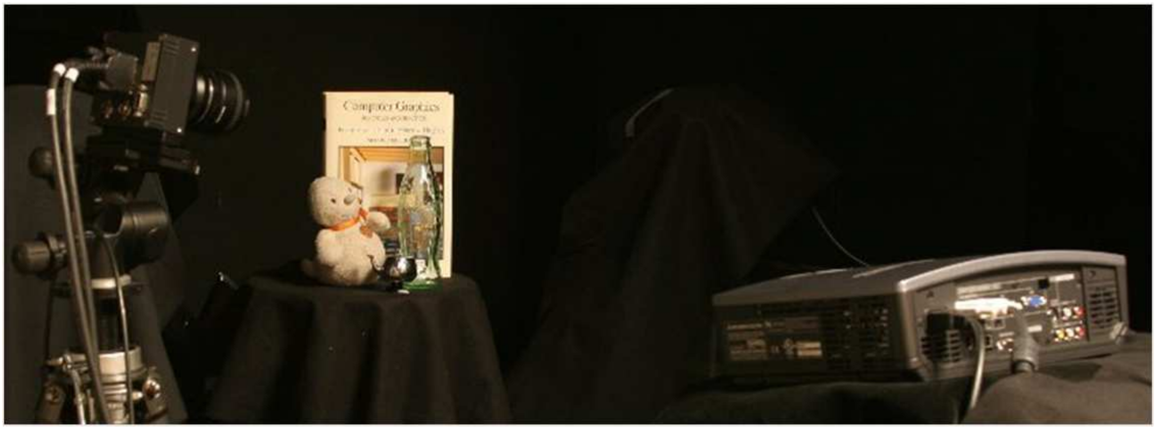


Figure 3.5: **Setup for dual photography.** A pattern loaded into projector on the right illuminates the scene and the camera on the left records the outgoing radiance. The camera used is a Basler A504kc (1280 × 1024 pixels) and the projector is a Mitsubishi X480U (1024 × 768 pixels). The setup is computer controlled, and we capture HDR images every 2.5 seconds.

for that level of the hierarchy. This is expressed by the dot product $\widehat{\mathbf{T}}_k^T \mathbf{c}''$ which results in a $4^{k-1} \times 1$ column vector. Referring to eq. (2.12), we see that this vector represents the $4^{k-1}-$pixel image that would be viewed at the projector under lighting $\mathbf{c}''$. Figure 3.4 shows a visualization of $f(\mathbf{T}_k{}^T \mathbf{c}'')$ of levels 1 through 10 for one of our scenes.

We must now add up the energy at each level to generate the final image. Because each of these vectors is of different dimensions, they must be resized to the final resolution of the image in order to be added up. We represent this resizing operation by the function f. The function f is needed to generate a continuous waveform from the discrete samples. In our present implementation, we use bi-cubic interpolation for f. Once the vectors have been resized, we simply add them to get the final image.

3.3 Acquisition Setup

The capture setup for the dual photography algorithm requires a projector and a camera. There is no restriction on the location of the camera and the projector. Also there is no geometric calibration required. Our particular capture setup is shown in Figure 3.5. The capture system is controlled by a computer. For a detailed description of the capture system architecture and the pre-processing steps involved before getting the image data, the reader is referred to Appendix B.

3.4 Results

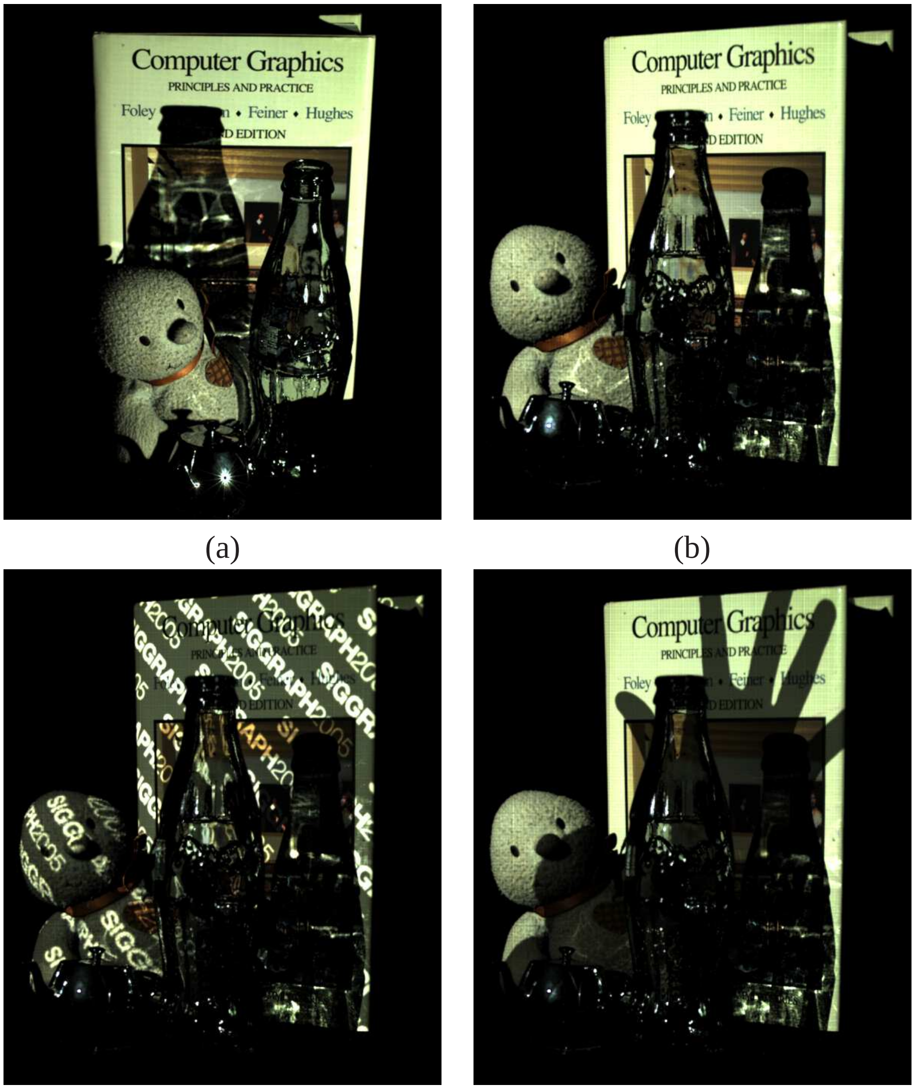
The technique presented allows us to efficiently capture the transport matrix $\widehat{\mathbf{T}}$ of a scene and measure many global illumination effects using only a moderate number of patterns and images. Figure 3.6 is an example of an image acquired with this technique. Note the capability of the algorithm to capture the complex refraction through the bottle. This image is 578×680 pixels and was acquired in a little over 2 hours. In contrast, a brute-force pixel scan would take almost 11 days at the same resolution (assuming 25 HDR images/min). Figure 3.7 shows two more scenes that were acquired using the hierarchical technique. To show that the algorithm accelerates our acquisition and results in a manageable size of the $\widehat{\mathbf{T}}$ matrix, we list relevant data for various scenes in Table 3.1. We compare it against calculated values for a brute-force pixel scan acquisition, assuming a capture rate of approximately 25 patterns/minute. The data is stored as three 32-bit floats for each matrix element. We can see that our technique is several orders of magnitude more efficient in both time and storage space, although further compression is still possible.

To characterize the effect of projector resolution on our hierarchical adaptive algorithm, we plot the number of acquired frames against projector resolution in Figure 3.8 for the box scene (Figure 3.4) and the bottle scene (Figure 3.6). As we increase the resolution exponentially, the curves approximate a straight line. This shows that the adaptive multiplexed illumination approach operates in $O(\log pq)$ time where pq is the projector resolution.

3.4.1 Scene Relighting

Once the transport matrix between the projector and the camera has been acquired, it can be used to relight the primal and dual images by multiplying $\widehat{\mathbf{T}}$ and $\widehat{\mathbf{T}}^T$ by the desired illumination vectors $\mathbf{p}'$ and $\mathbf{c}''$ respectively.

In Figure 3.6, we demonstrate this by relighting the dual image of the scene with a high frequency light pattern and can see that the caustics generated by the glass bottle vary



(a) (b)

(c) (d)

Figure 3.6: **Complex light transport example.** (a) Conventional photograph of a scene, illuminated by a projector with all its pixels turned on. (b) After measuring the light transport between the projector and the camera using structured illumination, our technique is able to synthesize a photorealistic image from the point of view of the projector. This image has the resolution of the projector and is illuminated by a light source at the position of the camera. The technique can capture subtle illumination effects such as caustics and self-shadowing. Note, for example, how the glass bottle in the primal image (a) appears as the caustic in the dual image (b) and vice versa. Because we have determined the complete light transport between the projector and camera, it is easy to relight the dual image using a synthetic light source (c) or a light modified matte captured later by the same camera (d).

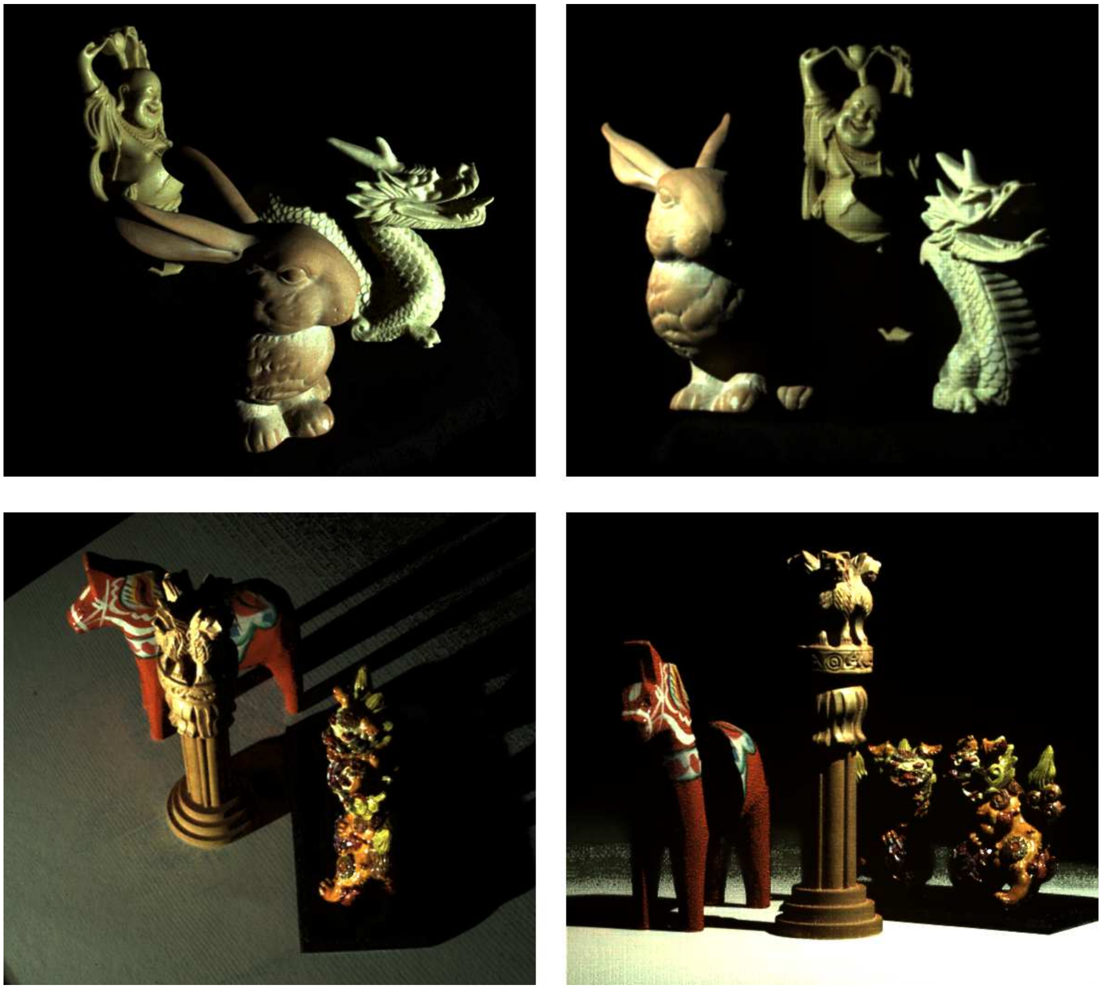


Figure 3.7: **Sample scenes.** The acquired primal image is on the left, the synthesized dual image on the right. Note for example the detail on the pillar in the dual image of the bottom row which is barely visible in the primal due to foreshortening.

	Brute-force scan		Dual photography		
SCENE	SIZE (TB)	TIME (days)	SIZE (MB)	TIME (min)	PATTERNS (#)
Fig. 3.4	3.7	7.3	179	14	352
Fig. 3.7 (top)	1.6	8.3	56	19	501
Fig. 3.7 (bottom)	1.4	8.3	139	15	369
Fig. 3.6	5.4	10.9	272	136	3397
Fig. 3.9 (80 positions)	114	362	6,675	1,761	19,140

Table 3.1: Table of relevant data (size, time and number of patterns) for different example scenes captured using the dual photography technique.

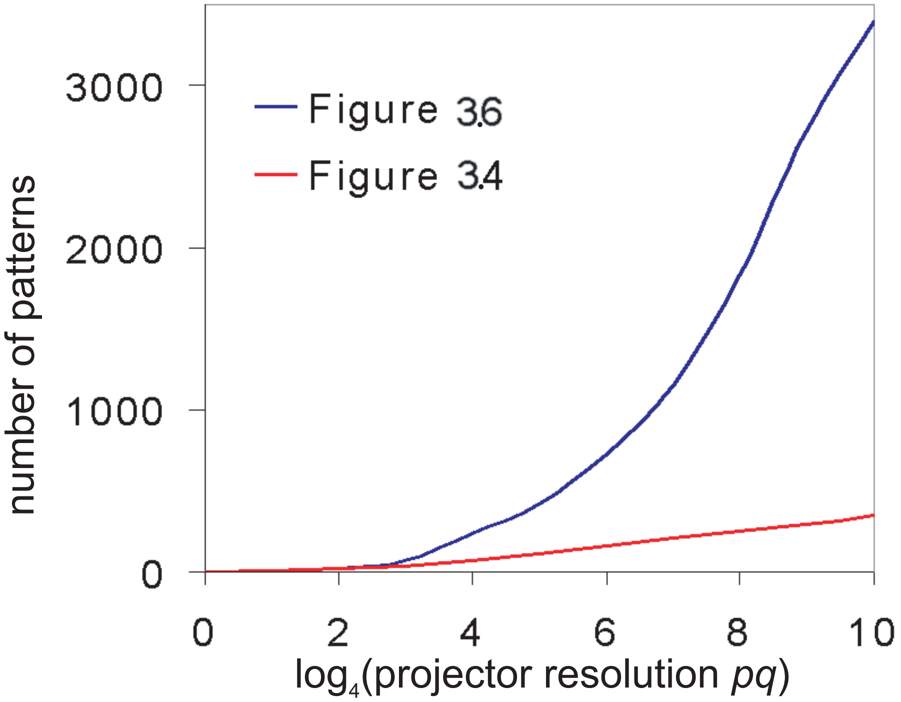


Figure 3.8: **Logarithmic behavior of adaptive algorithm.** Plot of the number of required patterns for the scenes in Figure 3.6 and Figure 3.4 against the projector resolution reveals that the algorithm operates in $O(\log pq)$ time.

spatially with illumination.

Since our adaptive algorithm is fast, we can also use a single camera to acquire the 6D reflectance field of a scene by moving it in a manner analogous to that of Masselus et al. [MPDW03] where they moved the projector. Figure 3.9 shows the relighting of a scene which was acquired in this manner by mounting the camera on a computerized gantry.

3.5 Discussion and Conclusions

In this chapter, we have developed an adaptive algorithm that looks for regions of the scene whose transport paths do not interact. This permits us to project many beams into the scene at once, letting us measure multiple entries of the transport matrix in parallel. Once we acquire the transport matrix, we have shown that the scene can be relit by multiplying the transport matrix by the appropriate vector. This allows us to illuminate a scene with a point light with directional control or an arbitrary light field.

There are some limitations of our technique, however. Scenes with significant global illumination would reduce the parallelism that the adaptive algorithm exploits. Since multiple projector pixels can affect overlapping regions in the camera after several bounces, they would be scheduled in separate passes. In the limit, the technique degenerates to a brute force scan.

Measuring inter-reflected transport paths accurately is also a problem, especially when the angle of view of the projector and the camera is large with respect to the scene. Figure 3.10 shows an example of a difficult scene to relight because the camera and the projector were at right angles to one another. In the next chapters we will use a data-sparse assumption on the matrix instead of sparseness assumptions and present the acquisition algorithm called symmetric photography based on this assumption in Chapter 5. This will alleviate the above mentioned problems.

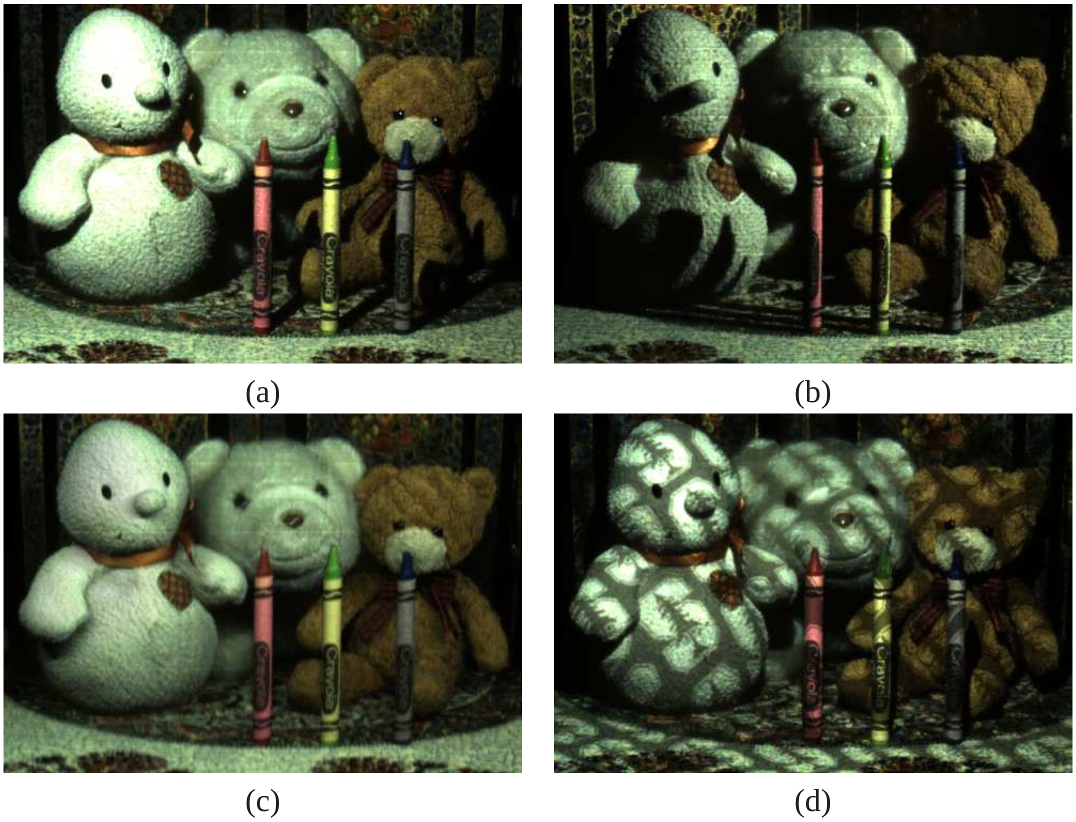


Figure 3.9: **Scene relit from multiple directions.** Using a camera mounted on a computerized gantry, we acquired the transport matrix for this scene at 80 different camera positions. We can then relight the dual image with a light source located at these positions, e.g. (a) one on the left or (b) on the right. By combining the contributions of these lights, we can illuminate the scene with an area light source (c) that casts soft shadows. Finally, because each transport is captured at high resolution, we can relight the scene using a high resolution matte as shown in (d).

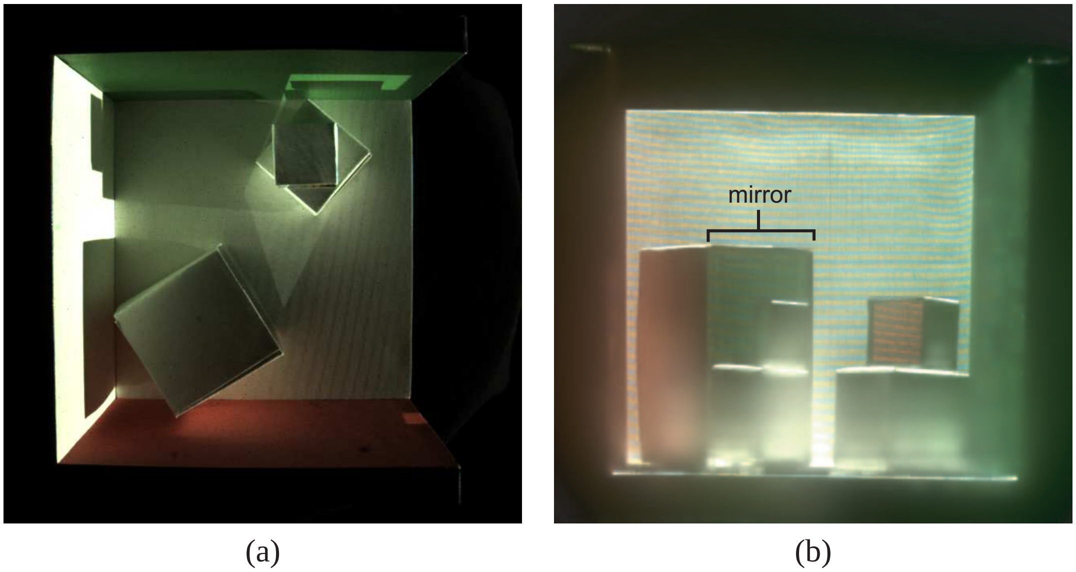


Figure 3.10: **Limits of the hierarchical assembly of the $\widehat{\mathbf{T}}$ matrix.** In this example, the camera is above the box looking down while the projector is on the right illuminating from the front, as can be seen in the primal image (a). Because of the large difference in the angle between the camera and projector viewpoints, there are many regions in the scene where there is no direct light transport. In these regions we are unable to resolve the transport matrix to the full resolution, and thus have to fall back to higher levels of the hierarchical tree. This results in a blurred dual image (b). Nevertheless, it is remarkable that the mirror reflection is captured, even though the mirror is barely visible in the primal image.

Chapter 4

Hierarchical Tensors: A Data Structure for Data-sparse Tensors

The dual photography technique described in Chapter 3 exploits the sparseness in the transport matrix (the fact that there are a lot of zeros) to speed-up the acquisition process. However, it has its limitations as the transport matrix is not necessarily sparse in many cases. But it turns out that the matrix is data-sparse, as explained in Section 2.2. We exploit this data-sparseness in the acquisition algorithm presented in Chapter 5. In this chapter, we will describe the data-structure that we will use to represent data-sparse matrices/tensors.

We introduce a new data structure called *hierarchical tensors* to represent data-sparse tensors. Hierarchical tensors are a generalization of hierarchical matrices (or $\mathscr{H}$-matrices) which we will explain first.

4.1 Hierarchical Matrices

Hierarchical matrices (or $\mathscr{H}$-matrices) were introduced by Hackbush [Hac99] in the applied mathematics community to represent arbitrary data-sparse matrices. The basic idea is to split a given matrix into a hierarchy of rectangular blocks and approximate each of the blocks by a low-rank matrix (see Figure 4.1). Specifically, at each level of the hierarchy, sub-blocks in the matrix are subdivided into 4 children (as in a quadtree). If a sub-block at

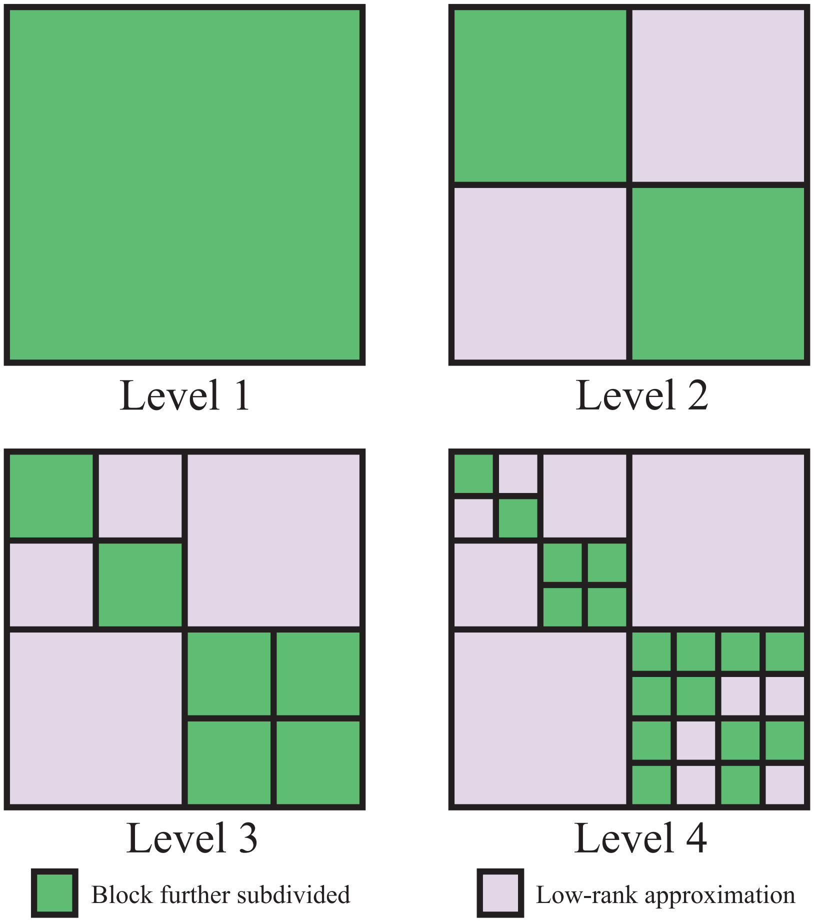


Figure 4.1: **Example of a hierarchical matrix.** This example shows 4 levels of a hierarchical matrix. The matrix is sub-divided like in a quadtree. Purple blocks are the sub-blocks for which a low-rank approximation is possible. They are not subdivided any further in the hierarchy. On the other hand, for green blocks a low-rank approximation is not possible. They are subdivided and investigated further. At level 1, the full matrix is investigated and it does not have a low-rank representation. Therefore, it is subdivided into 4 children which are investigated in level 2. At level 2, the top-right and bottom-left blocks are classified as low-rank and not subdivided any further whereas top-left and bottom-right blocks are scheduled for further investigation.

any level in the tree can be represented by a low-rank approximation, then it is not subdivided any further. Thus, a leaf node in the tree contains a low-rank approximation for the corresponding sub-block, which reduces to just a scalar value at the finest level in the hierarchy. Based on this structure, approximative algorithms for matrix arithmetic, inversion, preconditioning and even matrix equations can be introduced that work in almost optimal complexity. For a more complete discussion on $\mathscr{H}$-matrices, interested readers are referred to [BGH03].

4.2 Hierarchical Tensors

Consider the 4D reflectance field that describes the light transport for a single projector/camera pair. We have a 2D image representing the illumination pattern and a resulting 2D image captured by the camera. The connecting light transport can therefore be represented by a 4th-order tensor. One can alternatively flatten out the 2D image into a vector and represent the reflectance field with a matrix (a 2nd-order tensor), but that would destroy the spatial coherency present in a 2D image [WWS$^+$05]. To preserve coherency we represent the light transport by a 4th-order hierarchical tensor. Analogous to the hierarchical matrix representation, a node in the 4th-order hierarchical tensor is divided into 16 children at each level of the hierarchy. Thus, we call the hierarchical subdivision for a 4th-order tensor, a *sedectree*[1]. This is in continuation of the tradition of deriving the tree names from their Latin counterparts. e.g. *quadtree* from 4 nodes in a 2nd-order tensor subdivision and *octree* from 8 nodes in a 3rd-order tensor subdivision. Quadtree and octree were first introduced in the graphics community by Warnock [War69]; and Hunter and Steiglitz [Hun78] respectively and have been comprehensively surveyed in [Sam84].

Unlike in the case of a matrix where a unique low-rank approximation can always be obtained using SVD (Singular Value Decomposition), there is no analogous SVD for higher order tensors. Thus, there is no unique way to define the rank of a higher order tensor. There are two tensor decompositions that are commonly used. One is called the CANDECOMP-PARAFAC decomposition (CANonical DECOMPosition or PARAllel FACtors model) which was independently proposed by Carroll and Chang [CC70] and

[1] derived from *sedecim*, Latin equivalent of 16

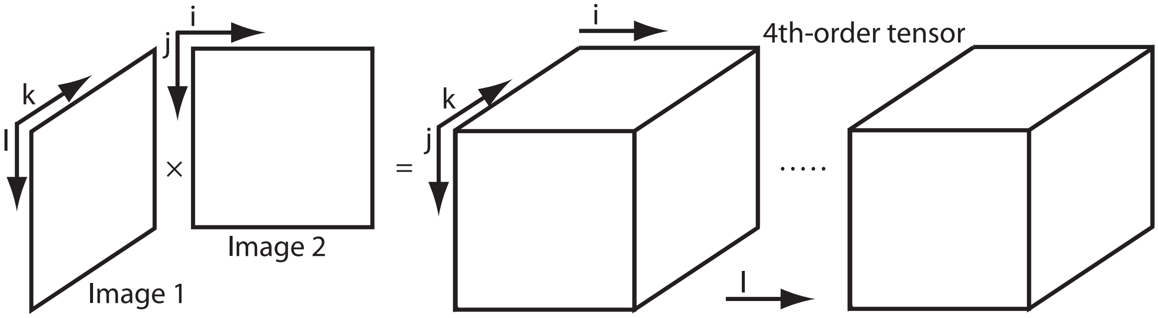


Figure 4.2: **Rank-1 approximation of a 4th-order tensor.** The figure illustrates how to represent a sub-block of a 4th-order tensor by a rank-1 approximation. The tensor is described by the tensor product of two 2D images. It is a rank-1 approximation because only one pair of images is needed.

Harshman [Har70]. This decomposition does not impose any orthogonality constraint on the vectors resulting from the decomposition but provides the concept of the rank of a tensor. The other commonly used tensor decomposition is called the HOSVD (Higher Order Singular Value Decomposition) which was proposed by [LMV00]. It is based on the Tucker decomposition form of a tensor proposed by [Tuc66]. Although this decomposition imposes orthogonality constraint on the vectors, there is no concept of tensor rank. HOSVD has been used in graphics community for efficient rendering [VT04] and compression [WWS$^+$05] from high dimensional data.

Under the general hierarchical tensor framework proposed here, any appropriate tensor decomposition can be used depending upon the demands of the application. In order to capture the data-sparseness in a tensor, for the purpose of this work, we represent the 4th-order tensor as a tensor product of two 2D images, one from the camera side and the other from the projector side, see Figure 4.2. We call it a rank-1 approximation, where the rank is defined by the number of 2D image pairs used for representing the tensor.

A hierarchical tensor offers many benefits for our purpose. Because of our rank-1 approximation, a sub-block of the 4th-order tensor can be stored compactly using just two 2D images. By using a clever hierarchical structure, the overhead of using a hierarchy can also be minimized. The worst case storage complexity of this representation is $(O(n^4))$ while the best case is $(O(n^2))$, where $n \times n$ is the image resolution. In practice, we will get something in between depending on the data-sparseness of the tensor. The data structure also provides constant time data-access during rendering from the data. This is because the depth of the tree in our hierarchical representation is constant.

4.3 Discussion and Conclusions

In this chapter we have introduced a new data structure called *hierarchical tensors* that can be used to efficiently represent data-sparse tensors. Hierarchical tensors are a generalization of hierarchical matrices. Besides providing an efficient representation for storage, it enables fast acquisition of the approximated transport matrix and fast rendering of the images from the captured matrix. Thus, the hierarchical tensor turns out to be a natural data-structure for our acquisition algorithm that we will explain in Chapter 5.

Chapter 5

Symmetric Photography: Exploiting Data-sparseness

In this chapter we will explain our symmetric photography technique for acquiring reflectance fields. Symmetric photography generalizes dual photography by exploiting data-sparseness rather than just sparseness. Here again, the goal is to be able to illuminate multiple projector pixels at the same time rather than illuminating them sequentially.

In order to understand how we can illuminate multiple projector pixels at the same time, let us assume that the transport matrix is:

$$\begin{bmatrix} \mathbf{U1} & \mathbf{M} \\ \mathbf{M^T} & \mathbf{U2} \end{bmatrix} = \begin{bmatrix} \mathbf{U1} & \mathbf{0} \\ \mathbf{0} & \mathbf{U2} \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{M} \\ \mathbf{M^T} & \mathbf{0} \end{bmatrix} \tag{5.1}$$

where **U1** and **U2** have not been measured yet. Note that here we are dealing with symmetric transport matrix. In Chapter 3 we utilized the fact that if $\mathbf{M} = \mathbf{0}$, then the unknown blocks **U1** and **U2** are radiometrically isolated, i.e. the projector pixels corresponding to **U1** do not affect the camera pixels corresponding to **U2** and vice versa. Thus, we can illuminate the projector pixels corresponding to **U1** and **U2** in parallel in such cases. Here, we observe that if the contents of **M** are known but not necessarily **0**, we can still radiometrically isolate **U1** and **U2** by subtracting the contribution of known **M** from the captured images. The RHS of eq. (5.1) should make this clear. We use this fact to illuminate the projector pixels corresponding to **U1** and **U2** in parallel when **M** is known.

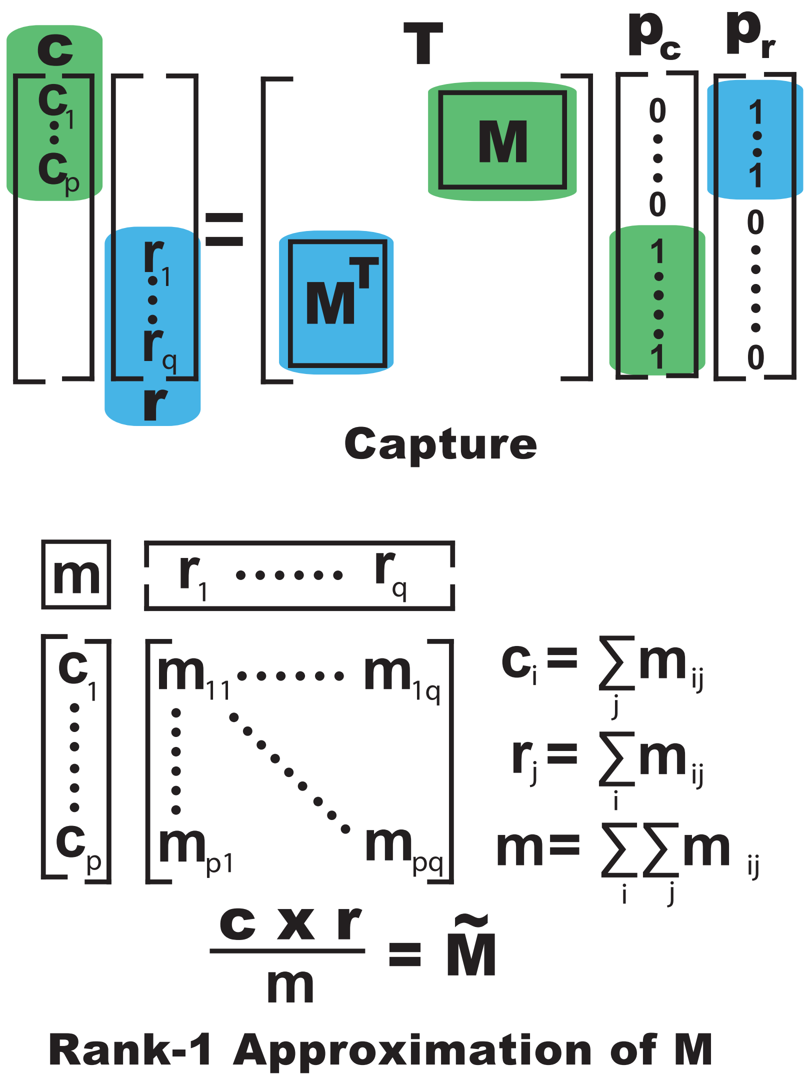


Figure 5.1: **Obtaining a rank-1 approximation.** An image captured by the camera is the sum of the columns in the transport matrix corresponding to the pixels illuminated by the projector. Therefore, on illuminating $\mathbf{p_c}$ the camera image **c** would be the sum of the columns of the matrix **M**; and on illuminating $\mathbf{p_r}$ the camera image **r** would be the sum of the columns of the matrix $\mathbf{M^T}$ or the sum of the rows of the matrix **M** as shown in the capture part. The tensor product of **c** and **r** after normalization with **m** provides a rank-1 approximation of **M**.

Now, consider a sub-block $\mathbf{M}$ of the transport matrix that is data-sparse and can be approximated by a rank-1 factorization. We can obtain this rank-1 factorization by just capturing two images. An image captured by the camera is the sum of the columns in the transport matrix corresponding to the pixels illuminated by the projector. Because of the symmetry of the transport matrix, this image is also the sum of corresponding rows in the matrix. Therefore, by shining just two projector patterns, $\mathbf{p_c}$ and $\mathbf{p_r}$, we can capture images such that one provides the sum of the columns, $\mathbf{c}$ and the other provides the sum of the rows, $\mathbf{r}$ of $\mathbf{M}$ ($\mathbf{c} = \mathbf{M}\mathbf{p_c}$ and $\mathbf{r} = \mathbf{M}^{\mathbf{T}}\mathbf{p_r}$). A tensor product of $\mathbf{c}$ and $\mathbf{r}$ after appropriate normalization directly provides a rank-1 factorization for $\mathbf{M}$ (see Figure 5.1). Thus, the whole sub-block can be constructed using just two illumination patterns. This is the key idea behind our algorithm. The algorithm tries to find such sub-blocks $\mathbf{M}$ in $\mathbf{T}$ that can be represented as a rank-1 approximation by a hierarchical subdivision strategy. Once measured, these sub-blocks can be used to parallelize the acquisition as described above.

5.1 Hierarchical Acquisition Scheme

Our acquisition algorithm follows the structure of the hierarchical tensor described in Chapter 4. At each level of the hierarchy we illuminate the scene with a few projector patterns. We use the captured images to decide which nodes of the tensor in the previous level of hierarchy are rank-1. Once a node has been determined to be rank-1, we do not subdivide it any further as its entries are known. The nodes which fail the rank-1 test are subdivided and scheduled for investigation during the next iteration. The whole process is repeated until we reach the pixel level. We initiate the acquisition by illuminating with a floodlit projector pattern. The captured floodlit image provides a possible rank-1 factorization of the root node of the hierarchical tensor. The root node is scheduled for investigation in the first iteration.

For each level, the first step is to decide what illumination patterns to use. In order to speed-up our acquisition, we need to minimize the number of these patterns. To achieve this, our algorithm must determine the set of projector blocks which can be illuminated in the same pattern. To determine this, we divide each scheduled node into 16 children and the 4 blocks in the projector corresponding to this subdivision are accumulated in a list

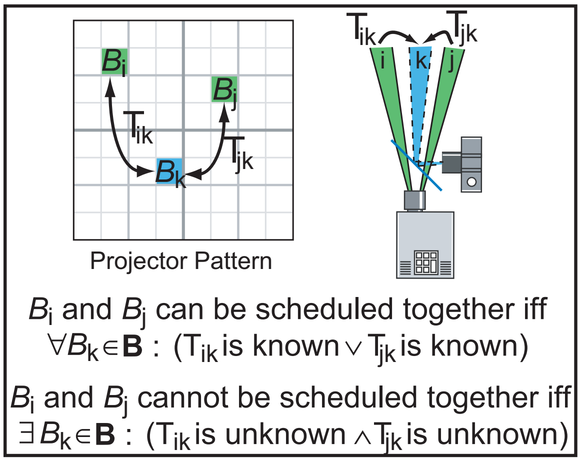

B_i
B_j
T_{ik}
T_{jk}
B_k
Projector Pattern
T_{ik}
T_{jk}
i
k
j
B_i and B_j can be scheduled together iff
$\forall B_k \in \mathbf{B}$: (T_{ik} is known $\vee$ T_{jk} is known)
B_i and B_j cannot be scheduled together iff
$\exists B_k \in \mathbf{B}$: (T_{ik} is unknown $\wedge$ T_{jk} is unknown)

Figure 5.2: **Determining block scheduling.** Two blocks B_i and B_j can be scheduled in the same frame if and only if, $\forall B_k \in \mathbf{B}$, the light transport T_{ik} or T_{jk} is known. Therefore, two blocks B_i and B_j cannot be scheduled in the same frame if and only if, $\exists B_k \in \mathbf{B}$, such that the light transports T_{ik} and T_{jk} are both unknown. This is because upon illuminating B_i and B_j simultaneously, the block B_k in the camera will measure the combined contribution of both T_{ik} and T_{jk}. If both of these are unknown at this point there is no way to separate them out.

$\mathbf{B} = \{B_1, B_2, ..., B_n\}$. Figure 5.2 describes the condition when two blocks B_i and B_j can be scheduled in parallel. It can be written as the following lemma:

Lemma 1 (Block Scheduling Lemma) *Two blocks B_i and B_j can be scheduled together if and only if, $\forall B_k \in \mathbf{B}$, at least one of T_{ik} or T_{jk} are known.*

We can use this lemma to derive a corollary for when two blocks B_i and B_j cannot be illuminated in parallel. We will use this corollary for our block scheduling algorithm.

Lemma 2 (Block Scheduling Lemma Corollary) *Two blocks B_i and B_j can not be scheduled together if and only if, $\exists B_k \in \mathbf{B}$, such that both T_{ik} and T_{jk} are not known.*

Since the direct light transport T_{ii} is not known until the bottom level in the hierarchy, any two blocks B_i and B_j for which T_{ij} is not known cannot be scheduled in parallel. For all such possible block pairs for which the light transport has not been measured yet, let us construct a set $\mathbf{C} = \{(B_i, B_j) : B_i, B_j \in \mathbf{B}\}$. Given these two sets, we define an undirected graph $\mathbf{G} = (\mathbf{B}, \mathbf{C})$, where $\mathbf{B}$ is the set of vertices in the graph and $\mathbf{C}$ is the set of edges.

Thus, the vertices in the graph have an edge between them if the light transport between the corresponding blocks is not known. In this graph, any two vertices B_i and B_j which do not have an edge between them, but have a direct edge with a common block B_k (as shown in Figure 5.2), also satisfy the lemma. Therefore, we cannot schedule them in parallel. Such blocks correspond to vertices at a distance two from each other in our graph $\mathbf{G}$. In order to capture these blocks as direct edges in a graph, we construct another graph $\mathbf{G}^2$ which is the square of graph $\mathbf{G}$ [Har01]. The square of a graph contains an edge between any two vertices which are at most distance two away from each other in the original graph. Thus, in the graph $\mathbf{G}^2$, any two vertices which are not connected can be scheduled together. We use a graph coloring algorithm on $\mathbf{G}^2$ to obtain subsets of $\mathbf{B}$ which can be illuminated in parallel. Once the images have been acquired, the known intra-block light transport is subtracted out for the blocks that were scheduled in the same frame.

In the next step, we use these measurements to test if the tensor nodes in the previous level of the hierarchy can be factorized using rank-1 approximation. We have a current rank-1 approximation for each node from the previous level in the hierarchy. The 8 measured images, corresponding to 4 blocks from the projector side and 4 blocks from the camera side of a node, are used as test cases to validate the current approximation (note that there are only 4 measured images if the tensor node is on-diagonal and hence symmetric). This is done by rendering estimate images for these blocks using the current rank-1 approximation. The 8 estimated images are compared against the corresponding measured images and an RMS (root mean square) error is calculated for the node. A low RMS error indicates our estimates are as good as our measurements and we declare the node as rank-1 and stop any further subdivision on this node. If on the other hand the RMS error is high, the 16 children we have measured become the new nodes. The 4 images from the projector side and the 4 images from the camera side are used to construct the 16 (4×4) rank-1 estimates for them. These nodes are scheduled for investigation in the next iteration.

A tensor node containing just a scalar value is trivially rank-1. Therefore, the whole process terminates when the size of the projector block reduces to a single pixel. Upon finishing, the scheme directly returns the hierarchical tensor for the reflectance field of the scene. For a complete overview of the algorithm, readers are referred to the pseudocode in Appendix C.2.

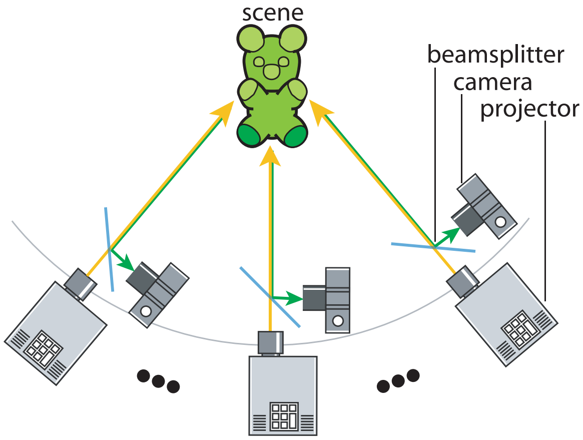


Figure 5.3: **Schematic of symmetric photography setup.** A coaxial array of projectors and cameras provides an ideal setup for symmetric photography. The projector array illuminates the scene with an incoming light field. Since the setup is coaxial, the camera array measures the corresponding outgoing light field.

Once the tensor has been created, the rendering from it is quite straightforward. Any incoming light pattern is partitioned according to the hierarchy and multiplied by the appropriate sub-block in the tensor. Since the size of the hierarchy is constant the worst case access time for any entry of the tensor is constant.

5.2 Acquisition Setup

In order to experimentally validate our ideas we need an acquisition system that is capable of simultaneously emitting and capturing along each ray in the light field. This suggests having a coaxial array of cameras and projectors. Figure 5.3 shows the schematic of such a setup. Our actual physical implementation is built using a single projector, a single camera, a beam-splitter and an array of planar mirrors. The projector and the camera are mounted coaxially using the beam splitter on an optical bench as shown in Figure 5.4, and the mirror array divides the projector/camera pixels into 9 coaxial pairs. Once the optical system has been mounted it needs to be calibrated. First, the center of projection of the camera and projector is aligned. The next task is to find the per pixel mapping between the projector and camera pixels. We use a calibration scheme similar to that used by Han and Perlin

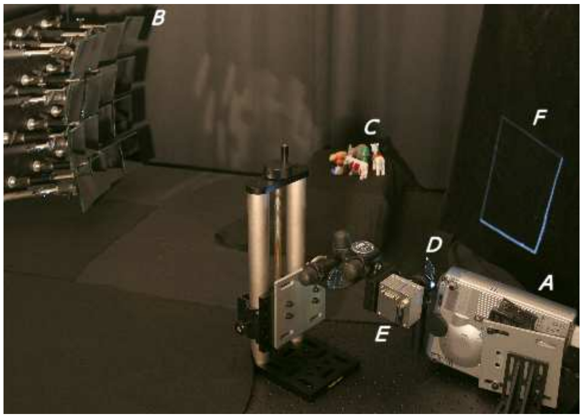


Figure 5.4: **Coaxial setup for capturing 8D reflectance fields.** A pattern loaded into projector at *A* illuminates a 3 × 3 subset of the 4 × 4 array of planar mirrors at *B*. This provides us with 9 virtual projectors which illuminate our scene at *C*. The light that returns from the scene is diverted by a beam-splitter at *D* towards a camera at *E*. Any stray light reflected from the beam-splitter lands in a light trap at *F*. The camera used is an Imperx IPX-1M48-L (984 × 1000 pixels) and the projector is a Mitsubishi XD60U (1024 × 768 pixels). The setup is computer controlled, and we capture HDR images every 2 seconds.

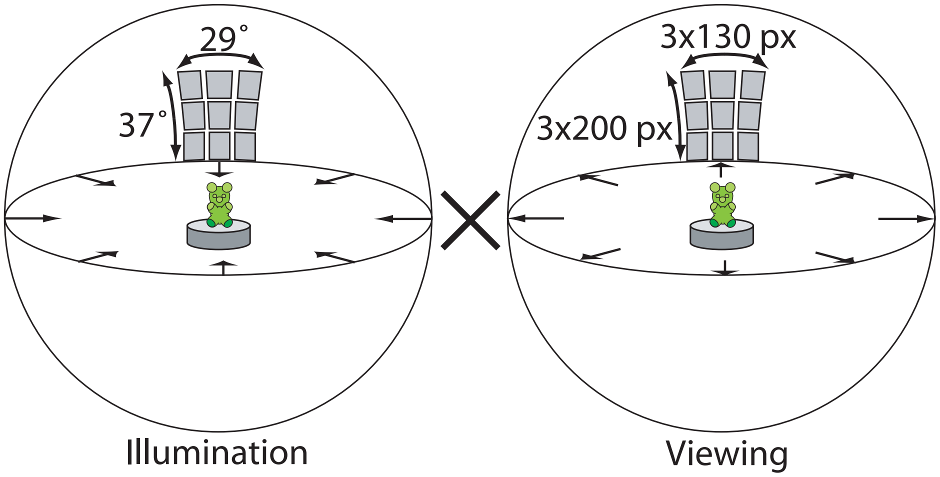


Figure 5.5: **Region of the sphere sampled by our setup.** Our setup spans an angular resolution of 37° × 29° on the sphere both for the illumination and view directions. The spatial resolution in each view is 130 × 200 pixels. This accounts for about 2% of the total rays in the light field at the current sampling rate of the capture system.

	Symmetric photography			Brute-force
SCENE	SIZE	TIME	PATTERNS	PATTERNS
	(MB)	(min)	(#)	(#)
Fig. 2.1(I)	255	44	809	108,500
Fig. 2.1(II)	371	70	1,085	108,500
Fig. 2.1(III)	334	65	1,081	108,500
Fig. 2.1(IV)	274	46	841	108,500
Fig. 5.6	337	151	2,417	91,176
Fig. 5.8	1,470	484	3,368	233,657

Table 5.1: Table of relevant data (size, time and number of patterns) for different example scenes captured using our technique. Note that the algorithm requires about 2 orders of magnitude fewer patterns than the brute-force scan.

[HP03] and Levoy et al. [LCV$^+$04] in their setup to find this mapping. Figure 5.5 illustrates the angular and spatial resolution of reflectance fields captured using out setup. The capture system is controlled by a computer. For a detailed description of the capture system architecture and the pre-processing steps involved before getting the image data, the reader is referred to Appendix B.

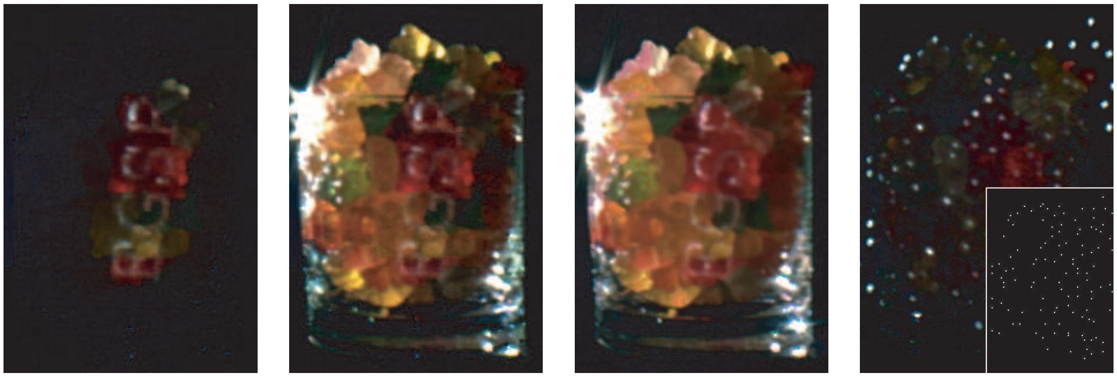
5.3 Results

We capture reflectance fields of several scenes using this technique. For reference, Table 5.1 provides statistics (size, time and number of patterns required for acquisition) for each of these datasets.

In Figure 2.1, we present the results of our measurement for four simple scenes consisting of planes. This experiment has been designed to elucidate the structure of the **T** matrix. A coaxial projector/camera pair is directly aimed at the scene in this case. The image resolution is 310×350 pixels. Note the storage, time and number of patterns required for the four scenes (listed in Table 5.1). A brute-force scan, in which each projector pixel is illuminated individually, to acquire these **T** matrices would take at least 100 times more images. Also, since the energy in the light after an indirect bounce is low, the camera would have to be exposed for a longer time interval to achieve good SNR during brute-force scanning. On the other hand, in our scheme, the indirect bounce light transport is resolved earlier in the hierarchy (see rows (c) and (d) in Figure 2.1). At higher levels of the hierarchy, we are illuminating with bigger projector blocks (and hence throwing more light into the scene than just from a single pixel), therefore we are able to get good SNR even with small exposure times. Also, note that the high frequency of the textures does not affect the data-sparseness of reflectance fields. The hierarchical subdivision follows almost the same strategy in all four cases as visualized in row (d). In row (e), we show the results of relighting the scene with a vertical bar. The smooth glow from one plane to the other in column (II), (III) and (IV) shows that we have measured the indirect bounce correctly.

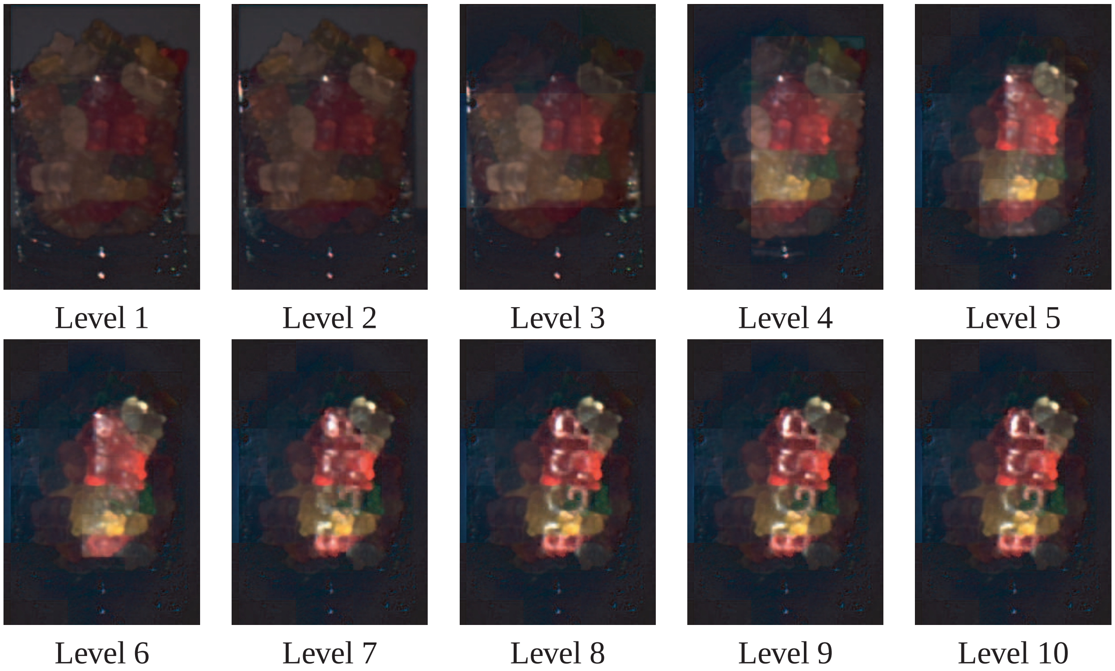
Figure 5.6 demonstrates that our technique works well for acquiring the reflectance fields of highly sub-surface scattering objects. The image (240×340 pixels) reconstructed from relighting with a spatially varying illumination pattern (see Figure 5.6(b)) is validated against the ground-truth image (see Figure 5.6(c)). We also demonstrate the result of reconstructing at different levels of the hierarchical tensor for this scene in Figure 5.7. This figure also explains the difference between our hierarchical tensor representation and a wavelet based representation.

Figure 5.8 shows the result of an 8D reflectance field acquired using our setup. The captured reflectance field can be used to view the scene from multiple positions (see Figure 5.8(b)) and also to relight the scene from multiple directions (see Figure 5.8(a)). The resolution of the reflectance field for this example is about $3 \times 3 \times 130 \times 200 \times 3 \times 3 \times 130 \times 200$. The total size of this dataset would be 610 GB if three 32-bit floats were used for each entry in the transport matrix. Our hierarchical tensor representation compresses it to 1.47 GB. A brute force approach would require 233,657 images to capture it. Our algorithm only needs 3,368 HDR images and takes around 8 hours to complete. In our current implementation, the processing time is comparable to the actual image capture time. We believe that the acquisition times can be reduced even further by implementing a parallelized version of our algorithm. Rendering a relit image from our datasets is efficient and takes less than a second on a typical workstation.



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

Figure 5.6: **Subsurface scattering scene.** The reflectance field of a glass full of gummy bears is captured using two coaxial projector/camera pairs placed 120° apart. (a) is the result of relighting the scene from the front projector, which is coaxial with the presented view, where the (synthetic) illumination consists of the letters "EGSR". Note that due to their sub-surface scattering property, even a single beam of light that falls on a gummy bear illuminates it completely, although unevenly. In (b) we simulate homogeneous backlighting from the second projector combined with the illumination used in (a). For validation, a ground-truth image (c) was captured by loading the same projector patterns into the real projectors. Our approach is able to faithfully capture and reconstruct the complex light transport in this scene. (d) shows a typical frame captured during the acquisition process with the corresponding projector pattern in the inset.



Level 1 Level 2 Level 3 Level 4 Level 5

Level 6 Level 7 Level 8 Level 9 Level 10

Figure 5.7: **Reconstruction results for different levels of hierarchy.** This example illustrates the relighting of the reflectance field of the gummy bears scene (Figure 5.6) with the illumination pattern used in Figure 5.6(a), if the acquisition was stopped at different levels of the hierarchy. Note that at every level, we still get a full resolution image. This is because we are approximating a node in the hierarchy as a tensor product of two 2-D images. Therefore, we sill have a measurement for each pixel in the image, though scaled incorrectly. This is different from wavelet based approaches where a single scalar value is assigned for a node in the hierarchy implying lower resolution in the image at lower levels. Note that at any level, the energy of the projected pattern is distributed over the whole block that it is illuminating. This is clear from the intensity variation among blocks, especially in the images at levels 3, 4, and 5.

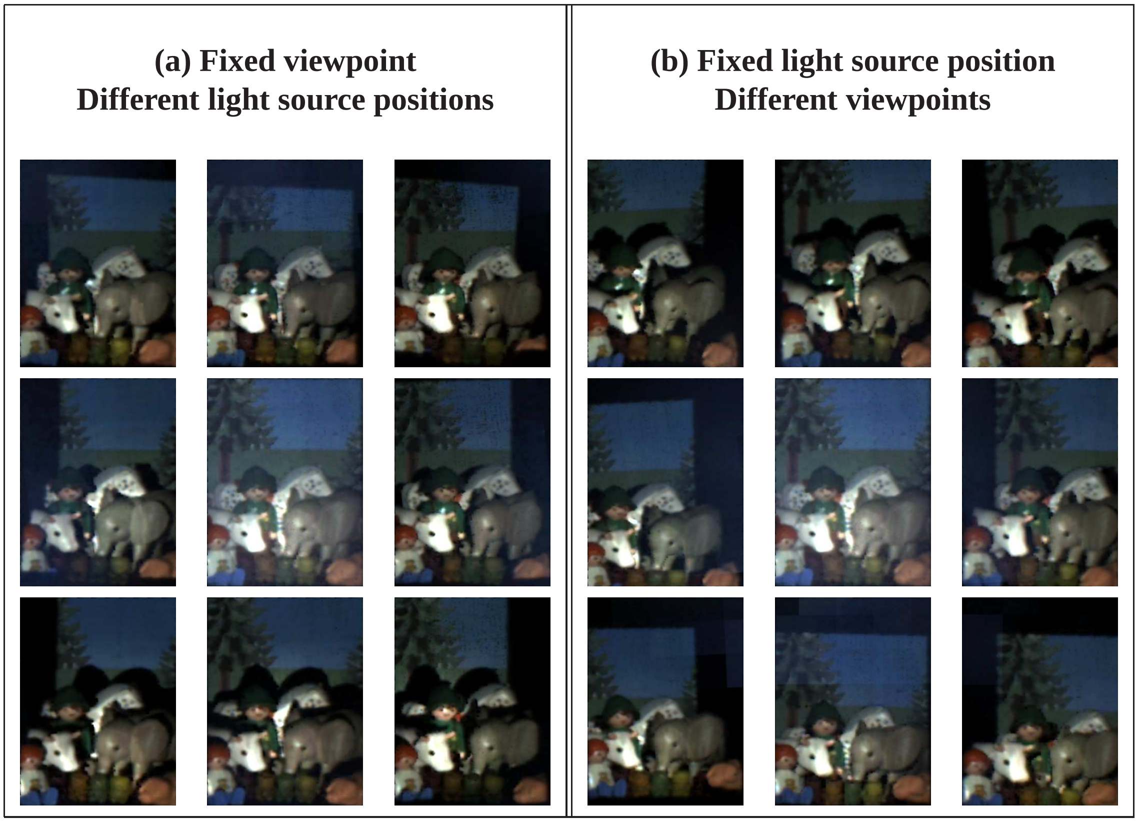
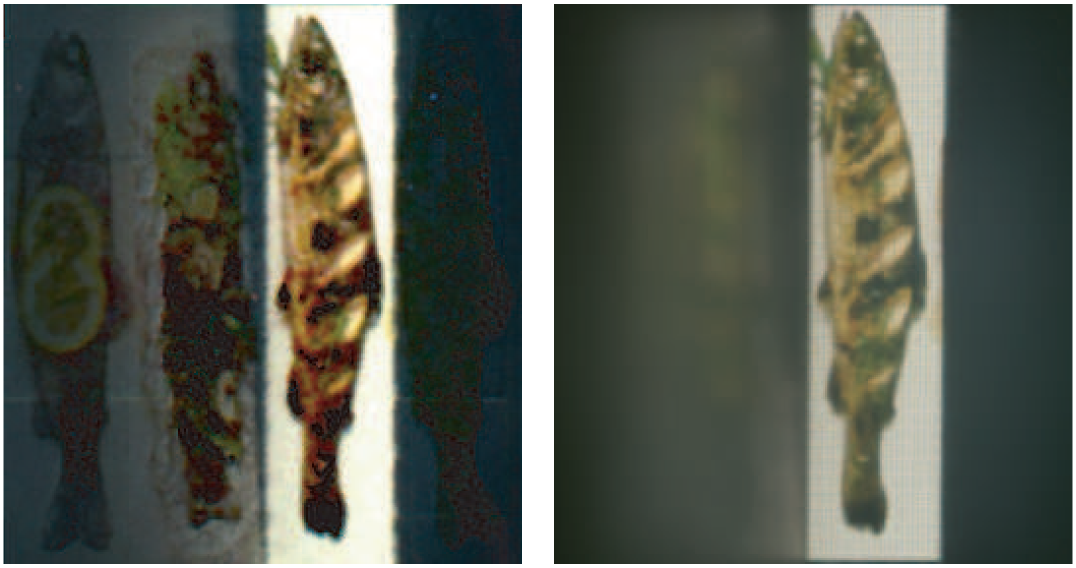

(a) Fixed viewpoint
Different light source positions
(b) Fixed light source position
Different viewpoints

Figure 5.8: **8D reflectance field of an example scene.** This reflectance field was captured using the setup described in Figure 5.4. A 3×3 grid of mirrors was used. In (a) we see images rendered from the viewpoint at the center of the grid with illumination coming from 9 different locations on the grid. Note that the shadows move appropriately depending upon the direction of incident light. (b) shows the images rendered from 9 different viewpoints on the grid with the illumination coming from the center. In this case one can notice the change in parallax with the viewpoint. Note that none of these images were directly captured during our acquisition. The center image in each set looks slightly brighter because the viewpoint and lighting are coincident in this case.

5.3.1 Comparison with Dual Photography

It is instructive to compare the symmetric photography technique against the dual photography technique presented earlier. Dual photography reduces the acquisition time by exploiting only sparseness (the fact that there are regions in a scene that are radiometrically independent of each other). These regions are detected and measured in parallel in dual photography. However, for a scene with many inter-reflections or sub-surface scattering, such regions are few and the technique performs poorly. In order to resolve the transport at full resolution, the technique would reduce to brute-force scanning for such scenes. Illuminating with single pixel for observing multiple scattering events has inherent SNR problems because the energy of indirect bounce light transport coefficients could be extremely low. The measurement system, which is limited by the black level of the projector and dark noise of the camera cannot pick up such low values. The scheme therefore stops refining at a higher level in the hierarchy and measures only a coarse approximation of the indirect bounce light transport. This essentially results in a low-frequency approximation for indirect bounce light transport. Thus, the fidelity of the images generated using symmetric photography is better than those generated using dual photography. The comparison of the two techniques in Figure 5.9 confirms this behavior. Since symmetric photography is probing the matrix from both sides, the high frequencies in indirect bounce light transport are still resolved whereas, dual photography can only produce a low frequency approximation of the same. Furthermore, while symmetric photography took just 841 HDR images for this scene, dual photography required 7382 HDR images. Note that upon comparing Tables 3.1 and 5.1, one might think that we are getting better speedups in case of dual photography, but the example scenes used in dual photography are easier than those used in symmetric photography. Symmetric photography will perform atleast as well on such scenes.

Finally, Figure 5.10 illustrates the relative percentage of rank-1 vs empty leaf nodes at various levels of the hierarchy for the transport matrices that we have captured. The empty leaf nodes correspond to sparse regions of the matrix while the rank-1 leaf nodes correspond to data-sparse regions of the matrix. While dual photography only exploits sparseness and hence culls away only empty leaf nodes at a particular level, symmetric photography exploits both data-sparseness and sparseness and culls away both rank-1 and



(a) Symmetric (b) Dual

Figure 5.9: **Symmetric vs. Dual Photography.** The figure illustrates the strength of the symmetric photography technique (a) when compared against the dual photography technique (b) of Chapter 3. The setup is similar to the book example of Figure 2.1 (IV). In both cases, the right half of the book is synthetically relit using the transport matrices captured by the respective techniques. Note that in the case of symmetric photography (a), the high frequencies in the left half of the book are faithfully resolved while in dual photography (b), the frequencies cannot be resolved and just appear as a blur. The light transport for (a) was acquired using 841 images, while that for (b) was acquired using 7382 images. The slight difference in overall color of the two images is due to the fact that the two images store the energies slightly differently.

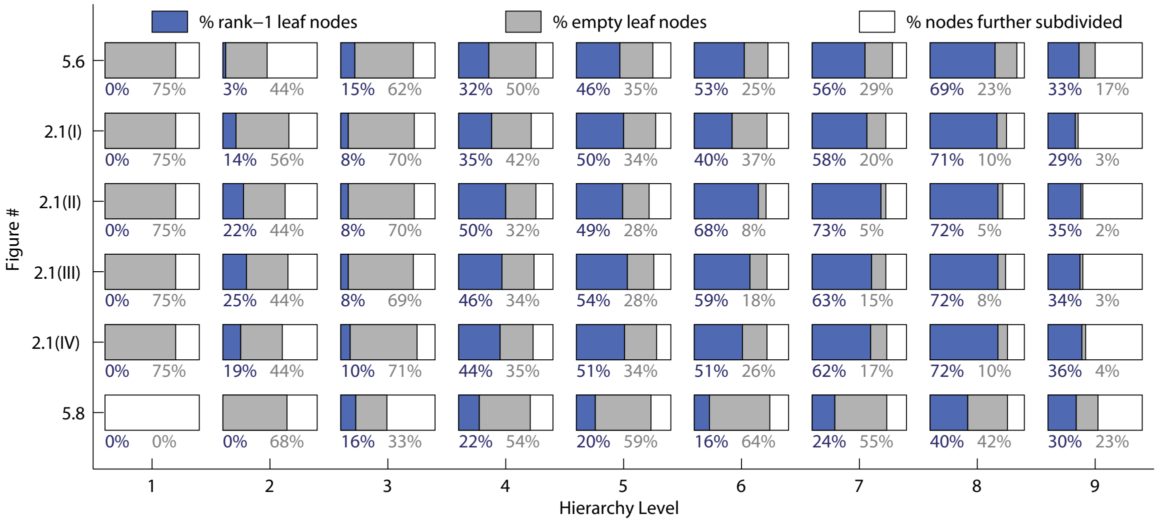


Figure 5.10: **Comparison of rank-1 vs empty leaf nodes.** The figure empirically compares the percentage of rank-1 vs. empty leaf nodes in the hierarchical tensor at different levels of the hierarchy for various scenes captured using our acquisition scheme. The blue area in each bar represents the percentage of rank-1 nodes while the gray area corresponds to the percentage of empty nodes. The white area represents the nodes which are subdivided at next level. Note that at levels 4, 5, 6, 7, 8 and 9 a significant fraction of leaf nodes are rank-1. Also note that for Figures 2.1(I) and 2.1(II), at levels 6, 7, and 8, there are far more empty nodes in 2.1(I) than in 2.1(II). This is what we expected as the transport matrix for 2.1(I) is sparser than that for 2.1(II).

empty leaf nodes. Note that between levels 4 and 9, there is a significant fraction of rank-1 nodes which are culled away by symmetric photography in addition to empty leaf nodes. This results in large reduction of nodes that still have to be investigated and results in a significantly faster acquisition as compared to dual photography.

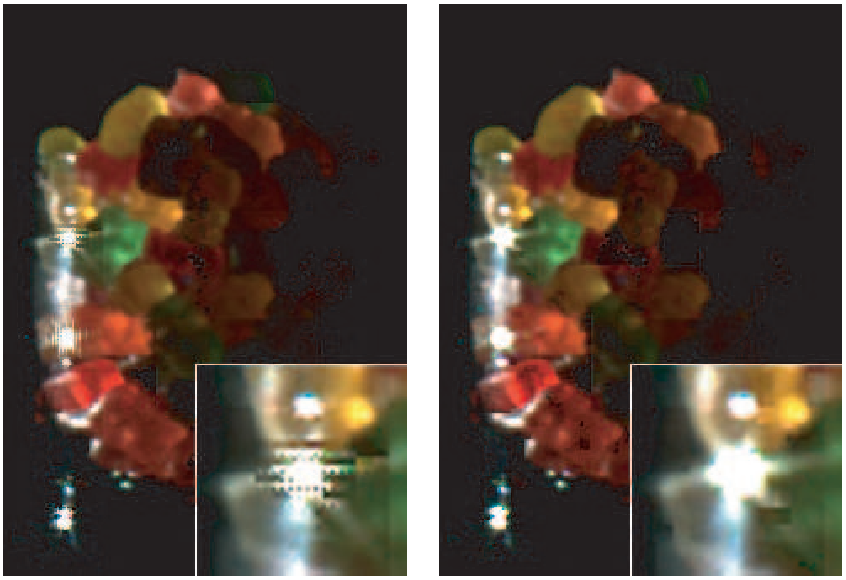
5.4 Discussion and Conclusions

In this chapter we have presented a framework for acquiring 8D reflectance fields. The method is based on the observation that reflectance fields are data-sparse. We exploit the data-sparseness to represent the transport matrix by local rank-1 approximations. The symmetry of the light transport allows us to measure these local rank-1 factorizations efficiently, as we can obtain measurements corresponding to both rows and columns of the transport matrix simultaneously. We have also introduced a new data structure called a hierarchical tensor (in Chapter 4) that can represent these local low-rank approximations efficiently. Based on these observations, we have developed a hierarchical acquisition algorithm, which looks for regions of data-sparseness in the matrix. Once a data-sparse region has been measured, we can use it to parallelize our acquisition resulting in tremendous speedup.

There are limitations in our current acquisition setup (Figure 5.4) that can corrupt our measurements. To get a coaxial setup we use a beam-splitter. Although we use a 1mm thin plate beam-splitter, it produces the slight double images inherent to plate beam-splitters. This, along with the light reflected back off the light trap, reduces the SNR in our measurements. The symmetry of our approach requires projector and camera to be pixel aligned. Any slight misalignment adds to the measurement noise. Cameras and projectors can also have different optical properties. This can introduce non-symmetries such as lens flare, resulting in artifacts in our reconstructed images (see Figure 5.11).

By way of improvements, in order to keep our implementation simple, we use a 4th order hierarchical tensor. This means that we are flattening out 2 of the 4 dimensions of the light field, thereby not exploiting the full coherency in the data. An implementation based on 8th order tensor should be able to exploit it and make the acquisition more efficient.

We introduce the hierarchical tensor as a data structure for storing reflectance fields. The concept may have implications for other high dimensional data-sparse datasets as well. The hierarchical representation also has some other benefits. It provides constant time access to the data during evaluation or rendering because of the finite depth of the hierarchical structure. At the same time, it maintains the spatial coherency in the data, making it attractive for parallel computation.



(a) (b)

Figure 5.11: **Artifacts due to non-symmetry in measurement.** The lens flare around the highlights in (b) is caused by the aperture in the camera. Since this effect does not occur in the incident illumination from the projector, the measurements are non-symmetric. Applying a strong threshold for the rank-1 test subdivides the region very finely and produces a corrupted result in the area of the highlights in (a). If the inconsistencies in measurement are stored at a higher subdivision level by choosing a looser threshold for the rank-1 test, these artifacts are less noticeable as in (b).

Chapter 6

Conclusions and Future Work

This dissertation describes a system for efficient acquisition of reflectance fields. To our knowledge this is the first system that acquires an 8D reflectance field, and has therefore advanced research in the area of reflectance field acquisition. The key challenge in acquiring reflectance fields efficiently is that a reflectance field is extremely large. Even if state of the art techniques are used for acquisition, the time required to acquire a reflectance field is still intractable. Our system reduces the storage and time requirements by at least two orders of magnitude. This reduction is enabled by leveraging two important properties of the transport matrix: symmetry and data-sparseness. Based on these properties, we have developed algorithms for acquiring the transport matrix data which allow us to represent the transport matrix using a low rank approximation. This approach results in a hierarchical adaptive acquisition algorithm.

In this work, we have demonstrated techniques for acquiring reflectance fields. However, the reflectance fields captured are sparse and incomplete. In order to smoothly change both the lighting and the viewpoint the acquired reflectance field needs to be dense. Regarding sparseness, techniques have been proposed for interpolating slices of the reflectance fields, both from the view direction [CW93] and from the illumination direction [CL05], but the problem of interpolating reflectance fields is still open. By applying similar flow based techniques to the transport matrix, one should be able to create densely sampled reflectance fields. One can also sample incoming and outgoing light fields more densely by replacing the small number of planar mirrors with an array of lenslets or mirrorlets

[UWH+03]. This will increase the number of viewpoints in the light field, but at the cost of image resolution.

Regarding completeness, if the mirror array setup is replicated to cover the whole sphere, then extrapolating from the numbers in Table 5.1, we would expect the transport matrix to be around 75GB in size and acquisition would take roughly two weeks. It should be noted that faster processing and use of an HDR video camera could significantly reduce this time significantly in the future.

We have built a prototype system for acquiring reflectance fields. The acquisition systems could be developed further with improvements in the hardware technology. Depending on the demands of the application, designs can also be adapted, e.g., a possible design could be based on Han and Perlin's [HP03] kaleidoscope.

The compression achieved by the hierarchical tensor representation depends on the coherency in the incoming and outgoing light fields. This coherency depends on the underlying parameterization used for representing the light field. The parameterization used in our work does not guarantee the best coherence. Instead, a parameterization based on surface light fields [WAA+00] would provide the best compression of the reflectance field data. However, this would require range scanning the object prior to reflectance field acquisition.

Finally, once a reflectance field has been acquired, we must be able to edit both the spatial and directional behavior of the reflectance data to change appearance. Recent research [LBAD+06] in this direction already looks promising and will further take off with easier systems for acquisition.

Appendix A

Proof of Symmetry of Light Transport

Here we prove that the pixel-to-pixel transport from the source (projector) to the destination (camera) is the same in both directions. Assume we have a flat surface patch S with arbitrary BRDF f_r viewed/illuminated by a camera/projector pair, as shown in Figure A.1. Let us assume that the projector is at point 1 with distance d_1, far enough from the surface so that the rays within a patch can be assumed to be parallel at angle θ_1. Similarly, the camera is at point 2 with distance d_2 and angle θ_2. We call the area illuminated by the single projector pixel S_1, and the region viewed by the camera pixel S_2.

If the projector produces radiant intensity I, the reflected radiance in direction θ_2 from a point in S_1 due to the projector pixel is:

$$L_s = I \frac{f_r(\theta_1 \rightarrow \theta_2) \cos\theta_1}{d_1^2} \tag{A.1}$$

The irradiance received by the camera pixel at position 2 is the integral of this radiance over the solid angle subtended by the intersection of S_1 and S_2:

$$E_{12} = \frac{L_s |S_1 \cap S_2| \cos\theta_2}{d_2^2} = \frac{I f_r(\theta_1 \rightarrow \theta_2) \cos\theta_1 |S_1 \cap S_2| \cos\theta_2}{d_1^2 d_2^2} \tag{A.2}$$

We specify the area of the intersection of S_1 and S_2 (denoted by $|S_1 \cap S_2|$) because the transfer of energy between the projector and camera pixel only happens in the region of intersection. This defines the transfer of energy between one pixel of the projector in position

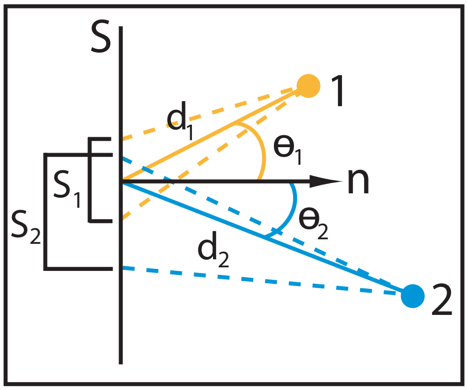


Figure A.1: **Proof of symmetry of light transport.** The transport of light from point 1 to point 2 via surface *S*.

1 and one pixel of the camera in position 2. Note that there exists a relationship between the surface area covered by a pixel and its solid angle (Ω), as given by the following equations for the projector and camera respectively:

$$\Omega_p = \frac{|S_1|\cos\theta_1}{d_1^2} \quad \text{and} \quad \Omega_c = \frac{|S_2|\cos\theta_2}{d_2^2} \tag{A.3}$$

Note that Ω_p and Ω_c are constant for our given projector and camera — they represent the solid angle for the pixel of each device. We now define a projection operator Π:

$$S_1 = \Pi_S^1\Omega_p \quad \text{and} \quad S_2 = \Pi_S^2\Omega_c \tag{A.4}$$

so that S_1 is the projection onto S from position 1 of the solid angle Ω_p, for example. We can now rewrite S_1 and S_2 as:

$$|S_1| = |\Pi_S^1\Omega_p| = \frac{\Omega_p d_1^2}{\cos\theta_1} \quad \text{and} \quad |S_2| = |\Pi_S^2\Omega_c| = \frac{\Omega_c d_2^2}{\cos\theta_2} \tag{A.5}$$

Thus Equation A.2 can be rewritten as:

$$E_{12} = \frac{I\, f_r(\theta_1 \rightarrow \theta_2)\cos\theta_1 |\Pi_S^1\Omega_p \cap \Pi_S^2\Omega_c| \cos\theta_2}{d_1^2 d_2^2} \tag{A.6}$$

We can now see that this equation will be the same if the camera is at point 1 and the projector is at point 2. The key is to remember that the virtual projector will take on the the camera parameters (in this case the Ω_c) and vice-versa. Thus the transfer of energy in the dual space is given by:

$$E_{21} = \frac{I f_r(\theta_2 \to \theta_1)\cos\theta_2 |\Pi_S^2 \Omega_c \cap \Pi_S^1 \Omega_p| \cos\theta_1}{d_2^2 d_1^2} \tag{A.7}$$

Because $f_r(\theta_1 \to \theta_2) = f_r(\theta_2 \to \theta_1)$ by Helmholtz reciprocity, we have $E_{12} = E_{21}$. This means that the pixel-to-pixel transport is equal in both directions.

Appendix B

System Overview

B.1 System Architecture

The algorithms presented in this dissertation require capturing thousands of images under patterned illumination from a projector. Therefore, we need to build a capture system which can handle the image capture process in a robust and scale able way. There are various requirements from the system.

The dynamic range of the scenes that we wish to acquire could be extremely high. In order to capture this entire range, we need to take multiple photographs of the scene under different exposure settings. Therefore, we would want to change the exposure time of the camera for every illumination pattern. This governed the choice of the camera we used. We use a camera which is computer controlled and the exposure time can be programmed for each acquired frame using the computer.

Second, we want high signal to noise ratio (SNR) from our capture system. For this purpose we use a high contrast projector. This ensures a significant difference between the lit and dark pixels of the projector resulting in high SNR. Furthermore, we subtract the image due to projector's black level to reduce the noise in the capture process.

Finally, we want the system to run completely automatically in a dark room without any interference. This is to allow for acquisitions which can take several hours to finish. Therefore, we have built a capture system in which the acquisition server (which controls the projector and the camera) can be controlled remotely. A daemon runs on the server machine which accepts requests from any client machine. A client machine can send any pattern that it wants the scene to be illuminated with. The daemon projects the pattern and captures images for various exposure settings. These images are then sent back to the requesting client machine which assembles them in to a single HDR image. The HDR combination process and other pre-processing steps are explained in the next section.

B.2 Pre-processing

The dynamic range of the scenes that we capture can be very high. This is because the light transport contains not only the high energy direct bounce effects but also very low energy secondary bounce effects. In order to capture this range completely, we take multiple images of the scene and combine them into a single high dynamic range image [DM97, RBS99]. Additionally, before combining the images for HDR, we subtract the projector's black level from our captured images, which was acquired by photographing the scene while projecting a black image. Subtracting this black-level image from every input image reduces the contribution by stray projector light to our matrix entries, as well as partially compensates for fixed-pattern noise in the camera. Upon experimentation, we found the black level to vary slightly with the number of pixels illuminated at a time, but this was not a problem for our acquisitions.

Another aspect of the measurement procedure that required care was the impact of the Bayer color mosaic in both cameras. These depend on having enough samples at the CCD to be able to properly interpolate the color components from the pixel values. We found that this introduced artifacts when illuminating the scene with individual projector pixels. A focused projector beam can illuminate very few pixels on the camera CCD, yielding errors when the samples are interpolated during demosaicing. When this happened, color contrast was significantly reduced and the images appeared darker than they should.

To remove these darkening and desaturation artifacts, we normalized the final images by forcing the individual images to add up to the floodlit image. First, all color values of the individual images were summed up so that we could see per-pixel how much of the total energy each image contained. Then the color energy of the floodlit image was distributed to the individual images in proportion to their contribution to the total energy.

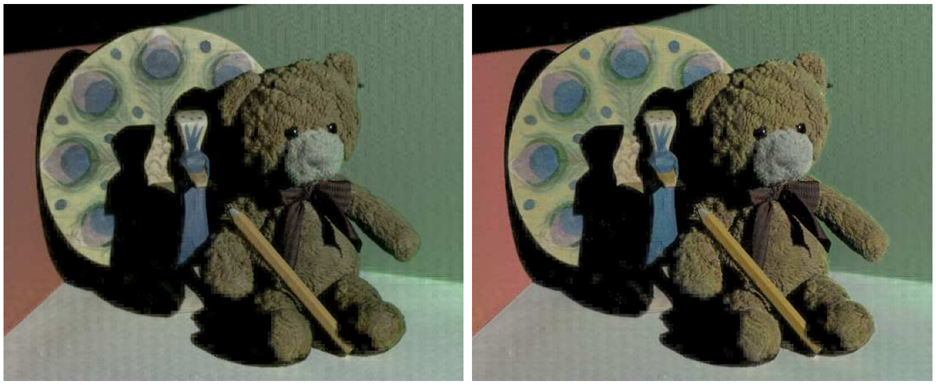


Figure B.1: **The effect of color normalization.** The sum of all images where individual pixels are illuminated (left) is dimmer and exhibits a reduced color contrast compared to the floodlit image (right). By normalization the color values of the floodlit image are distributed proportional to the observed pixel values.

This normalization improved the colors of the image, as demonstrated in Figure B.1.

Appendix C

Pseudocode

C.1 Pseudocode of Dual Photography Algorithm

```
Initialization() ;
repeat{
    // construct a conflict-free lists of blocks that can be processed in parallel
    ConstructConflictFreeLists() ;
    // illuminate scene with patterns constructed from each list and acquire with camera
    AcquireImages() ;
    // process images, store results, generate new lists of blocks for next iteration
    ProcessResults() ;
} until lowest level in hierarchy is reached
```

```
Initialization() {
      for each camera pixel $k$ {
            // initially assume every camera pixel is affected by block 0, the floodlit image
            $B_k = \{0\}$;
      }
      $C$ = empty; // initialize set of conflicts to empty
}
```

```
ConstructConflictFreeLists() {
    // form graph structure
    $B$ = Union($B_k$);                                  // nodes $B$, edges $C$
    ($L[0],\ldots,L[N-1]$) = GraphColor(graph($B$, $C$));   // N lists of nodes returned
}


AcquireImages() {
    // we now have $N$ conflict-free lists $L[]$'s
    for $i$ = 0 to $N-1$ {
        generate pattern $P[i]$ from $L[i]$;             // light pixels for all blocks in $L[i]$
        illuminate pattern $P[i]$;
        capture HDR image $I[i]$;
    }
}


ProcessResults() {
    $C$ = empty;
    for each camera pixel $k$ {
        $new\_B_k$ = empty;
        for $i$ = 0 to $N-1$ {
            // find block (if any) that affects current pixel
            current_block = intersect($B_k$, $L[i]$);      // because L[i] was conflict-free,
                                                   // this can be at most one block
            if (current_block = empty) {
                continue;                          // pixel $k$ not affected by $L[i]$
            }
            else {
                if (pixel $k$ in $I[i]$ = 0) {
                    continue;                      // no value measured, do nothing
                }
                else if (pixel $k$ in $I[i] <$ threshold) or last iteration {
                    // below the threshold so store the energy here.
                    // T() is the hierarchical representation of the matrix
                    // indexed by block in the subdivision tree and camera pixel $k$
                    $T$(current_block, $k$) = pixel $k$ in $I[i]$;
                    continue;                       // no further subdivision
                }
                else {
                    // request subdivision for this block
                    insert 4 children of current_block into $new\_B_k$;
                }
            }
        }
        // set $B_k$ for the next iteration
        $B_k$ = $new\_B_k$;
        // collect conflicts and add to $C$ for next iteration
        for each pair $(s,t)$ where $s$ and $t$ are in $B_k$ and $s \neq t$ {
            insert $(s,t)$ into $C$;                // $s$ and $t$ conflict and can't
                                                    // be measured in parallel
        }
    }
}
```

C.2 Pseudocode of Symmetric Photography Algorithm

```
**Initialization()** ;
repeat{
    // construct a conflict-free lists of blocks that can be processed in parallel
    **ConstructConflictFreeLists()** ;
    // illuminate scene with patterns constructed from each list and acquire with camera
    **AcquireImages()** ;
    // process images, store results, generate new lists of blocks for next iteration
    **ProcessResults()** ;
} until lowest level in hierarchy is reached
```

```
Initialization() {
    // capture image from floodlit illumination
    illuminate floodlit pattern $P[0]$;
    capture HDR image $I[0]$;
    // initialize the tensor root node as tensor product of $I[0]$
    $T_{00} = I[0] \times I[0]$;
    // schedule the root node, i.e. block 0 for investigation
    $old\_B = \{0\}$;
}
```

```
ConstructConflictFreeLists() {
    // form graph structure
    $B$ = empty;
    for each block $b$ in $old\_B$ {
        // request uniform subdivision for this block
        insert 4 children of $b$ into $B$;
    }
    // determine conflicts and add to $C$
    $C$ = empty;
    for each pair $(i, j)$ where $i$ and $j$ are in $B$ and $i \neq j$ {
        // $i$ and $j$ conflict and can't be measured in parallel
        if $T_{ij}$ is unknown {
            insert $(i, j)$ into $C$;
        }
    }
    graph $G = (B, C)$;                          // nodes $B$, edges $C$
    graph $G^2$ = square $G$;                    // construct the square graph of $G$
    $(L[0], \ldots, L[N-1])$ = GraphColor($G^2$);  // N lists of blocks returned
}
```

```
AcquireImages() {
    // we now have $N$ conflict-free lists $L[]$'s
    for $i = 0$ to $N - 1$ {
        generate pattern $P[i]$ from $L[i]$;            // light pixels for all blocks in $L[i]$
        illuminate pattern $P[i]$;
        capture HDR image $I[i]$;
        // compensate for known intra-block light transport T from previous level
        multiply known $T$ with $P[i]$ and subtract from $I[i]$;

        // generate estimate $I_e$ images using $T$
        $I_e[i] = T.P[i]$;
        multiply known $T$ with $P[i]$ and subtract from $I_e[i]$;
    }
}

ProcessResults() {
    $old\_B$ = empty;
    for each estimated tensor node $T_{ij}$ in $T$ {
        partition $b_i$ into 4 children $(b_{i1}, b_{i2}, b_{i3}, b_{i4})$;
        partition $b_j$ into 4 children $(b_{j1}, b_{j2}, b_{j3}, b_{j4})$;
        // calculate the rms error between measured and estimated images
        rms_error = 0;
        for $l = (i1, i2, i3, i4, j1, j2, j3, j4)$ {
            // calculate sum of squared difference (ssd)
            rms_error = rms_error + ssd(intersect($b_l$, $I[l]$), intersect($b_l$, $I_e[l]$));
        }
        // check whether rank-1 approximation valid
        if (rms_error $<$ threshold) {
                continue;                          // stop at previous level
        } else {
            for $l = (i1, i2, i3, i4)$ {
                for $m = (j1, j2, j3, j4)$ {
                    // set the rank-1 estimate for the children
                $T_{lm}$ = intersect($b_l$, $I[l]$) $\times$ intersect($b_m$, $I[m]$);
            }
        }
        // schedule the children for investigation
        $old\_B$ = union($new\_B$; ($b_{i1}, b_{i2}, b_{i3}, b_{i4}$));
        $old\_B$ = union($new\_B$; ($b_{j1}, b_{j2}, b_{j3}, b_{j4}$));
    }
  }
}
```

Bibliography

[AB91] Edward H. Adelson and James R. Bergen. The Plenoptic Function and the Elements of Early Vision. *M. Landy and J. A. Movshon, (eds) Computational Models of Visual Processing*, 1991. 9

[BGH03] Steffen Börm, Lars Graysedyck, and Wolfgang Hackbush. Introduction to Hierarchical Matrices with Applications. *Engineering Analysis with Boundary Elements*, 27(5):405–422, 2003. 16, 37

[CC70] J.D. Carroll and J. Chang. Analysis of Individual Differences in Multidimensional Scaling via an N-way Generalization of Eckart-Young Decomposition. *Psychometrika*, 35:283–319, 1970. 37

[Cha60] Subramaniam Chandrashekhar. *Radiative Transfer*. Dover Publications, New York, NY, 1960. 17

[CL05] Billy Chen and Hendrik P. A. Lensch. Light Source Interpolation for Sparsely Sampled Reflectance Fields. In *Vision Modeling and Visualization (VMV '05)*, pages 461–468, 2005. 59

[CW93] Shenchang Eric Chen and Lance Williams. View interpolation for Image Synthesis. In *SIGGRAPH '93*, pages 279–288, 1993. 59

[CZH$^+$00] Yung-Yu Chuang, Douglas E. Zongker, Joel Hindorff, Brian Curless, David H. Salesin, and Richard Szeliski. Environment Matting Extensions: Towards Higher Accuracy and Real-Time Capture. In *SIGGRAPH '00*, pages 121–130, 2000. 5

[Dan01] Kristin J. Dana. BRDF/BTF Measurement Device. In *IEEE International Conference on Computer Vision (ICCV '01)*, pages 460–466, 2001. 4

[DHS$^+$05] Frédo Durand, Nicolas Holzschuch, Cyril Soler, Eric Chan, and François X. Sillion. A Frequency Analysis of Light Transport. In *SIGGRAPH '05*, pages 1115–1126, 2005. 12

[DHT$^+$00] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the Reflectance Field of a Human Face. In *SIGGRAPH '00*, pages 145–156, 2000. 1, 5, 11

[DM97] Paul Debevec and Jitendra Malik. Recovering High Dynamic Range Radiance Maps from Photographs. In *SIGGRAPH '97*, pages 369–378, August 1997. 66

[DNGK97] Kristin J. Dana, Shree K. Nayar, Bram Van Ginneken, and Jan J. Koenderink. Reflectance and Texture of Real-World Surfaces. In *IEEE Computer Vision and Pattern Recognition (CVPR '97)*, pages 151–157, 1997. 3

[Ger36] Arun Gershun. The Light Field, Moscow. *Journal of Mathematics and Physics (1939)*, 1936. Vol. XVIII, MIT. Translated by P. Moon and G. Timoshenko. 9

[GGSC96] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The Lumigraph. In *SIGGRAPH '96*, pages 43–54, 1996. 5, 9

[GLL$^+$04] Michael Goesele, Hendrik P. A. Lensch, Jochen Lang, Christian Fuchs, and Hans-Peter Seidel. DISCO: Acquisition of Translucent Objects. In *SIGGRAPH '04*, pages 835–844, 2004. 6, 11

[GTLL06] Gaurav Garg, Eino-Ville Talvala, Marc Levoy, and Hendrik P. A. Lensch. Symmetric Photography: Exploiting Data-sparseness in Reflectance Fields. In *Eurographics Symposium on Rendering (EGSR '06)*, pages 251–262, 2006. 8

[Hac99] Wolfgang Hackbusch. A Sparse Matrix Arithmetic based on $\mathscr{H}$-Matrices. Part I: Introduction to $\mathscr{H}$-matrices. *Computing*, 62(2):89–108, 1999. 35

[Har70] Richard A. Harshman. Foundations of the PARAFAC Procedure: Model and Conditions for an Explanatory Multi-mode Factor Analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970. 38

[Har01] Frank Harary. *Graph Theory*. Narosa Publishing House, 2001. 45

[HCD01] Tim Hawkins, Jonathan Cohen, and Paul Debevec. A Photometric Approach to Digitizing Cultural Artifacts. In *Workshop on Virtual Reality, Archeology, and Cultural Heritage*, pages 333–342, 2001. 2

[Hec91] Paul S. Heckbert. *Simulating Global Illumination Using Adaptive Meshing*. PhD thesis, Department of Electrical Engineering and Computer Science, University of California, Berkeley, Berkeley, CA, 1991. 16

[HED05a] Tim Hawkins, Per Einarsson, and Paul Debevec. A Dual Light Stage. In *Eurographics Symposium on Rendering (EGSR '05)*, pages 91–98, 2005. 5

[HED05b] Tim Hawkins, Per Einarsson, and Paul Debevec. Acquisition of Time-varying Participating Media. In *SIGGRAPH '05*, pages 812–815, 2005. 4

[HP03] Jefferson Y. Han and Ken Perlin. Measuring Bidirectional Texture Reflectance with a Kaleidoscope. In *SIGGRAPH '03*, pages 741–748, 2003. 4, 48, 60

[Hun78] Gregory M. Hunter. *Efficient Computation and Data Structures for Graphics*. PhD thesis, Department of Electrical Engineering and Computer Science, Princeton Univeristy, Princeton, NJ, 1978. 37

[JMLH01] Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A Practical Model for Subsurface Light Transport. In *SIGGRAPH '01*, pages 511–518, 2001. 4, 5

[Kaj86] James T. Kajiya. The Rendering Equation. In *SIGGRAPH ’86*, pages 143–150, 1986. 13, 16

[KM99] Jan Kautz and Michael D. McCool. Interactive Rendering with Arbitrary BRDFs using Separable Approximations. In *Eurographics Workshop on Rendering (EWSR ’99)*, pages 247–260, 1999. 12

[LBAD$^+$06] Jason Lawrence, Aner Ben-Artzi, Christopher DeCoro, Wojciech Matusik, Hanspeter Pfister, Ravi Ramamoorthi, and Szymon Rusinkiewicz. Inverse Shade Trees for Non-parametric Material Representation and Editing. In *SIGGRAPH ’06*, pages 735–745, 2006. 60

[LCV$^+$04] Marc Levoy, Billy Chen, Vaibhav Vaish, Mark Horowitz, Ian McDowall, and Mark Bolas. Synthetic Aperture Confocal Imaging. In *SIGGRAPH ’04*, pages 825–834, 2004. 48

[LH96] Marc Levoy and Pat Hanrahan. Light Field Rendering. In *SIGGRAPH ’96*, pages 31–42, 1996. 2, 5, 9

[LHG$^+$05] Marc Levoy, Mark Horowitz, Gaurav Garg, Vaibhav Vaish, Eino-Ville Talvala, and Andrew Adams. Active Computational Imaging Using a Dense Array of Projectors and Cameras. *Proposal submitted to the National Science Foundation*, June 2005. 3

[LK02] Lutz Latta and Andreas Kolb. Homomorphic Factorization of BRDF-based Lighting Computation. In *SIGGRAPH ’02*, pages 509–516, 2002. 12

[LMV00] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A Multilinear Singular Value Decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000. 38

[MAA01] Michael D. McCool, Jason Ang, and Anis Ahmad. Homomorphic Factorization of BRDFs for High-Performance Rendering. In *SIGGRAPH ’01*, pages 171–178, 2001. 12

[MGW01] Tom Malzbender, Dan Gelb, and Hans Wolters. Polynomial Texture Maps. In *SIGGRAPH '01*, pages 519–528, 2001. 5

[MJC$^+$03] Stephen R. Marschner, Henrik Wann Jensen, Mike Cammarano, Steve Worley, and Pat Hanrahan. Light Scattering from Human Hair Fibers. In *SIGGRAPH '03*, pages 780–791, 2003. 4, 5

[MLP04] Wojciech Matusik, Matthew Loper, and Hanspeter Pfister. Progressively - Refined Reflectance Functions for Natural Illumination. In *Eurographics Symposium on Rendering (EGSR '04)*, pages 299–308, 2004. 6, 25

[MPDW03] Vincent Masselus, Pieter Peers, Philip Dutré, and Yves D. Willems. Relighting with 4D Incident Light Fields. In *SIGGRAPH '03*, pages 613–620, 2003. 6, 32

[MS81] Perry Moon and Domina E. Spencer. *The Photic Field*. MIT Press, Cambridge, MA, 1981. 9

[MWAM05] Stephen R. Marschner, Stephen H. Westin, Adam Arbree, and Jonathan T. Moon. Measuring and Modeling the Appearance of Finished Wood. In *SIGGRAPH '05*, pages 727–734, 2005. 4

[MWLT00] Stephen R. Marschner, Stephen H. Westin, Eric P. F. Lafortune, and Kenneth E. Torrance. Image-based Bidirectional Reflectance Distribution Function Measurement. *Applied Optics*, 39(16):2592–2600, 2000. 4

[NRH$^+$77] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. Geometrical Considerations and Nomenclature for Reflectance. *Final Report National Bureau of Standards*, October 1977. 3

[PD03] Pieter Peers and Philip Dutré. Wavelet Environment Matting. In *Eurographics Symposium on Rendering (EGSR '03)*, pages 157–166, 2003. 6, 25

[Ray00] John W. S. Rayleigh. On the Law of Reciprocity in Diffuse Reflexion. *Philosophical Magazine*, 49:324–325, 1900. 17

[RBS99] Mark A. Robertson, Sean Borman, and Robert L. Stevenson. Dynamic Range Improvement through Multiple Exposures. In *IEEE International Conference on Image Processing (ICIP '99)*, pages 159–163, October 1999. 66

[RH01] Ravi Ramamoorthi and Pat Hanrahan. A Signal-Processing Framework for Inverse Rendering. In *SIGGRAPH '01*, pages 117–128, 2001. 12

[Sam84] Hanan Samet. The Quadtree and Related Hierarchical Data Structures. *ACM Computing Surveys*, 16(2):187–260, 1984. 37

[SCG$^+$05] Pradeep Sen, Billy Chen, Gaurav Garg, Stephen R. Marschner, Mark Horowitz, Marc Levoy, and Hendrik P. A. Lensch. Dual Photography. In *SIGGRAPH '05*, pages 745–755, 2005. 8

[SMK05] Steven M. Seitz, Yasuyuki Matsushita, and Kiriakos N. Kutulakos. A Theory of Inverse Light Transport. In *IEEE International Conference on Computer Vision (ICCV '05)*, pages 1440–1447, 2005. 11, 12, 16

[SNB03] Yoav Y. Schechner, Shree K. Nayar, and Peter N. Belhumeur. A Theory of Multiplexed Illumination. In *IEEE International Conference on Computer Vision (ICCV '03)*, pages 808–815, 2003. 5

[TMRM03] Peter Tu, Paulo Mendonca, James Ross, and James Miller. Surface Registration with a Helmholtz Reciprocity Image Pair. In *IEEE Workshop on Color and Photometric Methods in Computer Vision*, 2003. 7

[Tuc66] L.R. Tucker. Some Mathematical Notes on Three-mode Factor Analysis. *Psychometrika*, 31:279–311, 1966. 38

[UWH$^+$03] Jonas Unger, Andreas Wenger, Tim Hawkins, Andrew Gardner, and Paul Debevec. Capturing and Rendering with Incident Light Fields. In *Eurographics Symposium on Rendering (EGSR '03)*, pages 141–149, 2003. 60

[Vea97] Eric Veach. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Department of Computer Science, Stanford University, Stanford, CA, 1997. 17

[vH56] Hermann von Helmholtz. *Treatise on Physiological Optics (1925)*. The Optical Society of America, Menasha, WI, 1856. Electronic edition (2001): University of Pennsylvania, http://www.psych.upenn.edu/backuslab/helmholtz. 17

[VT04] M. Alex O. Vasilescu and Demetri Terzopoulos. TensorTextures: Multilinear Image-Based Rendering. In *SIGGRAPH '04*, pages 336–342, 2004. 12, 38

[WAA$^+$00] Daniel N. Wood, Daniel I. Azuma, Ken Aldinger, Brian Curless, Tom Duchamp, David H. Salesin, and Werner Stuetzle. Surface Light Fields for 3D Photography. In *SIGGRAPH '00*, pages 287–296, 2000. 60

[War69] John E. Warnock. A Hidden-Surface Algorithm for Computer Generated Half-Tone Pictures. Technical Report TR 4-15, NTIS AD-753 671, Computer Science Department, University of Utah, Salt Lake City, UT, June 1969. 37

[War92] Gregory J. Ward. Measuring and Modeling Anisotropic Reflection. In *SIGGRAPH '92*, pages 265–272, 1992. 4

[Whi80] Turner Whitted. An Improved Illumination Model for Shaded Display. *Communications of the ACM*, 23(6):343–349, 1980. 17

[WJV$^+$05] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High Performance Imaging Using Large Camera Arrays. In *SIGGRAPH '05*, pages 765–776, 2005. 5

[WWS$^+$05] Hongcheng Wang, Qing Wu, Lin Shi, Yizhou Yu, and Narendra Ahuja. Out-of-Core Tensor Approximation of Multi-Dimensional Matrices of Visual Data. In *SIGGRAPH '05*, pages 527–535, 2005. 12, 37, 38

[ZBK02] Todd Zickler, Peter Belhumeur, and David Kriegman. Helmholtz Stereopsis: Exploiting Reciprocity for Surface Reconstruction. *International Journal of Computer Vision*, 49(2–3):215–227, 2002. 7

[ZDGG04] Dmitry Zotkin, Ramani Duraiswami, Elena Grassi, and Nail Gumerov. Fast Head Related Transfer Function Measurement via Reciprocity. Technical Report No. CS-4620, University of Maryland, 2004. 7

[ZWCS99] Douglas E. Zongker, Dawn M. Werner, Brian Curless, and David H. Salesin. Environment Matting and Compositing. In *SIGGRAPH '99*, pages 205–214, 1999. 5