Measurement, Modeling, and Synthesis of Time-Varying Appearance of Natural Phenomena

Jinwei Gu

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2010

ABSTRACT

Measurement, Modeling, and Synthesis of Time-Varying Appearance of Natural Phenomena

Jinwei Gu

Many natural phenomena evolve over time — often coupled with a change in their reflectance and geometry — and give rise to dramatic effects in their visual appearance. In computer graphics, such time-varying appearance phenomena are critical for synthesizing photo-realistic images. In computer vision, understanding the formation of time-varying appearance is important for image enhancement and photometric-based reconstruction. This thesis studies time-varying appearance for a variety of natural phenomena — opaque surfaces, transparent surfaces, and participating media — using captured data. We have two main goals: (1) to design efficient measurement methods for acquiring time-varying appearance from the real world, and (2) to build compact models for synthesizing or reversing the appearance effects in a controllable way.

We started with time-varying appearance for opaque surfaces. Using a computer-controlled dome equipped with 16 cameras and 160 light sources, we acquired the first database (with 28 samples) of time-and-space-varying reflectance, including a variety of natural processes — burning, drying, decay and corrosion. We also proposed a space time appearance factorization model which disassembles the high-dimensional appearance phenomena into components that can be independently modified and controlled for rendering.

We then focused on time-varying appearance of transparent objects. Real-world transparent objects are seldom clean — over time their surfaces will gradually be covered by a variety of contaminants, which produce the weathered appearance that is essential for photorealism. We derived a physically-based analytic reflectance model for recreating the weathered appearance in real time, and developed single-image based methods to measure contaminant texture patterns from real samples.

The understanding of the weathered appearance of transparent surfaces was also used for removing image artifacts caused by dirty camera lenses. By incorporating priors on natural images, we developed two fully-automatic methods to remove the attenuation and scattering artifacts caused by dirty camera lenses. These image enhancement methods can be used for post-processing existing photographs and videos as well as for recovering clean images for automatic imaging systems such as outdoor security cameras.

Finally, we studied time-varying appearance of volumetric phenomena, such as smoke and liquid. For generating realistic animations of such phenomena, it is critical to obtain the time-varying volume densities, which requires either intensive modeling or extremely high speed cameras and projectors. By using structured light and exploring the sparsity of such natural phenomena, we developed an acquisition system to recover the time-varying volume densities, which is about $4 \sim 6$ times more efficient than simple scanning. From the perspective of computer vision, our method provides a way to extend the applicable domain of structured light methods from 2D opaque surfaces to 3D volumes.

Table of Contents

List of Figures

List of Tables

Acknowledgments

First, I would like to express my sincere thanks to my three advisors — Peter N. Belhumeur, Shree K. Nayar, and Ravi Ramamoorthi — for their consistent support, guidance, inspiration, and advice over the last five years. It is such a treasurable and unique experience for me to be able to pursue my Ph.D. with the guidance of these three experts in computer vision, computer graphics, and physics. In addition to helping me on the technical aspects of my research, they also taught me how to select and appraise research topics that can make improvements for society, how to persevere with research while facing difficulties and failures, and how to efficiently communicate and present research work.

I would also like to thank my collaborators — Eitan Grinspun, Wojciech Matusik, and Chien-i Tu — with whom I worked on time-varying surface appearance (Chapter 3) and compressive structured light (Chapter 6). Thanks also to Wojciech Matusik with whom I had a great summer internship at Adobe working on an interesting project of printing spatially varying reflectance. Many thanks to Szymon Rusinkiewicz and Srinivasa Narasimhan for being on my thesis committee and for their comments.

My special thanks to Anne Fleming for making life at Columbia Vision and Graphics Center so pleasant and wonderful, and to my officemate, Neeraj Kumar, for many interesting and constructive discussions on various topics — research, programming, career, *etc.*I would also like to thank my colleagues Li Zhang, Kshitiz Garg, Sujit Kuthirummal, Gurunandan Krishnan, Bo Sun, Changyin Zhou, Oliver Cossairt, Alexander Berg, Dhruv Mahajan, Ryan Overbeck, Craig Donner, Kalyan Sunkavalli, Kevin Egan, Aner Ben-Artzi, Francesc Moreno-Noguer, Dmitri Bitouk, and Nandan Dixit. Thanks for all your help.

My gratitude also goes to the National Science Foundation (NSF), the Office of Naval Research (ONR), the Alfred P. Sloan Foundation, Adobe Systems Incorporated and Microsoft Research Asia for supporting my research.

Finally, I would like to present my sincerest gratitude to my family. Thanks to my mom, my wife Wei Jiang, and my brother for their love, patience, and encouragement throughout these years. This thesis is dedicated to them.

Chapter 1

Introduction

1.1 Time-Varying Appearance: What and Why

One of the fundamental problems in computer graphics, computer vision and digital photography is to understand and simulate the visual appearance of our real world. Every day we perceive the world through many different aspects of visual appearance — color and texture, shading and shadows, highlights and caustics, subsurface and volumetric scattering — all of which are caused by different interactions between light and objects.

Understanding and simulating appearance phenomena has been the topic of a great deal of research — particularly in the last 10 to 20 years. In this thesis, we refer to these research topics as *appearance modeling*. As shown in Figure 1.1, appearance modeling plays a central role in many related areas — it is critical for creating photo-realistic images for computer games and movies; it is the basis for many physics-based image restoration algorithms and photometric 3D reconstruction methods; and understanding the appearance of natural phenomena is indispensable to imaging science in applications such as remote sensing, digital photography, printing, and displays.

The appearance of an object depends on many factors — geometry and texture (spatial variation), color (spectral variation), and viewing and lighting conditions (directional variation). While extensive research has been devoted to understanding these factors, one important dimension that has long been neglected is the variation over *time*. Almost all objects evolve with time, and so does their appearance. As shown in Figure 1.2, the tem-

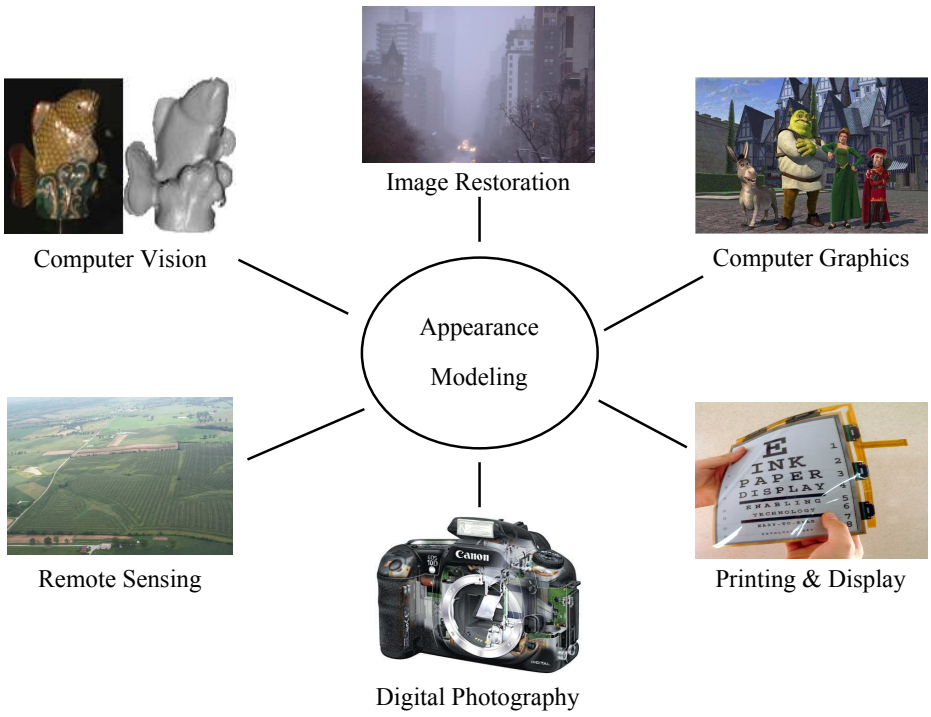


Figure 1.1: Appearance modeling plays a central role in many research areas.

poral variations in appearance are often caused by either chemical, mechanical or biological processes [Dorsey et al., 2008], such as the weathering of stones, the decaying of fruit, the accumulation of dust, and the convection of smoke. These processes form complex spatial patterns over time, often coupled with a change in reflectance or geometry, which gives rise to dramatic visual effects.

These temporal variations in appearance, referred to as *time-varying appearance*, are important for many tasks in computer graphics, computer vision, and digital photography:

- *Computer graphics*: Time-varying appearance — dust, rust, moss — implies objects undergoing a natural weathering process and therefore is important for the synthesis of photo-realistic images.

- *Computer vision*: To reconstruct the geometry of many dynamic natural phenomena, such as smoke, fire, clouds, or water, we need to understand how their appearance changes over time.

- *Digital photography*: To recover scene information from photographs that are taken

Figure 1.2: Examples of time-varying appearance of natural phenomena.

under undesirable conditions (*e.g.*, in bad weather or through defective optical imaging systems), it is important to understand how these appearance phenomena will affect image formation.

Despite its importance, time-varying appearance of natural phenomena has not yet been fully explored. First, previous studies have mostly focused on the physical simulation of specific phenomena, which is difficult to generalize and expensive to compute. Second, although some recent work has made progress towards a general solution by using captured data, due to the high dimensionality of time-varying appearance, these methods often have limited flexibility for artistic control. Finally, for complex dynamic natural phenomena, such as smoke, the acquisition can be quite challenging.

In this thesis, we address these problems with a series of studies and make the following contributions:

- acquired the first database of time-varying surface appearance covering a range of natural phenomena for both opaque and transparent surfaces.
- proposed the space-time factorization model for time-varying surface appearance, which offers intuitive control and editing abilities in rendering.
- derived a phenomenological reflectance model for rendering a weathered appearance of transparent surfaces in real time.
- developed automatic methods to reverse the weathered appearance, *i.e.*, to remove image artifacts caused by dirty camera lenses.

- presented an efficient structured light method to recover volume densities of dynamic inhomogeneous participating media.

Our definition of *time-varying appearance* consists of two categories. One category is dynamic processes that change in appearance over time, such as the drying of wood, rusting of iron, and convection of smoke, which are often represented as *videos*. The other category is appearance phenomena that are caused by aging over a period of time, such as weathered stone, decayed banana, surface scratches and cracks, which are also referred to as *weathered appearance* [Dorsey et al., 2008]. Weathered appearance is not strictly "time-varying" since it is often represented as *still images* rather than videos. Nevertheless, it is the result of time-varying processes. In order to synthesize weathered appearance, we often need to understand and simulate the time-varying processes that cause the weathering. Therefore, *time-varying appearance* and *weathered appearance* are interchangeable in this thesis.

In the remainder of this chapter, we first briefly review previous studies on time-varying appearance and discuss their limitations (Section 1.2). We then explain our approach and present an overview of the thesis (Section 1.3).

1.2 Related Work

The study of the appearance of natural phenomena has always aroused great interest among researchers — mathematicians and physicians seek to grasp the beauty of nature and find simple, elegant representations; biologists and psychologists try to understand the human visual system and perception of various appearance phenomena. In computer graphics and computer vision, in particular, our goal is to develop models that enable computer simulation of real-world phenomena and use these models to extract scene information. Over the years, many approaches have been developed in the computer graphics and computer vision communities to achieve this goal. These approaches are categorized into three groups — rule-based, physically-based, and data-driven — roughly in chronological order.

1.2.1 Rule-based: Texture Generation

In the early stage of computer graphics (1970s-1990s), much effort was devoted to the replication of the texture patterns of natural phenomena, such as marble, wood, cloud, fur, and rust. Texture patterns can be *manually* composed by artists from photographs or paintings [Demers, 2001]. While labor intensive, this method is still used in the film industry today.

Researchers have developed automatic algorithms for pattern generation based on a few characteristic *rules* of natural phenomena:

- *Self-similarity* has been observed in many places in nature, such as coastlines and snowflakes. A number of recursive algorithms have been proposed to generate fractal-like patterns, such as L-systems for plants [Prusinkiewicz and Lindenmayer, 1990] and fractal Brownian motion for landscapes and ocean waves [Barnsley et al., 1988]. By introducing the turbulence function (*i.e.*, a sum of noise components with a fractal power spectrum), Perlin [1985] proposed *procedural* methods to create realistic textures such as marble, fire, and fur [Perlin and Hoffert, 1989; Ebert et al., 1998]. Fractal methods have also been used together with heuristic rules to generate blemish patterns such as coffee stains and metal rusts [Becket and Badler, 1990].

- *Physical/chemical/biological principles* have been employed for pattern formation, where texture patterns are regarded as some features of a physical system and are formed either by the spontaneous breaking of physical equilibrium [Cross and Hohenberg, 1993] or by the diffusion of substances of some chemical [Turk, 1991; Witkin and Kass, 1991] or biological processes [Meinhardt, 1992].

- *Characteristics of surface geometry*, such as surface accessibility [Miller, 1994] and curvature [Hsu and Wong, 1995], have been used for synthesizing weathered appearance such as that resulting from rusting and dust accumulation.

These rule-based methods are intuitive and convenient to use. However, as with any procedural methods, they can only model the texture patterns for a small set of natural phenomena. Moreover, these methods are insufficient to deal with more complex appearance effects, such as angular variations (*i.e.*, lighting and view) and scattering.

1.2.2 Physically-based: Modeling and Simulation

Another approach to the study of appearance originated in physics (more precisely, radiometry and imaging science). In this approach, the primary goal is to *quantitatively* model the exact interaction between light and objects rather than the replication of texture patterns. These models are often derived from first principles, resulting in analytic formula with physically-meaningful parameters.

In computer graphics, traditional empirical models for surface reflection, such as the Phong model, have been replaced with physically-based models [Cook and Torrance, 1981; He et al., 1991; Oren and Nayar, 1994] that are derived by modeling the scattering of rough surfaces. Light transport theory in radiative transfer [Chandrasekhar, 1960; Ishimaru, 1978] has also been introduced to model subsurface scattering [Hanrahan and Krueger, 1993; Jensen et al., 2001] and volumetric scattering [Narasimhan and Nayar, 2003a] in participating media such as skin and smoke.

For time-varying appearance, researchers have developed models to simulate the underlying physical or chemical processes of specific types of dynamic natural phenomena, such as water flow [Dorsey et al., 1996], patina [Dorsey and Hanrahan, 1996], and stone corrosion [Dorsey et al., 1999; Merillou et al., 2001]. Chen et al. [2005] proposed to synthesize a weathering degree map by simulating the reflection and deposition of Gamma-tons on surfaces, which is an interesting combination of physical simulation and rule-based texture synthesis. Recent work in this area has been summarized in the surveys by Dorsey et al. [2008] and by Merillou and Ghazanfarpour [2008].

These physically-based models and simulations provide an analytic way to reproduce the appearance of natural phenomena in computer graphics. The realism of rendered images is often greater than rule-based texture synthesis. However, there are three main limitations of physically-based methods as they are used in computer graphics:

- *Limited realism.* Due to the requirement of extensive domain knowledge to understand the physical mechanisms behind the natural phenomena. The full generality of texture pattern formation remains beyond the reach of any particular mathematical model or physical simulation, especially for many complex natural phenomena.

- *Difficult to generalize and transfer.* A given model or simulation can only work for a specific phenomenon.

- *Computationally expensive.* Physical simulation often takes a long time to compute, which makes it difficult to tune parameters and control the final output.

Physically-based appearance models have also been extensively applied in computer vision and image processing. Using appearance models, researchers can develop computational algorithms or imaging methods to remove image artifacts and restore scene information from captured images taken under some undesirable conditions. The model for scattering of the atmosphere [Middleton, 1952] has been applied to enhance images taken on a foggy day and recover 3D structures [Narasimhan and Nayar, 2003b]. Polarization has been used for dehazing [Schechner et al., 2003a] and underwater imaging [Schechner and Karpel, 2005]. Physical models for rain have also been successfully used to remove rain from images and videos [Garg and Nayar, 2004]. However, the challenge of employing these models for computer vision tasks is to find compact approximations with fewer parameters to make estimation with sufficient accuracy.

1.2.3 Data-driven: Measurement and Synthesis

Data-driven methods for appearance modeling have received considerable attenuation in computer graphics since the mid-1990s. For these methods, cameras are used to acquire measurements from the real world and captured data is used to create novel images using interpolation. With the evolution of hardware (*e.g.*, cheaper image sensors and storage, more powerful graphics cards), data-driven methods introduce a standard pipeline to quickly reproduce realistic appearance in computers, even for complex natural phenomena.

The simplest and earliest data-driven application in computer graphics is probably texture synthesis, which uses a collection of texture samples to generate novel textures in a copy-and-paste manner [Efros and Freeman, 2001; Kwatra et al., 2003]. This idea was extended for temporal textures with repetition over time, such as fountains, waterfalls, and fire [Szummer and Picard, 1996; Schodl et al., 2000; Soatto et al., 2001]. However, these methods cannot be easily applied for our purpose, because for many natural phenomena

with time-varying appearance, such as drying, burning, and decaying, the visual appearance often evolves over time with no repetition.

Data-driven methods have also been applied to measure material properties, such as reflectance [Matusik et al., 2003; Lawrence et al., 2006], bi-directional texture function [Dana et al., 1999], and scattering parameters [Narasimhan et al., 2006]. In addition, researchers have proposed to directly acquire a light field [Levoy and Hanrahan, 1996; Gortler et al., 1996] or even the light transport matrix [Garg et al., 2006] for view interpolation and image relighting.

Compared to static appearance, there is much less data-driven work for time-varying appearance. Some researchers recently made a first attempt at measuring time-varying appearance. Koudelka [2004] studied time-varying textures (for a fixed lighting and view) using linear matrix factorization and extended static texture synthesis to time-varying texture using image quilting [Efros and Freeman, 2001]. Sun et al. [2006] measured time-varying surface reflectance of drying paints and household dust. Garg and Nayar [2006] acquired a database of rain streaks and the splash of water drops [Garg et al., 2007] for rendering rain. Lu et al. [2007, 2005] measured the texture patterns of several weathering processes and their corresponding context information (*e.g.*, temperature, lighting, humidity) for later synthesis.

While successful in many applications of computer graphics, data-driven methods also exhibit limitations, especially for complex appearance phenomena:

- *Expensive acquisition.* Since data-driven methods often sample appearance in a high-dimensional space, they often require a large number of synchronized light sources and cameras, such as the light stage for performance capture [Wenger et al., 2005]. Moreover, for measuring complex appearance phenomena such as smoke, special devices such as a ultra-high speed camera (5000fps) and a powerful laser are required [Hawkins et al., 2005]. For other complex phenomena such as subsurface scattering, it is still quite challenging to design and implement a measurement system [see Donner et al., 2008].

- *Cumbersome and redundant representation.* Traditionally, measurements are often represented as tables. As the dimensionality of appearance phenomena becomes

higher, this representation soon becomes cumbersome for rendering, storage, and transmission. Typical examples are the 4D light field and 8D light transport matrix, which might require tens of thousand megabytes to store for one scene. Existing compression techniques cannot completely solve the problem. While we are aware that there is much redundancy in the measured data, we have not yet found appropriate compact models for representation.

- *Limited controllability in rendering.* The tabulated representation, especially in a high-dimensional space, also imposes difficulties in rendering. While it is relatively simple to reproduce the measured data via interpolation, a more meaningful and difficult problem is to create novel appearance according to user controls. For time-varying appearance where spatial and temporal variations are coupled, factorization models are desirable.

1.3 Thesis Overview and Organization

In this thesis, our methodology for the study of time-varying appearance is data-driven. To address the limitations of the data-driven approach, we actively incorporate the physics of appearance phenomena into the data-driven pipeline, with three main objectives:

- to design *efficient acquisition methods* for measuring time-varying appearance of natural phenomena;
- to find *compact representations* for modeling complex, high-dimensional time-varying appearance;
- to develop *intuitive and controllable algorithms* for synthesizing or reversing time-varying appearance effects.

We call our methodology a *hybrid* approach. Using this hybrid approach, we systematically investigated the measurement, modeling, and synthesis of time-varying appearance for opaque surfaces (*e.g.*, wood, steel, stone), transparent surfaces (*e.g.*, glasses, windows, camera lenses), and inhomogeneous participating media (*e.g.*, smoke, mixing fluids).

In Chapter 2, we review some important concepts and definitions in appearance modeling, and place our work in the context of related research.

In Chapter 3, we study the time-varying appearance of opaque surfaces. Using a multi-light multi-camera dome, we acquired a database of 28 time-varying surfaces caused by burning, drying, decay and corrosion. We also proposed a nonlinear factorization model, which separates the spatial and the temporal variations from the captured data for artistic control.

Chapter 4 looks at the weathered appearance of transparent objects. Real-world transparent objects, such as windows, eyeglasses, camera lenses, and monitors, are seldom clean– over time their surfaces are gradually tainted by contaminants, such as dust, dirt, and lipids. Scattering from these contaminants causes visual effects that are essential for photorealism. We derived a physics-based analytic model for contaminated transparent surfaces, which enables not only real-time rendering but also the measurement of contamination patterns from a single image.

In Chapter 5, we turn the focus to how to "reverse" the appearance phenomena from captured images and videos. Specifically, we studied how to remove image artifacts caused by dust or dirt on camera lenses. These image artifacts are not only an annoyance for photographers, but also a hindrance to computer vision and digital forensics. By using multiple images and incorporating priors on natural images, we proposed several automatic methods to remove the artifacts from images and videos.

In Chapter 6, we investigate the recovery of volume densities for dynamic inhomogeneous participating media (*e.g.*, smoke, mixing fluids). Efficiency is the key for recovering such high-speed natural phenomena. We introduce a single-view method that reconstructs time-varying volume densities using structured light. The proposed method achieves higher efficiency by exploiting the sparsity in these natural phenomena.

Finally, Chapter 7 concludes the thesis with a summary of our contributions and some directions for further research.

Chapter 2

Background and Overview

In this chapter, we review some basic concepts and definitions in appearance modeling (Section 2.1). According to the dimensionality and variables involved in these concepts, we organize existing work into a taxonomy and place our thesis in the context (Section 2.2). For more details, please refer to the following relevant literature in computer graphics [Dorsey et al., 2008] and physics [Ishimaru, 1978; Chandrasekhar, 1960].

2.1 Definition and Nomenclature

Appearance of natural phenomena can be coarsely categorized into two sets. The first set is the appearance of *surfaces*, such as wood, stone, metal, or glass. When light hits the surfaces, it will be either reflected or refracted. The second set is the appearance of *participating media*, such as fog, smoke, liquid, or tissue, where small particles in the media interact with light as it passes through. For non-emissive participating media, the light will be either absorbed or scattered. For surfaces, the light interaction is localized, *i.e.*, the appearance of one point is independent with another point; for participating media, the light interactions at different points are correlated. There are separate concepts and algorithms in computer graphics for representing and reproducing the appearance for each set. That being said, the distinction between surfaces and participating media is mostly academic. In fact, surfaces are merely a limiting case of participating media.

2.1.1 Appearance of Surfaces

Before we define concepts for appearance of surfaces, let us first define concepts for describing light. What is light? Light is radiation energy. In radiometry and computer graphics, we mostly deal with geometry optics and assume light travels in straight line. A widely used concept for describing light is *radiance*, $L(\theta, \phi)$ which is defined as the radiation flux incident on a unit foreshortened area per unit solid angle:

$$L(\theta, \phi) = \frac{d\Phi}{dA \cdot \cos\theta \cdot d\omega} \qquad [\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1}]. \tag{2.1}$$

Radiance has a nice property in that it is preserved when light travels along straight lines, if there is no participating media [Brewster, 1992]. To understand the appearance of a given material, we need to know the relationship between the total incident flux and the reflected flux for each direction. This heuristics was first formally formulated by Nicodemus et al. [1977]. With a set of reasonable assumptions, they formulated that the surface appearance can be characterized with a 4D function, named *Bi-directional Reflectance Distribution Function*(BRDF), defined as the ratio between the exitant radiance and the incident irradiance:

$$f(\omega_i, \omega_o) = \frac{dL_o(\omega_o)}{L_i(\omega_i) \cdot \cos\theta_i \cdot d\omega_i} \qquad [\mathrm{sr}^{-1}]. \tag{2.2}$$

The geometry of the incident and exitant radiation can be represented by a simple polar diagram as in Figure 2.1. BRDF only characterizes reflection properties of surfaces, *i.e.*, the incident direction (θ_i, ϕ_i) and the reflected direction (θ_r, ϕ_r) are in the same hemisphere. This definition can be extended to *Bi-directional Transmittance Distribution Function*(BTDF) for situations where (θ_i, ϕ_i) and (θ_t, ϕ_t) are in different hemispheres. *Bi-directional Scattering Distribution Function* (BSDF) is a superset of BRDF and BTDF, where (θ_i, ϕ_i) and (θ_o, ϕ_o) can take values from the entire sphere instead of hemispheres.

There are three fundamental constraints for physically-valid BRDF functions:

- *Non-negativity:*

$$f(\omega_i, \omega_o) \geq 0. \tag{2.3}$$

- *Energy conservation:* this constraint states that the total reflected energy must not excess the total incident energy. Specifically, the total incident energy per unit area

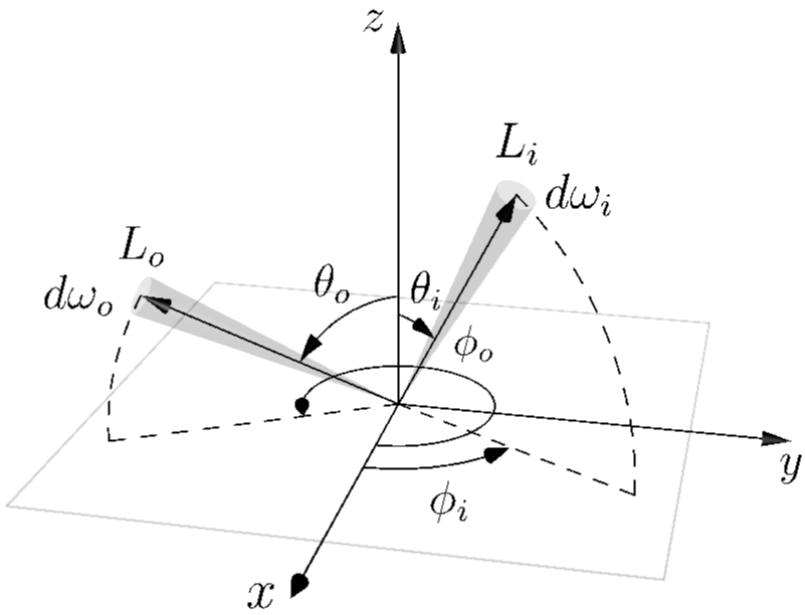


Figure 2.1: Definition of Bi-directional Reflectance Distribution Function (BRDF).

(*i.e.*, irradiance) E is

$$E = \int_{4\pi} L_i(\omega_i) \cdot \cos\theta_i \cdot d\omega_i \qquad [\mathrm{W} \cdot \mathrm{m}^{-2}], \tag{2.4}$$

and the total reflected energy per unit area (*i.e.*, total exitant radiance) M is

$$\begin{aligned} M &= \int_{4\pi} L_o(\omega_o) \cdot \cos\theta_o \cdot d\omega_o \\ &= \int_{4\pi} \left(\int_{4\pi} L_i(\omega_i) \cdot f(\omega_i, \omega_o) \cdot \cos\theta_i \cdot d\omega_i \right) \cdot \cos\theta_o \cdot d\omega_o \qquad [\mathrm{W} \cdot \mathrm{m}^{-2}]. \end{aligned} \tag{2.5}$$

For energy conservation, we must have $M \leq E$, and therefore we have the following constraint on the BRDF function:

$$\int_{4\pi} f(\omega_i, \omega_o) \cdot \cos\theta_o \cdot d\omega_o \leq 1. \tag{2.6}$$

- *Helmholtz reciprocity:* this property states the symmetry of surface reflectance. It says that if the incident direction and the reflected direction are swapped with each other, the amount of reflected light remains the same:

$$f(\omega_i, \omega_o) = f(\omega_o, \omega_i). \tag{2.7}$$

This equation assumes that both the incident direction and the reflected direction are within the same medium. For light interactions across two different media (*e.g.*, refraction), Veach [1998] extended the above equation to a generalized Helmholtz reciprocity as:

$$\frac{f(\omega_i, \omega_o)}{n_o^2} = \frac{f(\omega_o, \omega_i)}{n_i^2}, \tag{2.8}$$

where n_i is the refractive index of the medium where the incident radiance is coming from, n_o is the refractive index of the medium where the exitant radiance is going to. The generalized Helmholtz reciprocity is important to prove the BRDF reciprocity for surfaces composed of multiple layers, as we will show in Chapter 4.

As stated previously in Chapter 1, there are two ways to describe a BRDF function. One is *parametric* models where the BRDF function is some analytic formula that can be directly evaluated for any given incoming and outgoing directions. These formulas are either determined empirically (*e.g.*, Lambertian model, Phong model, Ward model [Ward, 1992], Lafortunate model [Lafortune and Torrance, 1997]) or derived physically based on first principles (*e.g.*, Cook-Torrance model [Cook and Torrance, 1981], Oren-Nayar model [Oren and Nayar, 1994], He-Torrance model [He et al., 1991]). Among the parameters commonly used for parametric BRDFs, there are three important parameters: k_d — the reflectance of the diffuse component of the BRDF; k_s — the reflectance of the specular component; and σ — surface roughness which is the mean slope of surface micro-facet.

The second way is *non-parametric* models where the BRDF functions are measured at a given set of incoming and outgoing directions and represented as tabulated data [Matusik et al., 2003]. Measured BRDFs are generally believed to have higher realism to real world materials.

BRDF is an important building block for modeling the angular dependency of surface appearance. It can be extended to include other variations, such as spatially-varying BRDF where two spatial coordinates are added to describe spatially-varying material properties. This thesis is to introduce the time dimension into BRDF for time-varying appearance. More details are discussed later in Section 2.2.

2.1.2 Appearance of Participating Media

A participating medium is one type of material that affects the transport of light through its volume. Examples include atmosphere, seawater, fog, smoke, and tissue. To describe the appearance of participating media, we need to formulate the absorption and the scattering of the interaction between light and the particles in the media. The atomic concept for characterizing these types of phenomena is scattering. Scattering has been thoroughly studied in physics [Chandrasekhar, 1960; Ishimaru, 1978] and was first introduced in computer graphics by Hanrahan and Krueger [1993].

In this thesis, we focus on non-emissive participating media. To describe the appearance of participating media, we need to introduce some definitions for scattering.

The first quantity we introduce is *density of particles*, or sometimes shortened as *density*, ρ, which represents the number of particles per unit volume. Its unit is m^{-3}.

To describe the absorption and scattering behavior for a given participating medium, we introduce *absorption cross section* σ_a and *scattering cross section* σ_s. These two quantities are characteristics of the participating medium. They have the same unit m^2. In particle physics, the concept of a cross section is used to express the likelihood of interaction between particles. When particles are thrown against a foil made of a certain substance, the cross section is a hypothetical area measure around the target particles that represents a surface. If a particle of the beam crosses this surface, there will be some kind of interaction.

There are two other related quantities: *absorption coefficient* μ_a, defined as $\mu_a = \rho \cdot \sigma_a$ and *scattering coefficient* μ_s, defined as $\mu_s = \rho \cdot \sigma_s$. The unit of both the absorption coefficient and the scattering coefficient is m^{-1}. The physical meaning of the absorption coefficient is the probability of the occurrence of absorption event when light travels one meter within the medium, and the scattering coefficient is the probability of scattering event per unit meter. A commonly-used concept in computer graphics is called *mean free path* ℓ, defined as

$$\ell = \frac{1}{\mu_a + \mu_s} \qquad [\mathrm{m}]. \tag{2.9}$$

As shown, the mean free path ℓ has a unit of length. Its physical meaning is the average distance covered by a particle between successive light interactions (*i.e.*, absorption or scattering). It is also the distance at which the power flux of the incident light decreases

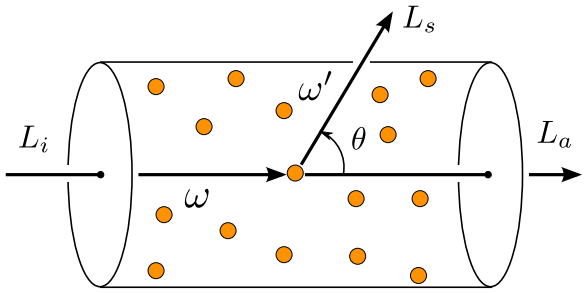


Figure 2.2: A diagram showing the scattering of participating media.

to $e^{-1} \approx 37\%$. Intuitively, the distance that we can "see through" a slab made of some participating medium is about one mean free path.

The *scattering albedo*, or sometimes shortened as *albedo* of a participating medium is defined as

$$W_0 = \frac{\sigma_s}{\sigma_s + \sigma_a}. \tag{2.10}$$

This is a dimensionless quantity. Its value is between 0 and 1. Higher values in albedo means light is more likely to scatter within the medium.

Another important dimensionless quantity is *optical thickness* τ, defined as

$$\tau = \int_0^d \rho \cdot (\sigma_a + \sigma_s) ds, \tag{2.11}$$

for a layer of participating medium with thickness of d. The optical thickness τ is exactly the same as the number of mean free paths through the layer.

With the quantities defined above, we can now formulate the absorption and scattering mathematically. As shown in Figure 2.2, consider a segment of the participating medium of length ds, for the incident light $L(\omega)$, it will attenuate the incident radiance because of absorption and out-scattering (*i.e.*, scattering in other directions), which is given by:

$$dL^-(\omega) = -\rho \cdot (\sigma_a + \sigma_s) \cdot ds \cdot L(\omega). \tag{2.12}$$

At the same time, the segment of participating medium will also gather light scattered from other directions (*i.e.*, in-scattering). To characterize in-scattering, we first introduce *phase function*. Phase function $p(\omega', \omega)$ describes the amount of light scattered from one direction ω' to another direction ω. There are a number of ways in which the phase function

may be normalized. Here we adopt the normalization in [Prahl, 1988], which requires the integral of the phase function over all angles to equal unity

$$\int_{4\pi} p(\omega', \omega) d\omega' = 1. \tag{2.13}$$

Therefore, the unit of the phase function is sr^{-1}. One commonly-used parametric model for the phase function is the Henyey-Greenstein phase function [Henyey and Greenstein, 1940]

$$p(\omega', \omega) = p_{\mathrm{HG}}(\theta) = \frac{1}{4\pi} \cdot \frac{1 - g^2}{(1 + g^2 - 2g\cos\theta)^{3/2}}, \tag{2.14}$$

where θ is the angle between the direction ω and the direction ω'. The Henyey-Greenstein phase function has one parameter g, which satisfies

$$g = \int_{4\pi} p_{\mathrm{HG}}(\theta) \cos\theta d\omega, \tag{2.15}$$

and thus g is often called the average of cosines. In this thesis, g is called the Henyey-Greenstein parameter. If $g = 0$ the scattering is isotropic, *i.e.*, $p(\omega', \omega) = \frac{1}{4\pi}$; if $g > 0$ the scattering is predominantly forward whereas if $g < 0$ the scattering is predominantly backward. Some other commonly-used phase functions can be found in [Prahl, 1988].

Given the phase function of the participating medium $p(\omega', \omega)$, the total amount of light scattered from other directions into the direction ω within the segment of the participating medium is given by

$$dL^+(\omega) = \rho \cdot \sigma_s \cdot ds \cdot \int_{4\pi} p(\omega', \omega) \cdot L(\omega') \cdot d\omega'. \tag{2.16}$$

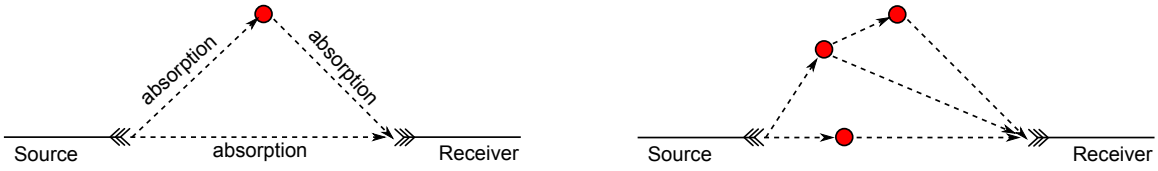
Combining Equations (2.12) and (2.16), the complete *light transport equation* is given by

$$\frac{dL(\omega)}{ds} = \frac{dL^-(\omega) + dL^+(\omega)}{ds} = -\rho \cdot (\sigma_a + \sigma_s) \cdot L(\omega) + \rho \cdot \sigma_s \int_{4\pi} p(\omega', \omega) \cdot L(\omega') d\omega', \tag{2.17}$$

for non-emissive participating media. In general, there is no analytic closed solution to this differential equation [Ishimaru, 1978; Hanrahan and Krueger, 1993]. However, analytic solutions exist for some special cases.

One important case is the solution for single scattering. As shown in Figure 2.3, single scattering means that the photons of the incident light will at most scatter once in the participating medium before they arrive at the receiver.[1] In contrast, multiple scattering

[1]This definition is the same as the "first-order multiple scattering" defined in [Ishimaru, 1978, Chapter4].



(a) Single scattering (b) Multiple scattering

Figure 2.3: A schematic diagram of single scattering and multiple scattering, excerpted from [Ishimaru, 1978, Chapter 4].

means multiple scattering events in the participating medium. Both single scattering and multiple scattering exist for almost all participating media. However, if the particles in a medium are tenuously distributed, or in other words *optically thin*, single scattering will become dominant. There is no known consensus about the definition of *optically thin*. In this thesis, we adopt the definition in [Ishimaru, 1978, Chapter 8-1] where an *optically thin* layer of participating medium (for which multiple scattering is negligible compared to single scattering) needs to satisfy: (1) the optical thickness is small, *i.e.*, $\tau \leq 0.4$; and (2) the particles are mostly absorbing, *i.e.*, $W_0 \leq 0.5$. Examples of single scattering include thin fog and haze, sea water, and thin dust layer. For other participating media with large optical thickness or high scattering, the models and conclusions derived for an optically thin layer are often the first-order approximations of the actual scattering phenomena.

Table 2.1 summarizes the symbols, the definitions and the units for the above quantities ("n/a" means dimensionless).

2.2 A Taxonomy of Appearance Modeling

Based on the physical quantities defined in the previous section, many extensions have been proposed recently for appearance modeling in computer graphics. We summarize recent work into a taxonomy, as shown in Figure 2.4. In this section, we walk through the taxonomy and establish the context for our work on time-varying appearance.

Theoretically, to fully characterize the appearance of a scene, we need to use the plenoptic function [Adelson and Bergen, 1991], which is a function of a 3D position (x, y, z), at a

Table 2.1: Commonly used symbols in appearance modeling.

Symbol	Unit	Definition
L	$\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1}$	Radiance
E	$\mathrm{W} \cdot \mathrm{m}^{-2}$	Irradiance
ω	sr	Solid angle or direction (θ, ϕ)
$f(\omega_i, \omega_o)$	sr^{-1}	BRDF
n	n/a	Refractive index
k_d	sr^{-1}	Diffuse reflectance
k_s	sr^{-1}	Specular reflectance
σ	n/a	Surface roughness
ρ	m^{-3}	Density of particles
σ_a	m^2	Absorption cross section
σ_s	m^2	Scattering cross section
μ_a	m^{-1}	Absorption coefficient, $\mu_a = \rho \cdot \sigma_a$
μ_s	m^{-1}	Scattering coefficient, $\mu_s = \rho \cdot \sigma_s$
ℓ	m	Mean free path, $\ell = 1/(\mu_a + \mu_s)$
W_0	n/a	Albedo (scattering albedo)
τ	n/a	Optical thickness (optical distance)
$p(\omega', \omega)$	sr^{-1}	Phase function

given direction (θ, ϕ), at a given wavelength λ at a given time t,

$$L(x, y, z, \theta, \phi, \lambda, t). \tag{2.18}$$

Therefore, in order to characterize the appearance of a scene under arbitrary lighting and viewing conditions, the most general scattering function has 14 dimensions, with 7 dimensions for each incoming and outgoing direction:

$$f(x_i, y_i, z_i, \theta_i, \phi_i, \lambda_i, t_i; x_o, y_o, z_o, \theta_o, \phi_o, \lambda_o, t_o). \tag{2.19}$$

This is shown as the root of the taxonomy in Figure 2.4.

2.2.1 Modeling Static Appearance

Let us first focus on existing work on static appearance. Starting from the top root level, we first omit the spectral λ and time t, and consider appearance at a given moment in RGB channels only. Since in a free space with no participating media the radiance of light is conserved along the path of light, the spatial variable can be reduced from (x, y, z) to (x, y). With these simplifications, the 14D function is simplified as an 8D scattering function:

$$f(x_i, y_i, \theta_i, \phi_i; x_o, y_o, \theta_o, \phi_o). \tag{2.20}$$

This 8D function is also known as the light transport matrix. Let us denote $\mathbf{x} = (x, y)$ and $\omega = (\theta, \phi)$, the 8D function can also be written as $f(\mathbf{x}_i, \omega_i; \mathbf{x}_o, \omega_o)$. This 8D function is usually sparse, and some recent work measured the 8D function based on the sparsity [Peers et al., 2009] and symmetry [Garg et al., 2006].

The eight variables are grouped into two spatial variables, $\mathbf{x}_i$ and $\mathbf{x}_o$ and two angular variables, ω_i and ω_o. Many previous studies in computer graphics fix one or two variables and further simplify the model. For example, if the light interaction is assumed to be localized (*i.e.*, surface appearance), we have $\mathbf{x}_i = \mathbf{x}_o = \mathbf{x}$ and the 8D transport function becomes spatially-varying BRDF [Lawrence et al., 2006] and bi-directional texture function (BTF) [Dana et al., 1999]. It can further be simplified to the 4D BRDF function as defined above. For isotropic reflection, the BRDF can be further simplified to a 3D function, $f(\theta_i, \theta_o, \phi_i - \phi_o)$.

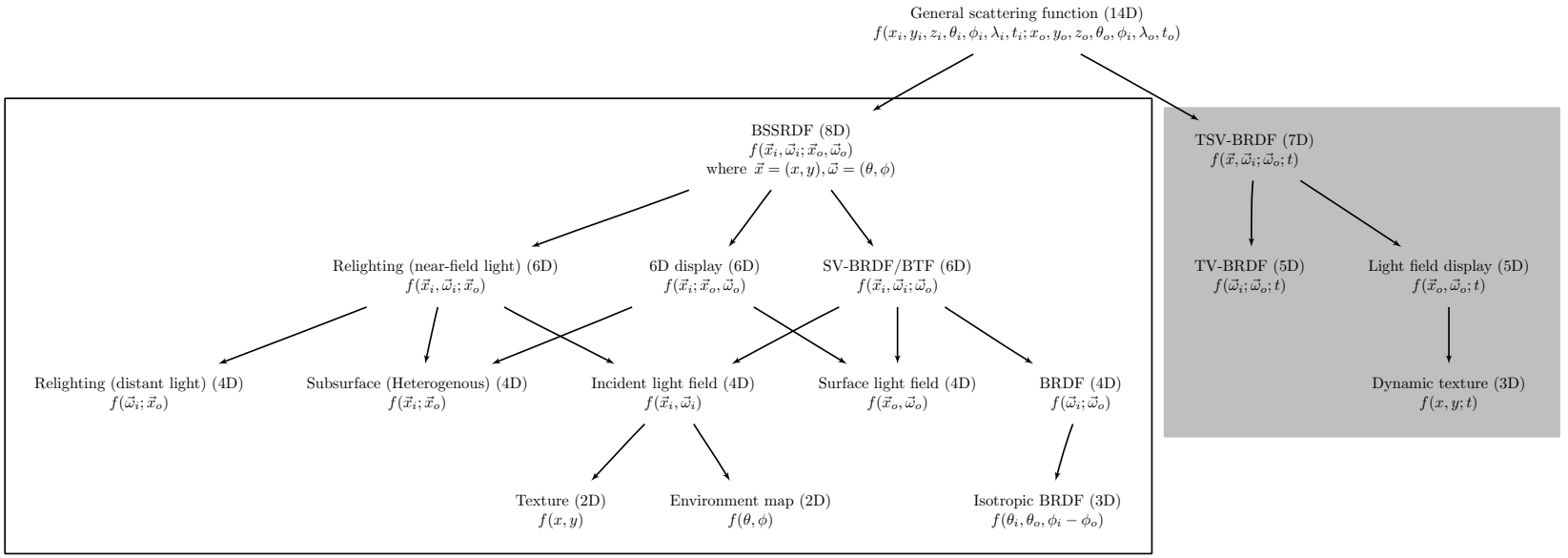


Figure 2.4: A taxonomy of appearance modeling.[a]

[a]This taxonomy is based on the course note by Szymon Rusinkiewicz and Ravi Ramamoorthi.

If we omit one of the angular variables, for example, ω_o, we will have a 6D function $f(\mathbf{x}_i, \omega_i; \mathbf{x}_o)$. This corresponds to image relighting with near-field lighting [Unger et al., 2003; Masselus et al., 2003], *i.e.*, 4D incident light field. There are several important subsets of this 6D function, as shown in Figure 2.4. For example, $f(\omega_i; \mathbf{x}_o)$ means the traditional image relighting under distant lighting [Nayar et al., 2004], *i.e.*, 2D environment map. By ignoring the two angular variables but taking into account the two spatial variables, Peers et al. [2006] proposed a 4D function $f(\mathbf{x}_i; \mathbf{x}_o)$ to describe heterogeneous subsurface scattering for materials such as wax or marble. The third 4D function is the 4D incident light field, $f(\mathbf{x}_i, \omega_i)$, which was introduced in computer graphics by Levoy and Hanrahan [1996]; Gortler et al. [1996] and is now widely used for view interpolation and other special effects. The 4D incident light field can be further simplified by fixing either the angular variable to be a 2D texture function, or the spatial variable to be a 2D environment map for distant lighting [Blinn and Newell, 1976; Miller and Hoffman, 1984]

If we omit the other angular variables, for example, ω_i, the corresponding 6D function $f(\mathbf{x}_i; \mathbf{x}_o, \omega_o)$ represents the outgoing light field caused by a directional light source. Imagine a film is illuminated with a programmable projector from behind and is viewed at different angles, its appearance can be described by this 6D function. Based on this concept, Fuchs et al. [2008] have implemented a prototype 6D display. If the incident light is fixed, this 6D function is simplified as a 4D surface light field [Wood et al., 2000] which describes the light field reflected from object surfaces.

2.2.2 Modeling Time-Varying Appearance

In theory, all the concepts for static appearance can be simply extended for time-varying appearance by adding one time variable t. Figure 2.4 lists some examples. The time and space-varying BRDF (TSV-BRDF) introduced in Chapter 3 (TSV-BRDF) is a 7D function $f(\mathbf{x}, \omega_i; \omega_o; t)$. Sun et al. [2006] studies time-varying BRDFs for paint drying and dust accumulation. In Chapter 4, we investigated the reflectance models for weathered transparent surfaces, which also belong to time-varying BRDF. Recently, some researchers developed novel display devices that can change the appearance over time. For instance, Jones et al. [2007] developed a dynamic light field display. Weyrich et al. [2009] and Matusik

et al. [2009] proposed ways to manufacture surfaces with designed surface reflectance.

As we can imagine, the temporal variations impose significant challenges for the study of these appearance phenomena. First, the acquisition must be efficient enough so that within the time of acquisition for the appearance at one moment, the appearance will not significantly altered. Second, since both the spatial and temporal variations need to be taken into account, we have to find compact models and representations for the appearance phenomena, for the purpose of either synthesizing time-varying appearance in a flexible and controllable way or reversing the weathering effect from captured images. In the remaining chapters, we will show how we address these problems for time-varying appearance on opaque surfaces, transparent surfaces, and dynamic participating media.

Chapter 3

Time-Varying Appearance for Opaque Surfaces

In this chapter, we studied the time-varying appearance of opaque surfaces for a range of natural phenomena. For opaque surfaces, their time-varying appearance is mainly due to the change in their reflectance function. We presented a complete pipeline from acquisition to modeling and to rendering.

3.1 Introduction

As discussed in Chapter 2, to fully characterize time-varying reflectance, one can imagine extending common appearance concepts, such as the BRDF or texture to include an additional time dimension. In this chapter, we extend Spatially-Varying BRDFs (SV-BRDFs) to Time and Space-Varying BRDFs (TSV-BRDFs). A general TSV-BRDF, $f(x, y, \theta_i, \phi_i, \theta_o, \phi_o, t)$, is a function of 7 dimensions — 2 each for spatial location, incident angle and outgoing directions, and 1 for time variation. For highly rough surfaces with large surface reliefs (*e.g.*, tree bark), the term Bi-Directional Texture Function (BTF) [Dana et al., 1999], and its time-varying extension TBTF are more appropriate. TBTF has the same dimensionality of TSV-BRDF but it includes inter-reflection, shadow, and masking. In this thesis, we focus primarily on flat surfaces or TSV-BRDFs.

A few previous works have studied some subsets of this 7D function. Sun et al. [2006]

studied time-varying BRDF, which is a 5D function without considering the spatial variation. Physical simulation has been employed for rendering the appearance resulting from several specific weathering processes [Dorsey et al., 1996; Dorsey and Hanrahan, 1996; Dorsey et al., 1999]. A few data-driven studies have been conducted for measuring and synthesizing time-varying textures [Koudelka, 2004; Lu et al., 2005, 2007] for a fixed lighting and view.

In contrast, our work is intended to perform a systematic data-driven study of the full 7D time-varying surface appearance, including a broad range of natural phenomena. Our work starts with acquiring data in the 7D space. Our goal is to find a compact representation for the acquired high dimensional data, which not only reproduces the measured data, but also gives the flexibility to create novel renderings. Our data-driven approach generalizes and complements physical simulation of specific phenomena, much as static data-driven reflectance models [Matusik et al., 2003; Dana et al., 1999] complement and extend specific physically-based analytic BRDFs.

3.2 Acquisition and Database

To develop an easy yet uniform way to acquire the full 7D TSV-BRDFs for a range of dynamic natural phenomena, we must consider the following two challenges:

- First, dynamic natural phenomena can have very different speeds in evolution, and we need proper control for each one. While some natural processes like drying occur over fairly short time scales (a few minutes), many others occur over a considerable duration under normal circumstances (several hours to days for decay of fruit skins, or a few months for corrosion of metals). In the case of burning and charring, we used a heat gun to carefully control the process. At each time interval, we uniformly heated the sample for a fixed duration of time (typically 30 seconds). For metal corrosion, we decided to speed up the process using specially prepared solutions [Hughes and Rowe, 1991]. We sprayed a chemical solution before each measurement and waited a few hours. Decay of organic samples takes several hours, and is fairly difficult to speed up — we decided to measure these processes without alteration.

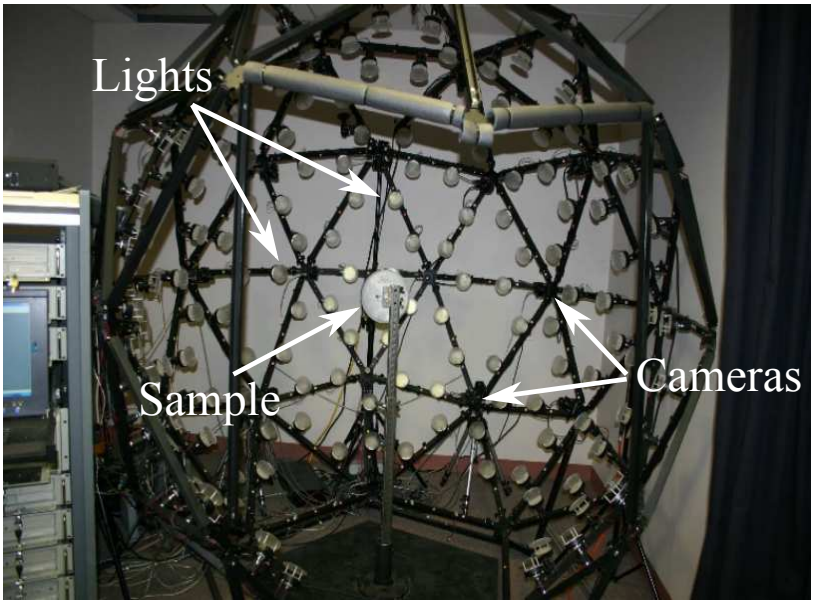


Figure 3.1: A photograph of the multi-light multi-camera dome used for the acquisition of the time-varying surface appearance database.

- A second difficulty is to design and build a measurement system that meets the following resolution requirements: (1) Spatial resolution — acquired samples should have at least 400×400 pixels. This means that the size of the imaging sensor has to be significantly larger (*e.g.*, at least one mega pixel) — it is usually difficult fit a sample perfectly into an image from different viewing directions. (2) Dynamic range — many of the processes (*e.g.*, drying or rusting) involve significant changes in specularity. (3) Light and view direction resolution — the sampling of the light and view directions should be sufficiently high to capture specular materials. (4) Temporal resolution — a complete acquisition at each time step, involving images with multiple lights, views and exposure settings needs to be conducted in a few seconds to avoid the sample changing significantly over this time. This rules out gantry-based systems, such as the spherical gantries at Stanford [Levoy and Hanrahan, 1996] and Cornell [Marschner et al., 2005] that typically can acquire very high quality and precisely calibrated images but take a few seconds to acquire even a single image.

We decided to use a multi-light multi-camera dome as shown in Figure 3.1. The dome skeleton is based on an icosahedron. We used 16 Basler cameras (resolution 1300×1030 pixels) placed on the icosahedron vertexes and 150 white LED light sources spaced evenly on the edges (approximately 80 of these lights lie in the visible hemisphere with respect to the flat sample, and therefore give useful images). This design is similar to the light stage [Debevec et al., 2002], but includes multiple cameras as well. The cameras and light sources were synchronized using a custom-built controller. The cameras were geometrically calibrated by moving a small LED diode in the working volume and detecting its 2D location in all cameras. A bundle adjustment was performed to obtain the precise geometric location and projection matrices for all cameras. We also performed a photometric calibration of the system, by capturing images of a perfectly white diffuse standard (spectralon) from all camera viewpoints under all light combinations. To obtain normalized BRDF values for each surface point, we divided each captured image by the corresponding observation of the white diffuse standard.

For acquisition, we placed a prepared sample (as close as possible to planar surfaces) in the center of the dome. At each time step, we captured a high dynamic range data set — we took images at two different exposures (typically 2 and 82 msec) for each light-camera pair — in total 4,800 photographs captured in 22 seconds. It took about 90 seconds to save the data to the hard disk. Therefore, the minimum time between two consecutive measurements was about 2 minutes. We typically captured 30 time frames for each sample. Once a complete time-varying appearance data set was captured, we re-sampled the data on a uniform grid (typically 400×400 pixels) for each light and view direction. Some of the measured data are listed in Figure 3.3, showing time variations for a single light source and view.

We have acquired 28 samples in total, covering five categories — burning and charring (wood, waffles), drying of smooth surfaces (wood, fabric), drying of rough surfaces (rock, granite), corrosion and rusting (steel, copper), and decay and ripening (apples, banana). Table 3.1 presents a list of the acquired samples and the corresponding sampling time. Figure 3.2 shows some examples of the database for their variations across time for a single light source and view. Figure 3.3 shows the acquired images for two separate views

Table 3.1: Description of the TSV-BRDF database, grouped into categories.

Type	Sample	Image Size	Time Frames	Time Interval (m)
Burning	Charred Wood 1 (#9)	512	11	2.1
	Charred Wood 2 (#40)	480	31	9.9
	Waffle Toasting (#41)	220	30	6.3
	Bread Toasting (#42)	260	30	5.9
Drying (Smooth Surfaces)	Light Wood 1 (#29)	420	14	3.1
	Light Wood 2 (#23)	300	34	2.3
	Orange Cloth (#22)	512	33	4.9
	Cotton Cloth (#30)	420	30	11.3
	Pattern Cloth (#34)	420	32	4.8
	White Felt (#24)	300	28	4.4
Drying (Rough Surfaces)	Dark Wood 1 (#21)	300	23	3.8
	Dark Wood 2 (#36)	280	34	4.0
	Paper Towel (#25)	300	32	7.0
	Brick (#27)	260	32	22.1
	Rock 1 (#19)	512	12	2.1
	Rock 2 (#20)	360	11	2.0
	Cardboard (#26)	420	29	7.0
	Granite (#37)	340	27	2.6
	Tree Bark (#38)	180	11	3.4
Corrosion	Rusting Steel 1 (#32)	300	30	7.3
	Rusting Steel 2 (#43)	460	35	10.8
	Cast Iron Rusting (#45)	460	30	13.9
	Copper with Patina (#44)	460	34	31.6
Decaying	Apple with Core (#28)	180	33	9.6
	Apple Slice (#35)	156	13	3.0
	Banana (#31)	220	33	11.3
	Potato (#39)	200	31	8.3
	Leaf under Humid Heat (#33)	300	30	12.6

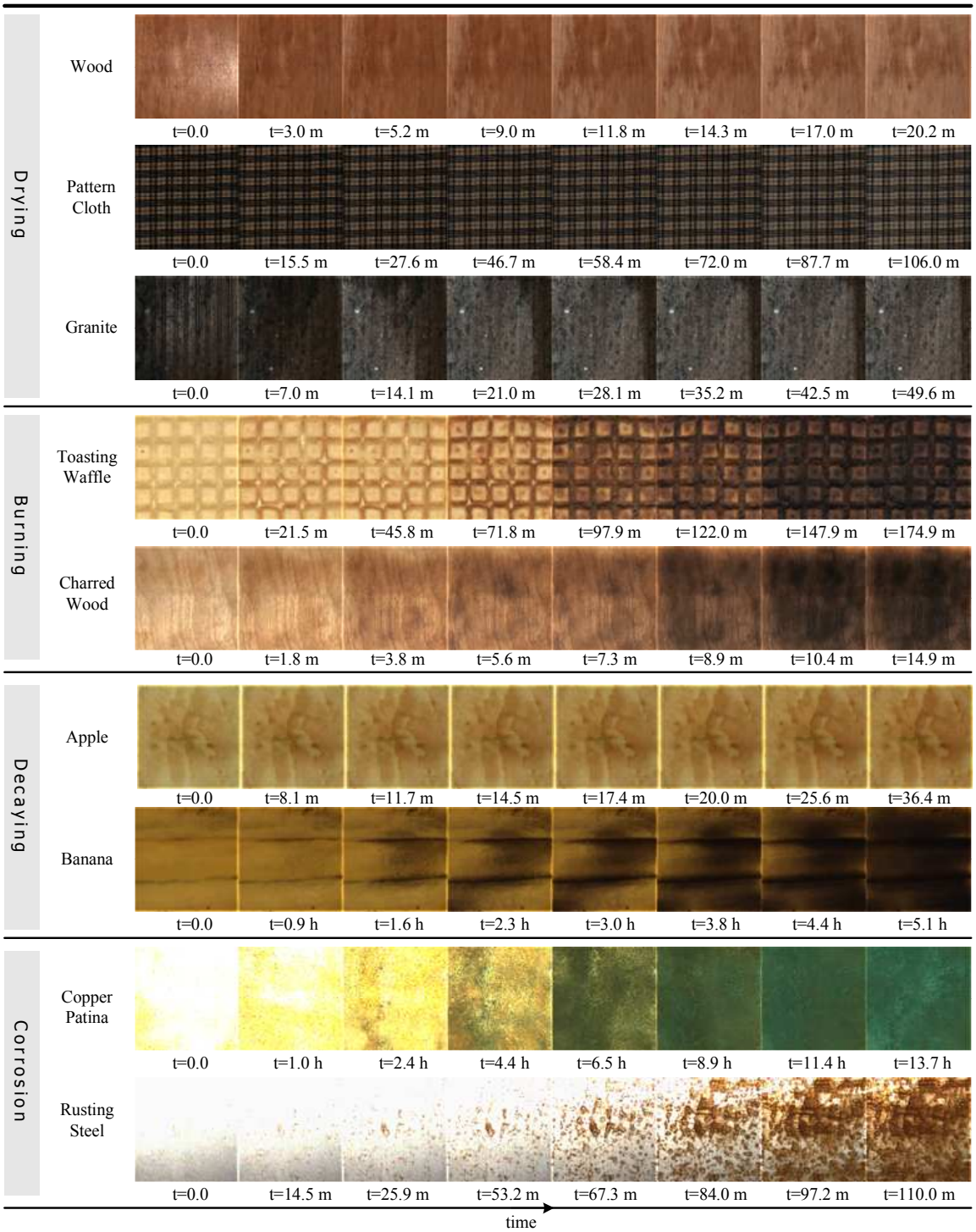


Figure 3.2: Some examples of the TSV-BRDF database, shown here with variation across time (in minutes m or hours h) for a single light source and view. We acquired images from 1280 light and view directions at each time step — some of these images are shown for one of the samples in Figure 3.3.

and time instances of the wood-drying sample. This database has been released online at `http://www.cs.columbia.edu/CAVE`.

3.3 TSV-BRDF: Non-parametric vs. Parametric

Once we acquired the database of time-varying surface appearance for certain lighting and view directions, the first desirable control is to smoothly vary lighting and view directions. Initially, we attempted to take a straightforward non-parametric approach to represent the BRDF at every point directly by the acquired raw data. For rendering we used the algorithm in [Vlasic et al., 2003] and did barycentric interpolation twice, once over view and then over lighting. However, as shown in Figure 3.4, because of limited sampling of the lighting and view directions, barycentric interpolation produces artifacts, especially at the boundary of shadow and the highlight regions.

Fully capturing the precise shape of sharp specularity at every point requires a large number of light sources and views that is not affordable for a database of time-varying surface appearance. Instead, we took the parametric approach, *i.e.*, fitting parametric reflectance models to each spatial location at each time step for all the samples, including specular lobes. While we might lose some subtleties by using a parametric model, this provides a practical solution for handling insufficient samples in a high-dimensional space and a compact representation that is directly usable for real-time or off-line rendering, as well as in further factorization and mathematical models of time-varying appearance. We used a simple combination of diffuse Lambertian and simplified Torrance-Sparrow reflectance, with the BRDF given by

$$f(\omega_i, \omega_o; x, y, t) = k_d(x, y, t) + \frac{k_s(x, y, t)}{4(\omega_i \cdot N)(\omega_o \cdot N)} \exp\left[-\left(\frac{\cos^{-1}(\omega_h \cdot N)}{\sigma(x, y, t)}\right)^2\right], \qquad (3.1)$$

where ω_i and ω_o are incident and outgoing directions, N is the surface normal and ω_h is the half-angle vector. The BRDF parameters are the diffuse intensity k_d, the specular intensity k_s and the surface roughness σ. Since k_d is an RGB color, we have a total of five parameters for each spatial location (x, y) and time t. We use the notation $p(x, y, t)$ for the parametric

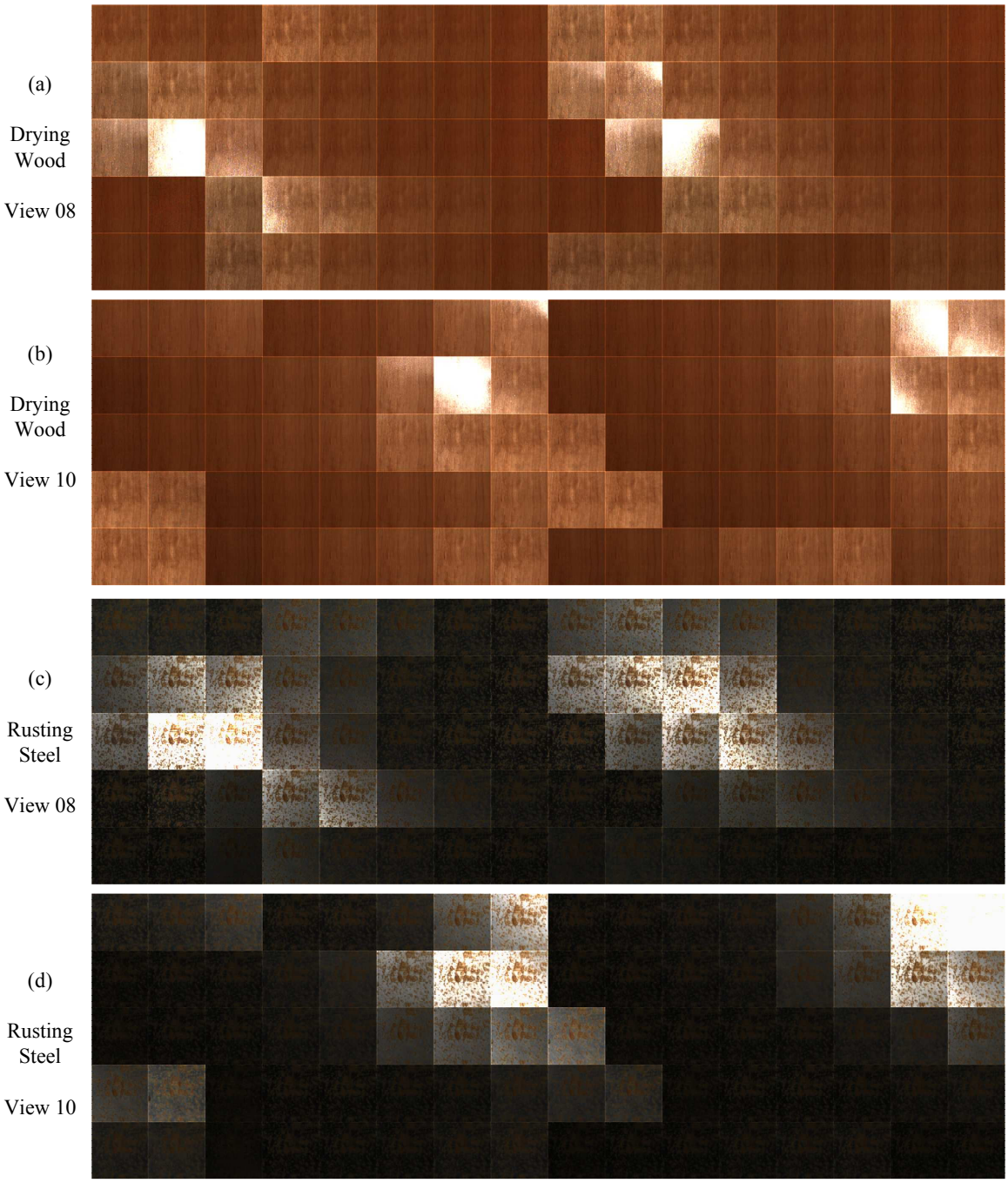
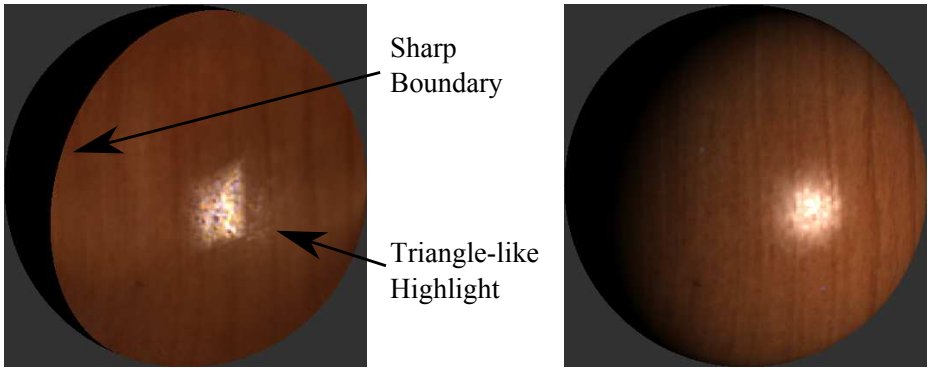


Figure 3.3: Acquired images of wood drying and steel rusting. We show two separate views/time instances, and 80 frontal lighting directions (*i.e.*, $\omega_i \cdot N > 0$) for the two time-varying samples.



(a) Barycentric Interpolation (b) Analytic Model

Figure 3.4: Rendering of TSV-BRDF with (a) barycentric interpolation or (b) parametric BRDF models. Compared with the barycentric interpolation, rendering with fitted parametric BRDF models generates more natural and accurate results, preserving the fine details of the wood grain, while eliminating artifacts in the highlights and boundaries.

fits to the TSV-BRDF

$$p(x,y,t) = \begin{pmatrix} k_d(x,y,t) \\ k_s(x,y,t) \\ \sigma(x,y,t) \end{pmatrix},$$

which is a vector of the five parameters. The angular dependence is implicit in the form of the specular term controlled by k_s and σ. We note that the BRDF model used to fit the raw data is independent of the STAF model in the remaining sections. Other kinds of parametric BRDF models could also be used if appropriate.

The diffuse and specular parameters are estimated separately in two steps, since for some materials there are only a few samples in the specular lobe. To fit the diffuse color k_d, we consider a frontal view with the highest resolution. At each spatial location, we average over only those light source directions where specular highlights are not present — conservatively, we require the light source and the reflected view directions to be separated by at least 30° — it works well for most of the samples in the database. We consider each time frame separately for the fitting.

Once k_d was estimated, we fitted the specular intensity k_s and the surface roughness σ by minimizing the ℓ^2-norm of the difference between the measured data and the model-based

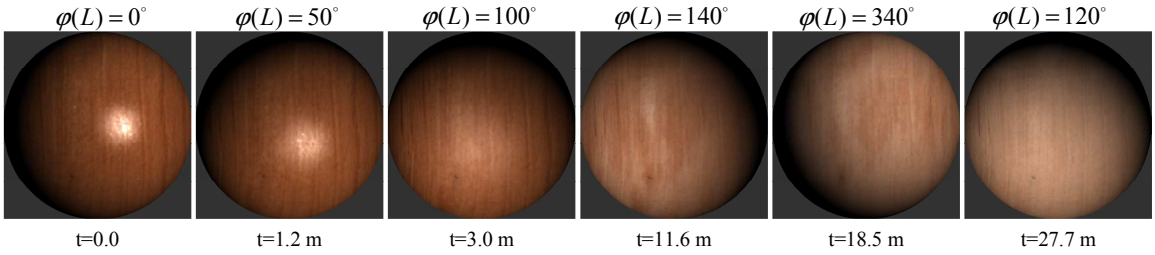


Figure 3.5: This example demonstrates a simple rendering with the TSV-BRDF database, which enables us to render with simultaneous changes in lighting and evolution with time. Note the diffuse spatial drying patterns, and the early dimming and diffusing of specularity. The elevation angle of the light with respect to the center is fixed at $\theta(L) = 30°$, while the azimuthal lighting angle varies as the sample dries.

BRDF at all sampled lighting and view directions for each point:

$$\min_{k_s,\sigma} \sum_{(\omega_i,\omega_o)} \|f_0(\omega_i,\omega_o) - f(\omega_i,\omega_o;k_d,k_s,\sigma)\|^2, \tag{3.2}$$

where $f_0(\omega_i,\omega_o)$ is the measured BRDF. For the optimization, we found it more robust to estimate σ with a linear search. For a given σ, we solve a linear system for k_s. The optimal σ and k_s were found with the minimum error.

As seen in Figures 3.4 and 3.5, we captured the important qualitative aspects of the specularity, without artifacts. For quantitative results, Figure 3.6 shows the means and the standard deviations of the Normalized Root Mean Square Error (NRMSE) of BRDF fitting for each of the 28 samples across all spatial locations and all time frames. As shown in the bar graph, for most samples in the database, the averaged NRMSE is less than 7%, which demonstrates the accuracy of the BRDF fitting. Appendix D lists the averaged NRMSE at different time frames for each of the 28 samples.

This parametric representation also reduces the size of the database. The average size of the raw data of one sample is about 30 GB. Fitting BRDF models for each time step will reduce the size to about 80 MB. With this parametric representation, our database of TSV-BRDFs can be texture-mapped onto arbitrary 3D objects and used directly for rendering with general lighting direction, viewing angle, and time variation. Indeed, our use of standard parametric models allows time-varying effects to be easily incorporated

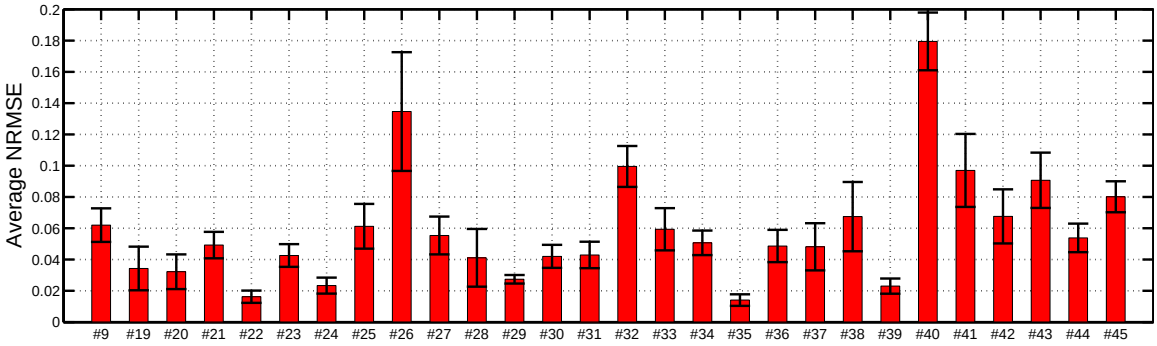


Figure 3.6: Average NRMSE for fitting measured TSV-BRDF data to the parametric model. The bar graph shows the means and the standard deviations of the NRMSE of BRDF fitting for each of the 28 samples across all spatial locations and time frames. The plot shows that most samples have the averaged NRMSE less than 7%.

in almost any interactive or off-line rendering system. As one example, Figure 3.5 shows drying wood texture-mapped onto a sphere. We show a sequence of frames, where we simultaneously change the lighting, and evolve the sample over time. Note the spatial drying patterns, as well as BRDF changes, wherein the initial sharp specularity quickly diffuses and dims over time.

3.4 Space-Time Appearance Factorization

While the fitted TSV-BRDFs are easy to use for rendering, they can only *reproduce* the captured data. There are many rendering applications where the user desires more controls to create novel renderings. For example, one may want to control the spatial drying patterns on a wooden floor to dry slower near recent wet footprints. Or one may want to remove the spatial drying patterns altogether allowing the surface to dry uniformly. The user might also want to change the underlying spatial texture, to create a different appearance for the wood grain. In order to obtain such flexibility in rendering, we hope to find a more compact model that can intuitively factorize the high-dimensional time-varying appearance into lower-dimensional, independent components for editing and controls.

3.4.1 The Model and Its Implications

The model we found is called Space-Time Appearance Factorization (STAF) model, which separates the spatial and temporal variations from TSV-BRDFs. In this section, we introduce the STAF model, explain the physical meaning of its components, and show how to estimate it from the TSV-BRDF. We also present results indicating its accuracy for the large variety of time-varying phenomena in our database.

The STAF model is based on the observation that many physical processes have an overall temporal behavior associated with them. For example, drying wood often gets lighter over time. However, different points can dry at different rates and with different offsets. For example, the points in a puddle start out wetter than others. Intuitively, we seek to *align* the time variations for different spatial locations, represented as $p(x, y, t)$, by deforming a single *temporal characteristic curve* $\phi(t)$ according to spatially-varying parameters for *rate* $R(x, y)$ and *offset* $O(x, y)$. The proposed model is formulated as

$$\begin{aligned} p(x, y, t) &= A(x, y)\phi(t') + D(x, y) \\ t' &= R(x, y)t - O(x, y). \end{aligned} \tag{3.3}$$

As mentioned earlier, $p(x, y, t)$ is a vector of five parameters (k_d, k_s, σ), and thus this equation is applied to each of the five parameters separately. This model is a data-driven nonlinear model, in which the factors A, D, R, O and ϕ are estimated directly from the acquired data. We now describe the meanings of the various terms.

$\phi(t')$ — **Temporal Characteristic Curve:** The overall time variation characteristic of the physical process is captured by the curve $\phi(t')$. The form of $\phi(t')$ can vary with the specific phenomenon. It could be exponential for some decays, sigmoid for drying and burning, a more complex polynomial form for rusting, or any other type of curve. Since our representation is data-driven, we can handle a variety of effects. In fact, one of the advantages of this data-driven model is that it can extract the overall temporal trends for different types of natural phenomena from captured data without knowing the physical mechanisms. We discuss this point in detail below.

$R(x, y)$ **and** $O(x, y)$ — **Spatial Rate and Offset:** Different spatial locations evolve differently. For example, regions close to a light source may dry faster. Some areas may also start out drier than others. We capture these effects with spatially varying rate $R(x, y)$ and offset $O(x, y)$ parameters. If $R(x, y)$ is large, the rate of change will be rapid. If $O(x, y)$ is positive, the point will start from an earlier state. The *effective time* t' for a given point is given by $t' = R(x, y)t - O(x, y)$, where we refer to t as the *global time*.

$A(x, y)$ **and** $D(x, y)$ — **Static SV-BRDFs:** $A(x, y)$ and $D(x, y)$ are static over time. The diffuse components correspond to standard spatial textures (*e.g.*, wood-grain) that remain fixed throughout the time variation. Consider the special case when $R(x, y) = 1$ and $O(x, y) = 0$ so all points evolve in the same way. Equation (3.3) becomes $A(x, y)\phi(t) + D(x, y)$. In this case, we simply interpolate from one texture (or more generally, SV-BRDF) to another with a weight $\phi(t)$. Therefore, $A(x, y)\phi(0) + D(x, y)$ can be interpreted as the "initial" appearance with $A(x, y)\phi(1) + D(x, y)$ being the "final" appearance.

We later found that the STAF model actually belongs to a family of the so-called *shape invariant model* which was first proposed by Lawton et al. [1972] and has been extensively studied in statistics for self-modeling nonlinear regression. The shape invariant model and its variations have also been successfully applied for analyzing medical data such as human height growth [Stützle et al., 1980; Kneip and Engel, 1995; Beath, 2007] and circadian rhythms [Wang et al., 2003]. A more general statistical method for analyzing the similarity between measurements is called *dynamic time warping*, which was originally proposed by Sakoe and Chiba [1978] to find the transformation between two speech signals. While the dynamic time warping was originally designed for discrete signals (which was solved via dynamic programming), it was extended by Wang and Gasser [1997, 1999] for continuous functions, which is solved via nonlinear optimization.

In our work, the goal is to align time-varying appearance curves (*i.e.*, time-varying BRDFs) at different spatial locations. The STAF model provides an intuitive way to analyze the statistical correlations between the temporal variations at different spatial locations. Its advantages are summarized as follows.

Separating Spatial and Temporal Variation: The STAF model in Equation (3.3) has factored spatial and temporal variation in a compact representation. The quantities (A,D,R,O) depend only on spatial location (x, y), and the temporal characteristic curve $\phi(t')$ only relates to time. Unlike linear decompositions, the STAF model is *non-linear*, because $\phi(t')$ is stretched and offset by the spatial rate and offset $R(x, y)$ and $O(x, y)$. A similar separation of spatial and temporal effects could not be accurately achieved by linear data reduction methods like PCA, nor would the terms in a linear model correspond to physically intuitive and editable factors.

Statistical Learning of the Overall Temporal Trends: One of the main obstacles in physical simulation is that for many complex natural phenomena, it is difficult to understand the underneath physical or chemical mechanisms and to design the simulation properly. The STAF model provides an intuitive way to *learn* the overall temporal trends (*i.e.*, the overall temporal curve $\phi(t)$) from the measured samples statistically. These learned trends can be used either to guide the design of the physical simulation or as an experimental verification for the derivations from first principles. For example, as shown in Figure 3.7, for the wood-drying process, the estimated overall temporal curve $\phi(t)$ is similar to a sigmoid curve. This is consistent both with the measured temporal appearance for the drying of stone [Lu et al., 2005] and the analytic derivation of the drying of wood [Jankowsky and Dos Santos, 2004].

Extrapolation: Another interesting aspect of the STAF model is its power to extrapolate beyond the time range of the acquired sequence. Let us normalize the global time t in the range of $[0 \dots 1]$. Consider the effective time $t' = R(x, y)t - O(x, y)$, which lies in the range $J(x, y) = [-O(x, y), R(x, y) - O(x, y)]$. If either $R(x, y)$ and/or $O(x, y)$ is large, this range can extend considerably beyond the global $[0 \dots 1]$ time. The valid domain of effective times for the full curve $\phi(t')$ is now

$$J = \bigcup_{(x,y)} J(x, y) = \left[\min_{(x,y)} \left(-O(x, y)\right), \max_{(x,y)} \left(R(x, y) - O(x, y)\right) \right], \tag{3.4}$$

which considers the minimum and maximum effective time t' over all points (x, y). By definition, the overall range of J is a superset of that for each point, enabling individual

pixels to be backed up or extended beyond the sequence captured, allowing time extrapolation. Intuitively, this is plausible because early-starting points could provide information for other similar points which start later by some offset.

3.4.2 An Iterative Factorization Algorithm

We use a simple iterative optimization to estimate the factors in Equation (3.3). Each iteration consists of two steps. In the first step, we fix the spatial parameters $A(x, y)$, $D(x, y)$, $R(x, y)$ and $O(x, y)$ and update $\phi(t')$ according to Equation (3.3). In the second step, we fix $\phi(t')$ and update $A(x, y)$, $D(x, y)$, $R(x, y)$ and $O(x, y)$ — this step requires non-linear optimization, but can be carried out separately for each spatial location (x, y). The details of the iterative algorithm are given below.

Initialization: Our inputs are discrete fits of the parameters $p(x, y, t)$ at pixels i and times j, which we denote $p_i(t_j)$. The pixel i corresponds to the spatial location (x_i, y_i). For each pixel, we construct a continuous curve $p_i(t)$ using the kernel-based method [Gasser et al., 1985]. Splines or local polynomial fitting can also be used. We are now ready to begin our iterative optimization. To initialize, we set $A_i = 1$, $D_i = 0$ and $R_i = 1$, $O_i = 0$ for all pixels.

Step 1 — Estimating $\phi(t')$: In each iteration, we first fix the spatial parameters A, D, R, O to estimate $\phi(t')$. We re-arrange Equation (3.3), writing $t = (t' + O_i)/R_i$, and we have

$$A_i\phi(t') + D_i = p_i\left(\frac{t' + O_i}{R_i}\right)$$
$$\phi(t') = \frac{p_i\left((t' + O_i)/R_i\right) - D_i}{A_i}, \tag{3.5}$$

for pixel i where $t' \in J_i$, where J_i is the range of the effective time for pixel i defined as $J_i = [-O_i, R_i - O_i]$. For robustness, and to consider the full effective time range, the estimate of $\phi(t')$ is computed as the average of multiple points,

$$\phi(t') = \frac{\sum_{i:t'\in J_i} p_i((t' + O_i)/R_i) - \sum_{i:t'\in J_i} D_i}{\sum_{i:t'\in J_i} A_i}. \tag{3.6}$$

Step 2 — Estimating A, D, R, O: We now fix the overall time curve $\phi(t')$ and estimate the four spatial parameters for each pixel. We formulate it as the following optimization problem for pixel i

$$\min \sum_{t_j \in J} \|p_i(t_j) - A_i \phi\left(R_i t_j - O_i\right) - D_i\|^2, \tag{3.7}$$

where J is the support of the curve $\phi(t')$. Note that this expression uses the discrete observations $p_i(t_j)$, finding spatial parameters that best match our input data. This is a nonlinear least-squares optimization problem, and we use the *lsqnonlin* function in Matlab, with Levenberg-Marquardt minimization.

Step 3 — Normalization: Before we start a new iteration, we need to normalize the estimates. This is because the STAF model involves a product, and requires normalization of the factors for uniqueness. We perform the following normalization

$$\begin{aligned} \frac{1}{N}\sum_{i=1}^{N} A_i &= 1, & \frac{1}{N}\sum_{i=1}^{N} D_i &= 0, \\ \frac{1}{N}\sum_{i=1}^{N} R_i &= 1, & \frac{1}{N}\sum_{i=1}^{N} O_i &= 0, \end{aligned} \tag{3.8}$$

where N is the total number of pixels. This simply says that the overall spatial textures need to be normalized so that the average rate is 1, while the average offset is 0. Suppose the the output of Step 2 (without normalization) are $\hat{A}_i$, $\hat{D}_i$, $\hat{R}_i$ and $\hat{O}_i$. The normalization is performed as follows:

$$\begin{aligned} A_i &= \frac{N}{\sum_{j=1}^{N} \hat{A}_j}\hat{A}_i, & D_i &= \hat{D}_i - \frac{A_i}{N}\sum_{j=1}^{N} \hat{D}_j, \\ R_i &= \frac{N}{\sum_{j=1}^{N} \hat{R}_j}\hat{R}_i, & O_i &= \hat{O}_i - \frac{R_i}{N}\sum_{j=1}^{N} \hat{O}_j. \end{aligned} \tag{3.9}$$

We can now start the next iteration of the optimization, returning to Step 1, until the averaged fitting error is sufficiently small.

Efficiency and Robustness: For efficiency, instead of using all the points on a sample, we (uniformly) randomly sample 400 points as the input to the iterative algorithm. Therefore, the iterative optimization itself takes only a few minutes. Once the final $\phi(t')$

is known, Step 2 (Equation (3.7)) is used to estimate A, D, R, O for all the points in full resolution. Since the average image size for each sample is large (400×400), and we must solve a nonlinear optimization for each pixel, this estimation can take a few hours, but the process is completely automated.

One difference between this estimation of the STAF model and existing algorithms for the shape invariant model and dynamic time warping is that we seek to estimate $\phi(t')$ in the full range $J = \bigcup_i J_i$ rather than the overlap range $\bigcap_i J_i$. This means in Step 2 (Equation (3.7)) we need to extrapolate both $p_i(t)$ and $\phi(t')$. We found that the kernel-based curve $p_i(t)$ and the estimated $\phi(t')$ from Step 1 cannot extrapolate well directly. Therefore, in Step 2 before the optimization, we fit $\phi(t')$ with a smooth polynomial model and use it to estimate A, D, R, and O.

Convergence: We are unable to perform a theoretical analysis on the convergence rate of the iterative algorithm. Theoretical analysis for the convergence of shape invariant modeling was performed by Kneip and Gasser [1988]. In practice, we found that for most samples in our TSV-BRDF database, $5 \sim 10$ iterations are sufficient to obtain accurate estimates of all parameters. Among the 28 samples in our database, there are three samples for which the iterative algorithm does not converge. These three samples are drying of tree bark, waffle toasting, and decaying of apple with core. For these samples, within the acquired sequence, not only their surface reflectance, but their surface geometry have also changed.

3.4.3 Experimental Results

The top row of Figure 3.7 shows five samples, with three spatial locations marked on each. The middle row shows curves for one of five BRDF parameters — the red diffuse component — over time. Similar results are obtained for the other four parameters. As can be seen, the curves from the three points for each sample are quite different, showing the difference in their time-varying appearance in the captured data. In the bottom row, we show the alignment of these separate time-varying curves by estimating the factored representation, where the estimated $\phi(t')$ is plotted in as the black curve. The horizontal axis is the *effective time*, t', while the vertical axis is the normalized function value $(p(x, y, t) - D(x, y))/A(x, y)$.

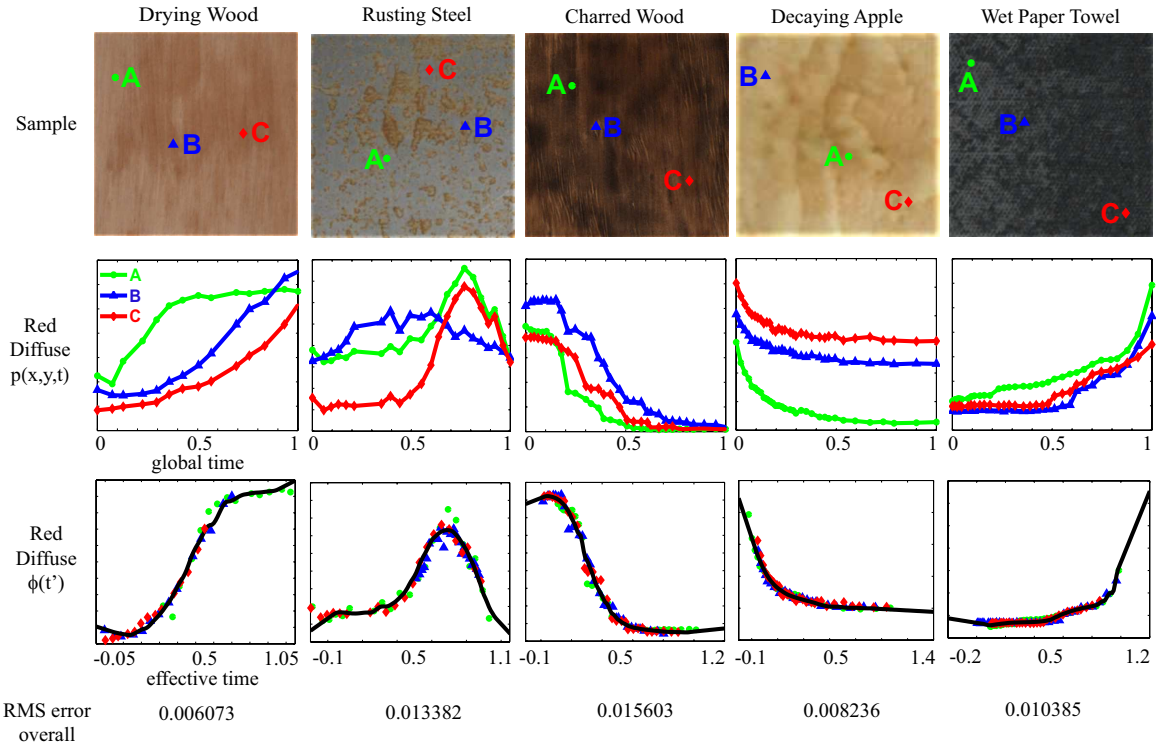


Figure 3.7: Estimating the factored representation. **Top:** A range of different phenomena, with 3 spatial locations marked on each sample. **Middle:** Time-varying curves $p(x, y, t)$ (for the red diffuse component) for spatial locations A, B and C. The curves are quite different for different points A, B and C. **Bottom:** We align these time-varying curves using our model. The data accurately matches the temporal characteristic curve $\phi(t')$ computed from all the points on the sample. The overall RMS image reconstruction error (across all temporal frames and spatial locations) is very low. This shows the generality of our model.

The green/red/blue dots overlaid on the black curves show which portions of the black curves $\phi(t')$ correspond to each of the three original curves in the second row. Note that the $\phi(t')$ curves extrapolate beyond the data as expected, having a larger range of effective times than $[0 \ldots 1]$.

If the model in Equation (3.3) could explain the measured data well, the curves from different spatial locations should all be aligned after the warping, fitting the overall temporal curve $\phi(t')$. Indeed, the time-aligned data in the bottom row of Figure 3.7 matches very well to the overall temporal curve. The overall RMS image reconstruction errors are computed across all temporal frames and spatial locations. The range of the image intensity is generally in $[0, 1]$, except for samples with strong specular components, such as the steel for which the intensity of the specular pixels is in $[0, 30]$. Note that Figure 3.7 shows a variety of phenomena, with a number of different data-driven forms and mathematical curve-types for the overall temporal curve $\phi(t')$.

The accuracy of our factored model is evaluated in Figure 3.8 for the wood-drying example in our database. As shown in the top two rows, the STAF model accurately reproduces the drying patterns in the original measured data. In the bottom row of Figure 3.8, in addition to the estimated overall temporal curves $\phi(t')$ for the five parameters for the TSV-BRDF, we also show the four estimated textures for the diffuse component of the TSV-BRDF. Since $A(x, y)$ and $D(x, y)$ are harder to interpret directly, we show the normalized initial appearance, $A(x, y)\phi(0) + D(x, y)$ and the normalized final appearance $A(x, y)\phi(1) + D(x, y)$. In the absence of rate and offset effects, the TSV-BRDF interpolates between these two SV-BRDFs, weighted by $\phi(t)$. We also show $R(x, y)$ that controls the rate at which different points dry. It corresponds closely with the spatial patterns observed at later frames. Finally, we show the offset texture $O(x, y)$. It is mostly close to 0, since we wet our sample uniformly before starting acquisition. However, it does show small non-uniformities and the slightly faster start to drying in the top left region. We also show the diffuse and specular $\phi(t')$ curves. The specular $k_s(t')$ decreases exponentially, changing more rapidly than diffuse color.

One of the principal benefits of our factored representation is that it enables a variety of rendering applications, as discussed in the next section. Figure 3.8 demonstrates a simple

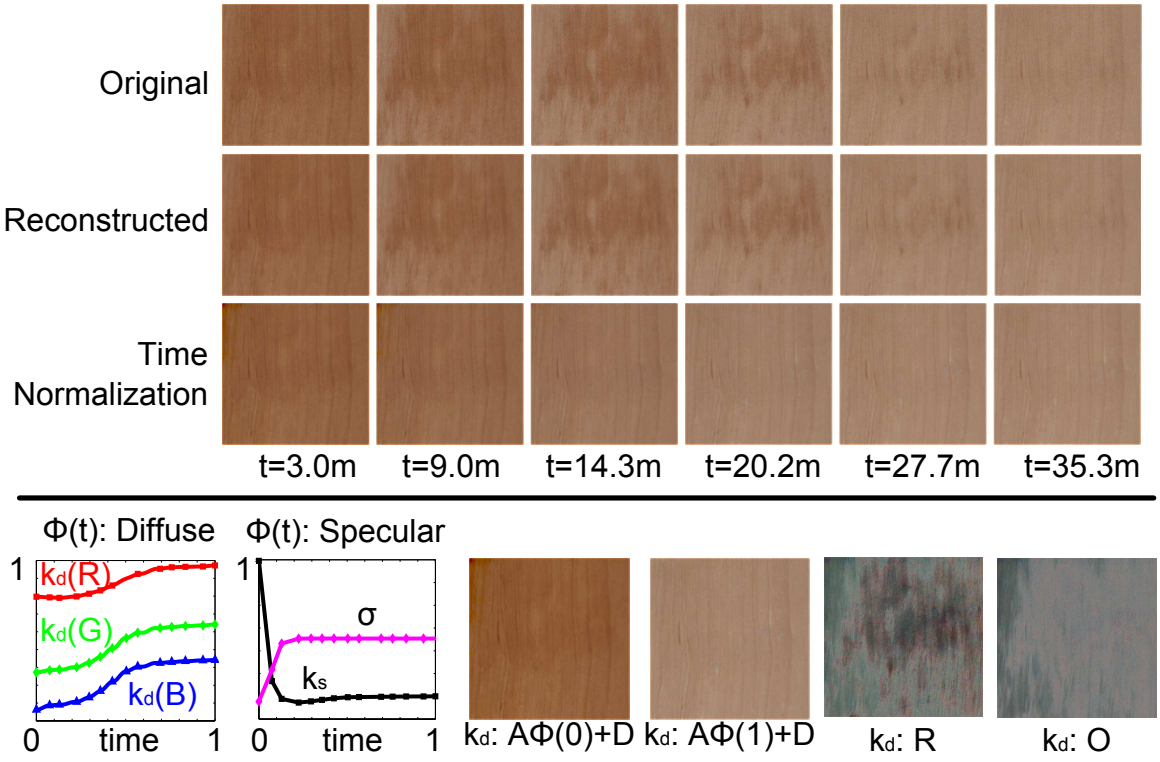


Figure 3.8: STAF factorization results for the drying wood sample. **Top:** the STAF model can accurately reconstruct the acquired time-varying appearance, and can also be used for time normalization wherein we keep the overall appearance changes but eliminate the spatial drying patterns. **Bottom:** we show the estimated $\phi(t')$ for both diffuse and specular parameters and the estimated spatial "textures" $A(x,y)$, $D(x,y)$, $R(x,y)$, and $O(x,y)$. In particular, we show the normalized initial appearance $A(x,y)\phi(0) + D(x,y)$, and the normalized final appearance $A(x,y)\phi(1) + D(x,y)$.

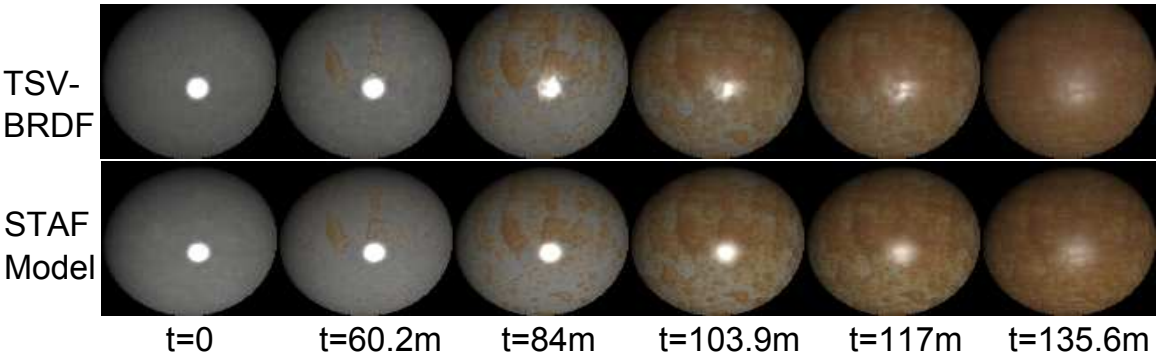


Figure 3.9: Comparison of the original TSV-BRDF (texture mapped onto a sphere and rendered) with the STAF model for several time frames for the rusting steel sample.

way in which we can separate space and time-varying effects by *time normalization*, making all points on the surface evolve at the same rate and in effect removing the spatial patterns formed over time. For this purpose, we leave $A(x, y)$, $D(x, y)$ and $\phi(t)$ unchanged, and set $O(x, y) = 0$ to eliminate offsets and $R(x, y) = 1$ to eliminate differences in rates. More flexible rendering applications are shown in the next section.

As another verification example, Figure 3.9 shows the renderings of the rusting steel sample using the STAF model and the original TSV-BRDF. As shown, the STAF model accurately reproduces the captured TSV-BRDF, which captures the dimming of the specular highlight, and the intricate spreading of the spatial rust patterns over time.

In terms of compression, as mentioned earlier, the average size of the raw data (high dynamic range images) of one sample is about 30 GB. Fitting parametric BRDF models for each time step reduces the size to about 80 MB. The STAF model can further reduce the size of one sample to about 6 MB on average — we only need to store four texture images $A(x, y)$, $D(x, y)$, $R(x, y)$, and $O(x, y)$ and the curve $\phi(t)$ for each of the five parameters in the TSV-BRDF model. Using other image compression techniques (*e.g.*, JPEG), we can reduce the size even further to about $1 \sim 2$ MB without producing noticeable artifacts.

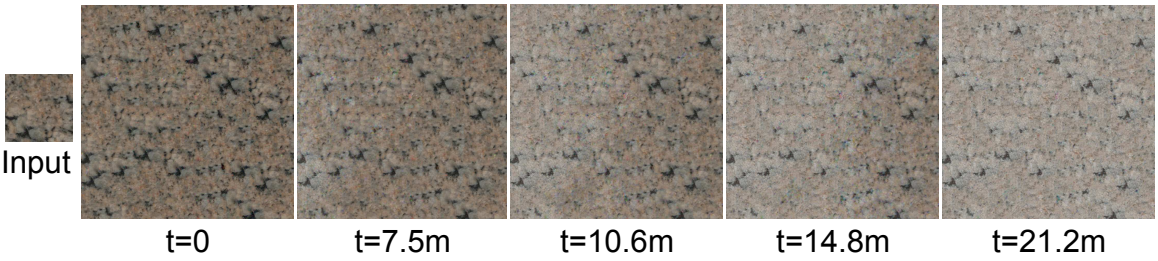


Figure 3.10: Time-varying texture synthesis can be reduced to 2D synthesis of static spatial textures $A(x, y)$ and $D(x, y)$ with our model. We choose to preserve the overall drying pattern from left to right in the original sample.

3.5 Synthesis of Time-Varying Surface Appearance

In this section, we show several applications to synthesize novel time-varying surface appearance based on the STAF model. Since the STAF model factors the space and time-varying effects, it allows us to edit and manipulate each term separately. The 3D renderings were done using the PBRT package [Pharr and Humphreys, 2004]. To focus on the texture and reflectance patterns and time variation, we mostly use simple geometry and lighting — it is straightforward to generalize to more complex scenes and lighting/rendering algorithms.

Time-Varying Texture Synthesis: Our database is acquired on small flat samples. For high quality rendering, we hope to synthesize larger time-varying samples. While there have been many successful algorithms for synthesizing 2D textures, synthesizing time-varying textures is quite challenging because there is usually no similarity over time (except for a few phenomena with time repetition such as water falls or waves, which are not the subject of this work). In other words, temporal coherence must be maintained during the synthesis. The STAF model allows us to use standard 2D texture synthesis methods, such as image quilting [Efros and Freeman, 2001], for creating larger time-varying textures.

Our method is to use image quilting to synthesize the initial and the final appearance, $I_0(x, y) = A(x, y)\phi(0) + D(x, y)$ and $I_1(x, y) = A(x, y)\phi(1) + D(x, y)$, and keep the temporal component $\phi(t')$ unchanged. Given the synthesized initial and final textures $I_0(x, y)$ and $I_1(x, y)$, we solve the two linear equations for the new $A(x, y)$ and $D(x, y)$. We can also apply texture synthesis to the rate map $R(x, y)$ and the offset texture $O(x, y)$ in a similar

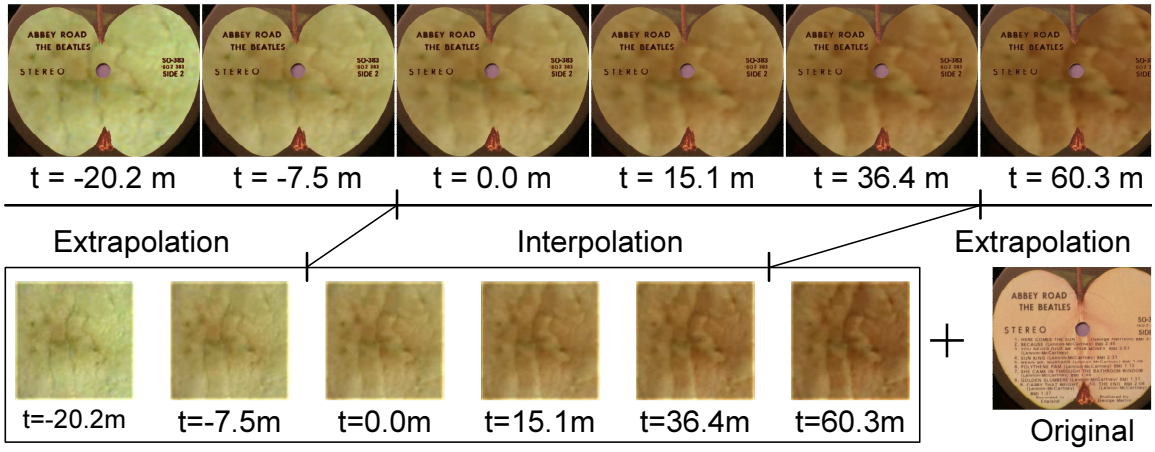


Figure 3.11: Time interpolation and extrapolation: decaying Apple Records logo, using our apple slice data set, and modulating by a static texture map of the logo from a Beatles record. This example demonstrates extrapolation, wherein we back up the decay process to considerably before actual start of acquisition—the decay is mostly complete at +30 minutes, and we back up to -20 minutes, getting a much greener look on the apple (we are also able to extrapolate beyond the final time frame).

fashion. Once we have the synthesize components, the synthesized time-varying texture $p(x, y, t)$ is given based on Equation (3.3).

Figure 3.10 shows a texture synthesis result of the drying rock example. In this case $R(x, y)$ and $O(x, y)$ are not textures in the conventional sense, but encode an overall variation over the surface, where the rock dries from left to right. Therefore in this example, we choose to preserve this overall effect and simply enlarge $R(x, y)$ and $O(x, y)$ with scaling.

Extrapolation: The temporal characteristic curve $\phi(t')$ extends beyond the actual global time range over which the data is acquired, allowing us to back up or extend the process beyond the acquired data for many pixels.

Figure 3.11 demonstrates extrapolation on the apple slice data set, to obtain virtual frames even before the actual start of acquisition. We use a photograph of an Apple Record logo to modulate the TSV-BRDF and create the effect of the cut apple logo decaying. For extrapolation, we evaluate the STAF model at its overall minimum and maximum value

Figure 3.12: Footprints drying on a wooden floor. We use the drying wood data set, controlling the rate and offset of drying as shown in the maps on the far right. Specifically, the prints dry faster toward the edges, and the left footprint has a lower offset (higher effective time) and so dries earlier.

as per Equation (3.4). In this data set, most of the decay actually happens in the first 30 minutes, and we use input from the corresponding 10 initial frames only for this figure. We show a significant backing up of the process for many pixels up to $t = -20$m, to the point where the apple is much greener.

Control: By changing rate and offset parameters $R(x, y)$ and $O(x, y)$, we can control the rate at which different points on the surface change, while still preserving the characteristic features of the time-varying process. For example, $R(x, y)$ and $O(x, y)$ can be designed by artists to achieve desired effects, or be set according to some environmental context, such as the amount of light or humidity.

Figure 3.12 shows how the drying wood can be controlled to create the appearance of drying footprints on a wooden floor. We set the offset map $O(x, y)$ so that the floor starts out dry (*i.e.*, $O(x, y) = -1$) and the lower left footprint dries earlier than the right (*i.e.*, it has a smaller offset $O(x, y) = 0$, compared to $O(x, y) = 0.3$ for the upper right footprint). We also set the rate map $R(x, y)$ to control the speed of drying, where $R(x, y)$ is set higher toward the edges and lower toward the center of the footprints. Specifically, we compute the distance transform $d(x, y)$ for points inside the footprint, and set $R(x, y) \sim d^{-1}(x, y)$. Finally, we use a 7×7 Gaussian filter on the resulting maps $R(x, y)$ and $O(x, y)$ to ensure smooth transitions, especially at the edges. The rendered images in Figure 3.12 show the desired effect.

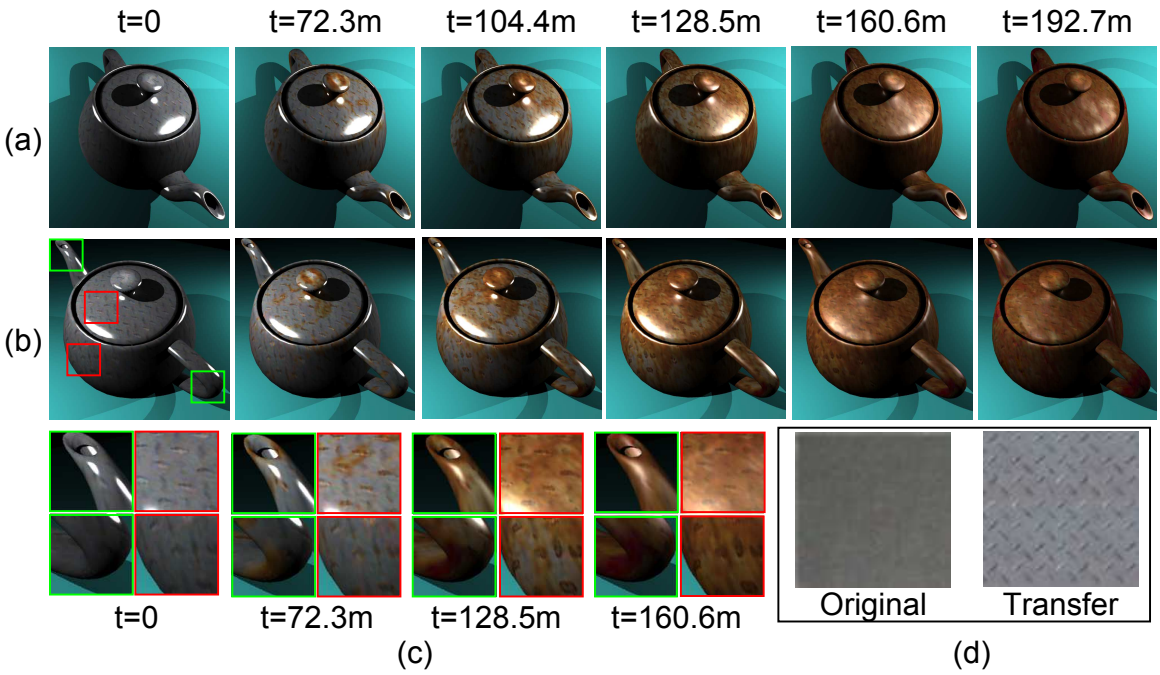


Figure 3.13: Rendering of a rusting teapot. We *transfer* the rust TSV-BRDF to a new static texture (shown at bottom right (d)). We show two views (a) and (b), with specular and diffuse effects. The insets (c) in the bottom left show the effects of control, where edges and high-curvature areas rust faster.

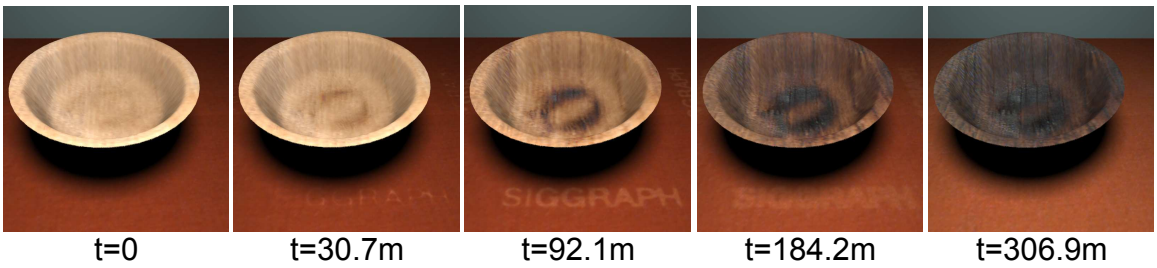


Figure 3.14: Creating ephemeral patterns by adjusting rates. The bowl uses the burning wood sample, and the table is drying orange cloth. Control is provided by a virtual heat source. We start with the static appearance, gradually evolving into the SIGGRAPH logo, and then into a fully charred and dry state.

Transfer: By changing $A(x, y)$ and $D(x, y)$ to those obtained from a new static photograph, we can transfer the time-varying animation, such as burning or rusting, to a new static object, while still preserving the essence of the data-driven appearance change.

In Figure 3.13, we *transfer* the rusting steel time-varying process to a new (not rusty) steel plate, using only a single image of its initial condition. The ratio of the new photograph $I_{new}(x, y)$ to the first frame of the original sample $I_0(x, y)$ is used to modulate both static textures $A_{new}(x, y) = A(x, y) \cdot I_{new}(x, y)/I_0(x, y)$ and $D_{new}(x, y) = D(x, y) \cdot I_{new}(x, y)/I_0(x, y)$. We then texture-map the time-varying pattern onto a 3D teapot. Note both the diffuse and specular effects as well as their temporal variations are preserved.

We also edit the rate map $R(x, y)$ to increase the rate of rusting in high curvature regions. In addition, we do edge detection on our static 2D image of the steel plate, to increase the rate near edges. The net rate $R(x, y) = \kappa(x, y)\mu(x, y)$ where $\mu(x, y)$ is an edge map and $\kappa(x, y)$ is the average curvature. The insets in the bottom row show that different parts of the object rust at different rates. We have full 3D rendering capabilities, and can see the teapot from different viewpoints while the appearance is evolving.

Figure 3.14 shows how user-specified patterns can be created in the otherwise natural time-varying processes, with implications for special effects and animations. We texture-map the burning wood onto a bowl model; the table cover is from our drying orange cloth data set. Control is effected through a virtual heat source, for both burning and drying. In addition, we manually modify the rate $R(x, y)$ to resemble the SIGGRAPH logo, for both

the bowl and the cloth. As time progresses, the patterns gradually appear on the bowl and table; when the charring and drying are complete, the patterns disappear.

Limitation: While the overall temporal curve $\phi(t)$ is rather independent with the four spatial textures $A(x, y)$, $R(x, y)$, $O(x, y)$, and $D(x, y)$, there could be strong correlations between some of these spatial textures for some time-varying samples, and therefore we might not be able to control each of the four textures independently for some samples. Often the correlation between $A(x, y)$ and $D(x, y)$ is strong since they control the initial and final appearance of the sample. The correlation between $A(x, y)$ (or $D(x, y)$) and $R(x, y)$ (or $O(x, y)$) is usually low for most samples, but for samples where the internal material structure is important for the time-varying appearance (*e.g.*, wood texture for wood drying), the correlation could be strong.

Figure 3.15 shows the normalized correlation among these four textures for all the samples in the database for the diffuse component k_d, the specular component k_s, and the surface roughness σ. We should take into account the correlations before we perform controls or edits to the components.

3.6 Summary and Discussion

We have presented a complete pipeline from acquisition to rendering for time and space-varying appearance or TSV-BRDFs. This leads to a new capability for computer graphics imagery, to include the dynamic evolution of surfaces and scenes. Our contributions include a newly acquired data set of time-lapse images for many natural processes from multiple light source and viewing directions, along with estimated parametric TSV-BRDFs. Our main technical contribution is a compact intuitive factored representation that separates spatially varying aspects from temporal variation, being accurate for a variety of natural phenomena. With this representation, we can generalize to a number of novel rendering tasks such as transfer, control, extrapolation and texture synthesis.

There are some limitations for the STAF model. (1) Current STAF model assumes there is one overall temporal curve for each sample. This assumption, however, does not hold for some natural phenomena. One example is the decaying apple slice with core, where there

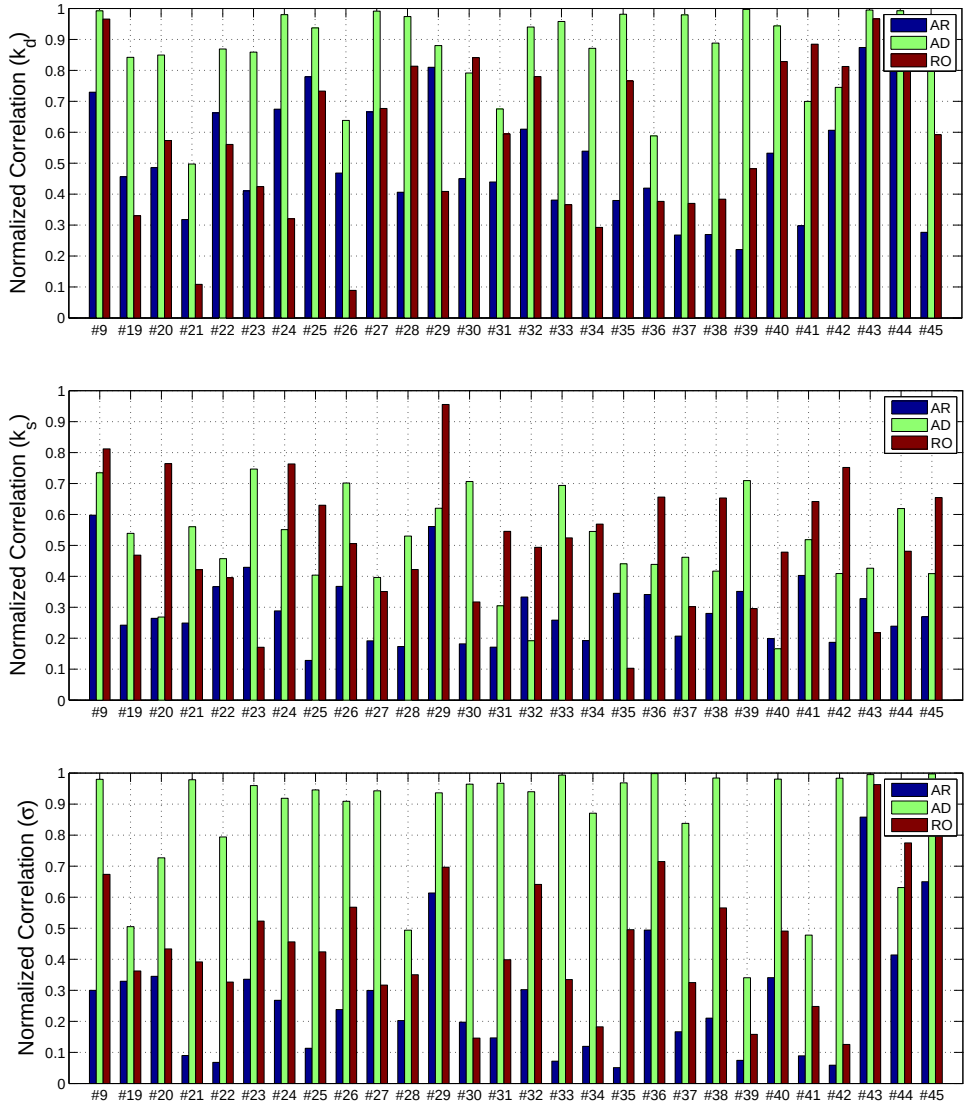


Figure 3.15: Normalized correlation among the four textures $A(x, y)$, $D(x, y)$, $R(x, y)$ and $O(x, y)$ for samples in the TSV-BRDF database, for the diffuse component k_d, the specular component k_s, and the surface roughness σ. High correlation means we cannot control or edit individual components separately. For most samples in the database, as expected, the correlation between $A(x, y)$ and $D(x, y)$ is high while the correlation between $A(x, y)$ (or $D(x, y)$) and $R(x, y)$ (or $O(x, y)$) is relatively low.

are multiple types of time-varying processes occurring such as the decay of the fruit and the core. Extending the STAF model to consist of multiple temporal characteristic curves is one possible solution. (2) In our study, we have assumed flat surfaces and simplified time-varying surface appearance as TSV-BRDF. For surfaces with large reliefs, such as tree bark, the inter-reflection and shadowing between neighboring spatial locations need to be taken into account explicitly using concepts such as time-varying BTF. These limitations are the subjects of our future research.

The idea of time-varying surface appearance extends beyond the data sets and models reported here. We currently represent the temporal characteristic curve $\phi(t')$, which is effectively the overall time-varying BRDF, in a purely data-driven way, without further analysis. One of our future goals is to understand the time evolution of $\phi(t')$, to develop time-varying BRDF models that would form the time-varying counterpart of common static BRDF models. In particular, we are interested in studying how the BRDF evolves with time for transparent surfaces, such as windows, glasses, monitors, camera lenses, as the surfaces accumulate dust, dirt and other types of contaminants over time. This subject is discussed in detail in the next chapter.

Chapter 4

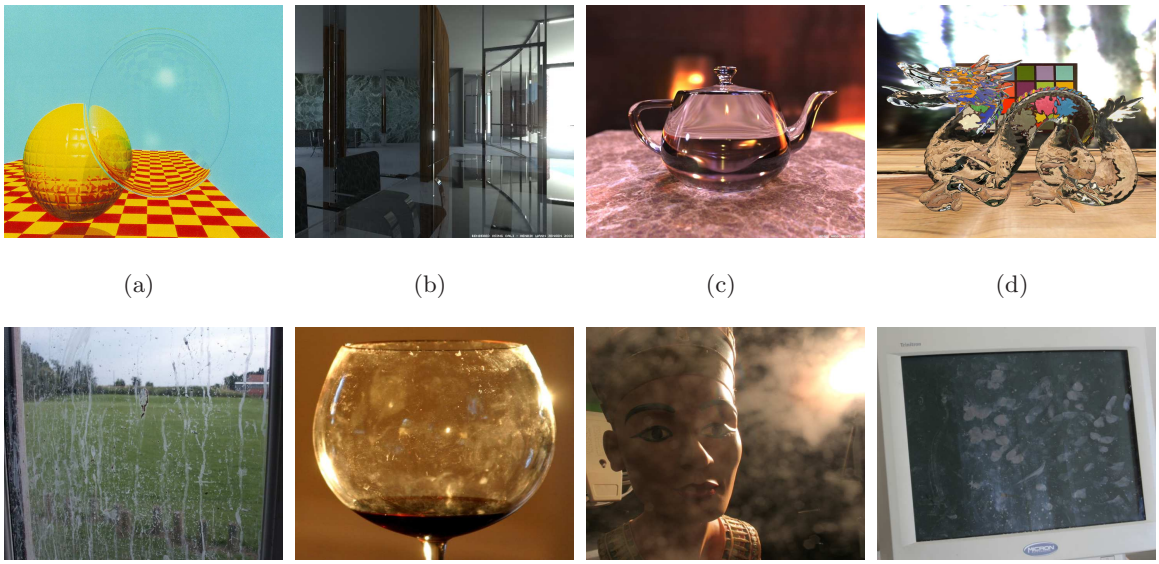
Weathered Appearance for Transparent Surfaces

In Chapter 3 we studied time-varying appearance for opaque surfaces. In this chapter, we turned our focus to time-varying appearance for transparent surfaces.

4.1 Introduction

In our daily lives, we are surrounded by transparent surfaces such as windows, windshields, monitors, reading glasses, drinking glasses, camera lenses, and mirrors. The rendering of the effects produced by clean transparent surfaces has been widely studied, and today's advanced rendering techniques, such as [Jensen, 2001], are able to accurately and efficiently render the most salient effects. However, rendered images of clean transparent surfaces tend to look too perfect.

Figure 4.1 show both the state-of-the-art renderings of transparent objects and photographs of read transparent objects. As seen, transparent objects in the real world almost always have surface contaminants such as dust (*e.g.*, pollen, skin cells, fabric fibers), dirt (*e.g.*, powder, salt, other minerals, organic materials, soil), and lipids (*e.g.*, fingerprints, lipstick, and other oils). These imperfections produce weathered appearance that implies the objects went through real weathering processes and convinces us the images are real [Dorsey and Hanrahan, 2000].



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

(e) (f) (g) (h)

Figure 4.1: **Top:** Rendered images of transparent objects in previous work — (a) refraction by ray tracing [Whitted, 1980]; (b) and (c) caustics and global illumination by photon mapping [Jensen, 2001]; and (d) interactive refraction [Wyman, 2005]. **Bottom:** Photographs of transparent objects in the real world — (e) a window with dirt; (f) a wine glass with stains; (g) an image taken with a dusty lens; and (h) a monitor with fingerprints. We note that the contaminants on the transparent objects produce imperfections in appearance that are important for image realism.

While such weathered appearance may seem simple to create in computer graphics at first glance (*e.g.*, alpha-blending some dirt texture with a rendered image of a transparent surface), it actually includes interesting scattering effects that require modeling of the light transport on the transparent surface. For example, in Figure 4.1(e) the dirt looks darker against the sky on the top, while it looks brighter against the ground on the bottom. In Figure 4.1(f), the stains on the wine glass alter the light transport through the surfaces and create an interesting soiled appearance. Dust on a camera lens scatters light from outside the field of view of the lens to the image detector to produce lens glare, as shown in Figure 4.1(g). Fingerprints or other contaminants become clearly visible on a monitor screen when the monitor is off, as seen in Figure 4.1(h). The visual effects also vary with viewing and illumination angles. For instance, contaminants on glass that are barely visible in a frontal view become clearly visible when viewed from a grazing angle.

Theoretically, as shown in Chapter 2, one could use Monte Carlo to simulate exactly the light transport and reproduce these effects by tracing a large number of photons scattering on the weathered transparent surface. In practice, however, Monte Carlo simulation often requires quite expensive computation and thus is not suitable for applications such as computer games and design.

Matusik et al. [2002] performed a data-driven study to acquire and render transparent objects with surface reflectance field, which achieves high realism but requires a complicated measurement system. In this chapter, our goal is to develop an efficient way to add the realistic weathered appearance on the rendering of transparent surfaces. We derived a physically-based analytic model for the reflectance and the transmittance for weathered transparent surfaces, based on the light transport theory in radiative transport theory [Chandrasekhar, 1960; Ishimaru, 1978]. The model captures many of the canonical visual effects common in the real world, such as the contrast reversals of the contaminated regions against the background, lens flare/blurring caused by stray light, and the angular dependency with the viewing/lighting direction.

In rendering contamination effects, we must also consider the spatial distribution or *texture* of the contaminant. To this end, we developed methods for measuring contamination patterns from real glass and plastic samples (Section 4.4). The measured data, together

with some procedural textures, can be used to synthesize contamination patterns for rendering various types of weathered appearance on transparent surfaces, as demonstrated in Section 4.5.

4.2 Light Transport of Contaminated Transparent Surfaces

In this section, we construct the simplified analytic model for light transport on a contaminated transparent surface. Below, we first define some characteristics of this specific type of material and explain our assumption of single scattering, and then show the derived models for its reflectance and transmittance.

4.2.1 Definition and Assumption

Contaminated transparent surfaces are one specific type of layered material consisting of two layers — the top layer is some type of contaminant (*e.g.*, household dust, dry powder, fingerprints) and the bottom layer is a transparent surface, as shown in Figure 4.2. All the scattering events happen with the contaminant layer. Also, depending on the material properties of the contaminant layer, the incident light might reflect and refract at the interface between the air and the contaminant layer. In our study, the physical thickness of the contaminant layers of interest ranges from a few micrometers (*e.g.*, oil film) to a few millimeters (*e.g.*, powder).

For illustrative purposes, we loosely define three types of contaminants that are commonly seen on transparent surfaces — dust, dirt, and lipids (*e.g.*, fingerprints), according to their material properties and thicknesses, as summarized in Table 4.1. *Dust* consists of smaller particles that are mostly forward scattering. Its density is usually very low and thus we assume there is no refraction between the air and the dust layer. Examples in this category include household dust, pollen, carbon blacks, and colloidal paints. *Dirts* are typically tiny pieces of bulk materials, such as metal oxides, dry powder, clay, and silts, which are often thicker and might have strong backward scattering. Since dirts have high densities, light will refract and reflect on the surface. *Lipids* means thin layers of some oily substance, such as fingerprints, lipstick, and oil mists. Lipids are usually very thin and mainly forward

Table 4.1: Definition of the three types of surface contamination.

	Interface with Air	Optical Thickness	Scattering
Dust	No, $n_1 = n_2$	Small	Mainly forward scattering
Dirt	Yes, $n_1 \neq n_2$	Medium	Both forward and backward scattering
Lipids	Yes, $n_1 \neq n_2$	Small	Mainly forward scattering

scattering, but they have high densities and therefore have an interface with the air. We emphasize that the distinctions are not strict. More details about the characterization of particles [Xu, 2002] and surface contamination [Kohli and Mittal, 2008] can be found in material science literature.

In our work, for the contaminant layers of our interest, we assume that multiple scattering is negligible compared with single scattering. This assumption holds for an *optically-thin* layer (defined in Section 2.1.2 on page 15). Fortunately, for many real-world transparent surfaces, the contaminant layer can be approximated as optically-thin layers [Kohli and Mittal, 2008], either because the layer is physically thin (*e.g.*, fingerprints), or because the contaminant has a very low density (*e.g.*, dust). For some dirt patterns with large optical thickness, the derived model can be regarded as a first-order approximation of the actual scattering phenomena.

4.2.2 Construction of the BRDF and BTDF Models

Some researchers have proposed models for the light transport of layered materials. Blinn [1982] derived a model for an opaque surface covered with a dust layer. Koenderink and Pont [2003] derived a similar model and used it to explain the velvety appearance of materials such as hairy skin and peaches. Both work assumed single scattering. Hanrahan and Krueger [1993] first systematically introduced light transport theory from radiative transfer literature [Chandrasekhar, 1960; Ishimaru, 1978] into computer graphics for modeling the reflection of layer surfaces, in which they simulated both single scattering and multiple scattering. Ershov et al. [2001] developed a recursive procedure to extend the analytic models for one layer to multiple layers for interactive rendering of car paints. Stam [2001]

Table 4.2: Notations of appearance models for weathered transparent surfaces.

L_i, L_r, and L_t	Radiance of the incident, reflected, and transmitted light
ω_i, ω_r, and ω_t	Direction of the incident, reflected, and transmitted light
N	Surface normal
R_p^{12} and T_p^{12}	Reflectivity and transmissivity at the air-contaminant interface
R_p^{23} and T_p^{23}	Reflectivity and transmissivity at the contaminant-glass interface
n_1, n_2, and n_3	Refractive index of air, the contaminant, and glass
g	Henyey-Greenstein parameter (average cosine)
$f_r(\cdot)$ and $f_t(\cdot)$	BRDF and BTDF of the contaminated transparent surface

proposed a semi-analytic model for multiple anisotropic scattering in a layer bounded by two rough surfaces.

In our study, we consider a specific type of layered material — a thin contaminant layer on a transparent surface. In addition to the scattering events modeled in the previous work, this specific material has its own set of scattering events. In particular, these events include the reflection at the air-contaminant interface, the scattering of the light reflected from the contaminated transparent surface, and the refraction through the contaminated surfaces. Our model for contaminated transparent surfaces includes additional terms for handling these light transport events. As shown later in Figure 4.6, theses events are important for accurately simulating the light transport for this material.

As shown in Figure 4.2(a), for the light $L_i(\omega_i)$ incident from the air onto the transparent medium, *i.e.*, $\omega_i \cdot N > 0$, there are four components for reflection and two components for transmission. Since we only consider single scattering, analytic solutions of the light transport equation (Equation (2.17)) exist for these components. The notations for some standard quantities have been defined previously in Table 2.1 on page 19. Other notations for the model derivation are summarized in Table 4.2.

Mirror Reflection from the Contaminant Surface $L_r^{(1)}(\omega_r)$: If the contaminant layer has an interface with the air ($n_1 \neq n_2$), there is a mirror-reflection at the surface of the

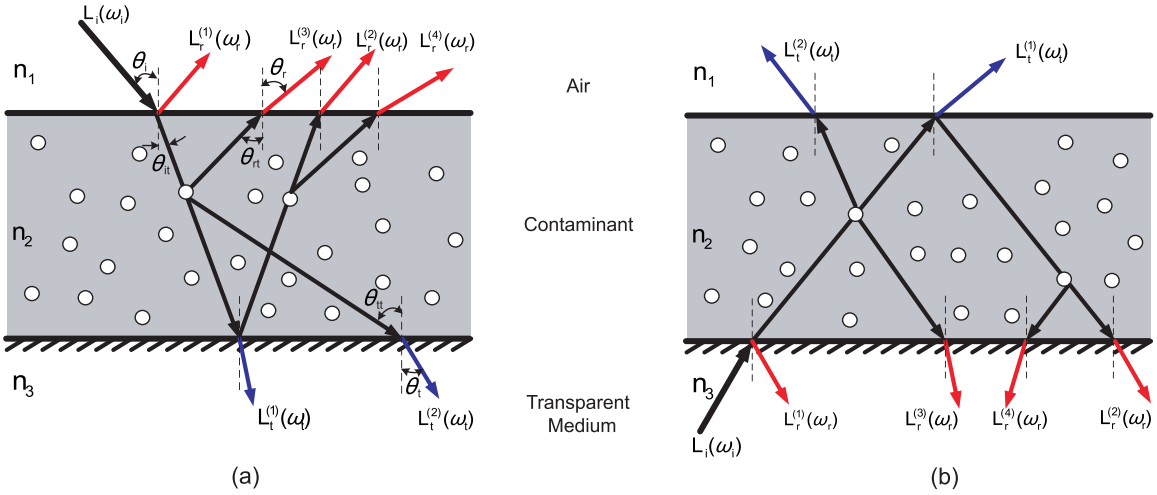


Figure 4.2: Light transport for a contaminated transparent surface: (a) Incident light from air to transparent medium; (b) from transparent medium to air.

contaminant, and

$$\frac{dL_r^{(1)}(\omega_i,\omega_r)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i} = R_p^{12}(\theta_i)\cdot\delta(\omega_i,\omega_r), \tag{4.1}$$

where $R_p^{12}(\theta_i)$ is the reflectivity (in terms of power) determined by Fresnel equations (see Table A.2 on page 153) , and $\delta(\omega_i,\omega_r)$ is the delta function defined as $\delta(\omega_i,\omega_r) = \delta(\sin^2\theta_i - \sin^2\theta_r)\delta(\phi_i - \phi_r \pm \pi)$.

Mirror Reflection from the Transparent Surface $L_r^{(2)}(\omega_r)$: This component models the light that passes through the contamination layer to the transparent surface and then is mirror reflected back toward the contaminant layer. As the light travels inside the contaminant layer it is attenuated exponentially with the optical path depth, leaving the *reduced intensity* [Ishimaru, 1978]. Note that the light will be attenuated twice — when it is transmitted into the contaminant and when it is reflected back. This gives

$$\frac{dL_r^{(2)}(\omega_i,\omega_r)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i} = e^{-\tau(\frac{1}{\cos\theta_{it}}+\frac{1}{\cos\theta_{rt}})}\cdot T_p^{12}(\theta_i)\cdot T_p^{12}(\theta_r)\cdot R_p^{23}(\theta_{it})\cdot\delta(\omega_i,\omega_r), \tag{4.2}$$

where τ is the optical thickness of the contaminant layer, ω_{it} and ω_{rt} are the directions of the refracted rays in the contaminant layer computed by Snell's law. Since we consider mirror reflection, we have $\omega_{it} = \omega_{rt}$. $T_p^{12}(\theta_i)$ and $T_p^{12}(\theta_r)$ are the transmissivity (in terms of power) determined by Fresnel equations (see Table A.2 on page 153).

Scattering of the Incident Light $L_r^{(3)}(\omega_r)$: This component is due to scattering and is called the *diffuse intensity* [Ishimaru, 1978]. For optically thin layers, we consider only single scattering, and thus have analytic solutions to the light transport equation (Equation (2.17))

$$\frac{dL_r^{(3)}(\omega_i,\omega_r)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i} = \frac{n_1^2}{n_2^2}\cdot T_p^{12}(\theta_i)\cdot T_p^{12}(\theta_r)\cdot\frac{1-e^{-\tau(\frac{1}{\cos\theta_{it}}+\frac{1}{\cos\theta_{rt}})}}{\cos\theta_{it}+\cos\theta_{rt}}\cdot W_0\cdot p(\pi-\theta_{it}-\theta_{rt}), \quad (4.3)$$

where $p(\theta)$ is the phase function, W_0 is the albedo of scattering of the contaminant.

Scattering of the Mirror-reflected Light $L_r^{(4)}(\omega_r)$: Similarly, this term involves single scattering of the light reflected from the transparent surface (bottom lit):

$$\begin{aligned}\frac{dL_r^{(4)}(\omega_i,\omega_r)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i} = \frac{n_1^2}{n_2^2}\cdot T_p^{12}(\theta_i)\cdot T_p^{12}(\theta_r)\cdot\frac{e^{-\frac{\tau}{\cos\theta_{it}}}-e^{-\frac{\tau}{\cos\theta_{rt}}}}{\cos\theta_{it}-\cos\theta_{rt}}\\ \cdot\left(R_p^{23}(\theta_{it})\cdot e^{-\frac{\tau}{\cos\theta_{it}}}+R_p^{23}(\theta_{rt})\cdot e^{-\frac{\tau}{\cos\theta_{rt}}}\right)\cdot W_0\cdot p(\theta_{it}-\theta_{rt}).\end{aligned} \quad (4.4)$$

Transmission of the Incident Light $L_t^{(1)}(\omega_t)$: This component models the light that passes through the contaminant layer and is transmitted into the transparent surface:

$$\frac{dL_t^{(1)}(\omega_i,\omega_t)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i} = \frac{n_3^2}{n_1^2}\cdot T_p^{12}(\theta_i)\cdot T_p^{23}(\theta_{it})\cdot e^{-\frac{\tau}{\cos\theta_{it}}}\cdot\delta(\sin^2\theta_i-\frac{n_3^2}{n_1^2}\sin^2\theta_t)\delta(\phi_i-\phi_t\pm\pi). \quad (4.5)$$

Scattering of the Incident Light $L_t^{(2)}(\omega_t)$: This component models the light scattered into the transparent surface:

$$\frac{dL_t^{(2)}(\omega_i,\omega_t)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i} = \frac{n_3^2}{n_2^2}\cdot T_p^{12}(\theta_i)\cdot T_p^{23}(\theta_t)\cdot\frac{e^{-\frac{\tau}{\cos\theta_{it}}}-e^{-\frac{\tau}{\cos\theta_{tt}}}}{\cos\theta_{it}-\cos\theta_{tt}}\cdot W_0\cdot p(\theta_{it}-\theta_{tt}). \quad (4.6)$$

Summary: The resulting BRDF model has both a specular term $f_{r,s}$ (from $L_r^{(1)}$ and $L_r^{(2)}$) and a diffuse term $f_{r,d}$ (from $L_r^{(3)}$ and $L_r^{(4)}$). The BTDF has both the attenuated term $f_{t,a}$ (from $L_t^{(1)}$) and the diffuse term $f_{t,d}$ (from $L_t^{(2)}$):

$$\begin{aligned}f_r(\omega_i,\omega_r) &= f_{r,s}(\omega_i,\omega_r)+f_{r,d}(\omega_i,\omega_r),\\ f_t(\omega_i,\omega_t) &= f_{t,s}(\omega_i,\omega_t)+f_{t,d}(\omega_i,\omega_t),\end{aligned} \quad (4.7)$$

where the four components are given by

$$f_{r,s}(\omega_i,\omega_r)=\frac{dL_r^{(1)}(\omega_i,\omega_r)+dL_r^{(2)}(\omega_i,\omega_r)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i},\quad f_{r,d}(\omega_i,\omega_r)=\frac{dL_r^{(3)}(\omega_i,\omega_r)+dL_r^{(4)}(\omega_i,\omega_r)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i},$$

$$f_{t,s}(\omega_i,\omega_t)=\frac{dL_t^{(1)}(\omega_i,\omega_t)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i},\qquad f_{t,d}(\omega_i,\omega_t)=\frac{dL_t^{(2)}(\omega_i,\omega_t)}{L_i(\omega_i)\cdot\cos\theta_i\cdot d\omega_i}.$$

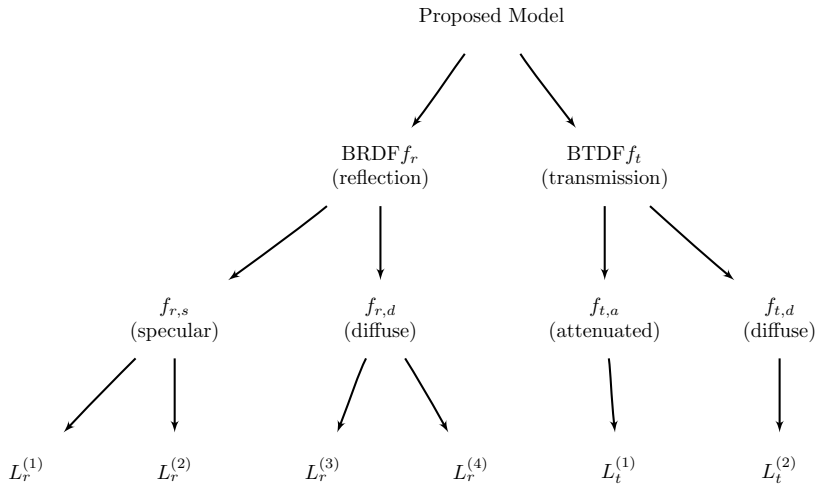


Figure 4.3: Relationship of the scattering components for the reflection and transmission of contaminated transparent surfaces.

These relationships are summarized in Figure 4.3. In Appendix B.2 on page 160, we have proved that the derived BRDF f_r and BTDF f_t models satisfy both the generalized Helmholtz reciprocity and the energy conservation constraints.

The Henyey-Greenstein function [Henyey and Greenstein, 1940] is used as the phase function. Therefore, there are four parameters in the model for characterizing the material properties for contaminated transparent surfaces (we assume the refractive index of the air n_1 and the refractive index of the glass n_3 are known) — τ, the optical thickness of the layer, usually is a texture of the contaminant pattern; g, the Henyey-Greenstein parameter of the phase function; W_0, the albedo of the contaminant; and n_2, the refractive index of the contaminant. Different kinds of contaminants can have parameters in different ranges, while their effects are captured with the same model.

Figure 4.4 plots the diffuse components, $f_{r,d}(\cdot)$ and $f_{t,d}(\cdot)$, with respect to the optical thickness τ and the Henyey-Greenstein parameter g of the phase function. It shows that as τ increases, the diffuse reflection $f_{r,d}(\cdot)$ steadily increases to a maximum value and keeps unchanged; in contrast, the diffuse transmission $f_{t,d}(\cdot)$ increases to a point, and then begins to decrease because of absorption and multiple scattering — this is consistent with our experience.

Figure 4.5 plots the diffuse components $f_{r,d}(\cdot)$ and $f_{t,d}(\cdot)$ with respect to the incident

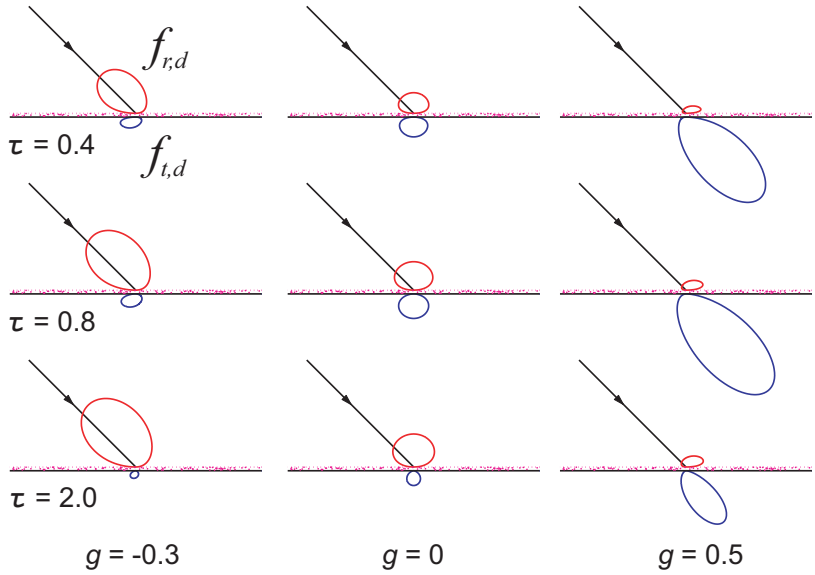


Figure 4.4: Plots of the diffuse components $f_{r,d}(\omega_i, \omega_r)$ (red) and $f_{t,d}(\omega_i, \omega_t)$ (blue) with respect to the optical thickness τ and the Henyey-Greenstein parameter g.

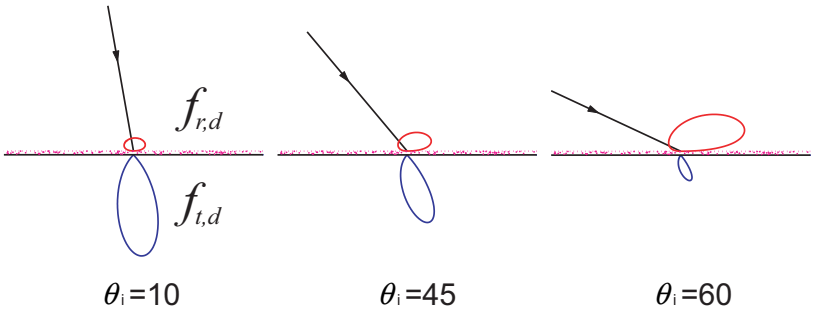


Figure 4.5: Plots of the diffuse components $f_{r,d}(\omega_i, \omega_r)$ (red) and $f_{t,d}(\omega_i, \omega_t)$ (blue), with respect to the incident angle θ_i: from left to right, θ_i increases from 10 to 40 to 65 degrees.

angle θ_i, which show the Fresnel effect — at more grazing angles, the diffuse reflection $f_{r,d}(\cdot)$ increases, while the transmission due to scattering $f_{t,d}(\cdot)$ decreases. This is also consistent with our experience, *i.e.*, the *bright contour* effect [Koenderink and Pont, 2003].

4.2.3 Model Validation and Comparison

Some components in the derived models above overlap with the models in previous work [Hanrahan and Krueger, 1993; Blinn, 1982]. [1] When $\tau = \infty$, the contaminant layer is semi-infinite. The BTDF is zero since there will be no light transmitted. For the BRDF, both $L_r^{(2)}$ and $L_r^{(4)}$ are zero since no light will be reflected on transparent surfaces, and thus the specular term $f_{r,s}$ only comes from $L_r^{(1)}$ and the diffuse term $f_{r,d}$ only comes from $L_r^{(3)}$ (Equation (4.3)) which reduces to

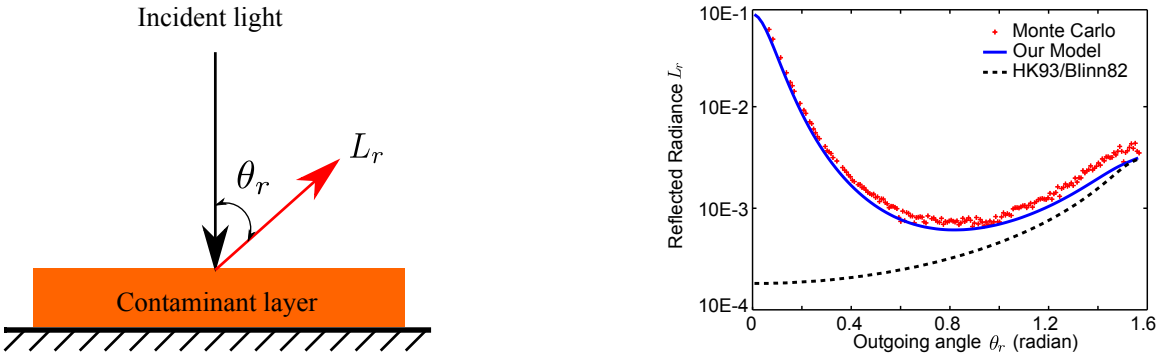
$$f_{r,d}(\omega_i, \omega_r) = \frac{n_1^2}{n_2^2} \cdot T_p^{12}(\theta_i) \cdot T_p^{12}(\theta_r) \cdot W_0 \cdot p(\pi - \theta_{it} - \theta_{rt}) \cdot \frac{1}{\cos\theta_{it} + \cos\theta_{rt}}. \tag{4.8}$$

This is consistent with the Lommel-Seeliger reflectance law and previous results in [Hanrahan and Krueger, 1993; Koenderink and Pont, 2003], with the assumption of isotropic scattering for semi-infinite scattering medium.

Nevertheless, there are some components that are in our model while not in [Blinn, 1982] (*i.e.*, $L_r^{(1)}$, $L_r^{(2)}$, $L_r^{(4)}$) and [Hanrahan and Krueger, 1993] (*i.e.*, $L_r^{(2)}$ and $L_r^{(4)}$). These additional components are used to model certain light transport events important for contaminated transparent surfaces, such as rendering certain kinds of contamination (*e.g.*, lipids and dirt) or rendering under certain lighting or view conditions (*e.g.*, stray light or viewing at grazing angles).

Figure 4.6 shows the simulation setup and results for a validation experiment. A collimated beam L_i is assumed to be normally incident on a transparent surface (refractive index $n_3 = 1.33$) covered with a uniform layer of dust, with the optical thickness $\tau = 0.2$, the albedo of scattering $W_0 = 0.5$, the Henyey-Greenstein parameter $g = 0.9$ and the refractive

[1] We also corrected some terms in [Hanrahan and Krueger, 1993; Blinn, 1982]. In [Blinn, 1982], $L_r^{(4)}$ is modeled as $\omega \cdot \phi(\pi)$, namely "a term for the forward scattering of the background through the cloud". This omits the angular dependency of the scattering. We correct it in Equation (4.4). $L_t^{(1)}$ in [Hanrahan and Krueger, 1993] (namely $L_{t,v}^{(0)}$ in Section 5.1) should be corrected by replacing τ with $\tau/\cos\theta$ as shown in Equation (4.5).



(a) (b)

Figure 4.6: Model validation and comparison. (a) A Monte Carlo simulation is performed to calculate reflection from a contaminated transparent surface. (b) The curves show the amount of reflected light $L_r(\theta_r)$ for each outgoing direction θ_r. Compared with the models in [Blinn, 1982; Hanrahan and Krueger, 1993], our derived model is seen to be more accurate and closer to the Monte Carlo simulation.

index $n_2 = 1$. We use the software package in [Wang et al., 1995] to perform the Monte Carlo simulation. The curves shown in Figure 4.6(b) represent the amount of reflected light $L_r(\theta_r)$ at each outgoing direction θ_r from 0 to $\pi/2$ for different evaluation methods. As shown, compared with the models in previous work [Blinn, 1982; Hanrahan and Krueger, 1993], the result of our model is more accurate and closer to the simulation.

4.2.4 Thin Transparent Slab with Contamination

The BRDF and BTDF model can be plugged directly into any raytracer to render contaminated transparent surfaces. However, since many transparent objects in the real world are thin slabs with two parallel interfaces with air, such as windows and eyeglasses, to make the rendering more efficient, we can further simplify the above BRDF/BTDF model since both the direction and the position of the incoming ray and the outgoing ray will not change. This is essentially a combination of the above model and the model for a clean transparent surface. If both sides of the slab have contaminants, we can also combine the above model twice to get a simplified model, each of which is for one side of the slab.

Specifically, as shown in Figure 4.7, for the case $\omega_i \cdot N > 0$ (*i.e.*, the incident light first

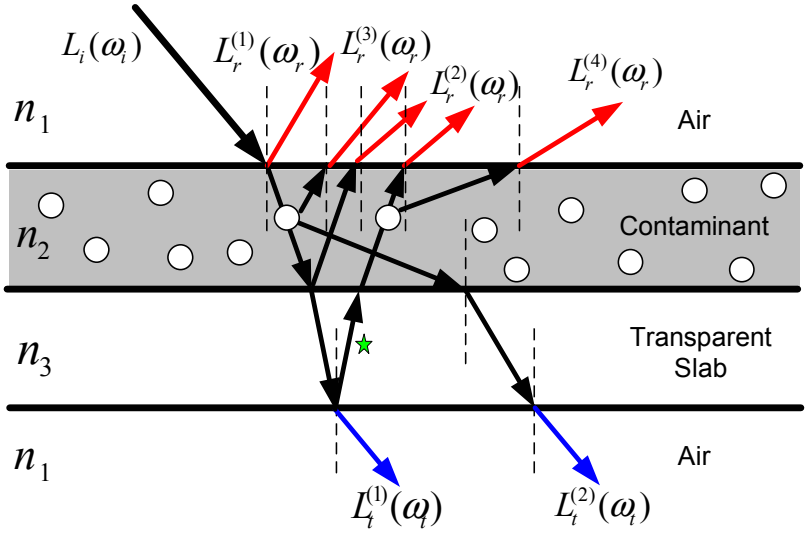


Figure 4.7: Light transport for a thin transparent slab with contamination, where the incident light goes from air into the slab and again into air.

strikes the contaminant layer and then transmits into the transparent slab and finally goes out to the air), for the BRDF, $L_r^{(1)}$ and $L_r^{(3)}$ are unchanged. For $L_r^{(2)}$ and $L_r^{(4)}$, we need to replace R_p^{23} by $R_p^{23} + T_p^{23}T_p^{32}R_p^{31}$ in Equations (4.2) and (4.4), since we need to take into account that a part of the transmitted light in the slab will be reflected back into the contaminant layer (the ray marked with $\star$ in Figure 4.7). For the BTDF, both $L_t^{(1)}$ and $L_t^{(2)}$ in Equations (4.5) and (4.6) need to be multiplied by an additional term T_p^{31} when the refracted rays come out from the slab to the air. The directions of the refracted rays, ω_{tt}, also need to be changed accordingly.

In Figure 4.8, we verify the accuracy of our BRDF/BTDF model by rendering (with a global illumination renderer) a simple scene of a dusty glass slab. Figure 4.8(a) is rendered using Monte Carlo simulation with 2048 samples per pixel needed to reduce noise, and it takes about 1 hour. Figure 4.8(b) is rendered with 4 samples per pixel using the derived BRDF/BTDF model, which takes about 10 seconds (this is essentially the same time as using any other analytic BRDF models). The RMS relative difference between these two images is 2.2%, which shows the accuracy of our model.

(a) Monte Carlo (b) Proposed Model

Figure 4.8: A uniform layer of dust rendered with (a) brute force Monte Carlo volumetric scattering and (b) our proposed BRDF/BTDF model. The RMS error between the two images is 2.2%, while the proposed analytic model is about 300 times more efficient for rendering.

4.3 Visual Effects Predicted by the Model

In Figure 4.9, we present a set of canonical examples corresponding closely to common real-world situations; in each a contaminant layer is placed on the surface of a thin transparent slab. The two leftmost columns describe the various viewing and illumination conditions considered here, as well as the placement of the contaminant layer. On the transparent slab, there are 2×3 patches of contaminants, as seen in the renderings in the rightmost column. From left to right, the contaminants are dust, dirt, and lipids (*e.g.*, fingerprints). These three kinds of contaminants cover a large range of the parameters and are selected to demonstrate the typical effects. The two patches in each column are identical. For dust and dirt, the optical thickness of the patches are uniform; for fingerprint, we crop a fingerprint texture and use it as the optical thickness. Finally, the middle column of Figure 4.9 shows scattering diagrams for the three types of contaminants, with the red components being reflection and the blue components being transmission. The parameters are chosen to demonstrate representative visual effects for each type of the contaminants.

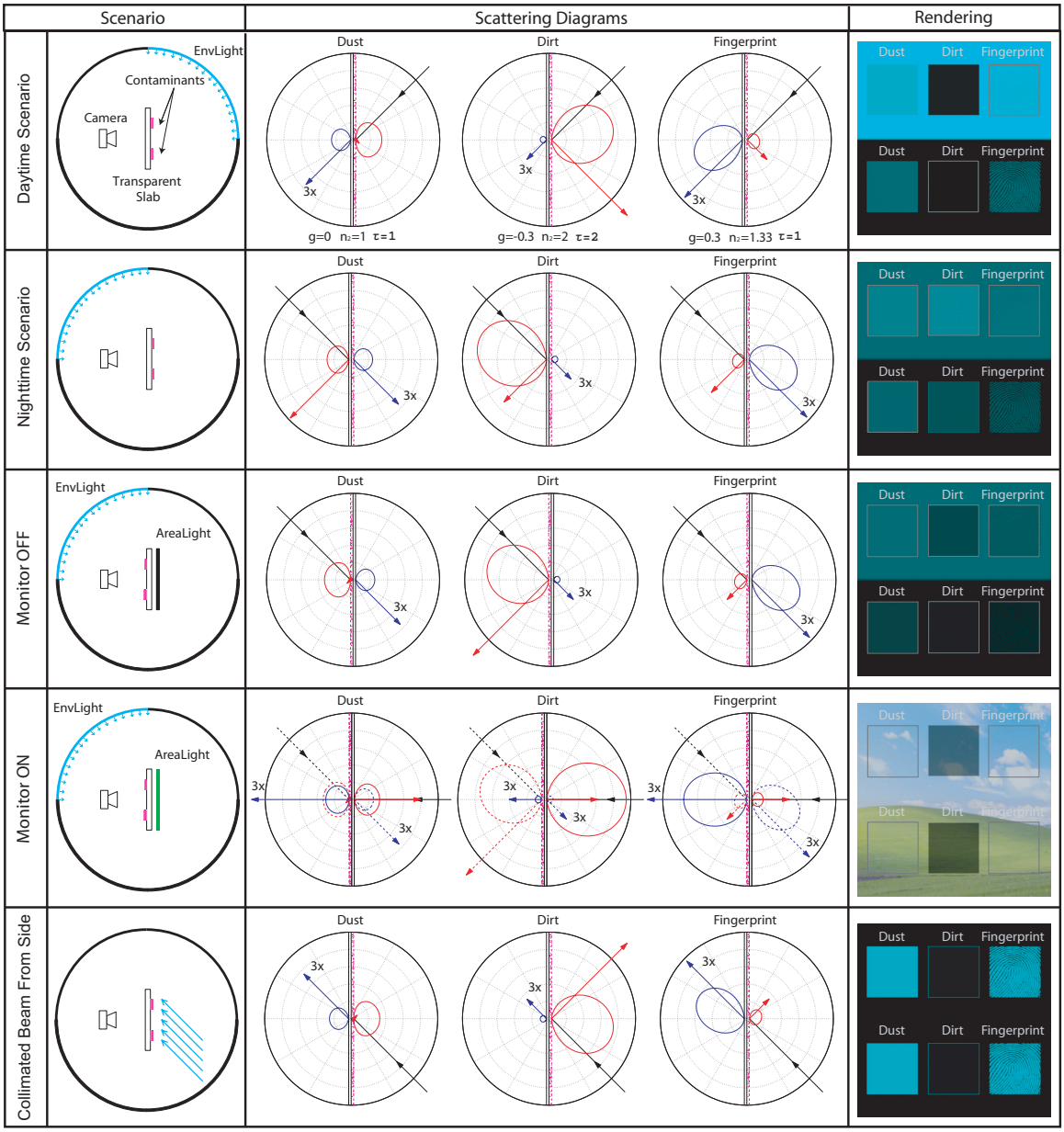


Figure 4.9: Visual effects predicted by our model, illustrated for a set of canonical scenarios common in the real world.

View of Window in Daytime: In the first row of Figure 4.9, we simulate the case when we are in a dark room and look through the window outward in daytime. Only the top region (or "sky") of the environment is (uniformly) bright. In the renderings, the upper half is bright and the bottom half is dark since only the upper half has environment light. However, the contaminated patches in the upper half look *darker* because of attenuation. By contrast, the contaminated patches (dust and fingerprint) in the bottom half look *brighter* because of scattering of the environment light. The dirt patches behave almost completely like an attenuator, with minimal scattering (consider the blue lobe in the middle scattering diagram). Therefore, that patch is black in both upper and lower regions.

Similar effects can be seen in real photographs (Figure 4.1(e)) and in rendering results (Figure 4.16 and Figure 4.21), where the contaminant looks darker against the sky, and brighter against the ground.

View of Window in Nighttime: The second row of Figure 4.9 corresponds to the situation when we are in a bright room (upper half is lighted) and look through the window in nighttime. Unlike the previous case, the contaminants in the upper half look *brighter* than the clean regions, because there is not only reflection from the slab but also scattering from the contaminant. The contaminants in the bottom half also look *brighter* because of scattering. This is true even for dirt–while the light transmitted through the window is attenuated as before, in this case we see primarily the light reflected from the dirt layer. Figure 4.16(c) shows this effect in rendering.

Monitor Turned Off: The third row in Figure 4.9 mimics the situation in which a computer monitor is turned off. Note that the contaminant is on the other side, the side closer to the camera. In this case, the contaminated regions in the bottom half still look brighter than the background due to scattering. The contaminants in the upper half look slightly darker than before because the reflected light from the monitor screen is attenuated by the contaminant before it comes to the camera. As the rendering results in Figure 4.17 (and the photograph in Figure 4.1(h)) show, the contaminant patterns are visible on the monitor when it is turned off.

Monitor Turned On: When the monitor is turned on, as shown in the fourth row in Figure 4.9, there is another textured area light source behind the screen (*i.e.*, the light from the monitor itself) in addition to the environment light. The area light is so close to the slab that the attenuated flux from the area light is much stronger than the flux scattered from the environment. Therefore, the difference between the clean regions and the regions with dust and fingerprint is not obvious, *i.e.*, the contaminants are almost invisible. For dirt, since its optical thickness is larger, its brightness will be attenuated more and thus it appears dark like an occluder.

With the previous example, this explains why we can see the dirts on the screen when the monitor is turned off while we usually cannot see them when the monitor is on, as shown in Figure 4.1(h) and Figure 4.17.

Collimated Beam from the Side: The fifth row in Figure 4.9 mimics the scenario of sunlight striking a window or lens. In this case, a collimated beam is incident on the transparent slab from the side. The camera sees no light from the collimated beam (*e.g.*, sunlight) in the clean regions because the transmission does not change the incident direction. In contrast, in the contaminated regions, the scattering due to the contaminants will redirect some of the light toward the camera. This is one of the reasons why a camera with a dusty lens or filter usually generates lens flare or a washed-out effect when there is strong stray light. Figure 4.1(g) and Figure 4.20 show photographs taken with a dusty lens and the corresponding rendering result.

4.4 Acquisition and Synthesis of Contamination Patterns

There exist a large variety of interesting contaminants on transparent surfaces in nature. For rendering photo-realistic images, we need to measure the weather appearance of these real samples. Among the four parameters in our derived BRDF and BTDF model, we developed methods to measure the optical thickness pattern $\tau(x, y)$ and the Henyey-Greenstein parameter g. For the other two parameters, the refractive index n_2 and the albedo of scattering W_0, their values are specified by users or from certain manuals [Stover, 1990]. To date, we have gathered 30 examples of representative materials, including many examples of dust, dirt, and

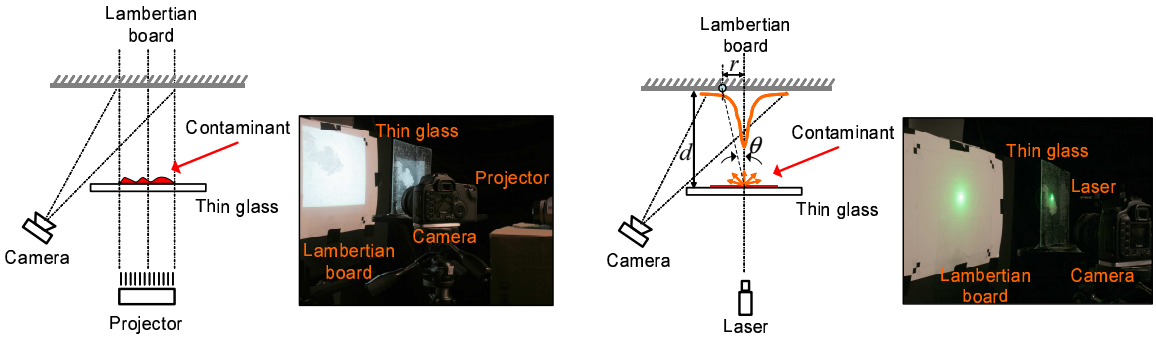


Figure 4.10: Setups used to measure (a) the optical thickness pattern $\tau(x, y)$ and (b) the Henyey-Greenstein parameter g from real contaminated glass samples.

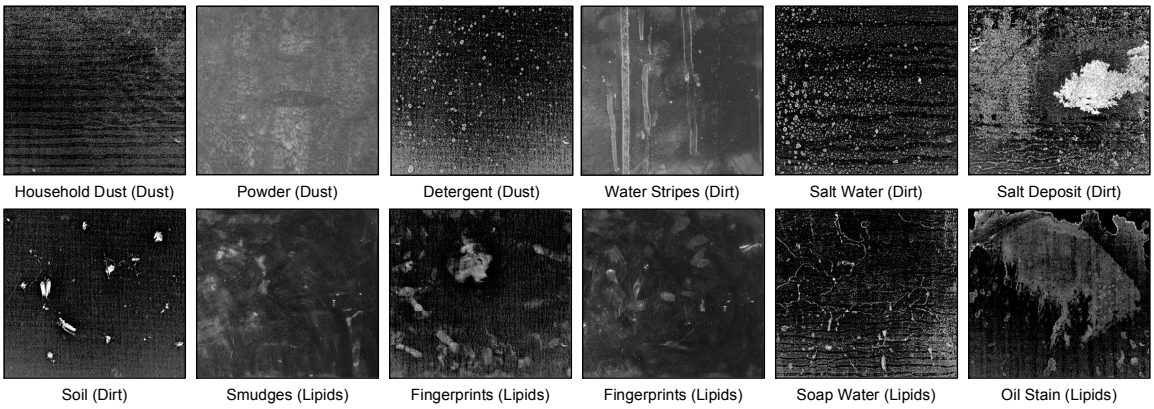


Figure 4.11: Examples of the acquired optical thickness patterns, measured with the setup in Figure 4.10(a). Intensity is proportional to $\tau(x, y)$.

lipids. The measured data has been released online at `http://www.cs.columbia.edu/CAVE`.

4.4.1 Measurement of Optical Thickness Pattern

To measure the optical thickness texture $\tau(x, y)$, we used the shadow map generated by attenuation from the contaminant layer. Figure 4.10(a) shows our setup. The projector illuminates a thin glass slab with contaminants on the far side. Behind it is a Lambertian board, and the camera is on the side. The camera is radiometrically calibrated beforehand. The intensity of each point in the shadow map comes from two parts: the attenuated transmitted light $L_t^{(1)}$ (Equation (4.5)) from this point, and the scattering component $L_t^{(2)}$

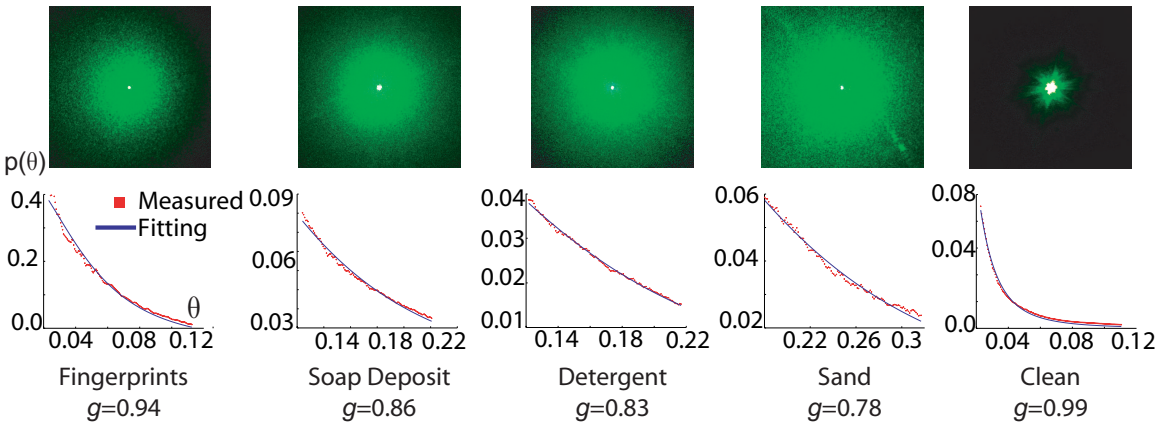


Figure 4.12: Examples of the measurements for estimating the Henyey-Greenstein parameter g. **Top:** acquired images. **Bottom:** the fitted phase functions.

(Equation (4.6)) from neighboring points. Since the albedo of the contaminant layer is assumed to be small (otherwise it would generate multiple scattering) and it is mostly forward scattering, there will be much less contribution from neighboring points due to scattering. Therefore, we have $L_t^{(1)} \gg L_t^{(2)}$ and thus the measured intensity of the shadow map, $I(x, y) \approx L_t^{(1)}$. Based on Equation (4.5), we now have $I(x, y) \propto e^{-\tau(x,y)}$ ($\cos\theta_{it} = 1$ since illumination is head on). Note that the attenuation is only related to $\tau(x, y)$ and is independent of g. The formula used to compute $\tau(x, y)$ is as follows:

$$\tau(x, y) = -\ln\frac{I(x, y)}{I_{\text{clean}}}, \tag{4.9}$$

where I_{clean} is the image intensity in the clean regions of the glass. Figure 4.11 shows some acquired textures with different kinds of contaminants. Image intensity is proportional to the optical thickness.

4.4.2 Measurement of Henyey-Greenstein Parameter

We developed a rather simple image-based method to measure the Henyey-Greenstein parameter g for various kinds of contaminants based on the derived BRDF/BTDF model, which requires only a single image and can be easily implemented. We aim to obtain qualitatively correct estimates of g for different materials.

The setup, shown in Figure 4.10(b), is similar to the one for measuring $\tau(x, y)$, except

in two places. First, we use a laser instead of a projector. Second, we focus on the light scattered from a small, uniform patch. Using the above method, we first measure the optical thickness τ of this patch. When the laser beam strikes the contaminants, part of it will scatter toward the Lambertian board and generate a falloff pattern on the board. The center region will have a very strong spike due to the attenuated laser, while the brightness of other regions is due to the scattering $L_t^{(2)}$. Based on Equation (4.6), the measured intensity for a point (x, y) is

$$I(x, y) = \beta \cdot p(\theta; g) \cdot \frac{e^{-\tau} - e^{-\tau/\cos\theta}}{1 - \cos\theta} \cdot \cos^4\theta, \tag{4.10}$$

where β is a scale factor, $p(\theta; g)$ is the phase function, and θ is the angle of the scattered ray from the normal. The derivation of this formula can be found in Appendix B.2.

We measured 10 samples in total which contain representative contaminants. For other samples, we used the g values from the corresponding category for rendering. Figure 4.12 shows some images for different kinds of contaminants. We also show the acquired image for clean glass where as expected the scattering is minimal. With the assumption of isotropic scattering, each circle around the center of the image corresponds to a single angle θ. We then compute the average $I(x, y)$ along the circle, and use it to fit the phase function. The second row in Figure 4.12 shows the fitting results. The third row shows the estimated g values.

4.4.3 Synthesis of Contamination Patterns from Captured Data

A simple tool was developed to synthesize novel contamination patterns from the captured data according to users' input. The synthesized texture, in which each channel corresponds to one kind of parameter of the contaminant, can be mapped on arbitrary transparent objects for rendering the weathered appearance.

We first extracted a collection of prototypes for each of the three types of contaminants from the measured raw data. In addition, to increase the variation of the atomic patterns, we also synthesized stain and dirt patterns using fractal Brownian motion [Barnsley et al., 1988], similarly as the synthesis of coffee stain in [Becket and Badler, 1990].

For synthesis, as shown in Figure 4.13, a user selects prototypes from each category for the desired contamination pattern and specifies the size of the output texture. The tool

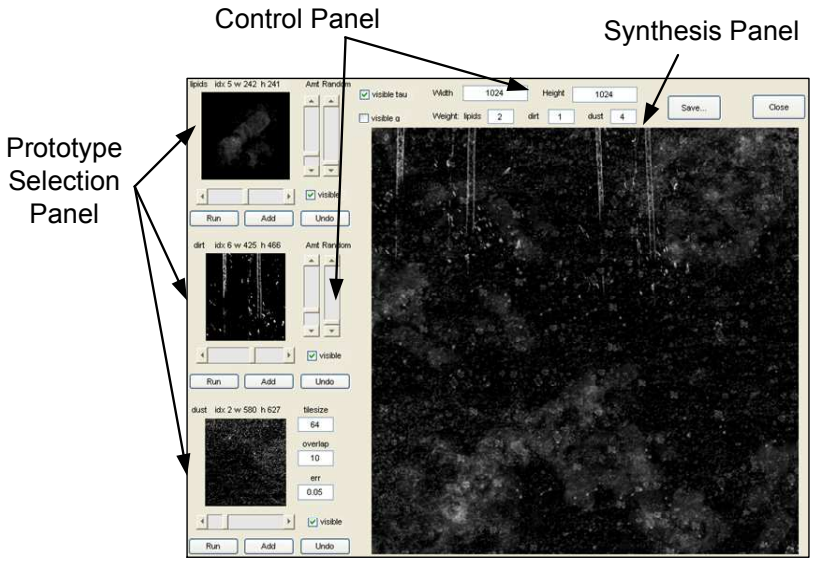


Figure 4.13: A synthesis tool to create contamination patterns from captured data.

then synthesizes a contamination layer for each prototype individually. For dust, we use image quilting [Efros and Freeman, 2001] for synthesis. For lipids, we randomly transform (scale and rotate) the prototype and randomly tile several copies on the canvas. For dirt, we use a similar random placement but we limit the range of rotation and translation since the water stripes usually are on the top of an image. The three synthesized layers are then alpha-blended together to generate the final texture. A user can control the amount and the range of transformation for each prototype and also add/remove more prototypes interactively.

We can also create time-varying contamination patterns to simulate a weathering process. Figure 4.15 shows an example. For rendering a window getting dirty over time, the time-varying contamination patterns are synthesized as a stochastic process. The accumulation of contaminants is assumed to be a Poisson process, *i.e.*, the number of *atomic* contaminant patterns appeared on the window during a time interval follows the Poisson distribution. For dust, the atomic patterns are randomly selected points on the window. For dirt and lipids, the atomic patterns are randomly selected from the above collection.

Figure 4.14: A glass sphere rendered with different thicknesses of a uniform layer of dust. As the optical thickness increases (from left to right), both the transmission and the reflection become smoother and give the sphere a more velvety appearance, especially near the boundary.

4.5 Rendering

We now show some rendering results of transparent objects with various kinds of contamination both in 3D scenes and on 2D photographs. In all the examples, sampling of the BRDF/BTDF models is done by sampling the Henyey-Greenstein phase function. The parameters of the contaminant layer are generated with the synthesis tool.

4.5.1 Rendering of 3D Scenes

Dusty Glass Sphere: Figure 4.14 shows a glass sphere covered with a uniform dust layer in a complex lighting environment while the optical thickness of the dust, τ, increases. As shown, the appearance of the glass sphere changes from completely transparent to semi-transparent and finally will become an opaque dusty surface. As τ increases, the sphere becomes smoother and softer, especially near the contour. This is consistent with the effects of dusty opaque surfaces since our model subsumes Blinn's dust model. However, we also capture the transmission effects of the dust layer on the glass sphere.

Dirt on a Glass Window: Figure 4.15 shows the time-varying weathered appearance of a window of a room. The accumulation of contaminants on the window are simulated as Poisson processes. The top row in Figure 4.15 shows the synthesized patterns at different times, and the middle row and the bottom row show the corresponding rendered images

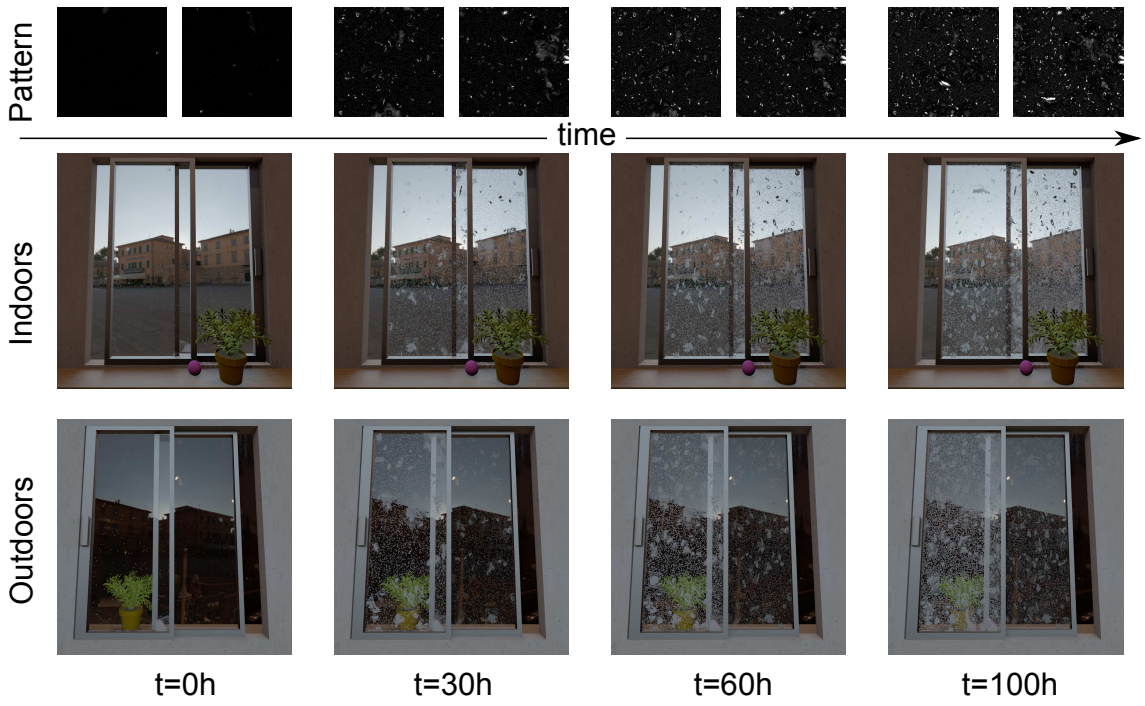
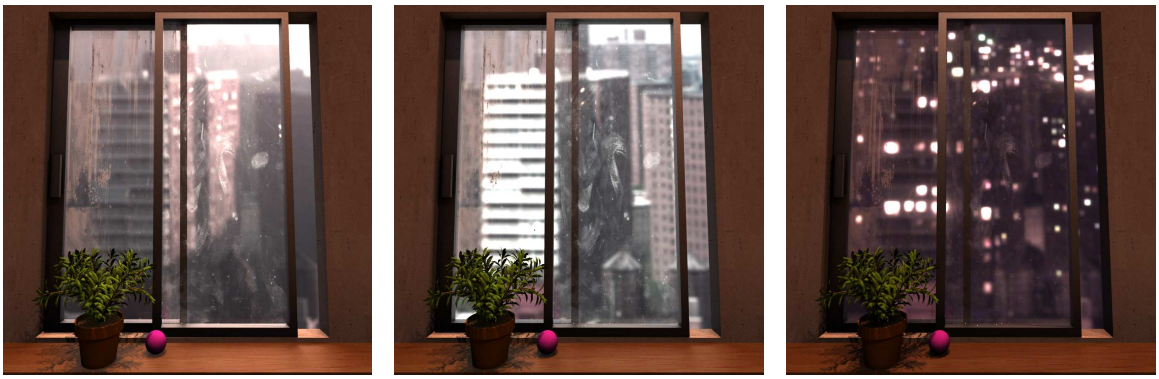


Figure 4.15: A contaminated window rendered with a background that changes with time of day. **Top**: the dirt pattern over time. **Middle**: the rendered scene viewed from indoors. **Bottom**: the rendered scene viewed from outdoors.



(a) Morning (6:00 AM) (b) Afternoon (2:00 PM) (c) Night(1:00 AM)

Figure 4.16: A contaminated window rendered with a background that changes with time of day. Note how the contaminants appear very differently for the different illuminations of the background.

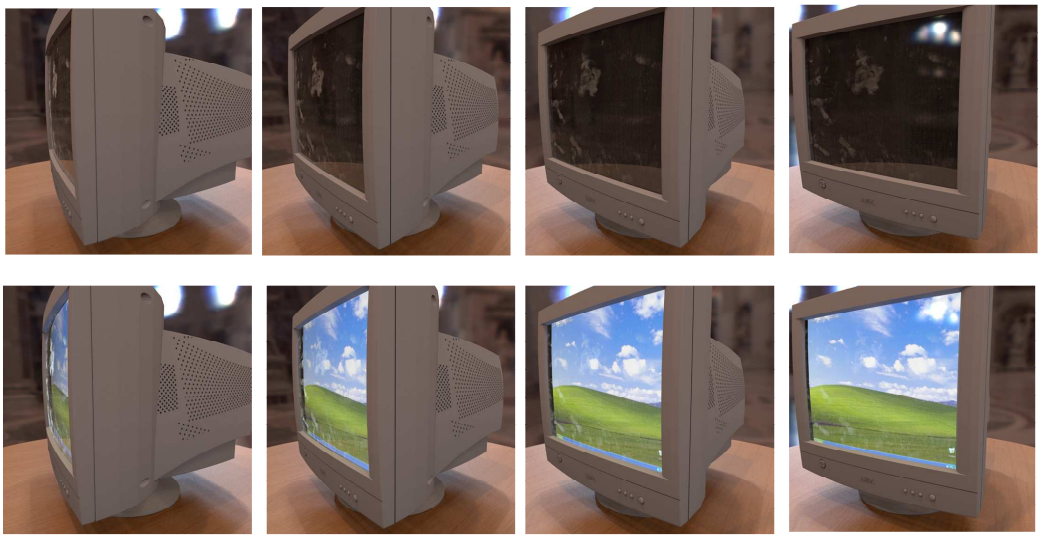


Figure 4.17: A monitor screen rendered with dust and fingerprints. The contaminants become more clearly visible when the monitor is off. Their brightness increases as the viewing angle approaches the grazing angle.

viewed from inside and outside of the room.

Figure 4.16 shows a view through the window while the outside scene (illumination) is changing over time. The outside scene is from the Weather and Illumination Database [Narasimhan et al., 2002] which records time-lapse images for natural scenes. Inside the room, we have a fixed environment light. The contamination pattern is composed of dried water streaks (dirt) in the left of the window, smudges and fingerprints (lipids) in the right, and a very thin layer of dust. As shown, against the sky the contaminant pattern looks darker because of attenuation; while against the building (or in the shadow of the building), the contamination pattern looks brighter because of scattering. Moreover, when viewing the window at different times of day, with different types of illumination, the contaminant pattern generates various effects, as predicted by our model in Figure 4.9.

Monitor On/Off: Figure 4.17 shows a monitor screen with contaminant. As predicted in Section 4.3, the contaminant pattern is much less visible when the monitor is turned on. Moreover, if the view or the illumination moves closer to the grazing angle, owing to the Fresnel effect and also the resulting longer path for photons to travel through the

Figure 4.18: A cognac glass rendered with different thicknesses of a uniform dust layer (increasing from left to right). The caustics and the base of the glass become dimmer and smoother as the dust thickness increases.

contaminant layer, the amount of reflected light will increase and and thus the contaminant pattern becomes more prominent, which is consistent with our experience.

Dirts on a Cognac Glass: Figure 4.18 shows dust accumulating on a cognac glass. We render the results with photon mapping [Jensen, 2001]. The body of the cognac glass has a similar appearance as the glass sphere example, while the base of the cognac glass and the caustics show interesting changes due to the scattering and the attenuation of the contaminant.

Figure 4.19 show the rendering of the cognac glass covered with a stain pattern (synthesized using the method in Section 4.4.3) with and without scattering. Notice the effects produced by the scattering of the contaminant, including the vertical "shadows" on the glass, the contrast reversal of the stain against the background, and the increased scattering at grazing angles, which are important for the realism of the rendering.

4.5.2 Composition of 2D Photographs

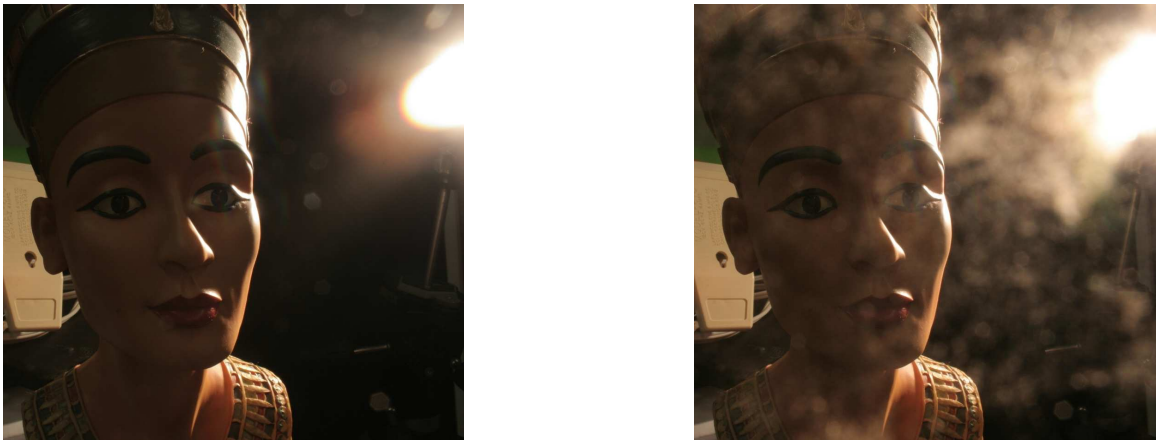
In addition to rendering 3D scenes, the derived models can be used as an *effects filter* to add the weathered appearance of transparent surfaces to 2D photographs.

Dust Scattering on a Camera Lens: Contaminants on camera lenses will scatter lights and generate effects such as lens flare from stray lights and blurring, as seen in

(a) With scattering

(b) Without scattering

Figure 4.19: A cognac glass rendered with stains. The glass is rendered (a) with and (b) without scattering from the stains. As shown, the scattering effects, embedded in the derived BRDF and BTDF models, are essential for the realism of the rendering.



(a)

(b)

Figure 4.20: Compared with the photograph taken with a dirty lens (Figure 4.1(g)), our model can be used to modify (a) a photograph of the scene taken with a clean camera lens to create the effects of (b) an image as if it were taken with a dirty camera lens.

Figure 4.21: Adding weathered appearance to the clean windows in real photographs: (a) Original photographs. (b) Synthesized contamination patterns. (c) Composed images.

Figure 4.1(g). Based on the derived model, we can use an image taken with a clean lens, such as Figure 4.20(a), and composite it as if it were taken with a dirty lens, as shown in Figure 4.20(b). The texture of the contamination pattern is created to be similar to that in Figure 4.1(g). While there is probably little chance that such composition is desirable in practice, this example is mainly to demonstrate that the derived BRDF/BTDF model is capable of accurately reproducing one type of appearance phenomena. A more interesting and meaningful problem is how to remove these dirty-lens image artifacts from captured photographs, which will be the topic of the next chapter.

Contaminating Clean Transparent Surfaces: Figure 4.21 show two examples where our goal is to make the window regions in the input photographs dirty. We first estimate the 3D positions of the window plane and camera from the four corner points of the window.

The outside illumination is assumed to be the image intensity in the window region which is synthesized and mapped onto a hemisphere. Using the derived BRDF and BTDF, we compute the image intensity in the regions of the window with contamination. In Figure 4.21, (a) are the photographs through clean windows, (b) are the synthesized contamination patterns, and (c) are the final composed images where the top image shows the composition as if the image were taken through a glass that had been sprayed by sea water and the bottom image shows the contrast reversal effects for dirty windows in real world, such as the brighter appearance on the bottom right and the darker appearance in the top regions.

4.6 Summary and Discussion

Contamination (such as dust, dirt, and lipids) often accumulates on transparent surfaces over time, thereby scattering light and creating the weathered appearance that is important for the rendering of transparent objects. In this chapter, we presented an efficient way to synthesize this weathered appearance. By modeling the contaminant as an optically-thin layer, we constructed an analytic BRDF/BTDF model that accurately captures most of the canonical visual effects and allows efficient rendering. We also developed single-image based methods to measure contamination patterns from real samples. We show that our model and the measurements can be used to synthesize realistic weathered appearance on 3D scenes and 2D photographs.

One future work is to extend our model to the rendering of contaminated translucent materials, such as lipids on skin, dust on marble, and dirt on transparent vessels holding participating media (*e.g.*, wine and various other liquids) where subsurface scattering needs to be considered.

Chapter 5

Removing Image Artifacts Due to Dirty Camera Lenses

In Chapter 4, we derived an analytic BRDF/BTDF model for reproducing weathered appearance for transparent surfaces. While imperfection is desirable in computer graphics, for many computer vision and imaging tasks, we often want to remove these artifacts from photographs. In this chapter, we studied one type of images artifacts that are commonly seen in digital photography — the artifacts caused by dirt or dust on camera lenses, and proposed two automatic methods to remove the image artifacts.

5.1 Introduction

The lenses of consumer digital cameras or telescopes, or the front windows of security cameras, often accumulate various types of contaminants over time (*e.g.*, fingerprints, dust, dirt). These contaminants attenuate incoming light and scatter light from other directions, which often results in artifacts in the captured image, as shown in Figure 4.1(g). The artifacts are an annoyance for photographers, and also damage important scene information for applications in computer vision or digital forensics.

Certainly, a simple solution is to clean the camera lens. However, this is impossible for existing images and videos, and impractical for some automated imaging systems, such as outdoor security cameras, remote sensing (*e.g.*, Mars Exploration Rover [Willson et al.,

2005]), astrophotography [Romanishin, 2006], or covert surveillance behind a fence. Therefore, it is necessary to develop computational algorithms to remove the dirty-lens artifacts from captured images.

Previous work on the removal of imperfections from images can roughly be grouped into two categories. The first category is *image in-painting*, where texture synthesis techniques are used to *fill holes* from neighboring regions [Bertalmio et al., 2000; Efros and Freeman, 2001; Sun et al., 2005; Liu et al., 2008]. These methods often require some user interaction to label the affected regions and synthesize the affected regions from their neighborhood, but they do not need know the image formation of the artifacts.

The second category is the so-called *physically-based* methods, which first understands and models the image formation of the artifacts and then solves for the original scene. Physically-based methods are generally expected to recover more faithful scene details, given that the image formation models are sufficiently accurate. Extensive work has been performed in this area, such as recovering images taken in bad weather [Narasimhan and Nayar, 2003b], dehazing [Schechner et al., 2001; Fattal, 2008; He et al., 2009], removing rain [Garg and Nayar, 2004], and underwater imaging [Schechner and Karpel, 2005].

For removing the artifacts caused by defective optic systems (*e.g.*, a camera), Talvala et al. [2007] studied camera-veiling glare which is a mostly uniform loss-of-contrast in the image and is caused by the inter-reflection among lenses. Raskar et al. [2008] proposed to use a light field camera for removing lens glare. Willson et al. [2005] studied lens dirt for simulating the artifacts. In astrophotography, for removing the artifacts caused by lens dirt, a standard procedure is to take flat field frames [Romanishin, 2006] and divide captured images with the flat field frames. Zhou and Lin [2007] proposed interactive methods to remove artifacts caused by the attenuation of sensor dust. Both methods considered only the attenuation of light caused by lens dust but not the scattering.

Our work follows the physically-based approach and seeks to recover the original scene on a point-wise basis without user interaction. This is possible because for lens dirt the artifacts are seriously defocused, and thus the original scene is (partially) visible in the captured image (except where very thick lens dirt completely blocks the light). While we draw inspiration from the related work listed above, we focus on different visual effects

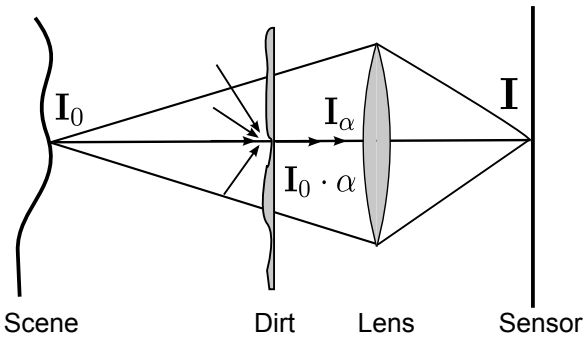


Figure 5.1: Image formation model for a dirty-lens camera: $\alpha \in [0, 1]$ is the attenuation pattern (0 = completely blocked). The captured image $I(x)$ includes the attenuated $I_0 \cdot \alpha$ and the scattered light I_α.

and consider both the attenuation and scattering effects caused by lens dirt. Moreover, physically-based methods and image in-painting methods are not mutually exclusive — the former can be used where the scene is partially visible, after which the latter can be used on the remaining completely-blocked regions.

5.2 Image Formation Model

Lens dirt can be modeled as an intermediate layer between the target scene and the camera. The captured image $I(x)$ consists of two components — (1) **attenuation** where scene radiance $I_0(x)$ is attenuated by the dirt; (2) **intensification** where the dirt scatters stray lights from the environment (*e.g.*, the sun) into the camera. Intuitively, the attenuation darkens the captured image while the intensification brightens it[1].

As shown in Figure 5.1, suppose $\alpha(x) \in [0, 1]$ is the attenuation pattern of the lens dirt, which describes how much light remains after traveling through the dirt layer ($\alpha(x) = 0$ means completely blocked), and $I_\alpha(x)$ is the intensification pattern (*i.e.*, the scattered radiance from the dirt). Assuming the camera is focused on the target scene, we have the following image formation model:

$$I(x) = I_0(x) \cdot (\alpha(x) * k(x)) + I_\alpha(x) * k(x), \tag{5.1}$$

[1]The darkening and brightening effects have been discussed in [Willson et al., 2005; Gu et al., 2007].

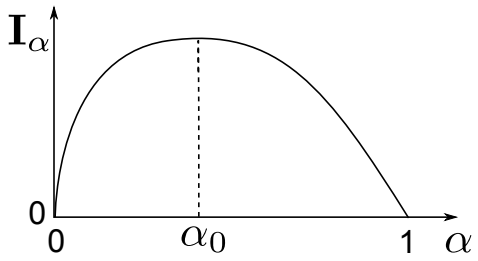


Figure 5.2: A schematic diagram showing the relationship between the attenuation pattern $\alpha(x)$ and the intensification pattern $I_\alpha(x)$.

where $k(x)$ is the defocus blur kernel for the dirt layer, and $*$ denotes convolution. There are several important features of Equation (5.1):

Point-wise multiplication preserves high frequencies: The term $I_0(x)\cdot(\alpha(x) * k(x))$ involves a point-wise multiplication of the original image $I_0(x)$ and the attenuation $\alpha(x) * k(x)$. This means that as long as $\alpha(x) * k(x) > 0$, all the high-frequency components in $I_0(x)$ will still be (partially) preserved in the captured image $I(x)$. It is this property that allows us to recover the original scene in a point-wise manner.

On the other hand, this property also shows a major limitation of our methods: in regions where $\alpha(x) * k(x) = 0$, no information on the original scene is captured in $I(x)$, and we have to revert to neighboring pixels for synthesizing $I_0(x)$. Fortunately, this is usually not a problem for dirty-lens artifacts — the defocus blur for lens dirt is so large that the artifacts are always highly blurred in captured images.

Relationship between $\alpha(x)$ and $I_\alpha(x)$: Both the attenuation pattern $\alpha(x)$ and the intensification pattern $I_\alpha(x)$ are caused by the same medium — the lens dirt. Therefore, they are not independent variables. Their relationship is governed by the radiative transfer equation (Equation (2.17) on page 17). Qualitatively, their relationship can be shown as the schematic diagram in Figure 5.2 — wherever the lens is clean, no light is blocked (*i.e.*, $\alpha = 1$), nor is there any intensification (*i.e.*, $I_\alpha = 0$); as the dirt increases, it begins to attenuate more light (*i.e.*, α decreases) but it will also scatter more stray light (*i.e.*, I_α increases); when the dirt layer is thick enough (*i.e.*, $\alpha < \alpha_0$), the amount of the scattered light reaches its peak value and starts to decrease because of absorption; when the dirt

becomes too thick and completely blocks the light (*i.e.*, $\alpha = 0$), the scattered light will also diminish (*i.e.*, $I_\alpha = 0$).

Although it is intuitive, this nonlinear relationship between $\alpha(x)$ and $I_\alpha(x)$ is often difficult to obtain for image enhancement, since it requires knowing the material properties of the lens dirt and camera settings. In Section 5.3, we show that under certain assumptions this relationship can be simplified, which can give us a simpler image formation model for image enhancement.

Solving for $I_0(x)$ with Multiple Images: Equation (5.1) shows that recovering the artifact-free image $I_0(x)$ from a single input image is ill-posed, because both $\alpha(x)$ and $I_\alpha(x)$ are unknown in general.

We demonstrate two methods to estimate $\alpha(x)$ and $I_\alpha(x)$ — (1) If we have access to the camera, the attenuation pattern and the intensification pattern of the lens dirt can be directly measured from multiple pictures (≥ 2) of a stripe pattern (Section 5.4); (2) For situations where we do not have access to the camera, we show that $\alpha(x)$ and $I_\alpha(x)$ can be estimated from multiple images taken with the same dirty lens camera, such as a video (Section 5.5). Once $\alpha(x)$ and $I_\alpha(x)$ of the lens dirt are known, we show that dirty-lens artifacts can be removed from individual photographs by enforcing the sparsity on the recovered images' gradients — one well known prior on natural images.

5.3 Model Simplification and Validation

In this section, we make a few assumptions about lens dirt and camera settings in order to simplify the image formation shown in Equation (5.1). The simplification is then validated with experimental results.

Model Simplification: Our first assumption is that lens dirt usually is an *optically thin* layer of participating media (see Section 2.1.2 on page 15). Therefore, the BRDF/BTDF model derived in Chapter 4 for single scattering can be used. For the attenuation pattern $\alpha(x)$, it is caused by the absorption of lens dirt and it corresponds to $L_t^{(1)}$ derived in

Section 4.2.2 (Equation (4.5) on page 60), based on which we have

$$\alpha(x) = k_1 \cdot \exp\left(-\tau(x)/\cos\theta(x)\right), \tag{5.2}$$

where $\tau(x)$ is the optical thickness of the lens dirt at pixel x, $\theta(x)$ is the angle between the camera optical axis and the ray from the optical center to pixel x, and k_1 is some constant.

The intensification $I_\alpha(x)$ is caused by the scattering of lens dirt, *i.e.*, the dirt will "gather" light from the environment. Therefore, $I_\alpha(x)$ is an integral of scattered light over the entire outside illumination:

$$I_\alpha(\omega_o) = \int_{\omega_i} f_t\left(\omega_i, \omega_o; \tau, W_0, g, n\right) \cdot L_i(\omega_i) \cos\theta_i d\omega_i, \tag{5.3}$$

where $L_i(\omega_i)$ is the illumination from the direction ω_i, $f_t(\cdot)$ represents how much light is scattered from the direction ω_i to the outgoing direction ω_o. W_0, g, and n are the albedo, the Henyey-Greenstein parameter, and the refractive index of the lens dirt. They are the material properties of the lens dirt, which are less likely to be spatially-varying over a camera lens. Therefore, we treat W_0, g, and n as constants for the lens dirt.

Since the scattered light $I_\alpha(x)$ corresponds to $L_t^{(2)}$ (Equation (4.6) on page 60), the function $f_t(\cdot)$ can thus be approximately simplified as

$$f_t\left(\omega_i, \omega_o; \tau\right) \approx k_2 \cdot \frac{\tau}{\cos\theta_o \cos\theta_i} \cdot p(\omega_i, \omega_o), \tag{5.4}$$

where $p(\cdot)$ is the phase function and k_2 is a scale factor determined by W_0 and n. This simplification is achieved because for small τ we have $e^{-\tau} \approx 1 - \tau$. We also note that $\theta_o = \theta(x)$.

Our second assumption is isotropic scattering (*i.e.*, $p(\cdot) = \frac{1}{4\pi}$). Moreover, we assume the field of view of the camera is narrow(*i.e.*, $\theta(x) \approx 0$). With these assumptions, $f_t(\cdot)$ now is only a function of the optical thickness pattern $\tau(\cdot)$. Thus, we have

$$I_\alpha(x) \approx f_t(\tau(x)) \cdot \int_{\omega_i} L_i(\omega_i) d\omega_i = f_t(\tau(x)) \cdot c, \tag{5.5}$$

where

$$c = \int_{\omega_i} L_i(\omega_i) d\omega_i, \tag{5.6}$$

is the aggregate of the outside illumination. The narrow field-of-view assumption can also be used to simplify $\alpha(x)$ (Equation (5.2)) as $\alpha(x) \approx k_1 \exp(-\tau(x))$. Finally, since α is a function of the optical thickness τ, $f_t(\tau)$ can be written instead as a function of α:

$$I_\alpha(x) = c \cdot f(\alpha(x)), \tag{5.7}$$

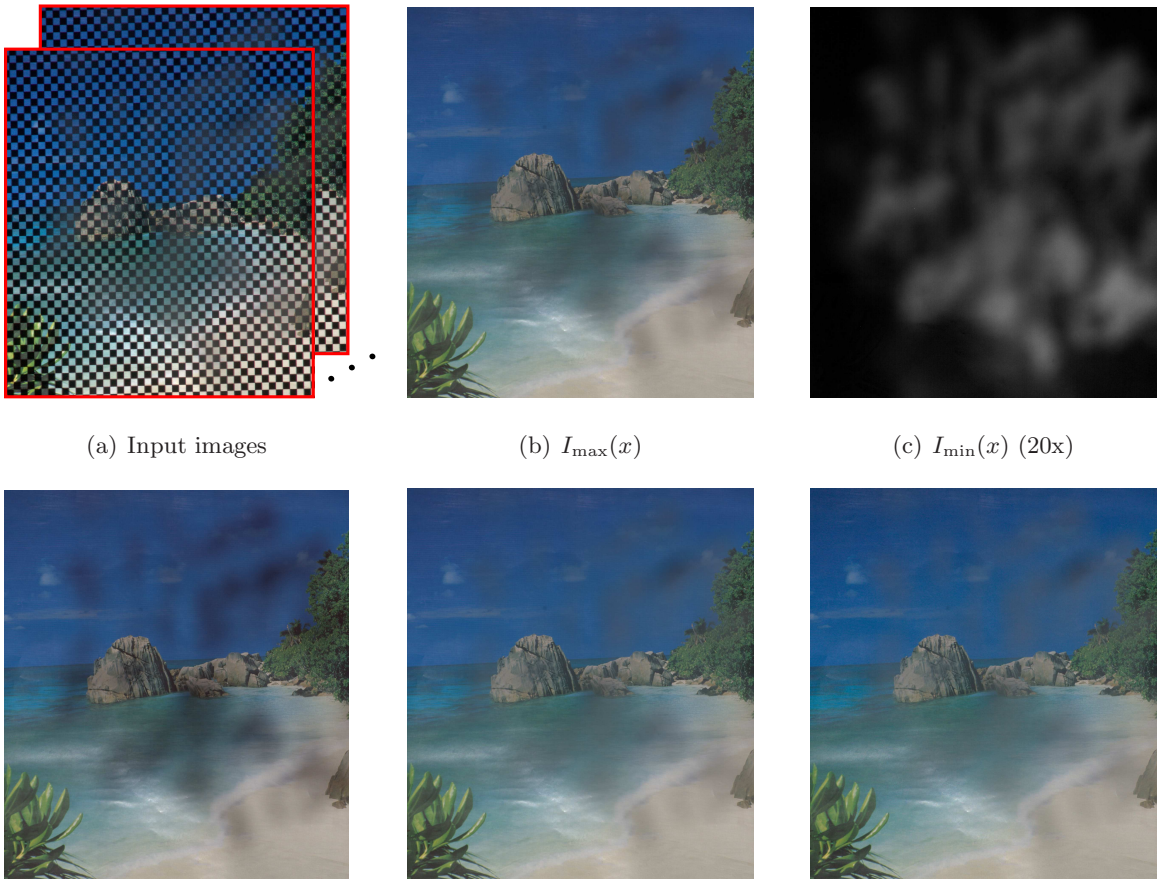
where we emphasize that both $I_\alpha(x)$ and $\alpha(x)$ are spatially-varying while c is a fixed vector. The function $f(\alpha(x))$ depends on the physical characteristics of the contaminants.

This relationship between $I_\alpha(x)$ and $\alpha(x)$ is important, because it indicates that only the aggregate of the outside illumination c is relevant for the intensification. In other words, the intensification $I_\alpha(x)$ is a global effect of the outside illumination, which is not directly related to the artifact-free image $I_0(x)$.

Validation Experiment: We validated the simplified model in Equation (5.7) by separating the attenuation and scattering components using an approach inspired by Nayar et al. [2006]. As shown in Figure 5.3, we project a sequence of slightly shifted checkerboard patterns and take pictures of the checkerboard modulated scene with a dirty lens camera. Scene points in black squares do not themselves emit light and thus their corresponding pixel intensities are caused by the scattering due to lens dirt. In contrast, pixel intensities of scene points in white squares include both the light emitted by themselves (after attenuation due to lens dirt), and the light scattered by the lens dirt. The amount of scattered light is fixed since the *integral* of the outside illumination is unchanged. Therefore, letting $I_{\max}(x)$ and $I_{\min}(x)$ denote the point-wise maximum and minimum over all the pictures, we have

$$I_{\max}(x) = I_0 \cdot (\alpha(x) * k(x)) + I_\alpha(x) * k(x) \qquad \text{and} \qquad I_{\min}(x) = I_\alpha(x) * k(x).$$

Their difference, $I_{\max}(x) - I_{\min}(x)$, should equal the attenuation pattern $\alpha(x)$ (up to a scale I_0). Figure 5.3 shows the separation results for these components, respectively. As can be seen, the intensification term, $I_{\min}(x)$, does not relate to the background scene. Moreover, according to the simplified model, $I_{\max}(x) + I_{\min}(x)$ should equal a fully-illuminated image (*i.e.*, an image captured when we project a white pattern). This is also verified and shown



(a) Input images (b) $I_{\max}(x)$ (c) $I_{\min}(x)$ (20x)

(d) $I_{\max}(x) - I_{\min}(x)$ (e) $I_{\max}(x) + I_{\min}(x)$ (f) Fully-illuminated image

Figure 5.3: Experimental validation of Equation (5.7). (a) A sequence of shifted checkerboard patterns are projected on a scene. (b) The point-wise maximum of the captured images, $I_{\max}(x)$, includes both the attenuation and the scattering. (c) The minimum of the captured images (amplified 20 times for demonstration), $I_{\min}$(x), directly measures the scattering of the lens dirt. (d) The attenuation can be simply computed as $I_{\max}(x) - I_{\min}(x)$. As shown in (c), the scattering is related only to the attenuation pattern, and not the background scene. (e) shows $I_{\max}(x) + I_{\min}(x)$, and (f) is the image captured when we project a white pattern on the scene. (e) should equal to (f) because the checkerboard patterns turn on half the projector pixels and thus the scattering in (c) is half of the scattering in (f) while the attenuation keeps the same. Indeed, we found (e) and (f) are closely matched with a mean absolute percentage error 0.6%.

in Figure 5.3(d)(f) with a mean absolute percentage error of 0.6%. Therefore, our simplified model is validated: only the aggregate of the outside illumination is relevant to the intensification, *i.e.*, $I_\alpha(x) = c \cdot f(\alpha(x))$.

Finally, we note that the defocus blur kernel $k(x)$ can be assumed to be fixed for a given dirty lens camera, since the distance from the contaminant layer to the optical center is usually fixed, and is much smaller than the distance from the scene to the optical center. Thus we can further simplify the model in Equation (5.1) by defining two variables — the attenuation map $a(x) := \alpha(x) * k(x)$ and the intensification map $b(x) := f(\alpha(x)) * k(x)$. The model can then be rewritten as:

$$I(x) = I_0(x) \cdot a(x) + c \cdot b(x), \tag{5.8}$$

where $a(x)$ and $b(x)$ are the characteristics for a given camera and will be independent of the scene, and c represents the aggregate of the outside illumination and is scene dependent.

Below, we propose two methods to estimate $a(x)$ and $b(x)$ for a given dirty-lens camera — we only need to do this once, after which the image artifacts can be removed from a single image taken with this camera.

5.4 Artifact Removal with Calibration

Suppose we have the access to the dirty lens camera. For example, we realize the photographs have artifacts afterwards but we still have the camera at hand (*e.g.*, consumer digital camera). Or, for some imaging systems (*e.g.*, telescopes), the camera can be controlled. In this case, we can use the same idea in the validation experiment, and directly measure $a(x)$ and $b(x)$ by taking a set of pictures of some structured pattern (*e.g.*, a checkerboard) at different positions

$$a(x) = I_{\max}(x) - I_{\min}(x) \qquad \text{and} \qquad b(x) = I_{\min}(x), \tag{5.9}$$

where $I_{\max}(x)$ and $I_{\min}(x)$ are the point-wise maximum and minimum over all the pictures. Note that the estimate $b(x)$ in Equation (5.9) has been scaled with the aggregate of the outside illumination of the calibration scene which is a constant. Ideally, if the black and

white pixels can be swapped exactly, only two pictures are needed. In practice, more pictures can be taken to suppress noise around edges.

In order to remove artifacts from a new image, we need to estimate the aggregate of the outside illumination c for the new scene. In fact, this is a fundamental issue — due to the limitation of the field of view of the camera, only a (small) portion of the environment light is captured in the input image. In other words, some information about the outside lighting conditions, especially stray lights such as the sun, is missing in the input. Biases in the estimation of c will often over-compensate the dirty-lens artifacts or do not completely remove the artifacts, as shown in Figure 5.9(b).

To solve this problem, we incorporate some prior on natural images for estimating c. Natural images are well known to have strong sparsity in their gradients [Rudin et al., 1992; Levin et al., 2007]. Since the dirty-lens artifacts are low frequency (because of camera defocus), if the artifacts were not completely removed, the recovered image would have retained the low frequency pattern, resulting in a non-sparse image gradient. Therefore, we can estimate c by enforcing the sparsity of the recovered image's gradient (*i.e.*, minimizing its ℓ-1 norm):

$$c^* = \arg\min_c \|\nabla I_0(x)\|_1, \tag{5.10}$$

where

$$I_0(x) = (I(x) - c \cdot b(x))/a(x). \tag{5.11}$$

We first perform a simulation to verify the accuracy of the proposed method. As shown in Figure 5.4, we generate a synthetic image with a dirty-lens artifact using the lens dirt pattern extracted in Figure 5.3(c). The aggregate of the outside illumination is set to be $c = 0.80$. Equation (5.10) is used to estimate c^* and recover the artifact-free image $I_0(x)$, as shown in Figure 5.4(b). The mean absolute percentage error of the recovered image is 0.44%. The plot in Figure 5.4(c) shows $\|\nabla I_0(x)\|_1$ at different c. The optimal value of c is estimated to be 0.802, an error of only 0.25%.

Figure 5.5 and Figure 5.6 show experimental results on real images taken with a Canon EOS 20D camera. The camera is equipped with a Canon EF 50mm f/1.8 lens contaminated with house dust powder and fingerprints (shown in the bottom right of Figure 5.5), thus introducing significant artifacts in the captured images, as shown in Figure 5.6(a). To

(a) Input image (b) Recovered image (c) Estimate of c^*

Figure 5.4: Simulation results for estimating the aggregate of outside illumination c based on Equation (5.10). (a) A synthetic image with a dirty-lens artifact is generated using the lens dirt pattern extracted from Figure 5.3(c). (b) Equation (5.10) is used to recover an artifact-free image, with a mean absolute percentage error of 0.44% compared to ground truth. (c) A plot of $\|\nabla I_0(x)\|_1$ versus c. The optimal estimate, c^*, has a 0.25% error compared to the ground truth.

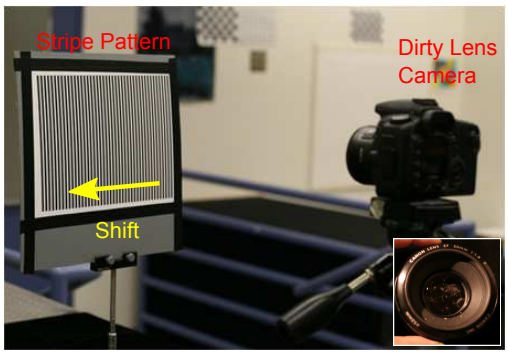
estimate the attenuation and scattering terms, we use a stripe pattern consisting of black and white vertical stripes printed on a piece of paper. Sixteen pictures are taken at randomly-shifted positions, as shown in Figure 5.5(b). The estimated attenuation map $a(x)$ and scattering map $b(x)$ are shown in Figure 5.5(c) and Figure 5.5(d), respectively.

Given this calibration, we use Equation (5.10) to estimate c and recover the artifact-free images. Four examples are given in Figure 5.6. As shown in the insets, the proposed method effectively removes the artifacts caused by the dirty camera lens and reveals more details in the original images. The method works well on photographs taken both indoors and outdoors, and across a large range of outside illuminations.

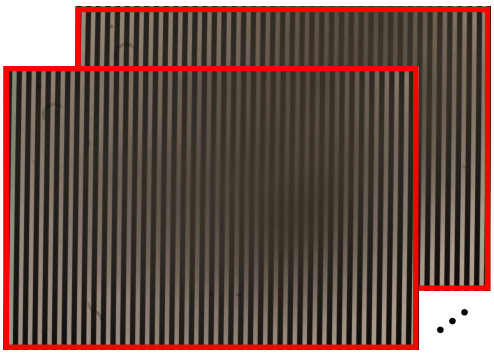
5.5 Artifact Removal without Calibration

In situations where we do not have the access to the camera to perform the above calibration (*e.g.*, for post-processing existing photographs or videos), we propose a method based on natural image statistics to estimate the attenuation map $a(x)$ and scattering map $b(x)$.

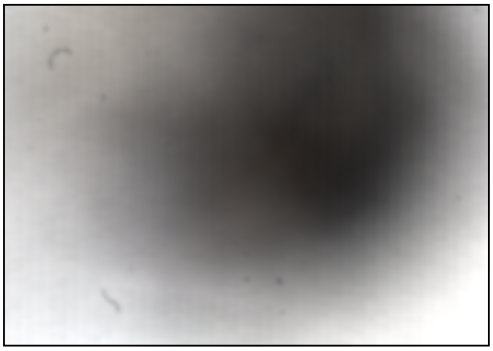
Let us first consider two neighboring pixels x_1 and x_2. Since both $a(x)$ and $b(x)$ are smoothly-varying due to the large camera defocus, we have $a(x_1) \approx a(x_2)$ and $b(x_1) \approx b(x_2)$,



(a) Calibration setup



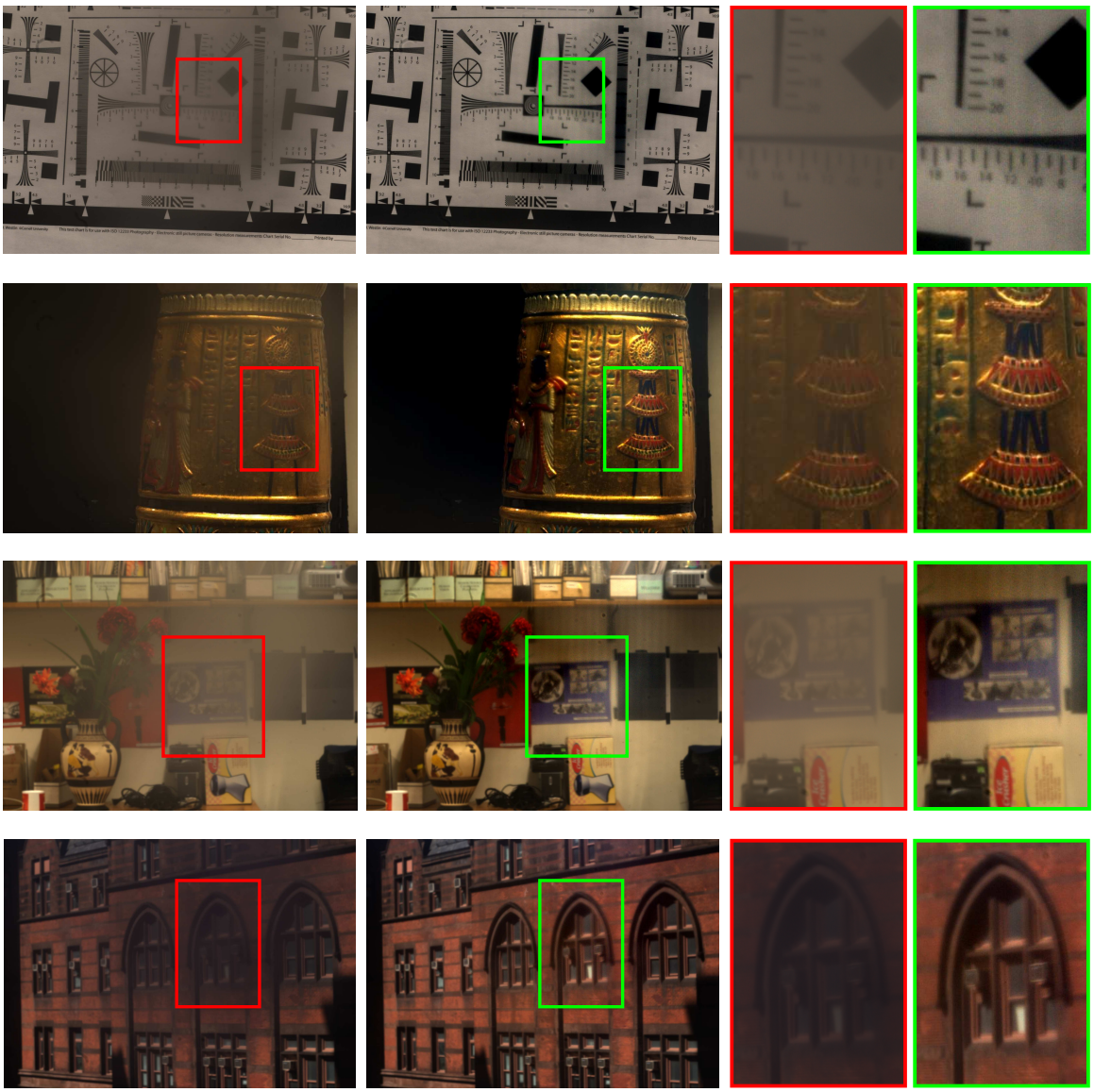
(b) Calibration images



(c) Attenuation map: $a(x)$

(d) Scattering map: $b(x)$

Figure 5.5: Calibration of a dirty-lens camera. (a) The dirty pattern on the lens can be measured by taking several pictures (≥ 2) of a stripe pattern, as shown in (b). With these calibration images, we can estimate (c) the attenuation map $a(x)$ and (d) the scattering map $b(x)$ for a dirty lens camera.



(a) Input images (b) Recovered images (c) Insets

Figure 5.6: Removal of the dirty-lens image artifacts with calibration. Given the attenuation map $a(x)$ and the scattering map $b(x)$ from the calibration, we can remove dirty-lens artifacts for each input image. (a) are four input images, and (b) are the recovered images. The optimal estimates of the aggregate outside illumination are $c^* = [1.37, 1.35, 1.41]$, $c^* = [1.19, 1.00, 0.74]$, $c^* = [1.82, 1.73, 1.58]$, and $c^* = [5.48, 5.41, 7.66]$, from top to bottom. (c) shows the insets of the input and recovered images.

and thus we have $I(x_1) - I(x_2) \approx (I_0(x_1) - I_0(x_2)) \cdot a(x_1)$. In other words, the magnitude of the image gradient has been attenuated by $a(x_1)$, which can be stated more formally as:

$$\begin{aligned}\nabla I(x) &= \nabla \left(I_0(x) \cdot a(x) + c \cdot b(x) \right) \\ &= (\nabla I_0(x)) \cdot a(x) + I_0(x) \cdot (\nabla a(x)) + c \cdot (\nabla b(x)) \\ &\approx (\nabla I_0(x)) \cdot a(x),\end{aligned}$$

since both $\nabla a(x) \approx 0$ and $\nabla b(x) \approx 0$ due to the large camera defocus. This relationship holds for every picture taken with the same dirty lens camera, and thus by computing the averaged magnitude of the image gradient over all frames of a video (or a collection of photographs), we have

$$\text{Avg}(|\nabla I(x)|) = \text{Avg}(|\nabla I_0(x)|) \cdot a(x), \tag{5.12}$$

where $\text{Avg}(\cdot)$ represents the averaging operation over multiple images. Similarly,

$$\text{Avg}(I(x)) = \text{Avg}(I_0(x)) \cdot a(x) + \bar{c} \cdot b(x), \tag{5.13}$$

where $\bar{c}$ is the averaged aggregate of the outside illumination over all frames and can be absorbed in the estimate of $b(x)$.

We now rely on natural image statistics to estimate both $\text{Avg}(I_0(x))$ and $\text{Avg}(|\nabla I_0(x)|)$. Since the probability distributions of image intensity and gradient for all pixels are similar to each other, the averaged values (*i.e.*, the expectations) for each of these pixels over a sequence of images should also be similar to each other. This has been shown extensively in previous studies [Burton and Moorhead, 1987; Torralba et al., 2008; Kuthirummal et al., 2008], where the averaged images are smoothly-varying. This heuristic enables us to estimate $\text{Avg}(I_0(x))$ and $\text{Avg}(|\nabla I_0(x)|)$ from $\text{Avg}(I(x))$ and $\text{Avg}(|\nabla I(x)|)$ via a simple iterative polynomial fitting.[2]

Specifically, we model $\text{Avg}(I_0(x))$ as a bivariate polynomial (x is a 2D variable representing the position of a pixel), $\sum_{i=0}^{3} \sum_{j=0}^{3} a_{i,j} u^i v^j$, where u, v are *normalized* pixel coordinates in $[-1, 1] \times [-1, 1]$. In the first iteration, we use all the pixels whose values are among

[2]For mounted surveillance cameras, the averaged images will not be smoothly-varying but will have certain structures of the scene, which can be used similarly as priors for the removal of lens dirt.

the top 50% of $\text{Avg}(I(x))$ for least-square fitting. We then take the difference between the fitted $\text{Avg}(I_0(x))$ and $\text{Avg}(I(x))$. Those pixels whose residuals are within a threshold (10% of its pixel value) are considered as "inliers" and used to perform least-square fitting in the next iteration. We found that this method converges in $100 \sim 200$ iterations and automatically finds the pixels in the dirt regions as "outliers". The same method is used for estimating $\text{Avg}(|\nabla I_0(x)|)$. Therefore, the attenuation map $a(x)$ and scattering map $b(x)$ can be computed from Equations (5.12) and (5.13):

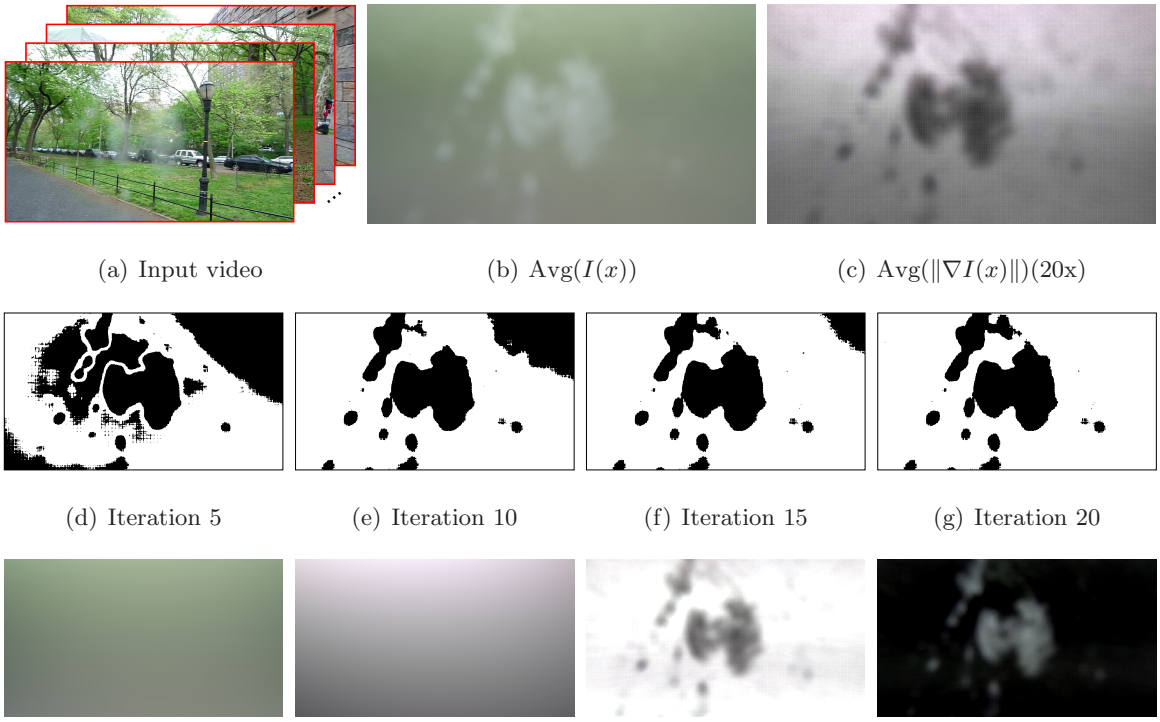
$$a(x) = \text{Avg}(|\nabla I(x)|)/\text{Avg}(|\nabla I_0(x)|) \quad \text{and} \quad b(x) = \text{Avg}(I(x)) - \text{Avg}(I_0(x)) \cdot a(x). \quad (5.14)$$

To remove artifacts for individual frames, we use the optimization shown in Equation (5.10) to estimate the aggregate of the outside illumination, c^*, independently for each frame.

Figure 5.7 and Figure 5.8 show some experimental results. A Panasonic Lumix DMC3 camcorder was used to take videos at 24fps inside a park on an overcast rainy day. The camcorder's lens is contaminated with fingerprints, dust, and rain drop deposit. A 5-minute clip is used for the computation, resulting in $5 \times 60 \times 24 = 7200$ frames. The image resolution is 1280×720. These frames are used to compute the averaged image and gradient magnitudes, as shown in Figures 5.7(b)(c), and to estimate the attenuation map $a(x)$ and scattering map $b(x)$, as shown in Figures 5.7(j)(k). Figure 5.8 shows several examples of artifact removal from images, where (a) and (c) show the original frames and (b) and (d) show the recovered images.

5.6 Further Post-Processing

For certain frames, the above algorithms sometimes introduce over-darkening and over-brightening artifacts in the dirt regions, as shown in Figure 5.9(b). There could be multiple causes — (1) Some of our assumptions might not hold for the scene, such as the dirt might not be optically-thin layer nor isotropic scattering, or the camera has a wide field of view. (2) The outside illumination might be extremely uneven. For example, some dirt regions might cover the bright sky while others cover the ground or bushes that are in shadow. Since many types of contaminants are forward scattering (*e.g.*, lipids, dermis, water droplets) [Ishimaru, 1978; Jacques et al., 1987], the unevenly distributed outside lighting



(a) Input video (b) Avg($I(x)$) (c) Avg($\|\nabla I(x)\|$)(20x)

(d) Iteration 5 (e) Iteration 10 (f) Iteration 15 (g) Iteration 20

(h) Avg($I_0(x)$) (i) Avg($\|\nabla I_0(x)\|$)(20x) (j) $a(x)$ (k) $b(x)$

Figure 5.7: Estimation of the attenuation map $a(x)$ and the scattering map $b(x)$ from a video taken with a dirty lens camera. (a) The input is a 5-min long video clip, consisting of 7200 frames. (b) The averaged image over all the frames. (c) The averaged image gradient (amplified 20 times for demonstration) over all the frames. (d)-(g) show the intermediate results of the iterative polynomial fitting where the black pixels in the images are the "outliers" corresponding to the "dirt" regions in captured images. (h) and (i) show the fitting results, and (j) and (k) show the estimated attenuation map $a(x)$ and the scattering map $b(x)$.

(a) Original (b) Recovered (c) Original (d) Recovered

Figure 5.8: Removal of the dirty-lens image artifacts from a video without calibration. Based on the estimated attenuation map and the scattering map, dirty-lens artifacts can be removed for each frame. We show several examples where (a) and (c) are the original frames, and (b) and (d) are the recovered images.

(a) Input frames (b) Original algorithm (c) Extended algorithm

Figure 5.9: Post-processing for unevenly distributed environment lighting. (a) The outside illumination for certain frames is unevenly distributed, causing the intensification $I_\alpha(x)$ to no longer be independent of the scene texture. (b) Consequently, for these frames, the original algorithm sometimes over-compensates for the dirt regions and causes over-darkening or over-brightening. (c) By modeling the scene texture term in $I_\alpha(x)$ as shown in Equation (5.16), we effectively overcome the problem of over-darkening and over-brightening ($w = 0.3$).

causes the intensification I_α no longer independent of the scene texture (Equation (5.7)). (3) While priors on natural images are correct in a statistical sense (*i.e.*, they work for most images), they might be invalid for some specific examples. When one of these causes happens, while the algorithm tries to remove artifacts for the entire image, it might *over-compensates* for some regions and causes the over-darkening and over-brightening.

There is not much we can do to solve the problem caused by (1) and (3) for automatic methods, unless we incorporate more general scattering models, or allow some user interaction to select the optimal c^* with a simple slider.

We now focused on the post-processing for solving the problem caused by (2), since extreme lighting condition is commonly seen in real photographs and videos. We extend the current model by explicitly modeling the forward scattering. We replace the phase function in Equation (5.4) with a simplified Delta-Eddington [Joseph et al., 1976] function $p(\omega_i, \omega_o) \approx (1-w) + w \cdot \delta(\omega_i - \omega_o)$, where w is the weight coefficient. Substituting this into Equation (5.3), we have:

$$I_\alpha(\omega_o) = f_1(\tau) \cdot (c + w \cdot I_0(\omega_o)), \tag{5.15}$$

where each outgoing direction ω_o corresponds to one pixel x, and thus it can be equivalently written as

$$I_\alpha(x) = (c + w \cdot I_0(x)) \cdot f(\alpha(x)). \tag{5.16}$$

Note that since the Delta-Eddington function is an approximation of the real scattering (which often results in a lower-frequency of the incident light $L_i(\omega_i)$), the weight w will not only be determined by the material properties of lens dirt, but it will also change according to the distribution of outside illumination $L_i(\omega_i)$.

Compared with Equation (5.7), the aggregated outside illumination is changed from c to $c + w \cdot I_0(x)$. Accordingly, Equation (5.11) is modified (approximately) as

$$I_0(x) = (I(x) - c \cdot b(x))/(a(x) + w \cdot b(x)). \tag{5.17}$$

The value of w is fixed by trial and error. We found $w = 0.2 \sim 0.8$ gives good results. Although w can be estimated for each frame together with c via Equation (5.10), estimating from individual frames might cause spurious flickering and is time-consuming. Also, w usually varies quite smoothly within the video sequence. Thus we assume w the same

value for an input video sequence and search its value by trial and error. As shown in Figure 5.9(c), this extension effectively overcomes the problem of over-darkening and over-brightening. Compared with the results of the original algorithm shown in Figure 5.9(b), although there are slightly more dirty-lens artifacts remained in the recovered images, the overall quality is higher.

5.7 Summary and Discussion

In this work, we studied image artifacts caused by taking pictures with dirty camera lenses, which can be modeled as an intermediate layer between the camera lens and the target scene that both *attenuates* the background scene and *intensifies* the resulting image. Based on this image formation model, we devised two automatic methods to remove the artifacts from images without user interaction, either with a simple calibration step, or by relying on the statistics of natural images for post-processing existing images or videos.

There are several limitations to our methods.

Not yet a single image approach: As shown in the image formation model in Equation (5.1), since there are four unknowns for one equation, it is challenging to recover the clean image from a single input image. To estimate the attenuation map $a(x)$ and the scattering map $b(x)$ of a dirty camera lens, the two proposed methods require either a calibration step or multiple images taken with the same dirty lens camera. Once $a(x)$ and $b(x)$ are known, we rely on natural image priors to find the optimal estimate of the aggregate of the outside illumination c and remove the artifacts for each input image. As explained in Section 5.6, the automatic estimation of c for some images might not be accurate.

To make the proposed algorithms more robust and also to develop single image based methods, one possible approach is to incorporate more natural image priors, such as the dark channel prior [He et al., 2009]. For certain applications such as consumer photograph software, we can also allow some user interaction to improve result quality.

Noise amplification: In our approach for the removal of dirty-lens artifacts, the artifact-free image $I_0(x)$ is computed as $(I(x) - c \cdot b(x))/a(x)$. Since $a(x)$ is less than 1 where there

is lens dirt, the amount of noise in those recovered pixels will be amplified by $1/a(x)$ times. For example, some color noise can be observed in the regions with the greatest amount of lens dirt in Figure 5.6 and Figure 5.8. In fact, this is a common and fundamental problem for almost all point-wise image enhancement operations [Treibitz and Schechner, 2009].

To suppress this amplification of noise during the process of removing dirty-lens artifacts, standard high-frequency preserving operations such as bilateral filtering can be performed afterwards. Moreover, it is recommended to take high dynamic range images when possible.

Chapter 6

Recovering Volume Densities for Dynamic Participating Media

In the previous chapters, we studied time-varying appearance for opaque and transparent surfaces. While the time-varying appearance for surfaces is usually due to the change in their surface reflectance/transmittance properties, for participating media, such as smoke, clouds, smog, and seawater, their time-varying appearance is mainly because of the change in their volume densities.

In computer graphics, fluid simulation has often been used to to reproduce these dynamic participating media [Stam, 1999; Fedkiw et al., 2001]. The drawback, however, is that it requires considerable amount of work for users to specify external forces (such as air turbulence) to generate realistic, intricate patterns over time (such as smoke ribbons). Therefore, in this chapter, we performed a "data-driven" approach and proposed a structured light method to measure the volume densities from real samples. We aimed at low-density participating media (such as diluted milk) for which we exploited its sparsity to achieve a higher efficiency for acquisition.

6.1 Introduction

Unlike in the case of an opaque object, for participating media, each pixel receives scattered light from all points along the line-of-sight within the volume. This makes it challenging to

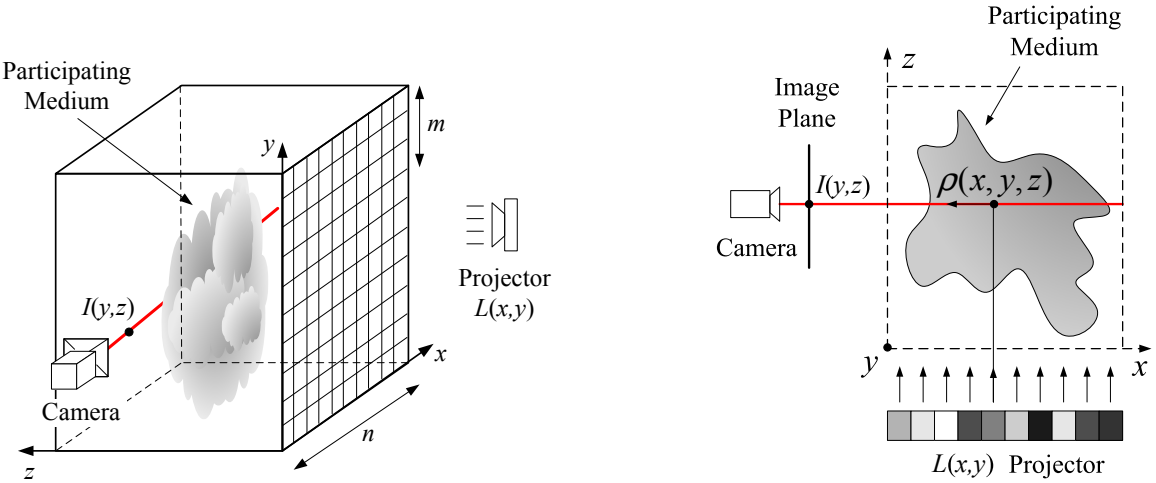
recover the volume density of a participating medium.

One typical approach is tomography, which acquires 2D projections of the 3D volume from multiple views and uses Fourier slice theorem to reconstruct the volume. Ihrke and Magnor [2004, 2006] used eight views to measure flames. Trifonov et al. [2006] proposed to recover transparent objects (immersed in water) with $72 \sim 360$ views. By assuming flames are 2D surfaces, Hasinoff and Kutulakos [2007] used two views to reconstruct flames using dynamic programming. A comprehensive survey of this area can be found in Ihrke et al. [2008]. While tomography is desirable for measuring static volumes, for recovering dynamic, fast-changing non-emissive volumes (*e.g.*, smoke), it requires multiple accurately calibrated and synchronized high-speed cameras and light sources, spanning around the working volume — this is both expensive to set up and also limits the total number of cameras (and thus the total number of 2D projections) we can place around.

Another approach is to recover volume densities from a single view, using active illumination. Hawkins et al. [2005] used a high-powered laser sheet and a high-speed camera (5000fps) to measure slices of smoke volumes via *scanning*, by trading off the temporal resolution for the spatial resolution of the volume. A similar technique called Laser-Induced Fluorescence (LIF) has also been used in the fluid imaging [Deusch and Dracos, 2001]. Fuchs et al. [2007] proposed the idea of shooting a set of static laser rays into the volume and using *interpolation* to reconstruct the volume — as expected the measurements are inherently sparse in this case and hence the recovered information is low in spatial resolution.

Our method belongs to the single view approach. We show that by using coded light patterns, one can make the measurement of a participating medium more efficient in terms of acquisition time as well as illumination power. In particular, we exploit the fact that the brightness measurements made at image pixels correspond to true line-integrals through the medium (see Figure 6.1(a)), and then solve for its volume density. We consider both spatially- and temporally-coded light patterns. Because the patterns are predetermined, measurement and reconstruction time are decoupled. We target low-density inhomogeneous media, for which the density function is *sparse* in an appropriately-chosen basis[1]; this allows

[1] "sparse" does *not* necessarily imply that the volume density must be sparsely distributed in space. It means that the density can be represented with a few non-zero coefficients in an appropriately-chosen basis,



(a) (b)

Figure 6.1: (a) Compressive structured light for recovering inhomogeneous participating media. Coded light is emitted along the z-axis to the volume while the camera acquires images as line-integrated measurements of the volume density along the x-axis. (b) Image formation for diluted participating medium. Image irradiance at one pixel, $I(y, z)$, depends on the integral along the x-axis of the projector's light, $L(x, y)$, and the medium density, $\rho(x, y, z)$, along a ray through the camera center (refer to Equations (6.3) and (6.4)).

us to harness compressive sensing techniques [Candes and Romberg, 2007; Donoho, 2006] that accurately reconstruct a signal from only a few measurements. We refer to our approach as *compressive structured light*.

We show that compressive structured light is more economical than a straightforward sequential scanning of a volume. Whereas the sampling rate of the latter is limited by the desired resolution, the sampling rate of the former is restricted by the sparsity of the data– a considerably more relaxed constraint for low-density phenomena. Since our approach requires fewer measurements, it naturally enables the recovery of dynamic participating media. An added advantage of compressive structured light, is that it requires the projection of multiplexed coded illumination which results in measurements with higher signal-to-noise ratio [Schechner et al., 2003b]. An important practical consequence is that light sources of

such as, wavelets, gradients, principal components, *etc.*

significantly lower power than in the case of sequential scanning can be used.

We have implemented our approach using a digital projector and a camera. A schematic diagram is shown in Figure 6.10(a). The projector and the camera are synchronized and both operate at 360fps. Using 24 coded light patterns, we are able to recover a $128\times128\times128$ volume at 15fps. Using this system, we have recovered various types of inhomogeneous participating media, including, multiple translucent layers, a 3D point cloud of a face etched in a glass cube, and the dynamic process of milk mixing with water.

6.2 Image Formation Model

Let us first derive the relationship between the volume density $\rho(x, y, z)$ and the image irradiance $I(y, z)$ of the camera under our camera/projector setting. We focus on non-emissive, diluted participating media, in which multiple scattering is assumed to be negligible.

As shown in Figure 6.1b, each camera pixel receives light scattered from a row of voxels along the line of sight in the volume (*i.e.*, the red line in Figure 6.1b). For simplicity, we assume the camera and the projector are placed sufficiently far from the working volume, and thus they are orthographic projection. The distortion caused by perspective projection can be corrected with a calibration step, if needed.

Consider one voxel $\rho(x, y, z)$ in the row. Light emitted from the projector is first *attenuated* as it travels from the projector to the voxel, *scattered* at the voxel, and then *attenuated* as it travels from the voxel to the camera. Assuming single scattering, the radiance sensed by the camera from this particular voxel is [Ishimaru, 1978]

$$L(x, y) \cdot \exp(-\tau_1) \cdot \sigma_s \cdot \rho(x, y, z) \cdot p(\theta) \cdot \exp(-\tau_2), \tag{6.1}$$

where $\rho(x, y, z)$ is the volume density (*i.e.*, density of particles) at the voxel, $p(\theta)$ is the phase function ($\theta = \pi/2$ since the camera and the projector are perpendicularly placed), and τ_1 and τ_2 are the optical thicknesses from the projector to the voxel and from the voxel to the camera; σ_s is the scattering cross section of the participating medium. Since σ_s and $p(\theta = \pi/2)$ are the same for all voxels, the above formula can be simplified to (up to a scale $\sigma_s \cdot p(\theta = \pi/2)$)

$$L(x, y) \cdot \exp\left(-(\tau_1 + \tau_2)\right) \cdot \rho(x, y, z). \tag{6.2}$$

The image irradiance, $I(y, z)$, which is the integral of the scattered light from all the voxels along the line, is therefore

$$I(y,z) = \int_x L(x,y) \cdot \exp\left(-(\tau_1 + \tau_2)\right) \cdot \rho(x,y,z) dx. \tag{6.3}$$

For highly diluted media (*i.e.*, $\rho \to 0$), because the optical thicknesses τ_1 and τ_2, which are proportional to the density ρ, are close to 0, the attenuation term usually can also be ignored (*i.e.*, $\exp\left(-(\tau_1 + \tau_2)\right) \approx 1$)for the recovery of volume densities [Hawkins et al., 2005; Fuchs et al., 2007]. In this case, Equation (6.3) is reduced to a linear projection of the illumination and the volume density,

$$I(y,z) \approx \int_x \rho(x,y,z) \cdot L(x,y)\, dx. \tag{6.4}$$

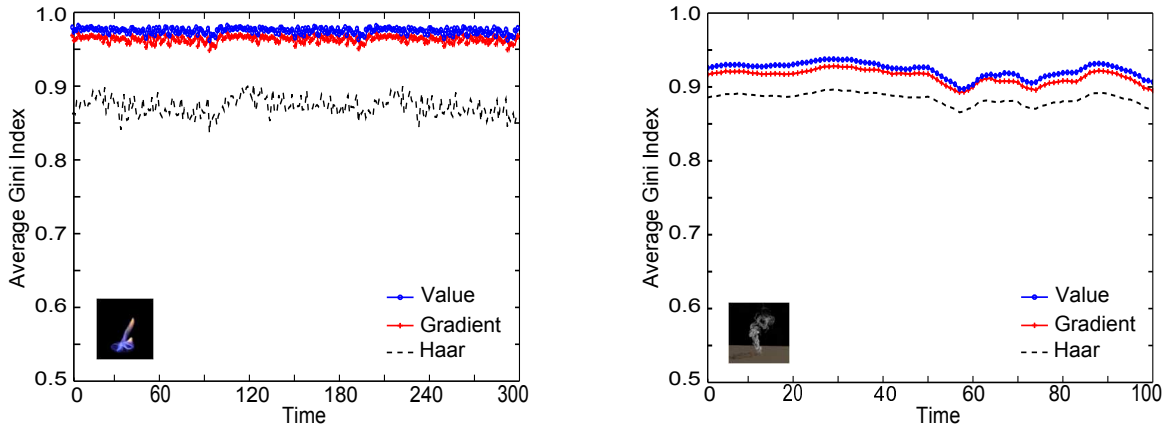
For denser media, we have to consider the attenuation term. We present a iterative method to correct for the attenuation in Section 6.4.3.

6.3 Sparsity of Participating Media

As mentioned in Section 6.1, the sparsity in the volume densities of diluted participating media (*e.g.*smoke, mixing liquids) has been employed for efficient acquisition. Qualitatively, the assumption of sparsity is reasonable since often only a small portion of the entire volume has non-zero density.

To quantitatively justify the use of the sparsity for reconstruction, in this section, we compute the sparsity of two sequences of time-varying participating media measured by other researchers. One sequence is flame measured by Ihrke and Magnor [2004] and the other sequence is smoke measured by Hawkins et al. [2005]. Although these are not the volumetric phenomena we recovered in our experiments, they have similar characteristics in the time-varying volume densities, and thus can give us some estimate about the sparsity of our subjects. Moreover, we compute the sparsity of the same signals at different bases — this will also give us insights of which space is the best for sparse reconstruction.

The key is how to compute sparsity for a given signal. While the definition of sparsity is the number of non-zero elements (*i.e.*, ℓ-0 norm), this is not suitable for signals with noise. While ℓ-1 norm has been used for sparse reconstruction, it is not normalized and thus not



(a) (b)

Figure 6.2: Sparsity (*i.e.*, the Gini index) of two measured dynamic participating media: (a) flame [Ihrke and Magnor, 2004]; (b) smoke [Hawkins et al., 2005]. For each volume in the two measured sequences, we compute the Gini index for each row of the volume and average them as a sparsity measure of the volume. Three types of bases are used — the value of the volume density itself, the gradient of the volume density, and the Haar wavelet transform of the volume density. As shown, these two dynamic phenomena have consistently large sparsity over time, especially in their values and gradients.

good for comparing the sparsity of two signals or two sets of bases. A thorough discussion about the measures for sparsity can be found in [Hurley and Richard, 2008], in which they suggested Gini index as the sparsity measure. Gini index was originally proposed in economics as a measure of the inequality of wealth [Gini, 1921]. It is normalized (its value in between 0 and 1), and it is robust to noise. Given a vector, $\mathbf{x} = [x_1, x_2, x_3, \cdots, x_N]$, we take its absolute value and sort the elements from smallest to largest, $|x|_{(1)} \leq |x|_{(2)} \leq \cdots \leq |x|_{(N)}$. The Gini index of the signal $\mathbf{x}$ is defined as:

$$G(\mathbf{x}) = 1 - \frac{2\sum_{k=1}^{N} |x|_{(k)} \cdot (N - k + 0.5)}{N \sum_{k=1}^{N} |x|_{(k)}}. \tag{6.5}$$

The higher $G(\mathbf{x})$ is, the more sparse the signal is. More properties of Gini index measure can be found in [Hurley and Richard, 2008].

We compute the Gini indices for all the rows of each volume and average them as the sparsity of the volume. Figure 6.2 shows the sparsity for the two data sets (Fig-

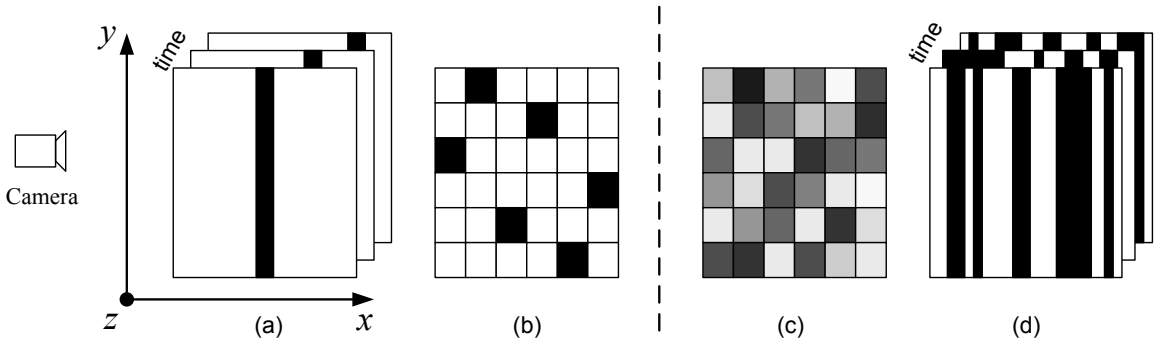


Figure 6.3: Different coding strategies of the light $L(x, y)$ at time t for recovering inhomogeneous participating media: (a) scan (one stripe turned on) [Deusch and Dracos, 2001; Hawkins et al., 2005]; (b) laser-lines interpolation (one pixel turned on per one row) [Fuchs et al., 2007]; (c) Spatial coding of compressive structured light (all pixels are turned on with random values per time frame); (d) Temporal coding of compressive structured light (random binary stripes are turned on per time frame). Compressive structured light, shown in (c) and (d), recovers the volume by reconstructing the 1D signal along x-axis from a few integral measurements. It utilizes the light more efficiently and thus makes the measurement process highly efficient both in acquisition time and illumination power.

ure 6.2(a) for the flame sequence [Ihrke and Magnor, 2004] and Figure 6.2(b) for the smoke sequence [Hawkins et al., 2005]). Three types of bases are used to compute the Gini index — the value of the volume density itself, the gradient of the volume density, and the Haar wavelet transform of the volume density. As shown in Figure 6.2, we found that: (1) over the entire sequence, the two dynamic participating media has consistently high sparsity (≥ 0.95). (2) Compared with the Haar wavelet, the volume densities have higher sparsity in their value and gradients. These observations help us design the acquisition system.

6.4 Compressive Structured Light

Unlike the conventional structured light methods for surface recovery where each camera pixel receives light reflected from one point, for participating media, each camera pixel receives light from all points along the line of sight within the volume. Thus each camera pixel is an integral measurement of one row of the volume density. Whereas conventional

structured light range finding methods seek to triangulate the position of a single point, compressed structured light seeks to reconstruct the 1D density "signal" from a few measured integrals of this signal.

This is clearly a more difficult problem. One way to avoid this problem is to break the integrals into pieces which can be measured directly. The price, however, is the deterioration of either spatial resolution or temporal resolution of the acquisition. Existing methods either illuminate a single slice at a time and scan the volume (see Figure 6.3(a) and [Deusch and Dracos, 2001; Hawkins et al., 2005]), thus sacrificing temporal resolution, or they illuminate a single pixel per row and use interpolation to reconstruct the volume (*e.g.*, Figure 6.3(b) and [Fuchs et al., 2007]), sacrificing spatial resolution.

In contrast, the proposed compressive structured light method uses the light much more efficiently, projecting coded light patterns that yield "signatures," or integral measurements, of the unknown volume density function.

The didactic illustration in Figure 6.1(a) depicts a simple lighting/viewpoint geometry under orthographic projection, with the camera viewpoint along the x-axis, and the projector emitting along the z-axis. Consider various coding strategies of the 3D light function $L(x, y, t)$: *Spatial* codes (Figure 6.3(c)) recover the volume from a single image by trading spatial resolution along one dimension; *Temporal* codes (Figure 6.3d) trade temporal resolution by emitting a sequence of vertical binary stripes (with no coding along y-axis), so that full spatial resolution is retained. All of the four methods shown in Figure 6.3 can be equally improved using color channels.

In the following, we will see that these compressive structured light codes yield high efficiency both in acquisition time and illumination power; this comes at the cost of a more sophisticated reconstruction process, to which we now turn our attention.

6.4.1 Coding and Formulation

To better visualize our formulation, consider first the case of spatial coding. Suppose we want to reconstruct a volume at the resolution $n \times n \times n$ (*e.g.*, $n = 100$). The camera and the projector have the resolution of $M \times M$ pixels (*e.g.*, $M = 1024$). Therefore, one row of voxels along the x-axis (refer to the red line in Figure 6.1a) will receive light from $m = M/n$

(*e.g.*, $m = 1024/100 = 10$) rows of the projector's pixels. The light scattered by these voxels in the viewing direction will then be measured, at each z-coordinate, by a vertical column of m camera pixels. Without loss of generality, we use $\mathbf{l}_1 = L(x,1), \cdots, \mathbf{l}_m = L(x,m)$ to denote the m rows of pixels from the projector, and $b_1 = I(1,z), \cdots, b_m = I(m,z)$ to denote the image irradiance of the m pixels in the camera image. Let $\mathbf{x} = [\rho_1, \cdots, \rho_n]^T$ be the vector of the voxel densities along the row. Assuming no attenuation for now, the image irradiance for each of these m pixels is a linear projection of the light and the voxels' density from Equation (6.4):

$$b_i = \mathbf{l}_i^T \mathbf{x}, \qquad i = 1, \cdots, m. \tag{6.6}$$

Rewriting these m equations in matrix form, we have:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{6.7}$$

where $\mathbf{A} = [\mathbf{l}_1, \cdots, \mathbf{l}_m]^T$ is a $m \times n$ matrix, $\mathbf{b} = [b_1, \cdots, b_m]^T$ is a $m \times 1$ vector.

Thus, if attenuation is not considered, the problem of recovering the volume is formulated as the problem of reconstructing the 1D signal $\mathbf{x}$ given the constraints $\mathbf{A}\mathbf{x} = \mathbf{b}$. To retain high spatial and temporal resolution, we often can only afford far fewer measurements than the number of unknowns, *i.e.*, $m < n$, which means the above equation is an underdetermined linear system and optimization is required to solve for the best $\mathbf{x}$ according to certain priors.

One benefit of this optimization-based reconstruction is high efficiency in acquisition, which we quantify using the *measurement cost*, m/n, where m is the number of the measurements and n is the number of unknowns (*i.e.*, the dimension of the signal). For example, the measurement cost of the scanning method [Deusch and Dracos, 2001; Hawkins et al., 2005] is one. We show that by exploiting the sparsity of the signal, we can reconstruct the volume with much lower measurement cost (about 1/8 to 1/4).

6.4.2 Reconstruction via Compressive Sensing

Solving the underdetermined linear system requires some prior (assumed) knowledge of the unknown signal, which can be represented as optimization functionals or constraints on the data. We consider several alternatives, as listed in Table 6.1. In addition to the

Table 6.1: Objective functionals used for volume reconstruction

Method	Optimization Functional	Constraints
Least Square (LS)	$\|\mathbf{Ax}-\mathbf{b}\|_2$	
Nonnegative Least Square (NLS)	$\|\mathbf{Ax}-\mathbf{b}\|_2$	$\mathbf{x}\geq 0$
CS-Value	$\|\mathbf{x}\|_1$	$\mathbf{Ax}=\mathbf{b},\ \mathbf{x}\geq 0$
CS-Gradient	$\|\mathbf{x}'\|_1$	$\mathbf{Ax}=\mathbf{b},\ \mathbf{x}\geq 0$
CS-Both	$\|\mathbf{x}\|_1+\lambda\|\mathbf{x}'\|_1$	$\mathbf{Ax}=\mathbf{b},\ \mathbf{x}\geq 0$

commonly-used Least Square (LS) and Nonnegative Least Square (NLS) approaches, we also consider functionals using ℓ_1-norms, as these bias toward sparse representation, based on our observations on the sparsity of some dynamic participating media (Section 6.3).

First, we observe that for many natural volumetric phenomena, often only a small portion of the entire volume is occupied by the participating media. For example, consider the beautiful ribbon patterns generated by smoke; similarly, sparsity was implicitly used to reconstruct (surface-like) flames [Hasinoff and Kutulakos, 2007]). This suggests the use of the ℓ_1-norm of the signal value (CS-Value).

Furthermore, from Figure 6.2 we also observe high sparsity in the gradients of the volume densities. The sparsity of *gradients* of natural images is well studied [Olshausen and Field, 1996; Simoncelli, 1997] and shown to work well for image restoration [Rudin et al., 1992]. In this vein, we consider the use of ℓ_1-norm on the signal's gradient (CS-Gradient).

Finally, consider a dynamic process, such as milk dissolving in water: here diffusion decreases the signal value's sparsity over time, but it increases the gradient sparsity. Motivated by this observation, we consider the linear combination of ℓ_1-norms of both the value and the gradient (CS-Both). The objective function is thus defined as $|\mathbf{x}|_1+\lambda|\mathbf{x}'|_1$, where λ is the weight. In order to obtain a one-to-one mapping from the gradient domain to the original signal, the gradient $\mathbf{x}'$ is defined as a $(n+1)\times 1$ vector consisting of both the differentials and the first and the last element of $\mathbf{x}$, *i.e.*, $\mathbf{x}'=[\rho_1,\rho_2-\rho_1,\rho_3-\rho_2,\cdots,\rho_n-\rho_{n-1},\rho_n]$, where $\mathbf{x}=[\rho_1,\cdots,\rho_n]$.

To understand these objective functionals intuitively, let us consider a simple example

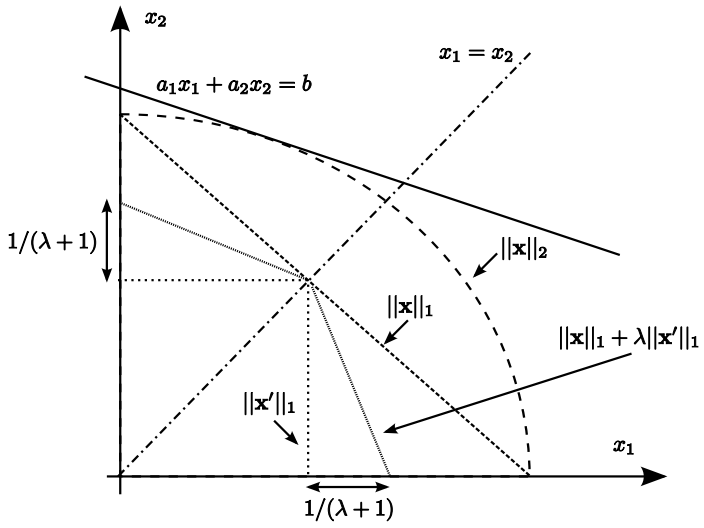


Figure 6.4: Geometric interpretation of the objective functionals for a 2×1 vector $\mathbf{x} = [x_1, x_2]$. Suppose we only have one measurement $a_1x_1 + a_2x_2 = b$. The least square solution minimizes $||\mathbf{x}||_2$, which is often non-sparse. The CS-Value solution minimizes $||\mathbf{x}||_1$ and will be on one of the two axes which is sparse. The CS-Gradient solution minimizes $||\mathbf{x}'||_1$, which in this case will be the intersection point with the diagonal line $x_1 = x_2$ and is sparse in its gradient. Finally, the CS-Both solution allows solutions both on the two axes and the diagonal lines. The exact solution depends on the relative sparsity of the signal in its value or its gradient.

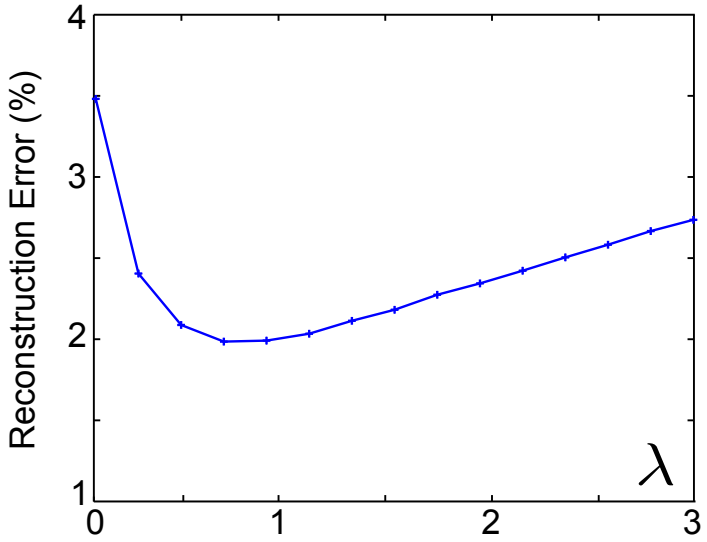


Figure 6.5: To find the optimal λ value for the functional $|\mathbf{x}|_1 + \lambda|\mathbf{x}'|_1$, we run simulation on the smoke sequence [Hawkins et al., 2005]. This plot shows the average reconstruction error for λ from 0.0 to 3.0, and we set $\lambda = 1$ in our experiments.

where $\mathbf{x}$ is a 2×1 vector, $\mathbf{x} = [x_1, x_2]$, and assume $x_1 \geq 0$, $x_2 \geq 0$. As shown in Figure 6.4, suppose we only have one measurement $a_1x_1 + a_2x_2 = b$ which tells us the solution of $\mathbf{x}$ should be on a line. The least square solution minimizes $||\mathbf{x}||_2$ and it will be the tangential point between the line $a_1x_1 + a_2x_2 = b$ and the circle, which is less likely to be sparse. The CS-Value solution minimizes $||\mathbf{x}||_1$, and it will be the intersection point between the line $a_1x_1 + a_2x_2 = b$ and the x_2 axis, which is sparse (*i.e.*, one element is zero). The CS-Gradient solution minimizes $||\mathbf{x}'||_1$, and in this case it will be the intersection point between the line $a_1x_1 + a_2x_2 = b$ and the diagonal line $x_1 = x_2$, which is sparse in the gradient. Finally, the CS-Both solution finds a trade-off between CS-Both and CS-Gradient. It allows sparse solutions both on the two axes and the diagonal line. Its exact shape depends on the weight λ, as shown in Figure 6.4. The optimal weight λ should be set to the ratio between the sparsity in the signal's gradient and the sparsity in the signal's value.

To find the optimal λ, we run simulation on the smoke sequence [Hawkins et al., 2005]. For each simulation, we fix the measurement cost $m/n = 1/4$ and generate the matrix $\mathbf{A}$ with Gaussian random variables. We then reconstruct the volume densities by minimizing the objective functional $|\mathbf{x}|_1 + \lambda|\mathbf{x}'|_1$ at multiple λ values. This simulation is performed for

all rows of each volume, and the averaged reconstruction error is computed for each λ value. As shown in Figure 6.5, $\lambda = 1$ gives the minimal reconstruction error overall, and thus we set $\lambda = 1$ in the following simulations and experiments.

6.4.3 An Iterative Algorithm for Attenuation Correction

Until now, we have not yet considered the attenuation in the image formation model in Equation (6.3). To take into account attenuation, we use a simple iterative linearization algorithm as follows:

1. Assume no attenuation, solve the optimization problem with techniques from Section 6.4.2 to get the initial reconstruction of the volume density $\rho^{(0)}(x, y, z)$.

2. At iteration k, compute the attenuated light as:

$$L^{(k)}(x, y, z) = \exp\left(-\left(\tau_1 + \tau_2\right)\right) \cdot L(x, y),$$

   where τ_1 and τ_2 are the optical thicknesses which can be computed using the volume density from the previous iteration $\rho^{(k-1)}(x, y, z)$ as

$$\tau_1 = (\sigma_a + \sigma_s) \int_z \rho^{(k-1)}(x, y, z)\, dz, \qquad \tau_2 = (\sigma_a + \sigma_s) \int_x \rho^{(k-1)}(x, y, z)\, dx, \tag{6.8}$$

   where σ_a and σ_s are the absorption cross section and the scattering cross section of the medium, which are assumed to be known They can be measured by a second camera taking the shadowgraph of the volume.

3. With the attenuated light $L^{(k)}(x, y, z)$, Equation (6.3) becomes a linear equation. We solve for $\rho^{(k)}(x, y, z)$ and go to next iteration until it converges. In practice, we found that the algorithm usually converges within 3-4 iterations.

Since our method accommodates the scanning method [Deusch and Dracos, 2001; Hawkins et al., 2005] and the interpolation method [Fuchs et al., 2007] as special cases, the iterative algorithm could be directly applied to these prior methods as well.

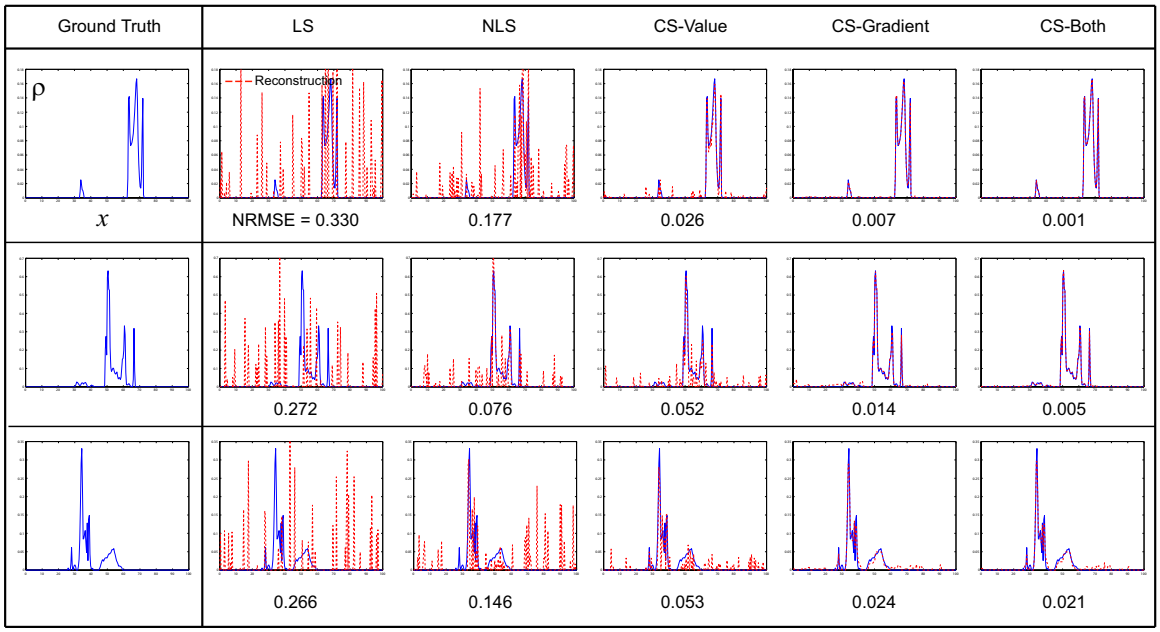


Figure 6.6: Comparison of the objective functionals for volumetric reconstruction. The first column is the original signal. The remaining columns show reconstruction results (red dashed lines) for different methods, given the measurement cost, m/n, is equal to 1/4. The value below each plot is the NRMSE (Normalized Root Mean Squared Error) of reconstruction.

6.5 Validation with Simulation

In this section, we perform simulations in order to compare the several objective functionals shown in Table 6.1 and validate their accuracy. Comparison of these reconstruction methods is first performed on 1D synthetic signals. These signals are randomly sampled rows from the volume density of smoke acquired in [Hawkins et al., 2005]. We restrict the measurement cost, m/n, to be 1/4. The measurement ensemble, $\mathbf{A}$, is generated in a way that each element is drawn independently from a normal distribution and each column is normalized to 1, which is effectively a white noise matrix and is known to be good for compressive sensing [Donoho, 2006]. Normalized Root Mean Squared Error (NRMSE) is used as the measure of error, which is defined as:

$$\text{NRMSE} = \frac{1}{x_{\max} - x_{\min}} \sqrt{\frac{\sum_{i=1}^{n} (\hat{x}_i - x_i)^2}{n}}, \tag{6.9}$$

where $\hat{x}_i$ and x_i are the i-th element of the reconstructed signal and the original signal, respectively, and $x_{\max} - x_{\min}$ is the range of the signal.

The reconstruction results are shown in Figure 6.6. The commonly-used LS performs the worst, since it merely minimizes the errors without using any prior on the data. With the nonnegative constraint added, NLS has better performance. CS-Value and CS-Gradient are better than NLS given that both use one more prior—the sparsity on the signal value or on the signal gradient. CS-Both($\lambda = 1$) outperforms other methods due to its adaptive ability. In our trials, the favorable performance of CS-Both was not very sensitive to changes of λ.

These observations carry over to the 3D setting (see Figure 6.7), where we reconstruct a $128 \times 128 \times 128$ volume; note that this requires 128×128 independent 1D reconstructions. The volume is generated from a triangular mesh of a horse and it is divided into $128 \times 128 \times 128$ voxels. For each voxel, if it is inside the mesh, the density is designed to be proportional to the distance from the center of the voxel to the center of the mesh, otherwise the density is 0 — we intentionally design the volume densities to be non-sparse (smooth-varying) in order to test the algorithm. Figure 6.7(a) shows the volume where blue corresponds to the lowest density while yellow corresponds to the highest density. A slice of the volume is shown in Figure 6.7(b).

Both spatial coding and temporal coding of compressive structured light are tested.

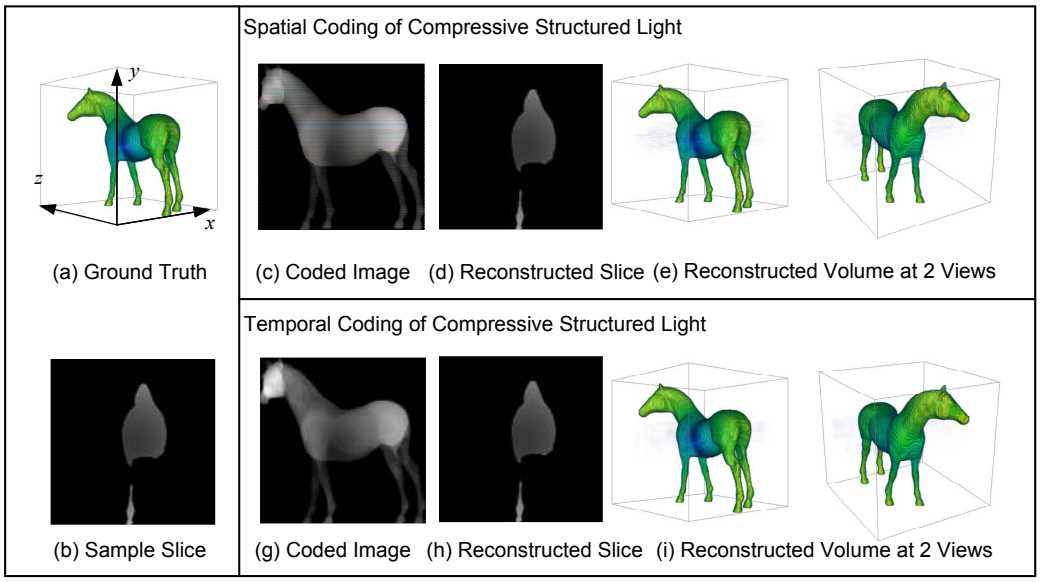
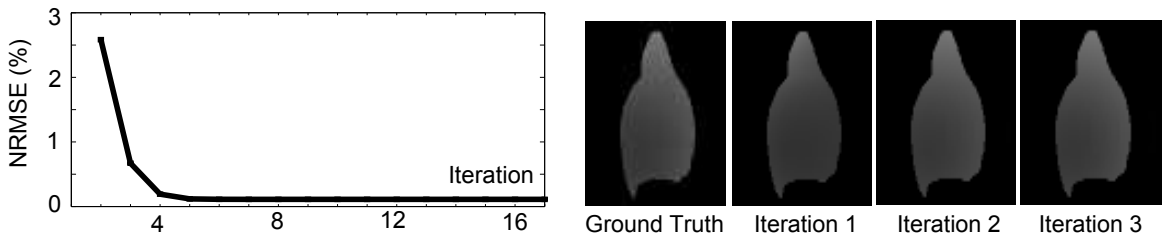


Figure 6.7: Simulation results of volumetric reconstruction using compressive structured light. (a) The original volume where blue means the lowest density and yellow means the highest density. (b) A slice of the volume. On the right, the top and the bottom row shows the reconstruction results for spatial coding and temporal coding, respectively. For each row, from left to right are the coded image acquired by the camera, the reconstruction of the slice, and the reconstructed volume under two different views.



(a) (b)

Figure 6.8: (a) Reconstruction errors and (b) slices with iterative attenuation correction.

The measurement cost, m/n, is fixed to 1/4. For spatial coding, we use a random color image with resolution of 1280×1280 as the coded light from the projector. This gives us $m = 1280/128 \times 3 = 30$ measurements to recover densities of 128 voxels on one row of the volume. Based on Equation (6.3), a single image is generated from the camera view and used for reconstruction. For temporal coding, we use random binary stripes as illumination and generate 32 images for reconstruction. CS-Both is used to reconstruct the volume for both cases. As shown in Figure 6.7, both methods accurately reconstruct the volume. Moreover, Figure 6.8 shows the reconstruction errors and reconstructed slices at different iterations of attenuation correction, which demonstrates the effectiveness of the iterative algorithm.

We also evaluate different reconstruction methods at various measurement costs from 1/16 to 1. The results are shown as a table in Figure 6.9. Conclusions similar to the ones from the previous 1D signal simulation can be drawn from these results: (1) As expected, all methods have improvements as the measurement cost increases. (2) Without using any prior of the data, LS is the worst for reconstruction with insufficient measurements. (3) CS-Gradient and CS-Both largely outperform other methods, especially for low measurement cost, which indicates strong sparsity in the signal's gradient. (4) CS-Both is better than CS-Gradient, especially at low measurement cost (*e.g.*, as shown in Figure 6.9 at $m/n = 1/16$). Based on these preliminary simulations, we chose to run our actual acquisition experiments with a measurement cost of 1/4 and the CS-Both optimization functional.

Table 6.2 shows the reconstruction error (NRMSE) for the CS-Both method for two coding patterns (*i.e.*, partial Hadamard code and random binary code) at different noise levels. As shown, the CS-Both method is robust across multiple noise levels. When the measurement cost is low, random binary code has better performance than the Hadamard code. If we have enough measurement (*i.e.*, the measurement cost is 1), the Hadamard code has better performance, as expected [Schechner et al., 2003b]. In our experiments, we used random binary coding pattern.

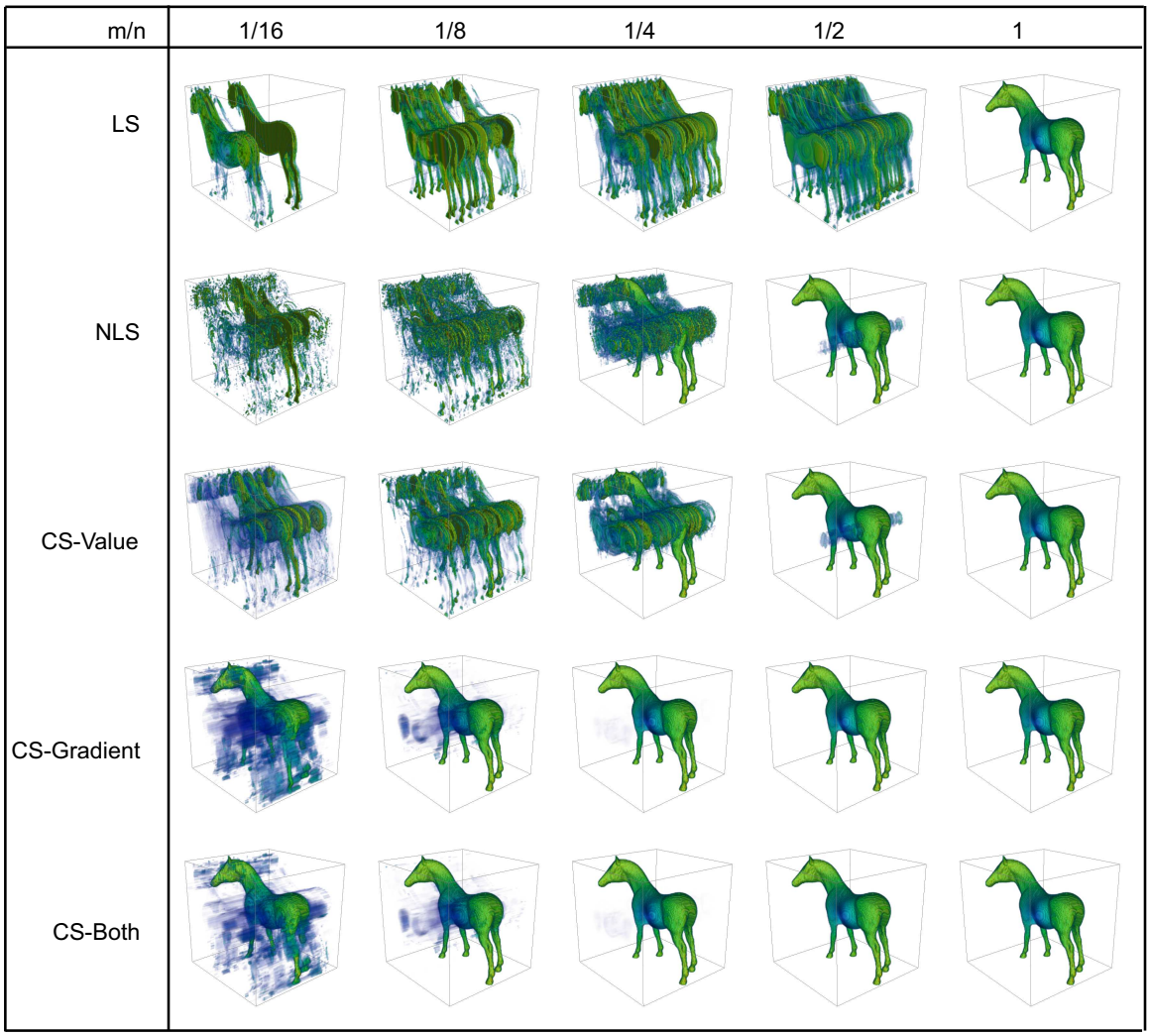


Figure 6.9: Comparison of the objective functionals at different measurement costs m/n. CS-Both outperforms other methods.

Table 6.2: NRMSE for two coding patterns at different noise levels

Noise Level	Pattern	Measurement Cost (m/n)				
		1/16	1/8	1/4	1/2	1
$\sigma = 0.001$	Hadamard	0.077	0.041	0.021	0.0007	2.47e-05
	Random	0.042	0.0063	0.0010	8.91e-05	3.13e-05
$\sigma = 0.005$	Hadamard	0.078	0.042	0.023	0.0014	0.0011
	Random	0.043	0.008	0.0026	0.0017	0.0013
$\sigma = 0.01$	Hadamard	0.074	0.045	0.022	0.0026	0.0020
	Random	0.044	0.011	0.0042	0.0031	0.0023

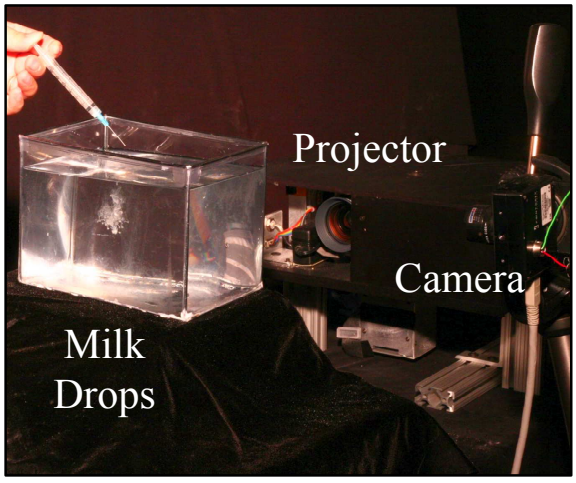


Figure 6.10: Experimental setup for compressive structured light. The projector and the camera are synchronized.

6.6 Experimental Results

We have implemented the temporal coding of compressive structured light for recovering inhomogeneous participating media. Our system consists of a 1024×768 DLP projector and a 640×480 Dragonfly Express 8-bit camera, positioned at right angles, both viewing the inhomogeneous participating medium (milk drops in water). The projector and the camera are synchronized and both operate at 360fps. The camera's resolution is set to 320×140 in order to achieve 360fps. Using 24 coded light patterns, we are able to recover a $128 \times 128 \times 128$ volume at 15fps. These light patterns consist of 128 vertical stripes. Each stripe is assigned 0 or 1 randomly with the probability of 0.5. In this way, about half the amount of the light is turned on for each measurement. We also tried alternative light patterns such as Hadamard codes, and found the random binary codes have better performance. The 24 light patterns correspond to 24 randomly chosen rows from a 127×127 Hadamard matrix.

We used this system to recover several types of inhomogeneous participating media, including, multiple translucent layers (Figure 6.11), a 3D point cloud of a face etched in a glass cube (Figure 6.12), and the dynamic process of milk mixing with water (Figure 6.13). The reconstructed volumes are visualized with the ray casting algorithm [Schroeder et al., 2006] in which the opacity function is set to the volume density.

6.6.1 Recovery of Static Volumes

We first perform reconstruction on static volumes. Figure 6.11 shows the results of an object consisting of two glass slabs with powder on both. The letters "EC" are drawn manually on the back plane and "CV" on the front plane by removing the powder. Thus we create a volume in which only two planes have non-zero density. A photograph of the object is shown in Figure 6.11a. We then project coded light patterns on the object and reconstruct the volume using the proposed method. Figure 6.11 shows one of the 24 captured images as well as the reconstructed volume at different views. We show the reconstructed volume with and without attenuation correction. It shows that attenuation correction improves the results by increasing the density on the back plane.

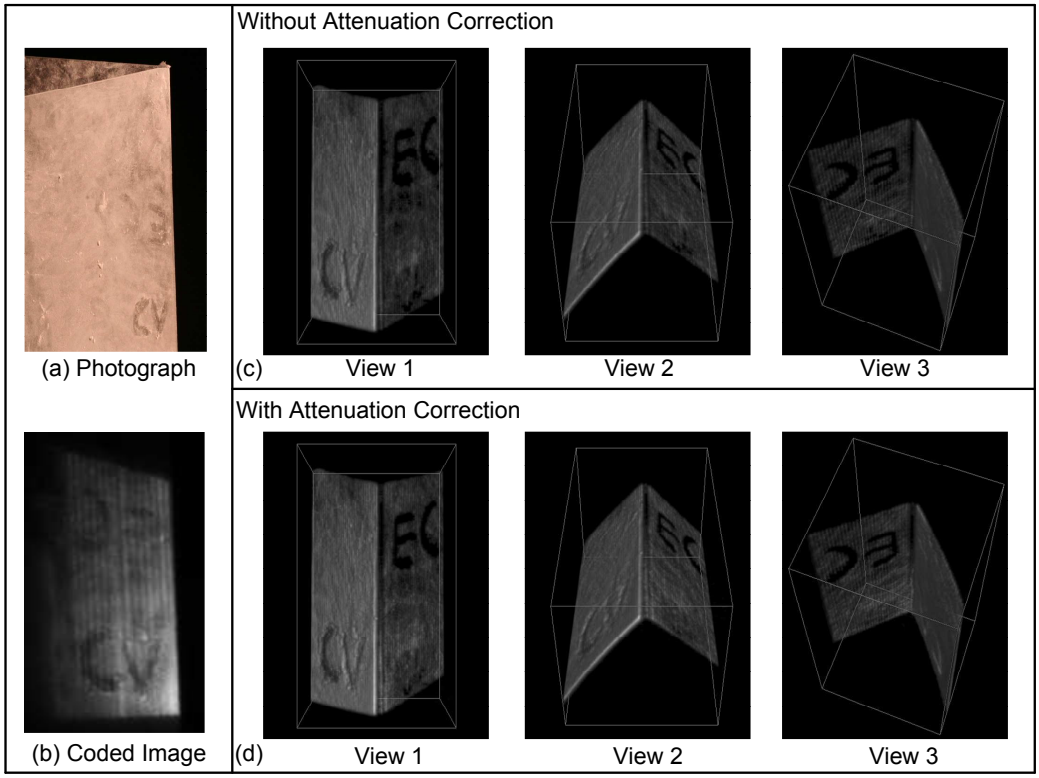


Figure 6.11: Reconstruction results of two planes. (a) A photograph of the object consisting of two glass slabs with powder. The letters "EC" are on the back slab and "CV" on the front slab. (b) One of the 24 images captured by the camera. Reconstructed volume at different views with (c) and without (d) attenuation correction.

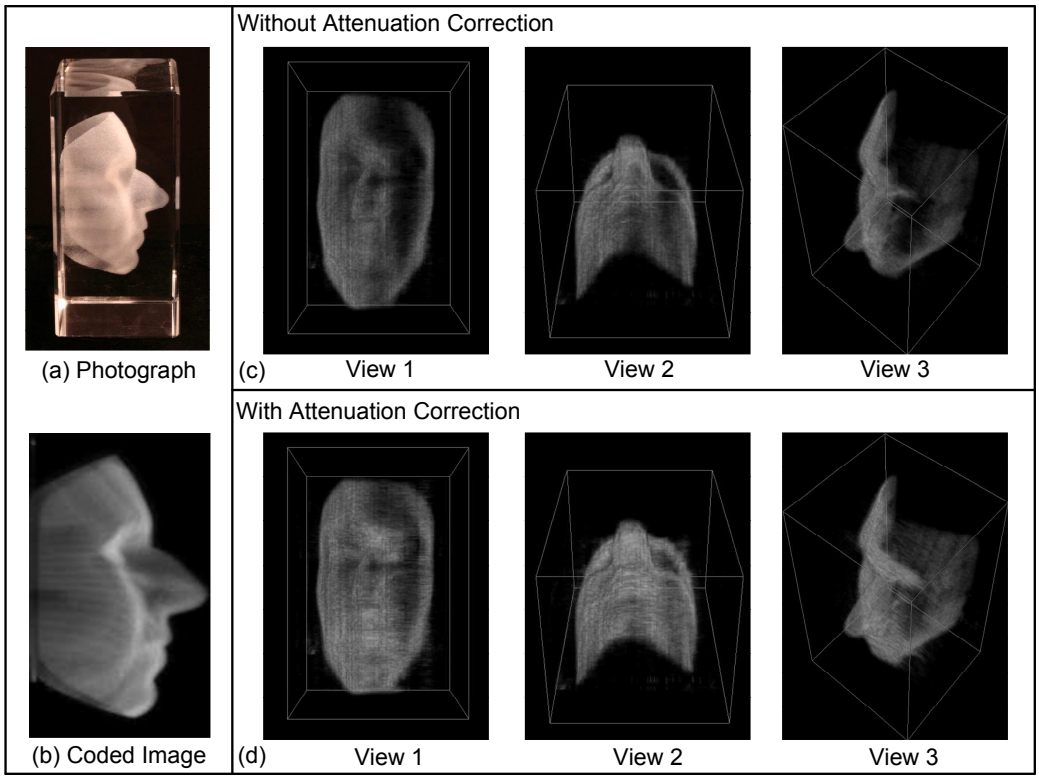


Figure 6.12: Reconstruction results of a 3D point cloud of a face etched in a glass cube. (a) A photograph of the object. (b) One of the 24 images captured by the camera. Reconstruction results are shown on the right side in the same manner as Figure 6.11.

Figure 6.12 shows the reconstruction for a 3D point cloud of a face etched in a glass cube. As seen, our method achieved good reconstruction of the volume. In this example, multiple scattering and attenuation within the point cloud are much stronger than the previous example. As a result, in the reconstructed volume, the half of the face not directly visible to the camera has a lower estimated density (*e.g.*, the relative darker area of the right eye and right forehead in Figure 6.12).

6.6.2 Recovery of Dynamic Volumes

Finally, we use our system to reconstruct time-varying volumes. We take the dynamic process of milk drops dissolving in water as an example. We use a syringe to drip milk drops into a water tank as shown in the adjacent figure. With the proposed method, we are able to reconstruct time-varying volumes with high spatial resolution ($128 \times 128 \times 250$) at 15fps, which recovers the interesting patterns of the dynamic process (see Figure 6.13).

6.7 Summary and Discussion

We proposed compressive structured light for recovering the volume densities of inhomogeneous participating media. Unlike conventional structured light range finding methods where coded light patterns are used to establish correspondence for triangulation, compressive structured light uses coded light as a way to generate measurements which are line-integrals of volume density. By exploiting the sparsity of the volume density, the volume can be accurately reconstructed from a few measurements. This makes the acquisition highly efficient both in acquisition time and illumination power, and thus enables the recovery of time-varying volumetric phenomena.

There are several limitations to our proposed compressive structured light formulation, which need to be addressed in the future:

1. *Multiple Scattering.* Although utilizing more light elements increases the efficiency of the acquisition, it will increase multiple scattering as well, which will cause biased reconstruction, as the artifacts shown in Figure 6.12. One potential way to alleviate this problem is to separate multiple/single scattering by using more complex light codes

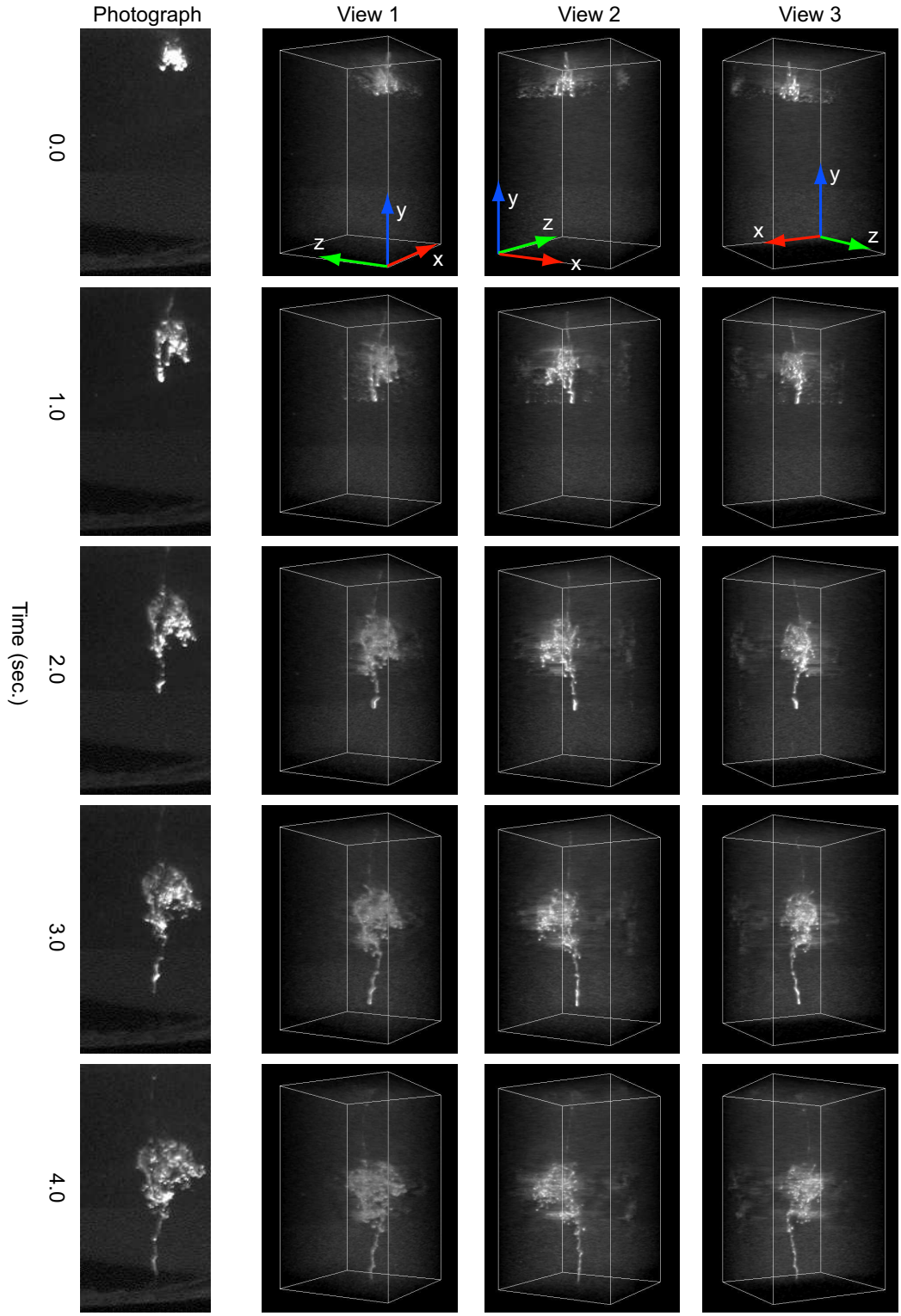


Figure 6.13: Reconstruction of milk drops dissolving in water. 24 images are used to reconstruct a $128 \times 128 \times 250$ volume at 15fps. Three views are shown. Each row corresponds to one instance in time. The leftmost column shows the corresponding photographs.

in a similar way to [Nayar et al., 2006]. For example, instead of using vertical stripes, one could use vertical 0/1-interleaved stripes and thus estimate the global illumination from neighboring pixels. This, however, requires trading off spatial resolution.

2. *Calibration for the Spatial Coding Method.* The spatial coding seems more desirable than the temporal coding due to its high temporal resolution (*i.e.*, volume reconstruction from one single image) and the easy access of high spatial resolution devices. However, it requires highly accurate calibration both geometrically and radiometrically. The defocus of both the projector and the camera needs to be considered as well. In contrast, the temporal coding method is more robust to noise and defocus and easy to calibrate.

We view compressive structured light as a general framework for coding the 3D light function $L(x, y, t)$ for reconstruction of signals from line-integral measurements. In this light, existing methods such as laser sheet scanning and laser line interpolation, as well as the spatial coding and temporal coding discussed in this work, can be considered as special cases. One interesting future direction is to design more complex coding strategies to improve the performance or apply the method to new problems.

Chapter 7

Conclusions

As shown in previous chapters, time-varying appearance of natural phenomena is important for both computer graphics and computer vision. In this thesis, we have shown a systematic study of time-varying appearance, including time-varying reflectance for opaque surfaces, time-varying refraction for weathered transparent surfaces, and dynamic volumetric phenomena.

7.1 Summary of Contributions

This thesis mostly follows the data-driven approach. The novel question we asked is how to make data-driven approach more *intelligent* and *efficient* from measurement to modeling to synthesis. In other words, we asked ourselves the following three questions during our study:

- **Measurement**: instead of directly measuring samples sparsely in high dimensional space, can we *code* the signal and measure them more *efficiently*? (Chapter 4 and Chapter 6).
- **Modeling**: instead of simply tabulating the acquired high-dimensional data, can we develop *compact* representations that are easy to control for rendering or convenient for removing the effect? (Chapter 3 and Chapter 5).
- **Synthesis**: instead of interpolation for reproducing the acquired data, can we develop

algorithms to create novel renderings for artistic control?(Chapter 3 and Chapter 4).

These three questions guided our study of time-varying appearance of natural phenomena, as summarized below.

Time-Varying Reflectance for Opaque Surfaces: In Chapter 3, we take a significant step toward measuring, modeling and rendering time-varying surface appearance. Using a computer-controlled dome, we have acquired the first time-varying appearance database of 28 samples, encompassing a variety of natural processes including burning, drying, decay and corrosion.

Our main technical contribution is a space-time appearance factorization model. This model separates the spatial variation and the temporal variation from measured data. Experiments show our model can accurately represent a variety of natural phenomena. It enables a number of novel rendering applications, such as transfer of the time-varying effect to a new static surface, accelerate temporal evolution, extrapolation beyond the acquired sequence, and texture synthesis of time-varying appearance.

Weathered Appearance for Transparent Surfaces: In Chapter 4, we studied the weathered appearance of transparent objects and constructed analytic models for its BRDF and BTDF for real-time rendering. Based on the proposed models, we also developed efficient acquisition methods to measure contamination patterns from a single image. The models and the measured data are used to reproduce the weathered appearance in both 3D renderings and 2D photographs.

The understanding of weathered appearance of transparent surfaces has been used to remove the artifacts from photographs caused by dirty camera lenses, as shown in Chapter 5. We took into account both the attenuation and the scattering of lens dirt and developed two automatic methods to remove the artifacts and enhance images — either by taking several pictures of a structured calibration pattern beforehand, or by leveraging natural image statistics.

Efficient Acquisition of Dynamic Volumetric Phenomena: In Chapter 6, we focused on the recovery of time-varying participating media, such as smoke, clouds, and

mixing fluids. For these natural phenomena, change in appearance over time is mainly caused by a time-varying volume density. We propose a method for recovering the volume densities using structured light. Whereas conventional structured light methods reconstruct 3D shapes based on triangulation (and thus are only applicable for opaque surfaces), our method projects patterns into the *volume* of a participating medium to produce *line-integral measurements* of the volume density. These integral measurements enable the use of *compressive sensing* to recover the volume density from only a few measurements — this makes the acquisition more efficient in terms of both acquisition time and illumination power.

7.2 Future Research

If we recall the history of appearance modeling in computer graphics and computer vision (Section 1.2 on page 4), we can see that there has been tremendous progress over the last twenty years. Many complex, high-dimensional appearance phenomena can be acquired to further facilitate photo-realistic rendering. Computer graphics has achieved such a high degree of realism that we began to see more and more augmented-reality applications in our daily life such as remote conferencing, flight simulators, and movies with a mix of real actors and CG. Researchers in computer vision continue to widen and deepen the use of appearance models for estimating scene information.

Each advance brings a new opportunity for the study of appearance of natural phenomena. Below, I summarize several important future directions.

Exploring Other Dimensions in Appearance Modeling: In this thesis, we have focused on the time-varying aspect of visual appearance. In addition to the dynamic natural phenomena studied in the thesis, there are many other natural phenomena that exhibit interesting time-varying visual appearance but have not yet been thoroughly investigated. For example, surface geometry might change over time for phenomena such as melting, roughening, or shrinking. Atmospheric turbulence due to time-varying refractive index will often cause artifacts in photographs and videos, especially for telescopy and astronomy. These effects are often very difficult to simulate or compensate with all the level of details, but are feasible to study in the data-driven manner.

Another important dimension for appearance modeling of natural phenomena is wavelength. Many natural phenomena exhibit strong, interesting multi-spectral appearance properties, such as flowers, plants, food, and skin. With multi-spectral material properties, one can simulate the appearance of these phenomena under various types of lighting conditions, which is useful not only for photo-realistic rendering but also for material recognition such as automatic inspection and medical imaging. So far, the acquisition of multi-spectral BRDF is still expensive in terms of acquisition time and spectral resolution. Moreover, there are few studies considering temporal variation in the multi-spectral appearance of natural phenomena. These might be promising research topics for further exploration.

Statistical Approach for Appearance Modeling: As discussed in Chapter 2, appearance models are usually high dimensional, such as the BRDF (4D) and the general light transport matrix (8D). Both acquisition and rendering of these high-dimensional functions are difficult.

On the other hand, the variations of the BRDFs of many real world materials are not so versatile to our human vision system — they mainly vary in the size of the specular lobes or whether the reflection is isotropic or anisotropic. The human visual system does a remarkable job distinguishing between the BRDFs of many real world materials, even from a single image [Adelson, 2001]. This suggests that the appearance of many natural phenomena might be embedded in some low-dimensional manifold. Moreover, there might exist some strong priors for material properties, just like the priors for natural images [Torralba and Oliva, 2003].

Therefore, one future direction for appearance modeling is to explore the statistical properties for the appearance of natural phenomena. We need to understand what is the space of real-world materials, and we need to summarize the priors. There have been some early studies where researchers proposed linear models such as PCA [Matusik et al., 2003], a matrix factorization model [Lawrence et al., 2006], and nonlinear models such as LLE [Matusik et al., 2003] to represent BRDFs. Their conclusions are encouraging. For example, in [Matusik et al., 2003] it was shown that many isotropic BRDFs can be represented using about 45 bases. Nonlinear statistical analysis on more general material

properties would be of great interest to a large audience — the insights of these studies would be valuable for real-time rendering, efficient acquisition, and material detection and recognition.

Study of Appearance in the Outdoors: Most current research on appearance modeling measures small samples in a laboratory environment under controlled lighting and imaging conditions. Although this simplifies data acquisition and processing, it limits the subjects of research and restricts the acquisition of many natural phenomena. Future study of appearance should focus on extending current acquisition and modeling techniques to the real, unconstrained environment. There are several possible research topics we can envision along this line:

- *Portable acquisition devices* which can quickly measure the material properties of samples outdoors in the field. Such devices are desirable especially if they can be equipped in personal digital devices (*e.g.*, cellular phones). With such devices, users can easily *capture* interesting appearance phenomena and *transfer* to the computer for design and rendering. We can also combine it with other data, such as 3D shape and GPS coordinates, for recognizing the sample and investigating its statistical properties. Structured light can be equipped in cellular phones or consumer digital cameras to assist the acquisition.

- *Simultaneous* recovery of shape and appearance for objects in the real world. Existing methods for appearance acquisition often are limited to small, flat samples, which separates the task for the recovery of shape and appearance for simplification. However, this approach is not always possible for many real world objects, such as *large scale* objects (*e.g.*, buildings), *irreplaceable* objects (*e.g.*, oil paintings), and *inaccessible* objects (*e.g.*, planetary exploration). Often, we need to recover both the shape and the appearance for these objects simultaneously under uncontrolled lighting conditions. It might be necessary to incorporate priors of reflectance and natural lighting to assist in this challenging task. There has been some early work using priors of light sources for single image based color constancy [Finlayson et al., 2001]. Lalonde et al. [2009] have collected a database of natural lighting using webcams, which could be used to

explore the priors of natural lighting.

- *Learning* high-dimensional appearance phenomena (*e.g.*, BRDF) from Internet image collections. Nowadays, enormous numbers of photographs of the real world have been taken by users and made available on the web. This new data source contains rich information about our world, not only shape but also appearance. Similar to the work done by Snavely et al. [2006] which extracts shape information from photographs for browsing photos, one can imagine recovering material properties from images taken under various natural lighting and view conditions without calibration. As a further step in image understanding, extracted material information could be used for various applications, such as browsing, recognition and retrieval, and synthesis of novel appearance.

Appearance Modeling for Printing and Display: As mentioned at the beginning of this thesis, appearance modeling serves as the bridge between the physical world and the computer world. Over the last decade, we have seen tremendous progress in measuring data from the physical world (*e.g.*, geometry, reflectance, and illumination) and in reusing the data for photo-realistic rendering. Today with various design and analysis tools, the next logical step is to go in the opposite direction. This includes manufacturing physical objects that have the desired appearance designed by artists in computers, as well as developing novel display devices to demonstrate complex appearance phenomena (*e.g.*, light field). A few recent studies [Weyrich et al., 2009; Matusik et al., 2009] have started printing designed BRDF on surfaces. Fabrication of more complex appearance phenomena, such as BTF and subsurface scattering, are good topics to explore next.

These directions are not isolated. In fact, improvements in one direction will often significantly facilitate other directions. With the ever constant evolution of more powerful image sensors and computers, I believe that the study of appearance will help to seamlessly intertwine real world and virtual world.

Bibliography

Edward H. Adelson. On seeing stuff: The perception of materials by humans and machines. In *Human Vision and Electronic Imaging VI (SPIE)*, volume 4299, pages 1–12, 2001.

Edward H. Adelson and James R. Bergen. The plenoptic function and the elements of early vision. *Computational Models of Visual Processing*, pages 3–20, 1991.

Michael Ashikhmin and Peter Shirley. An anisotropic Phong BRDF model. *Journal of Graphics, GPU, and Game Tools*, 5(2):25–32, 2000.

Michael F. Barnsley, Robert L. Devaney, Benoit B. Mandelbrot, Heinz-Otto Peitgen, Dietmar Saupe, and Richard F. Voss. *The Science of Fractal Images*. Springer-Verlag, 1988.

Ken J. Beath. Infant growth modeling using a shape invariant model with random effects. *Statistics in Medicine*, 26(12):2547 – 2564, 2007.

Welton Becket and Norman I. Badler. Imperfection for realistic image synthesis. *The Journal of Visualization and Computer Animation*, 1:26–32, 1990.

Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of SIGGRAPH*, pages 417–424, New York, NY, USA, 2000. ACM.

James F Blinn. Light reflection functions for simulation of clouds and dusty surfaces. *Computer Graphics*, 16(3):21–29, July 1982.

James F. Blinn and M. E. Newell. Texture and reflection in computer generated images. *Communications of the ACM*, 19(10):542–547, October 1976.

M. Quinn Brewster. *Thermal Radiative Transfer and Properties*. Wiley-Interscience, 1992.

G. J. Burton and Ian R. Moorhead. Color and spatial structure in natural scenes. *Applied Optics*, 26(1):157–170, 1987. doi: 10.1364/AO.26.000157.

Emmanuel J. Candes and Justin Romberg. Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969–985, 2007.

Emmanuel J. Candes, Justin Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

Subrahmanyan Chandrasekhar. *Radiative Transfer*. Courier Dover Publications, New York, 1960.

Yanyun Chen, Lin Xia, Tien-Tsin Wong, Xin Tong, Hujun Bao, Baining Guo, and Heung-Yeung Shum. Visual simulation of weathering by gammaton tracing. *ACM Transactions on Graphics (SIGGRAPH)*, 24(3):1127–1133, 2005.

Robert L Cook and Kenneth E Torrance. A reflectance model for computer graphics. *Computer*, 15:307–316, 1981.

M. C. Cross and P. Hohenberg. Pattern formation out of equilibrium. *Reviews of Modern Physics*, 65:851–1112, 1993.

Kristin J. Dana, Bram Van Ginneken, Shree K. Nayar, and Jan J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, January 1999.

Paul Debevec, Andreas Wenger, Chris Tchou, Andrew Gardner, Jamie Waese, and Tim Hawkins. A lighting reproduction approach to live-action compositing. *ACM Transactions on Graphics (SIGGRAPH)*, 21(3):547–556, 2002.

O. Demers. *Digital Texturing and Painting*. New Riders Publishing, 2001.

S. Deusch and T. Dracos. Time resolved 3D passive scalar concentration-field imagingby induced fluorescence (LIF) in moving liquids. *Measurement Science and Technology*, 12 (2):188–200, 2001.

Craig Donner, Tim Weyrich, Eugene d'Eon, Ravi Ramamoorthi, and Szymon Rusinkiewicz. A layered, heterogeneous reflectance model for acquiring and rendering human skin. *ACM Transactions on Graphics (SIGGRAPH)*, 2008.

David L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4): 1289–1306, 2006.

Julie Dorsey and Pat Hanrahan. Modeling and rendering of metallic patinas. In *Proceedings of SIGGRAPH*, pages 387–396, August 1996.

Julie Dorsey and Pat Hanrahan. Digital materials and virtual weathering. *Scientific American*, 282(2):46–53, Feb. 2000.

Julie Dorsey, Hans Kohling Pedersen, and Pat Hanrahan. Flow and changes in appearance. In *Proceedings of SIGGRAPH*, pages 411–420, August 1996.

Julie Dorsey, Alan Edelman, Henrik Wann Jensen, Justin Legakis, and Hans Kohling Pedersen. Modeling and rendering of weathered stone. In *Proceedings of SIGGRAPH*, pages 225–234, August 1999.

Julie Dorsey, Holly E. Rushmeier, and François X. Sillion. *Digital Modeling of Material Appearance*. Morgan Kaufmann, 2008.

David S. Ebert, F. Kenton Musgrave, Darwyn Peachey, Kenneth Perlin, and Steven Worley. *Texturing and Modeling: A Procedural Approach*. AP Professional, second edition edition, July 1998.

Alexei A. Efros and William T. Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of SIGGRAPH*, pages 341–346, New York, NY, USA, 2001. ACM.

Sergey Ershov, Konstantin Kolchin, and Karol Myszkowshi. Rendering pearlescent appearance based on paint-composition modeling. *Computer Graphics Forum (Eurographics)*, 20(3):227–238, 2001.

Raanan Fattal. Single image dehazing. *ACM Transactions on Graphics (SIGGRAPH)*, 27 (3):72:1–72:9, 2008.

Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. Visual simulation of smoke. In *Proceedings of SIGGRAPH*, pages 15–22, August 2001.

G. D. Finlayson, S. D. Hordley, and P. M. Hubel. Color by correlation: A simple, unifying framework for color constancy. *TPAMI*, 23(11):1209–1221, 2001. URL `http://www2.cmp.uea.ac.uk/Research/compvis/PhysicsBasedVision/`.

Christian Fuchs, Tongbo Chen, Michael Goesele, Holger Theisel, and Hans-Peter Seidel. Density estimation for dynamic volumes. *Computers and Graphics*, 31(2):205–211, 2007.

Martin Fuchs, Ramesh Raskar, Hans-Peter Seidel, and Hendrik P. A. Lensch. Towards passive 6D reflectance field displays. *ACM Transactions on Graphics (SIGGRAPH)*, 27 (3):58, 2008.

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)*, 2006.

K. Garg, G. Krishnan, and S.K. Nayar. Material based splashing of water drops. In *Eurographics Symposium on Rendering (EGSR)*, June 2007.

Kshitiz Garg and Shree K. Nayar. Detection and removal of rain from videos. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 528–535, 2004.

Kshitiz Garg and Shree K. Nayar. Photorealistic rendering of rain streaks. *ACM Transactions on Graphics (SIGGRAPH)*, 25(3):996–1002, 2006.

T. Gasser, H. G. Muller, and V. Mammitzsch. Kernels for nonparametric curve estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 47(2):238–252, 1985.

C. Gini. Measurement of inequality of incomes. *The Economic Journal*, pages 124–126, 1921.

Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. In *Proceedings of SIGGRAPH*, pages 43–54. ACM New York, NY, USA, 1996.

Jinwei Gu, Ravi Ramamoorthi, Peter N. Belhumeur, and Shree K. Nayar. Dirty glass: Rendering contamination on transparent surfaces. In *Eurographics Symposium on Rendering (EGSR)*, pages 159–170, June 2007.

Pat Hanrahan and Wolfgang Krueger. Reflection from layered surfaces due to subsurface scattering. In *Proceedings of SIGGRAPH*, pages 165–174, 1993.

Samuel W. Hasinoff and Kiriakos N. Kutulakos. Photo-consistent reconstruction of semi-transparent scenes by density sheet decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):870–885, 2007.

Tim Hawkins, Per Einarsson, and Paul Debevec. Acquisition of time-varying participating media. *ACM Transactions on Graphics (SIGGRAPH)*, 24(3):812–815, 2005.

Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009.

Xiao D He, Kenneth E Torrance, François X Sillion, and Donar P Greenberg. A comprehensive physical model for light reflection. *Computer Graphics*, 25:175–186, 1991.

L. G. Henyey and J. L. Greenstein. Diffuse radiation in the galaxy. *Annales d'Astrophysique*, 3:117, 1940.

Siu-Chi Hsu and Tien-Tsin Wong. Simulating dust accumulation. *IEEE Computer Graphics & Applications*, 15(1):18–22, January 1995.

Richard Hughes and Michael Rowe. *The Colouring, Bronzing and Patination of Metals*. Watson-Guptill Publications, New York, 1991.

Niall Hurley and Scott Richard. Comparing measures of sparsity. In *IEEE Workshop on Machine Learning for Signal Processing (MLSP)*, pages 55–60, Oct 2008.

Ivo Ihrke and Marcus Magnor. Image-based tomographic reconstruction of flames. In *ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA)*, pages 367–375, 2004.

Ivo Ihrke and Marcus Magnor. Adaptive grid optical tomography. *Graphical Models*, 68(5): 484–495, 2006.

Ivo Ihrke, Kiriakos N. Kutulakos, Hendrik P. A. Lensch, Marcus Magnor, and Wolfgang Heidrich. State of the art in transparent and specular object reconstruction. In *STAR Proceedings of Eurographics*, 2008.

Akira Ishimaru. *Wave Propagation and Scattering in Random Media*. IEEE/OUP Series on Electromagnetic Wave Theory. IEEE Press, New York, 1978. ISBN 0-7803-3409-4.

S. L. Jacques, C. A. Alter, and S. A. Prahl. Angular dependence of HeNe laser light scattering by human dermis. *Lasers Life Science*, 1:309–333, 1987.

Ivaldo P Jankowsky and Gllson Roberto V Dos Santos. Drying behavior and permeability of eucalyptus grandis lumber. In *Proceedings of International Drying Symposium (IDS)*, pages 1385–1389, 2004.

Hendrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A practical model for subsurface light transport. In *Proceedings of SIGGRAPH*, pages 511–518, 2001.

Henrik Wann Jensen. *Realistic Image Synthesis Using Photon Mapping*. A K Peters, Ltd., Wellesley, MA, 2001.

Andrew Jones, Ian McDowall, Hideshi Yamada, Mark Bolas, and Paul Debevec. Rendering for an interactive 360 light field display. *ACM Transactions on Graphics (SIGGRAPH)*, 26(3):40, 2007. URL `http://gl.ict.usc.edu/Research/3DDisplay/`.

J. H. Joseph, W. J. Wiscombe, and J. A. Weinman. The Delta-Eddington approximation for radiative flux transfer. *Journal of the Atmospheric Sciences*, 33(12):2452–2459, 1976.

Alois Kneip and Joachim Engel. Model estimation in nonlinear regression under shape invariance. *Annals of Statistics*, 23(2):551–570, April 1995.

Alois Kneip and Theo Gasser. Convergence and consistency results for self-modeling nonlinear regression. *The Annuals of Statistics*, 16(1):82–112, 1988.

Jan J Koenderink and Sylvia C Pont. The secret of velvety skin. *Machine Vision and Applications*, 14:260–268, 2003.

Rajiv Kohli and Kashmiri L. Mittal, editors. *Developments in Surface Contamination and Cleaning: Fundamentals and Applied Aspects*. William Andrew, 2008.

Melissa L Koudelka. *Capture, Analysis and Synthesis of Textured Surfaces With Variation in Illumination, Viewpoint and Time*. PhD thesis, Yale University, 2004.

Sujit Kuthirummal, Aseem Agarwala, Dan B. Goldman, and Shree K. Nayar. Priors for large photo collections and what they reveal about cameras. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 74–87, October 2008.

Vivek Kwatra, Arno Schodl, Irfan Essa, Greg Turk, and Aaron Bobick. Graphcut textures: Image and video synthesis using graph cuts. *ACM Transactions on Graphics (SIGGRAPH)*, 22(3):277–286, July 2003.

Eric P F Lafortune and Kenneth E Torrance. Non-linear approximation of reflectance functions. In *Proceedings of SIGGRAPH*, pages 117–126, New York, New York, USA, 1997. ACM Press. ISBN 0897918967. doi: 10.1145/258734.258801. URL `http://portal.acm.org/citation.cfm?doid=258734.258801`.

Jean-François Lalonde, Alexei A. Efros, and Srinivasa G. Narasimhan. Webcam clip art: Appearance and illuminant transfer from time-lapse sequences. 28(5), December 2009.

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. *ACM Transactions on Graphics (SIGGRAPH)*, 25 (3):735–745, 2006.

W. H. Lawton, E. A. Sylvestre, and M. S. Maggio. Self modeling nonlinear regression. *Technometrics*, 14(3):513–532, August 1972.

Anat Levin, Rob Fergus, Frédo Durand, and William T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics (SIGGRAPH)*, 26(3):70:1–70:10, 2007.

Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of SIGGRAPH*, 1996.

Yanxi Liu, Tamara Belkina, James H. Hays, and Roberto Lublinerman. Image de-fencing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2008.

Jianye Lu, Athinodoros S. Georghiades, Holly Rushmeier, Julie Dorsey, and Chen Xu. Synthesis of material drying history: Phenomenon modeling, transfering, and rendering. In *Eurographics Workshop on Natural Phenomena*, August 2005.

Jianye Lu, Athinodoros S. Georghiades, Andreas Glaser, Hongzhi Wu, Li-Yi Wei, Baining Guo, Julie Dorsey, and Holly Rushmeier. Context-aware textures. *ACM Transactions on Graphics*, 26:3:1–3:22, 2007.

Michael Lustig, David Donoho, and John M. Pauly. Sparse MRI: The application of compressed sensing for rapid MRI imaging. *Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007.

Stephen R. Marschner, Stephen H. Westin, Adam Arbree, and Jonathan T. Moon. Measuring and modeling the appearance of finish wood. *ACM Transactions on Graphics (SIGGRAPH)*, 24(3):724–734, 2005.

Vincent Masselus, Pieter Peers, Philip Dutre, and Yves D. Willems. Relighting with 4D incident light fields. *ACM Transactions on Graphics (SIGGRAPH)*, 22:613–620, 2003.

Wojciech Matusik, Hanspeter Pfister, Remo Ziegler, Addy Ngan, and Leonard McMillan. Acquisition and rendering of transparent and refractive objects. In *Eurographics Workshop on Rendering*, 2002.

Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Transactions on Graphics (SIGGRAPH)*, 22(3):759–769, 2003.

Wojciech Matusik, Boris Ajdin, Jinwei Gu, Jason Lawrence, Hendrik P.A. Lensch, Fabio Pellacini, and Szymon Rusinkiewicz. Printing spatially-varying reflectance. 27:128, 2009.

Hans Meinhardt. Pattern formation in biology: a comparison of models and experiments. *Reports on Progress in Physics*, 55:797–849, 1992.

S. Merillou and D. Ghazanfarpour. A survey of aging and weathering phenomena in computer graphics. *Computers & Graphics*, 32:159–174, 2008.

Stephane Merillou, Jean-Michal Dischler, and Djamchid Ghazanfarpour. Corrosion: Simulating and rendering. In *Graphics Interface*, Ottawa, Canada, 2001.

William Edgar Knowles Middleton. *Vision Through the Atmosphere*. University of Toronto Press, 1952.

Gavin Miller. Efficient algorithms for local and global accessibility shading. In *Proceedings of SIGGRAPH*, pages 319–326, 1994.

Gene S. Miller and C. Robert Hoffman. Illumination and reflection maps: Simulated objects in simulated and real environments. In *Course Notes for Advanced Computer Graphics Animation, SIGGRAPH 84.*, 1984.

Srinivasa G. Narasimhan and Shree K. Nayar. Shedding light on the weather. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 665–672, 2003a.

Srinivasa G. Narasimhan and Shree K. Nayar. Contrast restoration of weather degraded images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):713–724, 2003b.

Srinivasa G. Narasimhan, Chi Wang, and Shree K. Nayar. All the images of an outdoor scene. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume III, pages 148–162, May 2002.

Srinivasa G. Narasimhan, Mohit Gupta, Craig Donner, Ravi Ramamoorthi, Shree K. Nayar, and Henrik Wann Jensen. Acquiring scattering properties of participating media by dilution. *ACM Transactions on Graphics (SIGGRAPH)*, 25(3):1003–1012, 2006.

Shree K. Nayar, Peter N. Belhumeur, and Terry E. Boult. Lighting sensitive display. *ACM Transactions on Graphics*, 23(4):963–979, October 2004.

Shree K. Nayar, Gurunandan Krishnan, Michael D. Grossberg, and Ramesh Raskar. Fast separation of direct and global components of a scene using high frequency illumination. *ACM Transactions on Graphics (SIGGRAPH)*, 25(3):935–944, 2006.

F. E. Nicodemus, J. C. Richmond, I. W. Ginsberg, and T. Limperis. Geometrical considerations and nomenclature for reflectance. *National Bureau of Standards Monograph*, 160:201–231, October 1977. ISSN 1041-1135. doi: 10.1109/LPT.2009.2020494. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4812360`.

Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

Michael Oren and Shree K Nayar. Generalization of lambert's reflectance model. In *Proceedings of SIGGRAPH*, pages 239–246, 1994.

Pieter Peers, Karl vom Berge, Wojciech Matusik, Ravi Ramamoorthi, Jason Lawrence, , Szymon Rusinkiewicz, and Philip Dutre. A compact factored representation of heterogeneous subsurface scattering. *ACM Transactions on Graphics (SIGGRAPH)*, 25(3): 746–753, 2006.

Pieter Peers, Dhruv K. Mahajan, Bruce Lamond, Abhijeet Ghosh, Wojciech Matusik, Ravi Ramamoorthi, and Paul Debevec. Compressive light transport sensing. *ACM Transactions on Graphics*, 28(3:1-3:18):1289–1306, 2009.

Kenneth Perlin. An image synthesizer. In *Proceedings of SIGGRAPH*, 1985.

Kenneth Perlin and Eric M. Hoffert. Hypertexture. In *Proceedings of SIGGRAPH*, volume 23, pages 253–262, July 1989.

Matt Pharr and Greg Humphreys. *Physically Based Rendering: From Theory to Implementation*. Morgan Kaufmann, 2004.

Scott A. Prahl. *Light Transport in Tissue*. PhD thesis, University of Texas at Austin, 1988.

Przemyslaw Prusinkiewicz and Aristid Lindenmayer. *The Algorithmic Beauty of Plants*. Springer-Verlag, 1990.

Ramesh Raskar, Amit Agrawal, Cyrus A. Wilson, and Ashok Veeraraghavan. Glare aware photography: 4D ray sampling for reducing glare effects of camera lenses. *ACM Transactions on Graphics (SIGGRAPH)*, 27(3):56:1–56:10, 2008.

Bill Romanishin. *An Introduction to Astronomical Photometry Using CCDs*. 2006. URL `http://www.observatory.ou.edu/book2513.html`.

Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992. ISSN 0167-2789. doi: http://dx.doi.org/10.1016/0167-2789(92)90242-F.

H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustic Speech and Signal Processing*, 26:43–49, 1978.

Yoav Y. Schechner and Nir Karpel. Recovery of underwater visibility and structure by polarization analysis. *IEEE Journal of Oceanic Engineering*, 30(3):570–587, 2005.

Yoav Y. Schechner, Srinivasa G. Narasimhan, and Shree K. Nayar. Instant dehazing of images using polarization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 325–332, 2001.

Yoav Y. Schechner, Srinivasa G. Narasimhan, and Shree K. Nayar. Polarization-based vision through haze. *Applied Optics*, 42(3):511–525, 2003a.

Yoav Y. Schechner, Shree K. Nayar, and Peter N. Belhumeur. A theory of multiplexed illumination. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Nice, France, October 2003b.

Arno Schodl, Richard Szeliski, David H. Salesin, and Irfan Essa. Video textures. In *Proceedings of SIGGRAPH*, pages 489–498, New Orleans, Louisiana, July 2000.

Will Schroeder, Ken Martin, and Bill Lorensen. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Pearson Eduction, Inc., 4th edition, 2006.

Eero P Simoncelli. Statistical models for images: Compression, restoration and synthesis. In *Proceedings of Asilomar Conference on Signals, Systems and Computers*, pages 673–678, 1997.

Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3d. *ACM Transactions on Graphics (SIGGRAPH)*, 25(3):835–846, 2006. URL `http://phototour.cs.washington.edu/`.

Stefano Soatto, Gianfranco Doretto, and Yingnian Wu. Dynamic textures. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pages 439–446, Vancouver, Canada, July 2001.

Jos Stam. Stable fluids. In *Proceedings of SIGGRAPH*, pages 101–110, August 1999.

Jos Stam. An illumination model for a skin layer bounded by rough surfaces. In *Eurographics Symposium on Rendering (EGSR)*, pages 39–52, 2001.

John C. Stover. *Optical Scattering: Measurement and Analysis*. McGraw-Hill, New York, 1990.

W. Stützle, T. Gasser, L. Molinari, RH Largo, A. Prader, and PJ Huber. Shape-invariant modelling of human growth. *Annals of Human Biology*, 7(6):507–528, 1980.

Bo Sun, Kalyan Sunkavalli, Ravi Ramamoorthi, Peter N. Belhumeur, and Shree K. Nayar. Time-varying BRDFs. *IEEE Transactions on Visualization and Computer Graphics*, 13 (3):595–609, 2006.

Jian Sun, Lu Yuan, Jiaya Jia, and Heung-Yeung Shum. Image completion with structure propagation. *ACM Transactions on Graphics (SIGGRAPH)*, 24(3):861–868, 2005.

Martin Szummer and Rosalind W. Picard. Temporal texture modeling. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 823–826, Lausanne, Switzerland, September 1996.

D. Takhar, J. Laska, M. Walkin, M. Durate, and D.Baron. A new compressive imaging camera architecture using optical-domain compression. In *Computational Imaging IV at SPIE Electronic Imaging*, 2006.

Eino-Ville Talvala, Andrem Adams, Mark Horowitz, and Marc Levoy. Veiling glare in high dynamic range imaging. *ACM Transactions on Graphics (SIGGRAPH)*, 26(3):37:1–37:10, 2007.

Antonio Torralba and Aude Oliva. Statistics of natural image categories. *Network Computation in Neural Systems*, 14(3):391–412, August 2003. ISSN 0954-898X. URL `http://www.ncbi.nlm.nih.gov/pubmed/12938764`.

Antonio Torralba, Rob Fergus, and William T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.

Tali Treibitz and Yoav Y. Schechner. Recovery limits in pointwise degradation. In *Proceedings of IEEE International Conference on Computational Photography (ICCP)*, 2009.

Borislav Trifonov, Derek Bradley, and Wolfgang Heidrich. Tomographic reconstruction of transparent objects. In *Eurographics Symposium on Rendering (EGSR)*, pages 51–60, 2006.

Greg Turk. Generating textures on arbitrary surfaces using reaction-diffusion. In *ACM Transactions on Graphics (SIGGRAPH)*, pages 289–298, July 1991.

Jonas Unger, Andreas Wenger, Tim Hawkins, Andrew Gardner, and Paul Debevec. Capturing and rendering with incident light fields. In *Eurographics Symposium on Rendering (EGSR)*, 2003.

Eric Veach. *Robust Monte Carlo Methods for Light Transport Simulation.* PhD thesis, Standford University, 1998.

Daniel Vlasic, Hanspeter Pfister, Sergey Molinov, Radek Grzeszczuk, and Wojciech Matusik. Opacity light fields: Interactive rendering of surface light fields with view-dependent opacity. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D)*, pages 65–74, 2003.

Kongming Wang and Theo Gasser. Alignment of curves by dynamic time warping. *Annals of Statistics*, 25(3):1251–1276, 1997.

Kongming Wang and Theo Gasser. Synchronizing sample curves nonparametrically. *Annals of Statistics*, 27(2):439–460, 1999.

Lihong Wang, Steven L. Jacques, and Liqiong Zheng. MCML–Monte Carlo modeling of light transport in multi-layered tissues. *Computer Methods and Programs in Biomedicine*, 47: 131–146, 1995.

Yuedong Wang, Chunlei Ke, and Morton B. Brown. Shape-invariant modeling of circadian rhythms with random effects and smoothing spline ANOVA decompositions. *Biometrics*, 59(4):804–812, December 2003.

Gregory J Ward. Measuring and modeling anisotropic reflection. *Computer Graphics*, 26: 265–272, 1992.

Andreas Wenger, Andrew Gardner, Chris Tchou, Jonas Unger, Tim Hawkins, and Paul Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *ACM Transactions on Graphics (SIGGRAPH)*, 24(3):756–764, 2005.

Tim Weyrich, Pieter Peers, Wojciech Matusik, and Szymon Rusinkiewicz. Fabricating microgeometry for custom surface reflectance. *ACM Transactions on Graphics (SIGGRAPH)*, 28(3):32, 2009.

Turner Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 23:343–349, 1980.

R. Willett, M. Gehm, and D. Brady. Multiscale reconstruction for computational spectral imaging. In *Computational Imaging V at SPIE Electronic Imaging*, 2007.

Reg G. Willson, Mark W. Maimone, Andrew E. Johnson, and Larry M. Scherr. An optical model for image artifacts produced by dust particles on lenses. In *International Symposium on Artificial Intelligence, Robtics and Automation in Space (i-SAIRAS)*, 2005.

Andy Witkin and Michael Kass. Reaction-diffusion textures. In *Proceedings of SIGGRAPH*, pages 299–308, July 1991.

Daniel Wood, Daniel Azuma, Wyvern Aldinger, Brian Curless, Tom Duchamp, David Salesin, and Werner Stuetzle. Surface light fields for 3D photography. In *Proceedings of SIGGRAPH*, 2000. URL `http://grail.cs.washington.edu/projects/slf/`.

John Wright, Allen Y. Yang, Arvind Ganesh, Shankar S. Sastry, and Yi Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):210–227, 2009.

Chris Wyman. An approximate image-space approach for interactive refraction. *ACM Transactions on Graphics (SIGGRAPH)*, 24(3):1050–1053, August 2005.

Renliang Xu. *Particle Characterization: Light Scattering Methods*. Kluwer Academic, 2002.

Changyin Zhou and Stephen Lin. Removal of image arifacts due to sensor dust. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

Appendix A

Mirror Reflection and Refraction

When light strikes on the interface of two media, part of the light will be reflected and part of the light will be refracted into the media. In the simplest case of a perfectly smooth interface, both the reflection and refraction have singularity, *i.e.*., only one direction will see the reflected light and refracted light. Because of this singularity, it sometimes causes confusion in writing the formula of the BRDF and BTDF for this type of light-surface interaction. Moreover, many real-world surfaces often have a mirror reflecting layer on the top, such as finished wood or contaminated glass, and thus mirror reflection and refraction are necessary to construct more complex BRDF/BTDF models.

In this appendix, we summarize the physical principles and quantities involved in modeling mirror reflection and refraction. We derive their BRDF and BTDF and show how they both satisfy the two fundamental constraints — energy conservation and the generalized Helmholtz reciprocity. This appendix thus serves as background for the topics discussed in Chapter 4 and Chapter 5.

A.1 Snell's Law and Magnification Ratio

Let us first focus on the relationship of the incident angle and the outgoing angle for mirror reflection and refraction. As shown in Figure A.1, the polar angle θ_i of the incident light L_i and the polar angle θ_r of the mirror reflected light L_r have the following simple relationship:

$$\theta_r = \theta_i, \quad \text{and} \quad \phi_r = \phi_i \pm \pi. \tag{A.1}$$

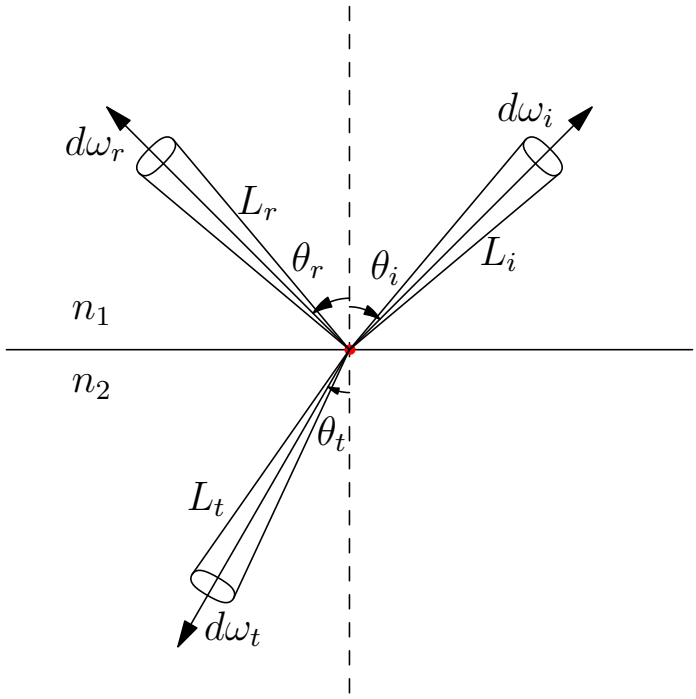


Figure A.1: Coordinates for mirror reflection and refraction.

For refraction, the relationship between θ_i and θ_t is governed by Snell's law (also known as the law of refraction):

$$n_1 \sin\theta_i = n_2 \sin\theta_t, \tag{A.2}$$

where n_1 and n_2 are the refraction indices of the two media. For the azimuth angles, we have:

$$\phi_t = \phi_i \pm \pi. \tag{A.3}$$

As shown in Figure A.2, the incident beam will become "wider" after it enters a more dense media (*e.g.*, from air into glass or water). Let A_i denote the cross section of the incident beam, A_t denote the cross section of the refracted beam, and A denote the area on the interface, we have:

$$\frac{A_i}{A_t} = \frac{A \cdot \cos\theta_i}{A \cdot \cos\theta_t} = \frac{\cos\theta_i}{\cos\theta_t}. \tag{A.4}$$

This ratio often is called "magnification factor" in optics.

Based on this equation, we can derive an important relationship between the solid angle of the incident beam, $d\omega_i$, and the solid angle of the refracted beam, $d\omega_t$. Based on the

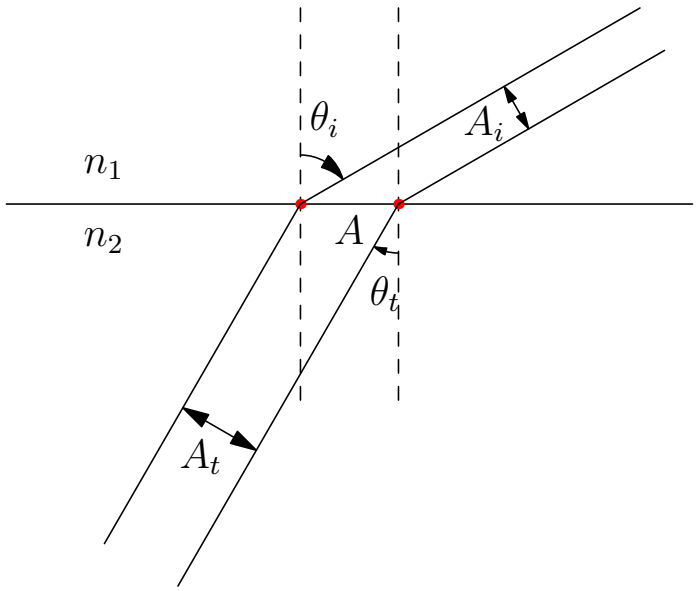


Figure A.2: Magnification in refraction.

definition of solid angle, we have

$$d\omega = \sin\theta \cdot d\theta d\phi. \tag{A.5}$$

From Snell's law, since $\phi_t = \phi_i \pm \pi$, we have $d\phi_t = d\phi_i$. Also, by taking the differential on Snell's law equation, we have

$$n_1 \cdot \cos\theta_i \cdot d\theta_i = n_2 \cdot \cos\theta_t \cdot d\theta_t. \tag{A.6}$$

Based on these two equations, we have

$$\begin{aligned} \frac{d\omega_i}{d\omega_t} &= \frac{\sin\theta_i \cdot d\theta_i \cdot d\phi_i}{\sin\theta_t \cdot d\theta_t \cdot d\phi_t} = \frac{\sin\theta_i \cdot d\theta_i}{\sin\theta_t \cdot d\theta_t} \\ &= \frac{\sin\theta_i \cdot n_2 \cos\theta_t}{\sin\theta_t \cdot n_1 \cos\theta_i} = \frac{n_2 \cdot n_2 \cos\theta_t}{n_1 \cdot n_1 \cos\theta_i} = \frac{n_2^2 \cos\theta_t}{n_1^2 \cos\theta_i}. \end{aligned} \tag{A.7}$$

This relationship between the two solid angles will be useful for the derivation of the BRDF and BTDF for mirror reflection and refraction, as shown below.

A.2 Fresnel Equations

Next, we focus on the relationship between the "energy" of the incident light and the outgoing light for mirror reflection and refraction. This relationship is described by Fresnel

equations, which are derived based on Maxwell equations [Ishimaru, 1978]:

$$r_\perp = \frac{E_{0t,\perp}}{E_{0i}} = \frac{\cos\theta_i - \sqrt{n^2 - \sin^2\theta_i}}{\cos\theta + \sqrt{n^2 - \sin^2\theta}} = -\frac{\sin(\theta_i - \theta_t)}{\sin(\theta_i + \theta_t)}, \tag{A.8}$$

$$r_\parallel = \frac{E_{0r,\parallel}}{E_{0i}} = \frac{n^2\cos\theta_i - \sqrt{n^2 - \sin^2\theta_i}}{n^2\cos\theta + \sqrt{n^2 - \sin^2\theta}} = \frac{\tan(\theta_i - \theta_t)}{\tan(\theta_i + \theta_t)}, \tag{A.9}$$

$$t_\perp = \frac{E_{0t,\perp}}{E_{0i}} = \frac{2\cos\theta_i}{\cos\theta_i + \sqrt{n^2 - \sin^2\theta_i}} = \frac{2\sin\theta_t\cos\theta_i}{\sin(\theta_i + \theta_t)}, \tag{A.10}$$

$$t_\parallel = \frac{E_{0t,\parallel}}{E_{0i}} = \frac{2n\cos\theta_i}{n^2\cos\theta_i + \sqrt{n^2 - \sin^2\theta_i}} = \frac{2\sin\theta_t\cos\theta_i}{\sin(\theta_i + \theta_t)\cos(\theta_i - \theta_t)}, \tag{A.11}$$

where $n = n_2/n_1$, and E_{0i}, E_{0r}, and E_{0t} are the field complex amplitudes of the incident, reflected, and transmitted electromagnetic waves. These equations give the ratios between the field complex amplitudes between the incident, reflected, and transmitted electromagnetic waves, for parallel polarized ($\parallel$) and perpendicular polarized light ($\perp$), respectively. These ratios are often referred to as "reflection coefficients" (*i.e.*, $r_\perp$ and $r_\parallel$) and "transmission coefficients" (*i.e.*, $t_\perp$ and $t_\parallel$).

Based on Maxwell's equations, the flux density of light can be computed from electromagnetic waves' field complex amplitude as follows:

$$I = \left(n\frac{\epsilon_0 c_0}{2}\right)|E_0|^2 \qquad [\mathrm{W}\cdot\mathrm{m}^{-2}] \tag{A.12}$$

where n is refraction index, c_0 is the speed of light, ϵ_0 is the permittivity of the matter. Based on this equation, we can compute the ratio between the power of the reflected light $d\Phi_r$ and the power of the incident light $d\Phi_i$, *i.e.*, the reflectance in terms of power (also called "reflectivity"), is

$$R_p = \frac{d\Phi_r}{d\Phi_i} = \frac{I_r \cdot A_i}{I_i \cdot A_i} = \frac{I_r}{I_i} = |r|^2. \tag{A.13}$$

For unpolarized light,

$$R_p = \frac{1}{2}(r_\perp^2 + r_\parallel^2). \tag{A.14}$$

The transmittance in terms of power (also called "transmissivity"), T_p, which is the ratio between the power of the refracted light $d\Phi_t$ and the power of the incident light $d\Phi_i$, can be derived based on Equation (A.4):

$$T_p = \frac{d\Phi_t}{d\Phi_i} = \frac{I_t \cdot A_t}{I_i \cdot A_i} = |t|^2 \cdot \frac{n_2^2\cos\theta_t}{n_1^2\cos\theta_i}. \tag{A.15}$$

For unpolarized light,

$$T_p = \frac{1}{2}(t_\perp^2 + t_\parallel^2)\frac{n_2^2 \cos\theta_t}{n_1^2 \cos\theta_i}. \tag{A.16}$$

By substituting the above equations into the equation below, one can show that the energy is conserved as expected,

$$R_p + T_p = 1. \tag{A.17}$$

In computer graphics and computer vision, we often care about the relationship between the radiance of the incident light and the reflected/refracted light. To derive this relationship, we go back to the definition of radiance, and we have:

$$\begin{aligned} d\Phi_i &= L_i \cdot dA \cdot \cos\theta_i \cdot d\omega_i, \\ d\Phi_r &= L_r \cdot dA \cos\theta_r \cdot d\omega_r = R_p \cdot d\Phi_i, \\ d\Phi_t &= L_t \cdot dA \cos\theta_t \cdot d\omega_t = T_p \cdot d\Phi_i. \end{aligned}$$

We have derived the relationship between $d\omega_i$, $d\omega_r$ and $d\omega_t$ that $d\omega_i = d\omega_r$ and $d\omega_i/d\omega_t = n_2^2 \cos\theta_t/(n_1^2 \cos\theta_i)$. Substitute them into the above equation, we have:

$$L_r = R_p \cdot L_i, \tag{A.18}$$

and

$$L_t = \frac{\cos\theta_i}{\cos\theta_t} \cdot \frac{d\omega_i}{d\omega_t} \cdot T_p \cdot L_i = \frac{n_2^2}{n_1^2} T_p \cdot L_i. \tag{A.19}$$

Therefore, the reflectance in terms of radiance is:

$$R = R_p, \tag{A.20}$$

and the transmittance in terms of radiance is:

$$T = \frac{n_2^2}{n_1^2} T_p. \tag{A.21}$$

These quantities for reflection and refraction are summarized in Table A.2.

A.3 BRDF and BTDF of Mirror Reflection and Refraction

BRDF for Mirror Reflection: From [Nicodemus et al., 1977] Appendix C, page 44, a perfectly specular (mirror-like) or regular reflecting element dA is one for which each

Table A.1: Summary of physical quantities for refraction and reflection

	$s-$ polarized($\perp$)	$p-$ polarized($\parallel$)	unpolarized
Reflection coefficient	$r_s = \frac{n_1\cos\theta_i - n_2\cos\theta_t}{n_1\cos\theta_i + n_2\cos\theta_t}$	$r_p = \frac{n_1\cos\theta_t - n_2\cos\theta_i}{n_1\cos\theta_t + n_2\cos\theta_i}$	$r^2 = \frac{1}{2}(r_s^2 + r_p^2)$
Transmission coefficient	$t_s = \frac{2n_1\cos\theta_i}{n_1\cos\theta_i + n_2\cos\theta_t}$	$t_p = \frac{2n_1\cos\theta_i}{n_1\cos\theta_t + n_2\cos\theta_i}$	$t^2 = \frac{1}{2}(t_s^2 + t_p^2)$
Reflectance (Reflectivity)	$R_p = \lvert r_s\rvert^2$	$R_p = \lvert r_p\rvert^2$	$R_p = \lvert r\rvert^2$
Transmittance (Transmissivity)	$T_p = \frac{n_2^2\cos\theta_t}{n_1^2\cos\theta_i}\lvert t_s\rvert^2$	$T_p = \frac{n_2^2\cos\theta_t}{n_1^2\cos\theta_i}\lvert t_p\rvert^2$	$T_p = \frac{n_2^2\cos\theta_t}{n_1^2\cos\theta_i}\lvert t\rvert^2$
Reflectance (Radiance)	$R = \lvert r_s\rvert^2$	$R = \lvert r_p\rvert^2$	$R = \lvert r\rvert^2$
Transmittance (Radiance)	$T = \frac{n_2^3\cos\theta_t}{n_1^3\cos\theta_i}\lvert t_s\rvert^2$	$T = \frac{n_2^3\cos\theta_t}{n_1^3\cos\theta_i}\lvert t_p\rvert^2$	$T = \frac{n_2^3\cos\theta_t}{n_1^3\cos\theta_i}\lvert t\rvert^2$

ray of incident flux produces only a corresponding reflected ray in the specular direction $(\theta_r = \theta_i; \phi_r = \phi_i \pm \pi)$ [rad], so that (assume lossless reflector)

$$L_r(\theta_r, \phi_r) = L_i(\theta_r, \phi_r \pm \pi) \quad [\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1}]. \tag{A.22}$$

From the definition of BRDF, we know it needs to satisfy the following equation

$$L_r(\theta_r, \phi_r) = \int_\omega L_i(\theta_i, \phi_i) \cdot f_r(\theta_i, \phi_i; \theta_r, \phi_r) \cdot \cos\theta_i \cdot d\omega_i \quad [\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1}]. \tag{A.23}$$

By using the Dirac delta-function $\delta(\cdot)$, the BRDF for an ideal (lossless) specular reflector can be written as:

$$f_r(\theta_i, \phi_i; \theta_r, \phi_r) = 2 \cdot \delta(\sin^2\theta_r - \sin^2\theta_i) \cdot \delta(\phi_r - \phi_i \pm \pi) \quad [\mathrm{sr}^{-1}]. \tag{A.24}$$

We note that Helmholtz reciprocity holds for this BRDF, since by swapping θ_i and θ_r the formula remains the same.

BTDF for Refraction: We now investigate the BTDF for refraction. Similar to the mirror reflection, for refraction, each ray of incident flux produces only a corresponding refracted ray in the refracted direction, so we have

$$L_t(\theta_t, \phi_t) = T_{12} \cdot L_i(\sin^{-1}(n_2 \sin\theta_t / n_1), \phi_t \pm \pi) \quad [\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1}], \tag{A.25}$$

where $T_{12} = n_2^2 T_p / n_1^2$, and T_p is the transmittance defined as $1 - R_p$. By using the Dirac delta-function, the BTDF for refraction can be written as:

$$f_t(\theta_i, \phi_i; \theta_t, \phi_t) = 2 \cdot T_{12} \cdot \delta(n_2^2 \sin^2\theta_t / n_1^2 - \sin^2\theta_i) \cdot \delta(\phi_t - \phi_i \pm \pi) \quad [\mathrm{sr}^{-1}]. \tag{A.26}$$

To show that this function satisfy the generalized Helmholtz reciprocity, we need to prove

$$\frac{f_t(\theta_i, \phi_i; \theta_t, \phi_t)}{n_2^2} = \frac{f_t(\theta_t, \phi_t; \theta_i, \phi_i)}{n_1^2} \tag{A.27}$$

We note that the reflectance R_p and thus the transmittance $T_p = 1 - R_p$ are the same in both $f_t(\theta_i, \phi_i; \theta_t, \phi_t)$ and $f_t(\theta_t, \phi_t; \theta_i, \phi_i)$ so that T_p can be canceled from both sides. Moreover, for the Dirac delta-function, $\delta(x)$, we have

$$\int \delta(ax)dx = (1/a) \cdot \int \delta(ax)d(ax) = 1/a, \tag{A.28}$$

and thus we have

$$\delta(ax) = \delta(x)/a. \tag{A.29}$$

Therefore, we have

$$\delta(n_2^2 \sin^2\theta_t/n_1^2 - \sin^2\theta_i) = \frac{n_1^2}{n_2^2}\delta(\sin^2\theta_t - n_1^2\sin^2\theta_i/n_2^2), \tag{A.30}$$

which means

$$\frac{f_t(\theta_i,\phi_i;\theta_t,\phi_t)}{f_t(\theta_i,\phi_i;\theta_t,\phi_t)} = \frac{T_{12}\cdot\delta(n_2^2\sin^2\theta_t/n_1^2 - \sin^2\theta_i)}{T_{21}\cdot\delta(\sin^2\theta_t - n_1^2\sin^2\theta_i/n_2^2)} = \frac{n_2^2}{n_1^2}\cdot\frac{n_2^2}{n_1^2}\cdot\frac{n_1^2}{n_2^2} = \frac{n_2^2}{n_1^2}. \tag{A.31}$$

Therefore, we have

$$\frac{f_t(\theta_i,\phi_i;\theta_t,\phi_t)}{n_2^2} = \frac{f_t(\theta_i,\phi_i;\theta_t,\phi_t)}{n_1^2}. \tag{A.32}$$

This proves the generalized Helmholtz reciprocity holds for the BTDF of refraction.

A.4 BRDF for a Laminated Surface

In this section, we look at one special type of surfaces that are commonly present in our daily lives — a surface composed with an opaque surface underneath and a transparent layer on the top. Examples include photo frames, laminated floors, and various types of reflective plastics. One interesting effect of these surfaces is the Fresnel effect — when viewing at a glancing angle, we will see more of the mirror reflection and less of the underlying color. Ashikhmin and Shirley [2000] studied this phenomena for the reflectance of a diffuse surface covered with a glossy specular layer on the top. Here we focus on the surface composed with a general opaque surface (both diffuse or specular) underneath and a transparent layer on the top. Our goal is to show that the BRDF for the composed surface will also satisfy the Helmholtz reciprocity constraint.

Figure A.3 shows a diagram of an opaque surface covered with a transparent layer on the top. We assume the BRDF for the opaque surface is $f_0(\cdot)$, which satisfies the Helmholtz reciprocity. With a transparent layer on the top, there will be two components in the reflected light — the perfect reflection from the top layer, which we have shown satisfies the Helmholtz reciprocity earlier, and the light that transmits into the top layer and reflects from the bottom layer (*i.e.*, the opaque surface) and then refracts to the outside. Below, we

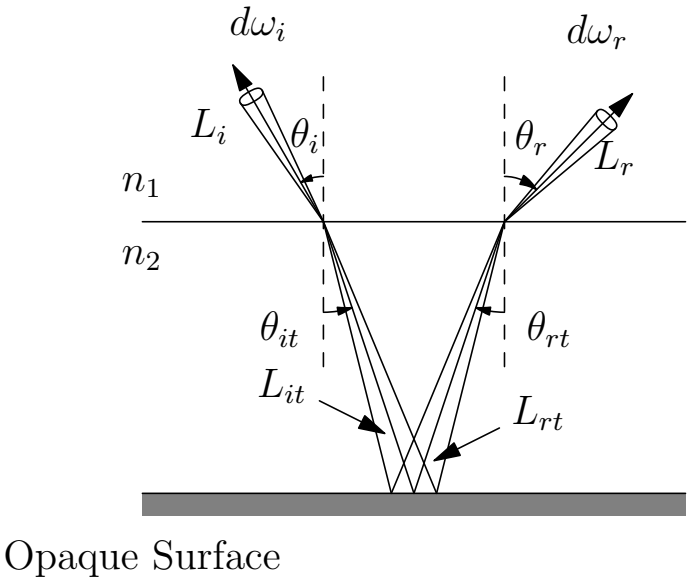


Figure A.3: Surfaces composed of a transparent top layer and an opaque surface at the bottom layer.

show that the second component results in a BRDF that satisfies the generalized Helmholtz reciprocity.

Let n_1 and n_2 be the refraction indices of air and the top layer. For a beam of incident light $L_i(\theta_i, \phi_i)$, we have

$$L_{it}(\theta_{it}, \phi_{it}) = \frac{n_2^2}{n_1^2} \cdot T_p(\theta_i) \cdot L_i(\theta_i, \phi_i). \tag{A.33}$$

and

$$L_{rt}(\theta_{rt}, \phi_{rt}) = \frac{n_2^2}{n_1^2} \cdot T_p(\theta_r) \cdot L_r(\theta_r, \phi_r). \tag{A.34}$$

Therefore,

$$dL_r(\theta_r, \phi_r) = \frac{n_1^2}{n_2^2} \cdot T_p(\theta_r) \cdot dL_{rt}(\theta_{rt}, \phi_{rt}). \tag{A.35}$$

As a result, the reflected radiance of the opaque surface in the direction of (θ_{rt}, ϕ_{rt}) is:

$$dL_{rt}(\theta_{rt}, \phi_{rt}) = dE_{it} \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi rt) \tag{A.36}$$

This irradiance, dE_{it}, by definition, also equals to

$$dE_{it} = L_{it}(\theta_{it}, \phi_{it}) \cdot \cos\theta_{it} \cdot d\omega_{it} \tag{A.37}$$

From the definition of BRDF, we have

$$
\begin{aligned}
f(\theta_i, \phi_i; \theta_r, \phi_r) &= \frac{dL_r(\theta_r, \phi_r)}{L_i(\theta_i, \phi_i)\cos\theta_i d\omega_i} \\
&= \frac{\frac{n_1^2}{n_2^2} \cdot T_p(\theta_r) \cdot dL_{rt}(\theta_{rt}, \phi_{rt})}{L_i(\theta_i, \phi_i)\cos\theta_i d\omega_i} \\
&= \frac{\frac{n_1^2}{n_2^2} \cdot T_p(\theta_r) \cdot dE_{it} \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi rt)}{L_i(\theta_i, \phi_i)\cos\theta_i d\omega_i} \\
&= \frac{\frac{n_1^2}{n_2^2} \cdot T_p(\theta_r) \cdot L_{it}(\theta_{it}, \phi_{it}) \cdot \cos\theta_{it} \cdot d\omega_{it} \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi_{rt})}{L_i(\theta_i, \phi_i)\cos\theta_i d\omega_i} \\
&= \frac{\frac{n_1^2}{n_2^2} \cdot T_p(\theta_r) \cdot \frac{n_2^2}{n_1^2} \cdot T_p(\theta_i) \cdot L_i(\theta_i, \phi_i) \cdot \cos\theta_{it} \cdot d\omega_{it} \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi_{rt})}{L_i(\theta_i, \phi_i)\cos\theta_i d\omega_i} \\
&= \frac{\cos\theta_{it} \cdot d\omega_{it}}{\cos\theta_i \cdot d\omega_i} \cdot T_p(\theta_r) \cdot T_p(\theta_i) \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi_{rt}) \\
&= \frac{\cos\theta_{it} \cdot n_1^2 \cos\theta_i}{\cos\theta_i \cdot n_2^2 \cos\theta_{it}} \cdot T_p(\theta_r) \cdot T_p(\theta_i) \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi_{rt}) \\
&= \frac{n_1^2}{n_2^2} \cdot T_p(\theta_r) \cdot T_p(\theta_i) \cdot f_0(\theta_{it}, \phi_{it}; \theta_{rt}, \phi_{rt}).
\end{aligned}
$$

The above derivation uses the relationship we derived earlier in Equation (A.7)

$$\frac{d\omega_{it}}{d\omega_i} = \frac{n_1^2 \cos\theta_i}{n_2^2 \cos\theta_{it}}.$$

Since $f_0(\cdot)$ satisfies the Helmholtz reciprocity, we have proved that $f(\cdot)$ also satisfies the reciprocity,*i.e.*,

$$f(\theta_i, \phi_i; \theta_r, \phi_r) = f(\theta_r, \phi_r; \theta_i, \phi_i). \tag{A.38}$$

Appendix B

Reflectance Models for Contaminated Transparent Objects

In this appendix, we present the details for the reflectance models and the equations for measurement for contaminated transparent objects which we derived earlier in Chapter 4. We also prove that the derived BRDF and BTDF models for contaminated transparent objects also satisfy the generalized Helmholtz reciprocity.

B.1 BRDF and BTDF of Contaminated Glass

The BRDF and BTDF models derived in Chapter 4 for contaminated transparent surfaces are briefly summarized as follows. As shown in Section 4.2 on page 56, there are four components for reflection (*i.e.*, $dL_r^{(k)}$, $k = 1, 2, 3, 4$) and two components for refraction (*i.e.*, $dL_t^{(k)}$, $k = 1, 2$). The formulas for the BRDF and BTDF models are elaborated below.

For reflection,

$$\begin{aligned}
f_{r,s}(\omega_i,\omega_r) &= R_p^{12}(\theta_i)\cdot\delta(\sin^2\theta_i-\sin^2\theta_r)\delta(\phi_i-\phi_r\pm\pi),\\
&\quad + e^{-\tau(\frac{1}{\cos\theta_{it}}+\frac{1}{\cos\theta_{rt}})}\cdot T_p^{12}(\theta_i)T_p^{12}(\theta_r)R_p^{23}(\theta_{it})\\
&\quad \cdot\delta(\sin^2\theta_i-\sin^2\theta_r)\delta(\phi_i-\phi_r\pm\pi),\\
f_{r,d}(\omega_i,\omega_r) &= \frac{n_1^2}{n_2^2}\cdot T_p^{12}(\theta_i)T_p^{21}(\theta_r)\cdot\frac{1-e^{-\tau(\frac{1}{\cos\theta_{it}}+\frac{1}{\cos\theta_{rt}})}}{\cos\theta_{it}+\cos\theta_{rt}}\cdot W_0\cdot p(\pi-\theta_{it}-\theta_{rt})\\
&\quad + \frac{n_1^2}{n_2^2}\cdot T_p^{12}(\theta_i)T_p^{21}(\theta_r)\cdot\frac{e^{-\frac{\tau}{\cos\theta_{it}}}-e^{-\frac{\tau}{\cos\theta_{rt}}}}{\cos\theta_{it}-\cos\theta_{rt}}\\
&\quad \cdot\left(R_p^{23}(\theta_{it})\cdot e^{-\frac{\tau}{\cos\theta_{it}}}+R_p^{23}(\theta_{rt})\cdot e^{-\frac{\tau}{\cos\theta_{rt}}}\right)\cdot W_0\cdot p(\theta_{it}-\theta_{rt}).
\end{aligned}$$

For refraction,

$$\begin{aligned}
f_{t,s}(\omega_i,\omega_t) &= \frac{n_3^2}{n_1^2}T_p^{12}(\theta_i)T_p^{23}(\theta_{it})\cdot e^{-\tau/\cos\theta_{it}}\cdot\delta(\sin^2\theta_i-\frac{n_3^2}{n_1^2}\sin^2\theta_t)\delta(\phi_i-\phi_t\pm\pi),\\
f_{t,d}(\omega_i,\omega_t) &= \frac{n_3^2}{n_2^2}T_p^{12}(\theta_i)T_p^{23}(\theta_t)\cdot\frac{e^{-\tau/\cos\theta_{it}}-e^{-\tau/\cos\theta_{tt}}}{\cos\theta_{it}-\cos\theta_{tt}}\cdot W_0\cdot p(\theta_{it}-\theta_{tt}).
\end{aligned}$$

From the above equations, it is easy to see that $f_{r,s}(\omega_i,\omega_r)$ and $f_{r,d}(\omega_i,\omega_r)$ satisfy the Helmholtz reciprocity, because if we swap θ_i and θ_r, θ_{it} and θ_{rt} are also swapped and the equations remain the same. For the two refraction components, we need to show they satisfy the generalized Helmholtz reciprocity, *i.e.*,

$$\frac{f_t(\omega_i,\omega_t)}{n_3^2}=\frac{f_t(\omega_t,\omega_i)}{n_1^2}.$$

This property holds because for $f_{t,s}(\omega_i,\omega_t)$, we have

$$\begin{aligned}
\frac{f_{t,s}(\omega_i,\omega_t)}{n_3^2} &= \frac{1}{n_1^2}T_p^{12}(\theta_i)T_p^{23}(\theta_{it})\cdot e^{-\tau/\cos\theta_{it}}\cdot\delta(\sin^2\theta_i-\frac{n_3^2}{n_1^2}\sin^2\theta_t)\delta(\phi_i-\phi_t\pm\pi)\\
&= \frac{1}{n_1^2}T_p^{12}(\theta_i)T_p^{23}(\theta_{it})\cdot e^{-\tau/\cos\theta_{it}}\cdot\frac{n_1^2}{n_3^2}\delta(\frac{n_1^2}{n_3^2}\sin^2\theta_i-\sin^2\theta_t)\delta(\phi_i-\phi_t\pm\pi)\\
&= \frac{1}{n_3^2}T_p^{12}(\theta_i)T_p^{23}(\theta_{it})\cdot e^{-\tau/\cos\theta_{it}}\cdot\delta(\sin^2\theta_t-\frac{n_1^2}{n_3^2}\sin^2\theta_i)\delta(\phi_t-\phi_i\pm\pi)\\
&= \frac{f_{t,s}(\omega_t,\omega_i)}{n_1^2},
\end{aligned}$$

and

$$\begin{aligned}\frac{f_{t,d}(\omega_i,\omega_t)}{n_3^2} &= \frac{1}{n_2^2}T_p^{12}(\theta_i)T_p^{23}(\theta_t)\cdot\frac{e^{-\tau/\cos\theta_{it}}-e^{-\tau/\cos\theta_{tt}}}{\cos\theta_{it}-\cos\theta_{tt}}\cdot W_0\cdot p(\theta_{it}-\theta_{tt})\\ &= \frac{1}{n_2^2}T_p^{12}(\theta_i)T_p^{23}(\theta_t)\cdot\frac{e^{-\tau/\cos\theta_{tt}}-e^{-\tau/\cos\theta_{it}}}{\cos\theta_{tt}-\cos\theta_{it}}\cdot W_0\cdot p(\theta_{tt}-\theta_{it})\\ &= \frac{f_{t,d}(\omega_t,\omega_i)}{n_1^2}.\end{aligned}$$

B.2 Derivation of Equation (4.10) in Section 4.4

In this section we show the derivation of Equation (4.10) for the measurement of scattering in Section 4.4 on page 69. As shown in Figure 4.10(b), for a point (x, y) on the Lambertian board, its outgoing radiance is proportional to the irradiance

$$L_c(x,y) = \rho\cdot E(x,y),$$

where ρ is the albedo of the board, and the irradiance $E(x, y)$ can be computed as

$$E(x,y) = L_t^{(2)}(\omega_t)\cos\theta d\omega_t,$$

where $L_t^{(2)}(\omega_t)$ is the outgoing radiance of the contaminant due to scattering of the incident beam, and the solid angle $d\omega_t$ is given by

$$d\omega_t = \frac{dA_p\cos\theta}{(D\cos\theta)^2},$$

where dA_p is the area of the contaminant layer illuminated by the beam and D is the distance from the slab to the board. Based on these equations and Equation (4.6) in Section 4.2 for $L_t^{(2)}(\omega_t)$, we can derive that (note that the incident direction ω_i is perpendicular to the slab and thus $\cos\theta_i = \cos\theta_i' = 1$):

$$I(x,y)\propto L_c(x,y)\propto p(\theta;g)\cdot\frac{e^{-\tau}-e^{-\tau/\cos\theta}}{1-\cos\theta}\cdot\cos^4\theta,$$

where we approximate the angle of the refracted ray inside the contaminant layer as θ as well. From this equation, we fit the phase function $p(\theta; g)$ with a nonlinear optimization algorithm to estimate g.

Appendix C

An Introduction to Compressive Sensing

In this appendix, we give a brief introduction on compressive sensing for recovering sparse signals, which has been used in Chapter 6. Compressive sensing [Candes and Romberg, 2007; Donoho, 2006] is a nascent field of applied mathematics with a variety of successful applications including imaging [Takhar et al., 2006; Willett et al., 2007], medical visualization [Lustig et al., 2007], and face recognition Wright et al. [2009]. It offers a theoretical framework to reconstruct "sparse" signals from far fewer samples than required by the conventional Shannon sampling theorem. Our work builds on the basic formulation of compressive sensing, which we augment with auxiliary terms specific to the reconstruction of volume density.

In its simplest form, compressive sensing seeks a solution of the *underdetermined* linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{C.1}$$

where $\mathbf{x} \in \mathbf{R}^n$ is a sparse signal, $\mathbf{A}$ is an $m \times n$ matrix called the "measurement ensemble", and $\mathbf{b}$ is the vector of m measurements, with $m < n$.

Compressive sensing theory asserts that one can recover the signal from far fewer measurements than the dimension of the signal, if the signal is *sparse*—it is represented with few non-zero coefficients in a suitable basis—and the measurements are uncorrelated, in the

sense that each measurement is an inner product of the signal with a *test function* that has a necessarily *dense* representation in the chosen basis. Equivalently, the measurement ensemble $\mathbf{A}$ needs to satisfy the *restrictive isometry condition (RIC)*. Strict definitions of sparsity and RIC can be found in [Candes et al., 2006]. Several ensembles are known to meet the RIC:

1. *Gaussian Random Variables* where each element of the matrix $\mathbf{A}$ is independently drawn from a normal distribution and each column is normalized to unit length.

2. *Bernoulli Random Variables* where each element is independently drawn from a Bernoulli distribution of +1 or -1 with the probability of 0.5.

3. *Partial Fourier Ensembles* where the ensemble consists of randomly selected Fourier basis functions.

Given a measurement ensemble matrix $\mathbf{A}$, compressive sampling theory predicts that $\mathbf{x}$ is the minimizer of

$$\min\|\mathbf{x}\|_1, \quad \text{subject to} \quad \mathbf{Ax} = \mathbf{b}. \tag{C.2}$$

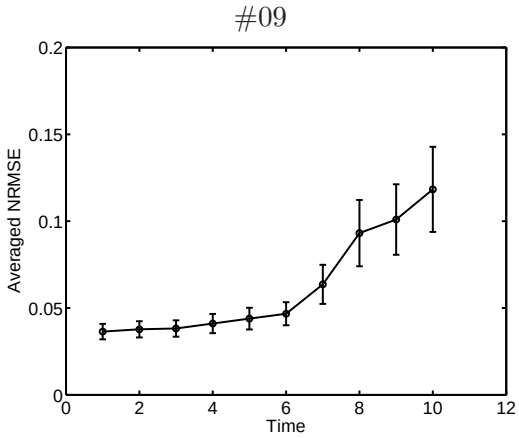
The above reconstruction strategy has been shown to work well for sparse signal estimation, even from a noisy measurement [Candes et al., 2006]. In Chapter 6 for recovering the volume densities of dynamic participating media, we augmented the basic problem above with auxiliary terms that enforce the nonnegative constraint for the signal, and that exploit the sparsity not only of the signal value but also its gradient. Indeed, our evaluation (Section 6.5, Section 6.6) shows that using the sparsity of the gradient is important for accurate and efficient reconstruction.

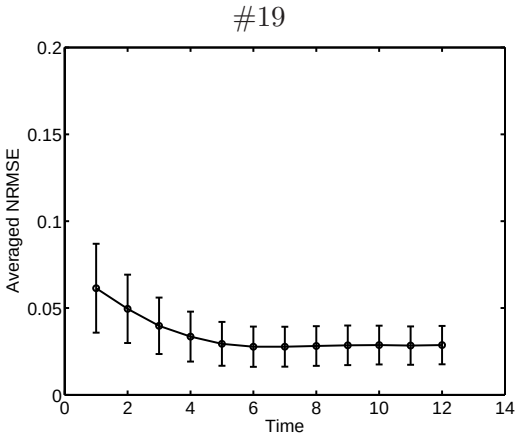
Appendix D

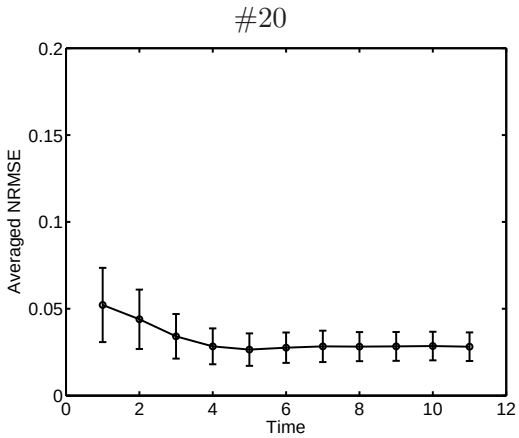
Fitting Measured TSV-BRDF Data to Parametric Model

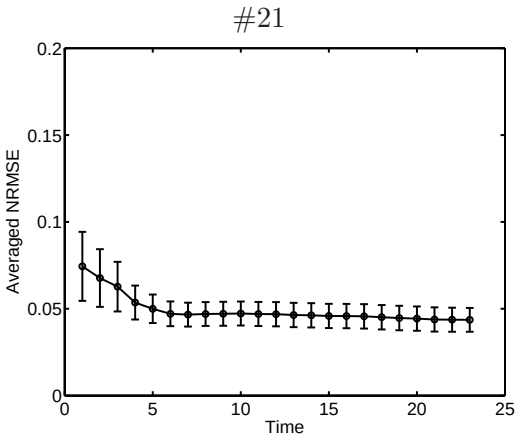
In this appendix, we list the average Normalized Root Measure Square Error (NRMSE) for fitting the acquired TSV-BRDF data to the parametric reflection model. More details can be found in Section 3.3 on page 30.

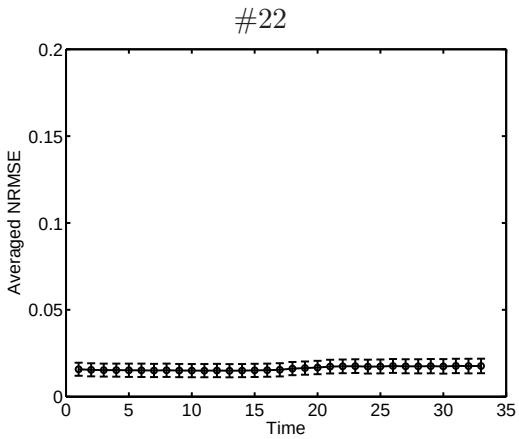
We show the results for the 28 samples in our database. In each plot, the x-axis is the index of time frames, and the y-axis is the averaged NRMSE computed across all the spatial locations on each sample. The standard deviation is also included. These plots shows that most samples in our database have the average NRMSE less than 7% across all time frames, which shows the accuracy of the BRDF fitting.

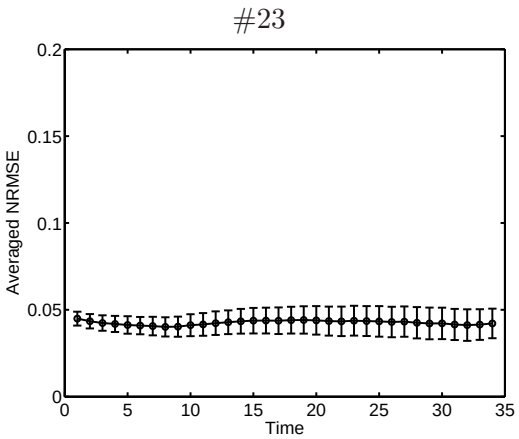

#09
Averaged NRMSE
Time


#19
Averaged NRMSE
Time

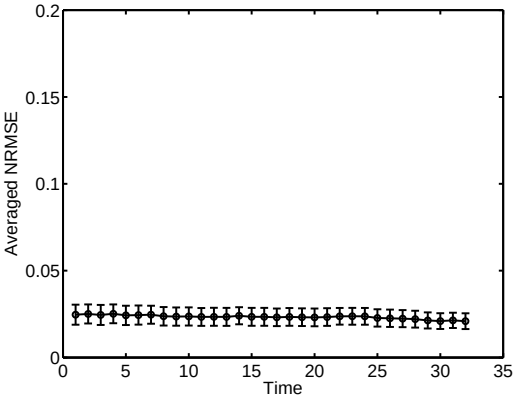

#20
Averaged NRMSE
Time


#21
Averaged NRMSE
Time

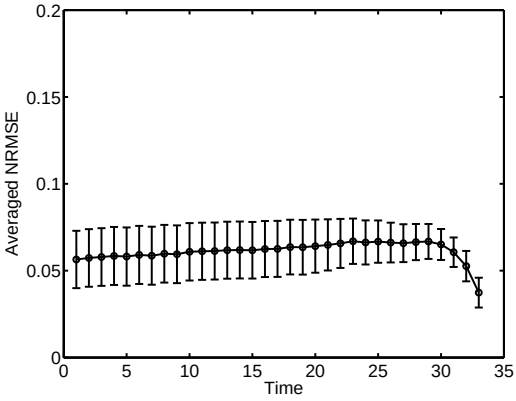

#22
Averaged NRMSE
Time


#23
Averaged NRMSE
Time

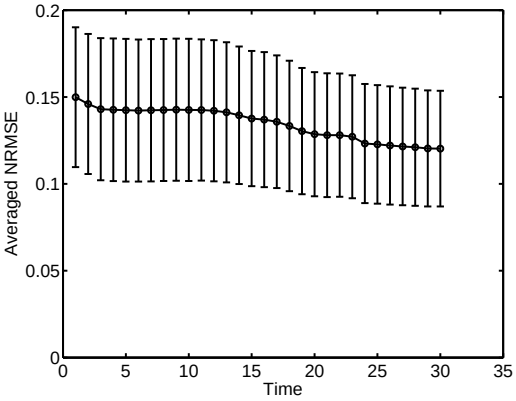
#24



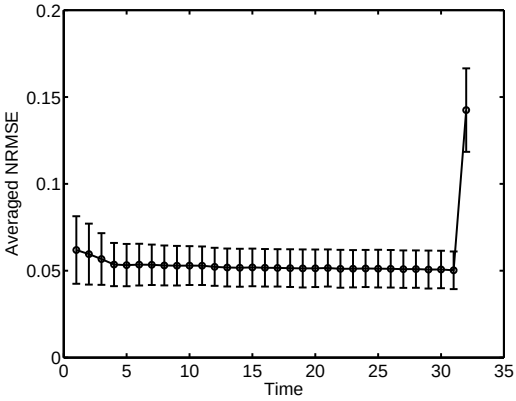
#25



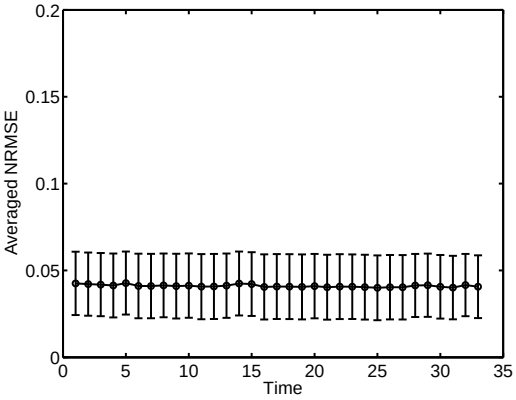
#26



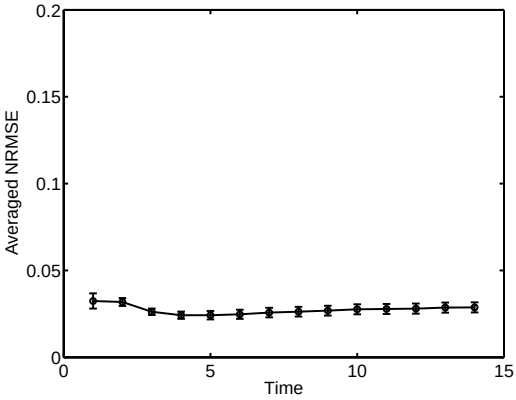
#27



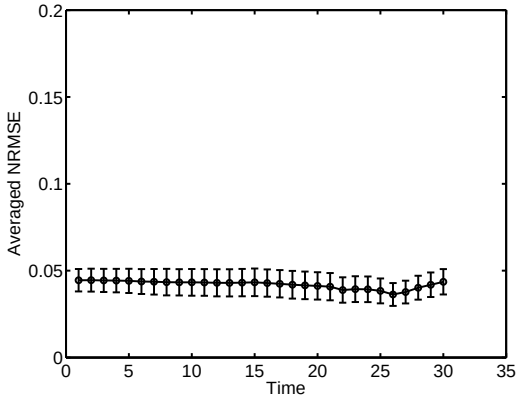
#28



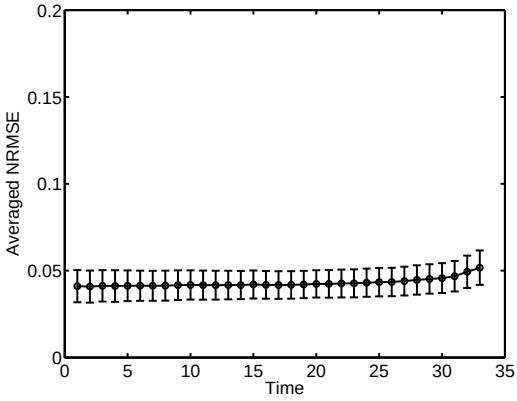
#29



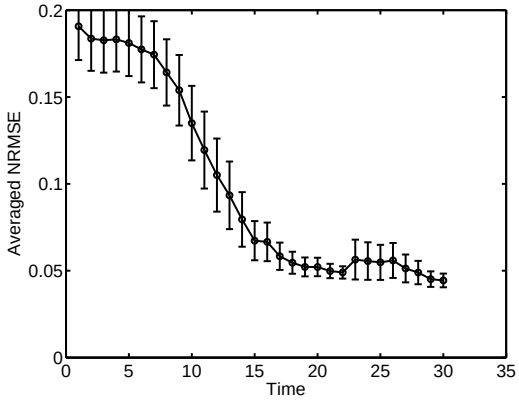
#30



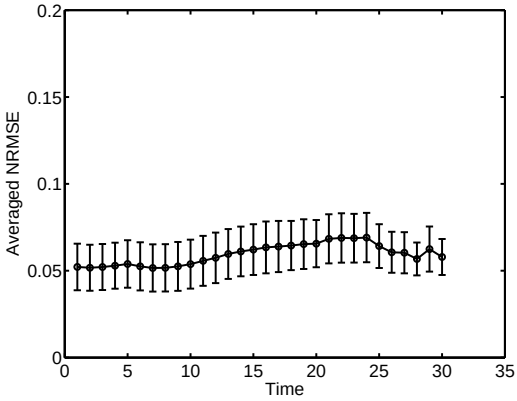
#31



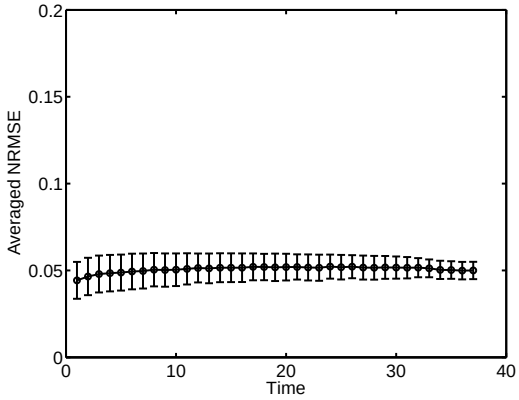
#32



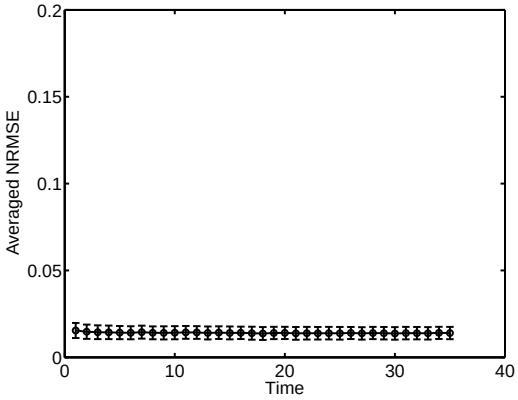
#33



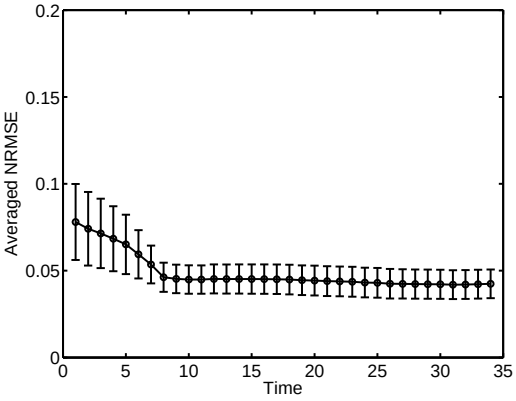
#34



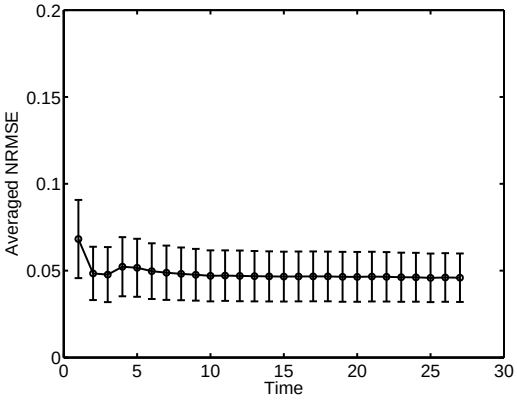
#35



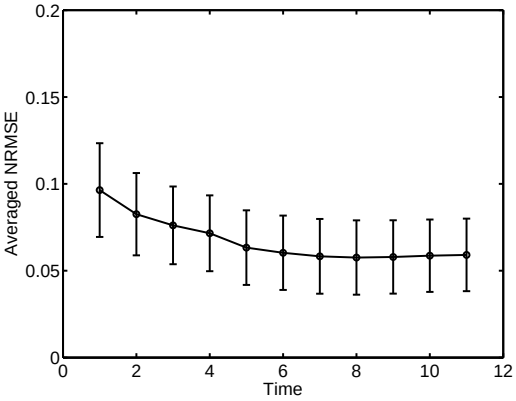
#36



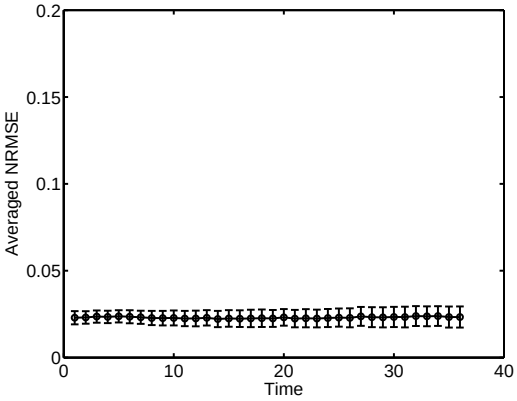
#37



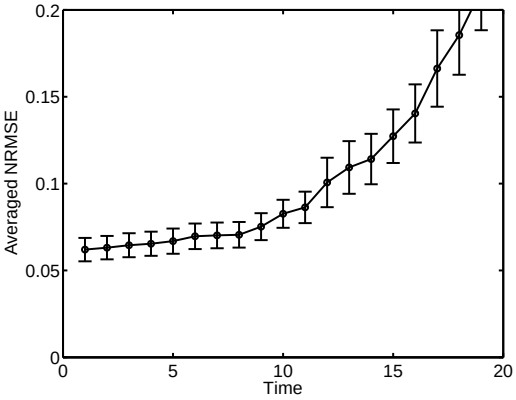
#38



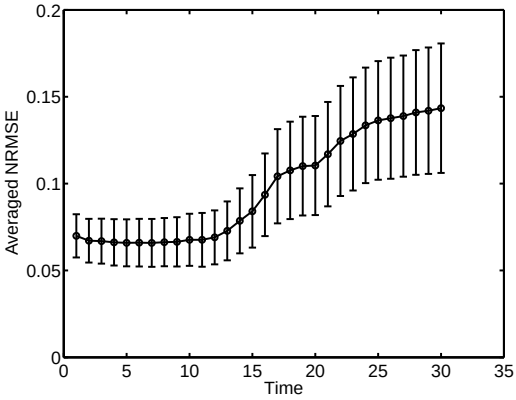
#39

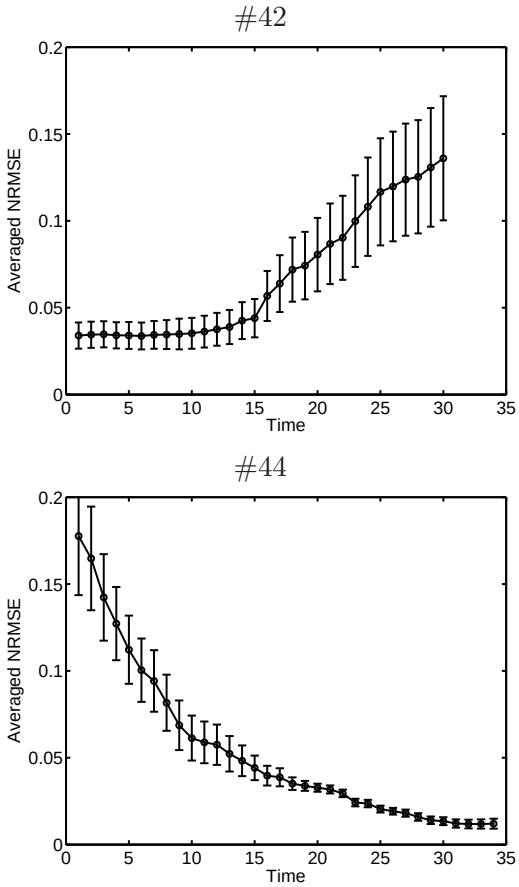


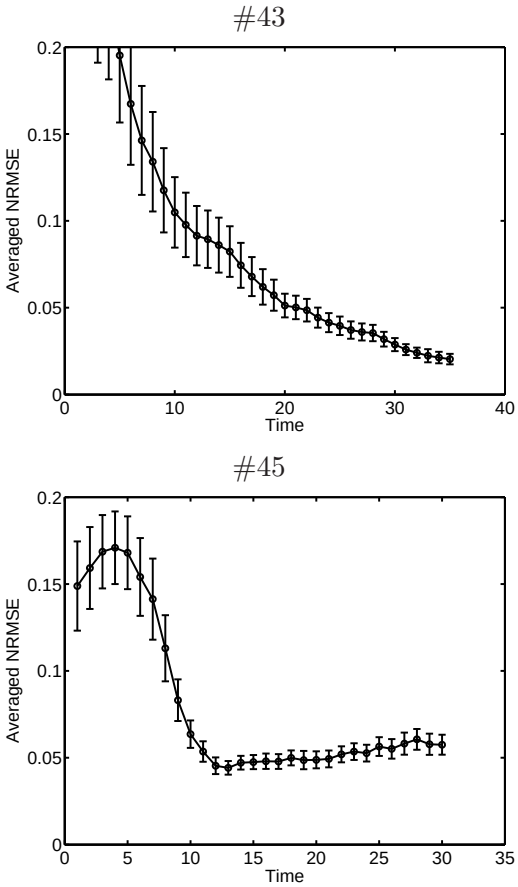
#40



#41




#42
Averaged NRMSE
Time
0
0.05
0.1
0.15
0.2
0 5 10 15 20 25 30 35


#43
Averaged NRMSE
Time
0
0.05
0.1
0.15
0.2
0 10 20 30 40

#44
Averaged NRMSE
Time
0
0.05
0.1
0.15
0.2
0 5 10 15 20 25 30 35

#45
Averaged NRMSE
Time
0
0.05
0.1
0.15
0.2
0 5 10 15 20 25 30 35