# Modeling Illumination for Scene Recovery through the Motion, Occlusion and Strobing of Light-Sources

Sanjeev Jagannatha Koppal

**CMU-RI-TR-09-37**

*Submitted in partial fulfillment of the*
*requirements for the degree of*
*Doctor of Philosophy in Robotics*

The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

August, 2009

# Abstract

Recent display applications for entertainment and business have made available new types of illumination using LEDs, DMDs and LCDs, which are bright, energy efficient and cheap. Some of these devices are programmable and allow spatio-temporal control of the emitted light rays. With the advent of such digital light-sources, illumination is becoming a flexible, configurable medium. This has impacted computer vision and spurred techniques that control illumination for analysis of indoor areas, industrial environments, stage/studio sets, underwater scenes, underground locations and outdoor scenes at night. In such methods, the light-source's programmability is often exploited to create easily detectable features, such as bright stripes or binary patterns.

In this thesis, we extract illumination-based features for three new scenarios by exploiting the motion, occlusion and strobing of light-sources. First, we move a light-source in a smooth and random path. For a static scene, this creates a continuous set of intensities at each pixel. We exploit this continuity to detect brightness maxima and minima at scene points and show how these extrema features relate to geometric cues, such as surface normals and depths. For our second approach, we occlude a light-source using moving opaque masks. Each pixel's brightness minima, due to the mask shadow, corresponds to a set of blocked incident light rays. These shadow features can be used to render the scene from the light-source's point-of-view. The third technique exploits the flickering emitted by a strobing source as a temporal feature that is easily detected, even for scenes with fast moving objects. This enables active vision for dynamic scenes, which we demonstrate using DLP illumination.

Our methods work with a variety of indoor and outdoor materials, real-world textures and glossy/metallic objects. We are not restricted to distant point sources and show results with outdoor illumination, sources with intensity fall-off and area/line sources such as indoor fixtures. None of our approaches require a complex or calibrated setup. Some even allow the light-source to be hand-waved in an unstructured manner. In addition, all of our techniques are easy to implement, requiring only a few lines of code. The work in this thesis demonstrates, for the first time, results such as iso-normal clustering of indoor and outdoor scenes, reciprocal views from general, non-programmable sources and very high-speed marker-less motion-capture and photography. Finally, almost all natural and artificial light-sources in our world either undergo motion (or are mobile), get occluded or exhibit strobing (at some frequency). Therefore our algorithms have relevance for real-world illumination and any applications or extensions of these will have significant impact.

# Contents

# Acknowledgements

This work became possible due to my advisor, Srinivasa Narasimhan, whose guidance has been invaluable. I would also like to thank my thesis committee, Martial Hebert, Alexei Efros and Steven Seitz, for their insights. I am also grateful to Shuntaro Yamazaki and Mohit Gupta, whose collaboration, technical discussions and proof-reading proved crucial at many times. Finally, this work was supported by NSF Awards #CCF-0541230, #CCF-0541307 and #IIS-0643628 and by ONR contracts #N00014-06-1-0762, #N00014-08-1-0330 and DURIP N00014-06-1-0762.

*to my father, for being a fighter: disease or desert conflict.*
*to my mother, for forging my strength and soul.*
*to my sister, for showing me courage.*
*and to my best friend, for her smiles.*

*The principal person in a picture is light.*

Edouard Manet

*Work. Finish. Publish.*

Michael Faraday

# Chapter 1

# Introduction: The Ray Geometry of Illumination

Long before cameras existed, humans thought deeply about illumination. Planning the optimal use of natural light was a key task in early architecture, and it remains important today. Artificial light was derived solely from combustion, and the desire to control such man-made illumination led to advancements in optics, of which the Lighthouse at Alexandria was an ancient and stunning example ([69]). Less marvelous but equally intricate are designs expressed in the craft of making light-fixtures. These abound in the ruins of classical cities and led to the field of modern lighting design. In addition, other areas of endeavor such as astronomy, art, navigation, war and communication developed theories and models of light ([70]).

However, the most important breakthrough came hundreds of years later, when work by a series of scientists in the 18th and 19th centuries created a theory of electricity. The search for applications soon led to the first incandescent lamp, and a lighting industry started to emerge. Many of the civilization-changing inventions from this time, from cinema projectors to skyscrapers, depended on electric lights. War advanced the field: one of the first battlefield stereo devices in WW-II was a pair of searchlights looking skywards, allowing anti-aircraft gunners to calculate the height of illuminated planes ([36]).

In the decades after the war, light-sources based on solid-state technology were invented and illumination began to become digital in nature. The use of lasers became wide-spread as a means of creating digital storage devices with much higher capacity that magnetic disks would allow. MEMS technologies and other innovations were and continue to be spurred by display and entertainment applications as well as recent energy efficiency pressures. The new forms of digital light-sources are important since they allow computer control of different illumination characteristics. This trend of continued overlap between illumination and computation has implications for a multitude of fields and, in particular, we concern ourselves here with the impact on computer vision and computer graphics.

## 1.1 Lighting in Computer Vision

In parallel to light-source development, cameras transformed from analog, unwieldy machines into devices creating digital images processable by computer. Thus started the field of computer vision, which seeks all the information contained in any set of images or video, with goals such as scene reconstruction, object recognition, motion tracking, novel view synthesis and image relighting. This thesis is concerned with the broad class of computer vision techniques which use illumination changes to extract scene information. The input to these algorithms are typically images of a scene taken under varying lighting conditions, and have been recently termed as computational illumination techniques.

Even before the advent of digital illumination, classical methods in computer vision, such as photometric stereo, demonstrated the value of images taken under different lighting directions. Recently, with the popularity of pro-

grammable digital projectors, a broad range of techniques have been proposed. Illumination-based methods can be classified into two groups: techniques that have an alternate camera-based solution, and techniques which recover scene properties which simply cannot be obtained in the camera domain. The former includes grid coding for scene reconstruction, which is especially useful for textureless surfaces where finding correspondences is hard. The latter spans both older methods, such as photometric surface normal estimation, and more recent spatial-frequency methods, such as separation of direct and global illumination.

Table 1.1 shows examples from both classes of approaches, along with the light-sources that are modeled. The light-source types derive from a seminal work ([49]) that classified sources by parameterizing the emitted rays as they pass through two planes. Since the location of the ray as it strikes each plane can be represented by four numbers, all types of illumination lie on a 4-D light-source hypercube. What is immediately striking is that a significant portion of the work in vision and graphics has been done for very limited types of light-sources. These are singularities or vertices of the 4-D hypercube, such as both near and distant point sources. Some approaches (such as recovering light-transport) are only possible with the control provided by a programmable light-source such as a laser or projector. In addition, a few methods require the light-source to be completely calibrated. We would like to move beyond these restrictions and model complex, general illumination.

## 1.2 Ray Geometry of Light-sources vs. Cameras

A camera can be completely defined by the geometry of its viewing rays ([28]). Given correspondence between rays from multiple cameras, a vast array of multi-view techniques can be applied to extract information from images. Similar ray geometric models exist for light-sources, where the analogous problem to camera correspondences is finding the incident light ray at each scene point. However, this *ray correspondence* is only easily obtainable for programmable light-sources, such as projectors. Therefore, light-source ray geometry is usually ignored by most illumination-based approaches. Instead, these methods favor modeling photometry for scene recovery. This is a hard problem, which partially explains why fewer algorithms exist for complex illumination in Table 1.1. In contrast, we argue that the issue of ray correspondence should not be bound with the problem of photometric modeling.

At first, the separation of photometry and ray correspondence may seem a difficult problem to solve. However, structured light techniques easily achieve this by projecting an illumination pattern onto the scene. The key assumption here is that the bright intensity or high spatial-frequency of the pattern will dominate the scene reflectance, allowing easy detection. This assumption may not hold for degenerate scenes (such as made of mirrors or glass), but is broadly applicable. Making a similar argument, we exploit the motion, occlusion and strobing of light-sources to create easily detectable features, despite complex

scene reflectance or dynamic objects. The challenge here is to decide which features to compute and to explain why they relate to incident ray geometry. This distinguishes our techniques from the trivial thresholding required for detecting line stripes and binary patterns in structured light approaches.

In our setup, each pixel in an image (or set of images) is assigned a set of light rays incident at the corresponding scene point. Typically, both light-source shape and the scene properties determine the incident ray at each scene point. However, we demonstrate that illumination variation due to light-source motion, occlusion and strobing encode incident ray information directly in the measured intensities. Once we obtain this incident ray information, we can design algorithms with fewer reflection model assumptions or restrictions to a particular light-source type. In addition, modeling illumination through ray geometry allows us to leverage the large set of approaches in the camera-domain to suggest algorithms in the light-source domain. This is possible since, in terms of ray geometry, light-sources and cameras share many similarities.

In fact, light-sources are duals to cameras: they emit light rays instead of collecting them. Previous work has acknowledged this idea for programmable sources, such as projectors, whose equations of projection are inverse to those of perspective cameras. In addition Figure 1.1 outlines a set of such properties. For example, consider a pin-hole camera, which is a collection of rays from the scene that pass through a single point. A near point light-source can be analogously described as a collection of rays emanating from a single location. Similar parallels exist between orthographic cameras and distant light sources. Area and linear sources have analogies in large aperture cameras. The concept of focal length for a point-light source is easily demonstrated by creating sharp and blurred shadows. Light-source fluctuation is measured over time, exactly as camera frame-rate. In addition, light-sources can have varied spectral emittance while a camera can record multi-spectral information. Finally, even the intensity fall-off from a light-source can be compared to a stereo-rig whose disparity is inversely related to the depth from the baseline.

While the duality of light-sources and cameras is a classic idea in vision and graphics, we use it to frame the search for new computer vision algorithms. For every algorithm that works purely in the camera-domain, there are two analogous illumination techniques. The first, which we term the primary approach, uses the light-source to achieve exactly the same result as the camera-domain method. The second, which we term the dual approach, is able to utilize what is different about the light-source to create a new result that would not be possible in the camera-domain. The second class of approaches is interesting because they show how light-sources can achieve a task better than the camera itself. Table 1.4 lists some selected camera-centric techniques and shows the primary and dual applications of each, with our approaches given in bold. We discuss these in detail in the next section.

| Vision and Graphics Applications | Traditional Sources | | | | Non-traditional Sources | | |
|---|---|---|---|---|---|---|---|
| | Distant Point $(\infty, \infty, 0, 0)$ | Near Point $(0, 0, \infty, \infty)$ | Projector/Laser $(u, v, proj(u), proj(v))$ | Environment Map $(u, v, \infty, \infty)$ | Line/Curve $(u, f(v), \infty, \infty)$ | Area $(f_1(u), f_2(v), \infty, \infty)$ | General Source $(f_1(u), f_2(v), f_3(s), f_4(t))$ |
| Surface Normal Estimation | ✓ | ✓ | ✓ | ✓ | | | |
| Scene Depth Recovery | | ✓ | ✓ | | | | |
| Camera Calibration | | ✓ | ✓ | ✓ | | | |
| Light Transport Estimation | | | ✓ | | | | |
| Image-based Relighting | ✓ | ✓ | ✓ | ✓ calibrated | ✓ | ✓ calibrated | |
| Artistic Visual Effects | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |



Table 1.1: **Selected Illumination Variation Methods in Computer Vision:** Note that most of the algorithms either use calibrated point sources or utilize a programmable light-source such as a projector. Extending current techniques to general lighting requires modeling complex light-sources. We achieve this by obtaining incident ray correspondence at scene points through the motion, occlusion and strobing of light-sources.

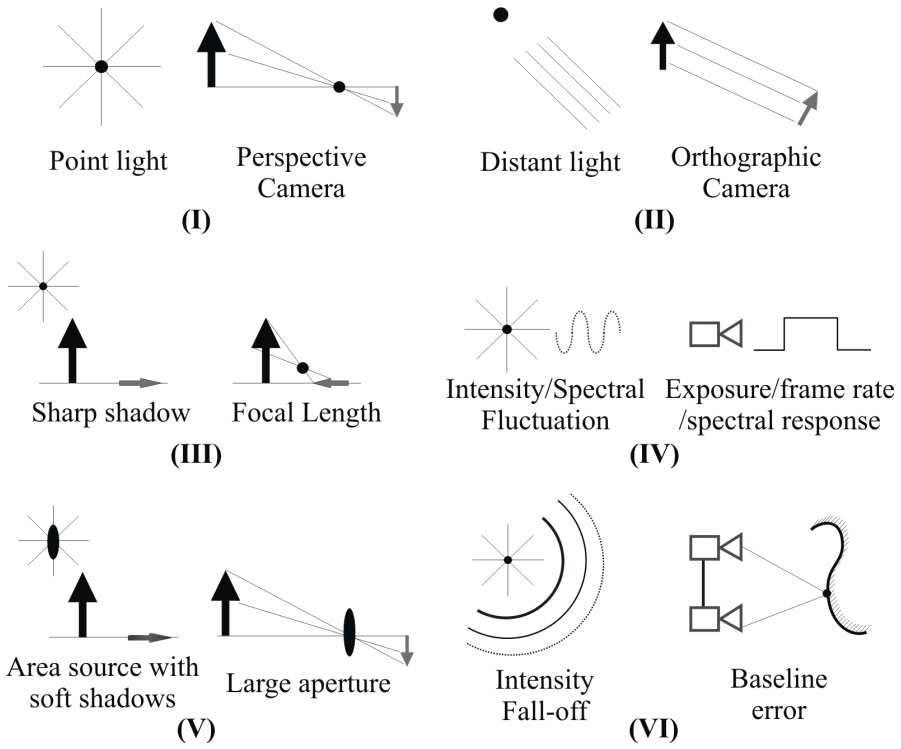



Figure 1.1: **Camera-Light source similarities:** Light sources emanate rays while cameras collect them. In (I) and (II) we illustrate this duality for point light sources, which are analogs to perspective cameras, and distant light sources, which are analogs to orthographic cameras. This duality exists for all geometric shapes of light sources, as well as for other properties such as (III) focal length, (IV) exposure time, (V) aperture size and (VI) baseline error.

## 1.3 Modeling Light-source Motion, Occlusion and Strobing

In the first approach we consider what cues can be obtained from the uncalibrated smooth **motion** of a light source. The observed intensities at scene point, which we term an appearance profile, depend heavily on how the incident ray geometry changes. For smooth changes in the ray direction, we will demonstrate that the locations of maxima and minima in the profile contain geometric information. Informally speaking, scene points that 'light up' and 'go dark' at the same time should experience similar changes in incident ray directions. The advantage of using extrema over a feature such as, say, shadowgrams or reciprocity pairs, is that they can be easily detected in images of scenes without any complex set up or calibration. In fact, all our examples are illuminated by a light-source waved by hand. Furthermore, since we require relative intensity measurements to find brightness extrema locations, photometric camera calibration is not needed. We show that surface normal information can be obtained from varying distant lighting, and depth cues from near lighting.

The second method we present will look at the **occlusion** of light-sources, which span a much richer set of shapes or caustics, when compared to cameras. We capture this geometry through the shadow-field of a non-programmable light source. The shadow-field is obtained by using an opaque mask to block rays from the light source. Therefore, our representation is purely geometric and not photometric, since incident light rays are detected by not measuring them. Ray reciprocity allows us to describe these rays as if they were viewed by a virtual camera. The recovered light ray geometry depends on the mask shape, the light source type and the relative motion between the two and we have shown perspective, orthographic, catadioptric, pushbroom, and cross-slit geometries. In addition, we show that recursively applying the algorithm allows the estimation of off-diagonal elements from the light-transport matrix.

In the third approach we exploit the temporal dithering or **strobing** of certain types of illumination. Incident rays across the scene (and within a scene point) can be disambiguated quickly by observing the rate and extent of their dithering. This disambiguation can enable almost any active vision technique for dynamic scenes. We present a broad framework for fast active vision using Digital Light Processing (DLP) projectors. Our key idea is to exploit the flickering of illumination due to the DMD chip in a DLP projector, as observed by a high-speed camera. The dithering encodes each brightness value uniquely and may be used in conjunction with virtually any active vision technique achieving significant speedups in performance.

## 1.4 Related Work

Lighting is a powerful cue for scene analysis using photometry methods. When lighting is known it is possible to extract scene structure. Most reconstruction techniques first choose a reflectance model and then estimate its parameters from images. This is the basis for Woodham's classical photometric stereo ([111]) for lambertian scenes, as well as several extensions for non-lambertian low parameter BRDFs, such as the micro-facet model and the dichromatic model ([46],[73],[26],[101],[9],[95],[55],[83]) or near point lighting ([37],[72],[54],[8],[7], [99]). On the other hand, if the lighting is unknown, its estimation usually requires additional, fundamental information about the scene. The class of 'inverse rendering' algorithms that estimate low parameter BRDFs and lighting ([87],[56]) using scanned 3D scene geometry. Ramamoorthi's thesis ([76]) provides a formal analysis of when exactly inverse rendering is possible for general BRDFs and lighting that are represented using Spherical Harmonics. Hertzmann and Seitz, ([34]), recover the geometry of objects by estimating combinations of "basis example spheres" that best describe scene BRDFs.

Multi-view stereo algorithms are widely used in computer vision ([30],[92]). A difficult problem commonly faced in these methods is matching correspondences between different views. This is especially challenging for regions without texture (where matching is ambiguous) or with complex material properties (where matching requires knowledge of the BRDF). Structured light techniques provide a single view alternative to multi-view stereo. Scene point locations

are obtained by intersecting viewing rays from a camera with rays from a light source. Although these approaches avoid the correspondence problem, they require a programmable source to obtain incident light rays at each scene point. Much work has been done on obtaining scene structure by either using projectors or lasers ([110],[39],[6],[35],[119],[86]). In addition, techniques have been proposed that exploit a projector's control of the incident light field to recover the light-transport matrix ([93], [94]) or portions of it such as the global component ([65]) even in the case of large scene depths ([29]).

Illumination masking follows a 'less is more' trend where coded camera apertures capture additional light-field information ([77]). Shadows can be used for spatio-temporal correspondences across cameras ([14]). Light-field rendering ([53],[27]) obtains new views by interpolating between densely sampled images, with applications such as view synthesis ([118]), all-focus images ([66]), seeing past obstacles ([104]) and mosaicing both multi-viewpoint images ([91]) and new cameras (cross-slit ([123]), pushbroom ([117])). Shadow-based approaches have been popular since they are invariant to material properties ([79], [88], [50]). Linear masks have been previously used for scene point triangulation ([3], [4],[51]), while shadowgram methods reconstruct intricate objects ([114]) and multi-flash approaches obtain occluding edges ([105]).

Projectors are commonly used as programmable light sources for a variety of active vision techniques including structured light range finding [110, 119, 15, 11, 116, 89], photometry-based reconstruction [121, 34], relighting [109], light transport analysis [65, 104] and depth from defocus [120]. The intensity and color of the scene's illumination can be controlled over space and time depending on the task at hand. For instance, projecting a set of colored striped patterns onto a scene alleviates the problem of finding correspondences for 3D reconstruction [119]. Dual photography ([94], [23]) exploits Helmholtz reciprocity ([32], [121]) between every projector-camera pixel pair to render the scene from the projector's point-of-view. Since the full light transport is measured, any scene relighting technique can be applied.

Digital Light Processing (DLP) technology (http://www.dlp.com) based projectors contain a key component called the Digital Micromirror Device (DMD). These mirrors can switch between orientations in a few microseconds, enabling high precision control of illumination. As a result, the DMD device has found applications in areas ranging widely from microscopy to chemistry to holographic displays [17]. Nayar et al. [64] re-engineer a DLP projector into a DMD-camera and demonstrate the notion of programmable imaging for applications including adaptive dynamic range and optical filtering and matching. Based on the theory of compressive sampling, a single pixel camera has been implemented where the DMD device used to compute optical projections of scene radiance [106]. Jones et al. [42] modify a DLP projector using custom made FPGA-based circuitry to obtain 1-bit projection at 4800Hz for high speed stereoscopic light field displays. McDowall and Bolas [58] use a specially re-programmed high speed projector based on Multiuse Light Engine (MULE) technology to achieve range finding at kilohertz rates.

| Camera-domain Approach | Application | Light-source centric Approach | Primary Lighting Application | Dual Lighting Application |
|---|---|---|---|---|
| Arbitrary Views with correspondences | Scene Reconstruction View Synthesis | Many Illumination Directions (calibrated) | Scene Reconstruction and Relighting | View Synthesis (Dual Photography, **Shadow Cameras**) |
| Line Motion | Depth Estimation (EPIs) | Linear light-source Motion | **Depth Cues** | |
| Arbitrary Views without correspondences | Visual Hull | Many Illumination Directions (uncalibrated) | **Surface Normal Clusters** | |
| Coded/Synthetic aperture | Scene Reconstruction and Deblurring | Occluded Illumination | Scene Reconstruction Illumination deblurring | **Shadow Cameras** |
| Exposure coding (flutter shutter) | Deblurring images | Strobing Illumination | **Fast Active Lighting** | **DLP Photography** |

Table 1.2: **Selected Camera-domain Algorithms and their Corresponding Light-source Domain methods:** In the first column we list some important camera domain methods. The duality of light-sources and cameras suggests corresponding light-source centric algorithms. These either achieve a similar goal as the camera-domain methods (we call this a primary application) or they give a completely new result for the scene (which we call a dual application). The methods introduced in this thesis are shown in bold.

## 1.5 Thesis Organization

Besides the introduction, this document has three chapters detailing technical contributions, followed by a chapter on future work. Each chapter begins with an *abstract paragraph* that motivates the broad problem and provides an outline of the content. In addition, each chapter contains a *summary section* providing technical summaries of the methods described, required input and output, comparisons to competing methods, implementation issues such as run-time, overall advantages and disadvantages and potential for real-world application. **The abstract paragraph and the summary section in each chapter should be of interest to those skimming the document.**

Chapter 2 investigates the intensity at every pixel for a camera viewing a scene under smoothly moving illumination. We introduce the concept of an appearance profile and show empirically that its maxima and minima locations contain surface normal information. An application is demonstrated for texture transfer of non-linear effects such as specularities on velvet. We also look at the appearance profiles for near-point lighting and show their maxima contain depth information. In the last section, we discuss how the appearance profile can be modeled using higher-order Taylor polynomials.

Chapter 3 introduces the idea of shadow cameras, which are created by the relative motion of a light-source and occluding mask. The image of the scene from the point-of-view of the shadow camera is provided by helmholtz reciprocity. We develop a series of geometries for shadow cameras, from perspective and orthographic to multi-view and cross-slit. We demonstrate applications for removing distortions from dioptric and catadioptric cameras, as well as a rendering application, where we demonstrate the capture of the geometry of light-sources. We end the chapter by developing the shadow-camera idea further and investigating the second order minima detected at each scene point. This suggests the recovery of off-diagonal information in the light-transport matrix, and we show a simple example to separate specular interreflection.

Chapter 4 introduces DLP projectors and details temporal dithering due to the DMD chip when the projector illumination is observed with a high-speed camera. We show a series of applications for high-speed active vision. These include structured light reconstruction, illumination multiplexing and demultiplexing for photometric stereo and colorizing and direct-global separation. We also briefly introduce a motion deblurring framework which we term 'flutter-flash'. The next part of the chapter extends the flutter-flash idea to created strobe photographs of fast-moving scenes. We show how strobe photographs under DLP illumination have more high-frequency information that those taken under fluorescent illumination. We show applications such as summarizations and motion illustrations of fast scenes.

In Chapter 5, we discuss future applications of this work and attempt to predict how current trends in illumination will affect a variety of fields. We choose to focus on possible applications for hand-held devices, tracking, light-fixtures and architecture. We conclude by discussing the idea of camera analogies, of which the set of light-sources is an example.

# Chapter 2

# Motion: Geometric Cues from Smoothly Varying Illumination

Moving light-sources exist all around us: the day sky contains both highly constrained motion (such as the sun) and random patterns (such as from clouds). At night, humans carry illumination in their modes of transport (such as headlights in cars). Previous work has modeled the location and extent of these light-source, but in this section we investigate the motion of the light-source itself. For the first time, we demonstrate modeling the derivatives of observed static scene appearance as the illumination changes. We restrict ourselves to point light sources, either distant (like the sun) or near (such as an LED).

Our key idea is to associate with each scene point (observed as a pixel) an incident light ray. In general, estimating the ray from observed intensities is hard, since the light is affected by scene BRDF before being viewed by the camera. However, as the light-source moves, each pixel generates a sequence of intensities, which we term an appearance profile. Inflections points (peaks and troughs) in the intensity profile over time can be corresponded to inflections or changes in the direction of the incident light ray. These intensity peaks and troughs can be detected directly from images without modeling scene reflectance or material properties. Depending on the type of light-source and its motion, these intensity extrema provide scene information. We first show that distant light source motion allows us to group scene points according to their surface normal. In Section 2.2, we demonstrate that a near point source moving in a line or plane can group scene points according to depth information. Finally, in Section 2.3, we turn to modeling the appearance profile itself as a Taylor series polynomial, allowing image rendering and relighting.

## 2.1 Smoothly Moving Distant Lighting

A static scene under varying distant lighting gives rise to a smooth intensity or appearance profile at each scene point. We first show empirical evidence suggesting that the locations of the *extrema* of the appearance profile provide a strong cue for the scene point's surface normal. Based on this observation we derive a simple transformation of the appearance profiles and introduce a distance metric that can be used with any unsupervised clustering algorithm to obtain iso-normal clusters of a scene. We also present a formal analysis for the extrema occurring in surfaces that can be modeled using linearly separable model, which represents a large class of BRDFs. We show results of applying our algorithm to difficult indoor and outdoor scenes containing a variety of complex geometry and materials.

### 2.1.1 Appearance Profiles and their Extrema

Extrema are said to be *shared* between two appearance profiles if they occur at the same time instance in both profiles. Intuitively, shared brightness extrema are important since scene points that share the same normal should 'light up' and 'go dark' at the same time. An important factor that determines where extrema occur in a profile is the path of the light source. Consider a distant

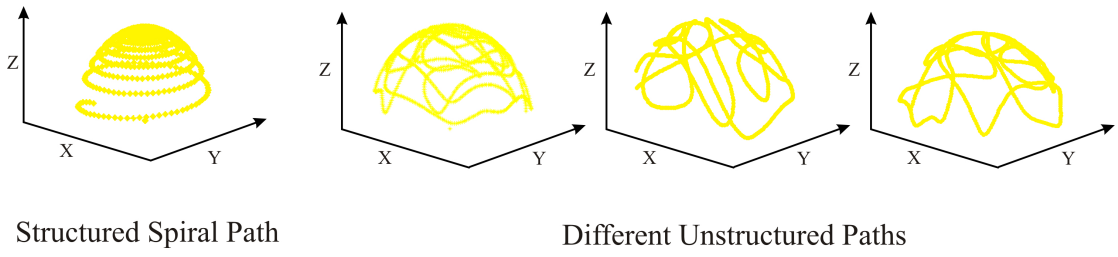



Figure 2.1: **Structured versus unstructured paths for the light-source.** Engineered setups such as a gantry may produce paths such as the spiral shape shown on the left. In contrast, in our method the user hand-waves the light source. This results in unstructured paths, such as the three shown on the right.

point light source that is being waved by a user. The trajectory of the light source is *unstructured* and it contrasts with the light source paths created by gantry setups used in many previous papers (such as [103]). While a structured path may have a regular shape, such as a spiral (see Figure 2.1), an unstructured path created by 'hand-waving' the light source may not have any standard, recognizable shape. Since the light source changes its position smoothly, but randomly, at every time-step the intensities at every scene point are generated stochastically. The empirical evidence in this section will show that iso-normal profiles produced by such unstructured paths share many extrema.

### 2.1.2 Shared Extrema in Iso-normal Profiles

We investigate the extrema present in profiles generated by BRDF simulations, rendered scenes, measured BRDFs and real textures. These experiments provide strong empirical evidence that shared extrema locations can be exploited to obtain iso-normal clusters.

**BRDF Simulations:** In Figure 2.2 profiles are generated for 50 unique surface normals that were sampled uniformly from the hemisphere of directions. The simulations consist of four BRDF models: Lambertian, Oren-Nayar, Torrance-Sparrow and Oren-Nayar + Torrance-Sparrow. The user creates a smooth, unstructured path for the light source on the hemisphere of directions. The material properties were varied (roughness $\sigma$ from 0 to 1 and albedo $\rho$ from 0 to 1) for each of these models producing over 20,000 profiles for each normal. The top row of Figure 2.2 shows some of these profiles for a particular surface normal which demonstrate significant variation in the profile shape due to changing material properties. Despite these differences extrema locations were found that were common to over 95% of a normal's profiles and therefore insensitive to the changes in albedo and roughness. Furthermore, these shared extrema locations were unique to a particular normal. In the bottom row of Figure 2.2, profiles from two different normals are shown, with their shared extrema locations marked on the x-axis.

**Rendered Scene:** A scene was rendered using a commonly used ray-tracing tool ([75]) generating profiles under conditions similar to real scenes, with ef-

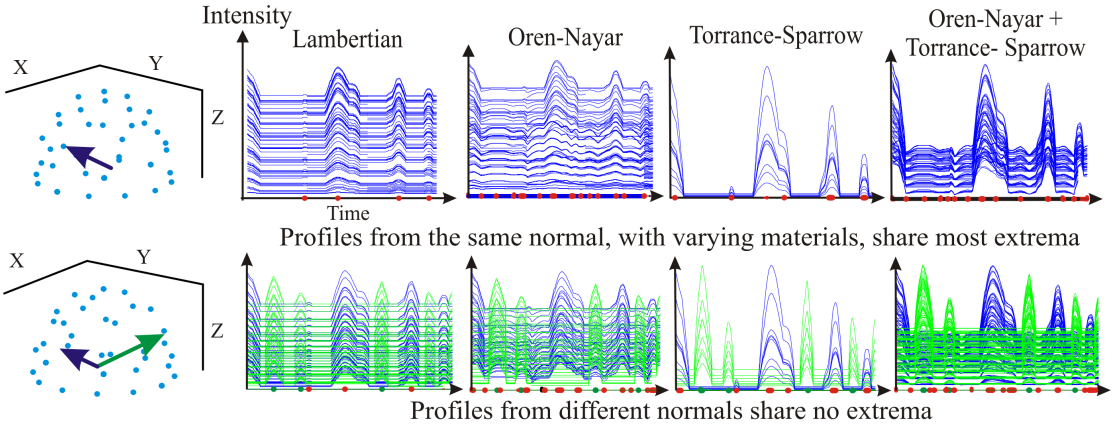



Figure 2.2: **Simulations showing the link between extrema and surface normal:** Appearance profiles are simulated for four BRDFs over a range of 20,000 material properties (only a few are shown for clarity). Profiles are only shown for two normals, although we simulated profiles for 50 (marked by blue dots on the hemisphere, on the left). In the graphs above, the extrema location of a profile is marked on the time axis by a colored dot. Note that profiles from the same local normal (top row) share most of the extrema locations, whereas profiles from different normals (bottom row) do not.

fects such as cast shadows and interreflections. In Figure 2.3 such a scene is shown consisting of three piecewise planar objects: a pyramid, a box, and a diamond. This was rendered using the Oren-Nayar and Torrance-Sparrow models whose material properties were varied (roughness $\sigma$ from 0 to 1 and albedo $\rho$ from 0 to 1) to create 100 instances of the scene, two of which are shown in Figure 2.3. A light source moving in an unstructured path was simulated, producing 40 renderings of each such scene instance. There are nine unique normals in our scene and, unlike the previous scenario, each normal was associated with at least 2,000 profiles. On the left of Figure 2.3, four images of the rendered scene are shown with different material properties. When the objects are rendered with the Oren-Nayar model, increased roughness makes the objects appear flatter and darker. Similarly sharp highlights in the Torrance-Sparrow model, such as on the right facet of the green pyramid, become blurred as roughness increases. On the right of Figure 2.3 profiles from three different normals are plotted and, for clarity, only the maxima locations are marked on the x-axis. Even though all the profiles vary significantly with change in material properties, the iso-normal profiles share the same extrema.

**Measured BRDFs:** The two previous experiments dealt with profiles generated from artificial scenes. Next, 25 real materials were selected from a BRDF database measured by Matusik et al ([57]). To create appearance profiles from these measured BRDFs, the light source path was simulated by the user, similar to the previous cases. For each material, the hemisphere of directions was sampled uniformly to create 25 unique normals. Each normal was used to generate profiles whose material properties corresponded to the measured BRDFs.

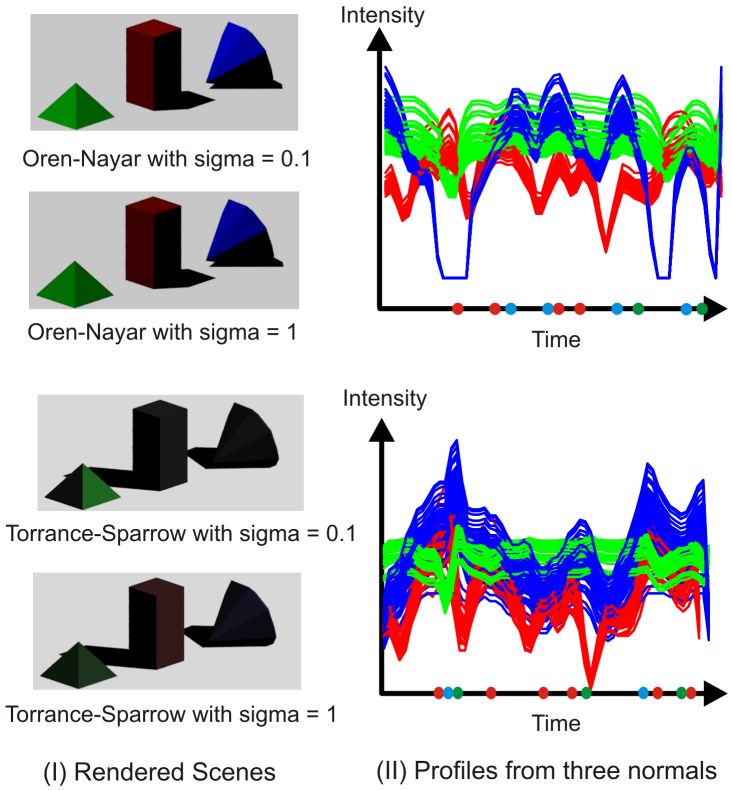



Figure 2.3: **Profiles from iso-planar regions of rendered scenes show the link between extrema and surface Normal:** Three piecewise planar shapes are rendered using both the Oren-Nayar general lambertian model, as well as the Torrance-Sparrow off-specular model under smoothly moving distance light source. Varying the roughness parameters $\sigma$ in both models produced 2,000 profiles, of which we show a few here. Profiles from three normals in the rendered images are shown, and iso-normal profiles have the same color. These experiments indicate that iso-normal profiles share the same extrema. The common maxima for each of normal is marked using the same profile color.

For each of these normals, over 90% of the profiles shared their extrema, and in Figure 2.4 we show 5 such profiles from two such different normals. These experiments demonstrate that the extrema feature has significance beyond the commonly used BRDF models.

**Real Textures:** In addition to the above simulations, real experiments were conducted with four anisotropic textures: silk, crepe satin, mink velvet, and royal velvet. These materials cannot be modeled by the Oren-Nayar and Torrance-Sparrow BRDFs. The textures were attached on a planar board so that they all shared the same surface normal, as shown in Figure 2.5. The profiles from these textures were measured for six different orientations of the board, and some of these are shown in the figure. These materials have complex appearance effects; for example, the maroon mink velvet exhibits strong vertical specularities which change in position and width as the orientation of the board is changed. Even though the profiles are drastically different from each other, they still share some extrema because they have the same surface normal.

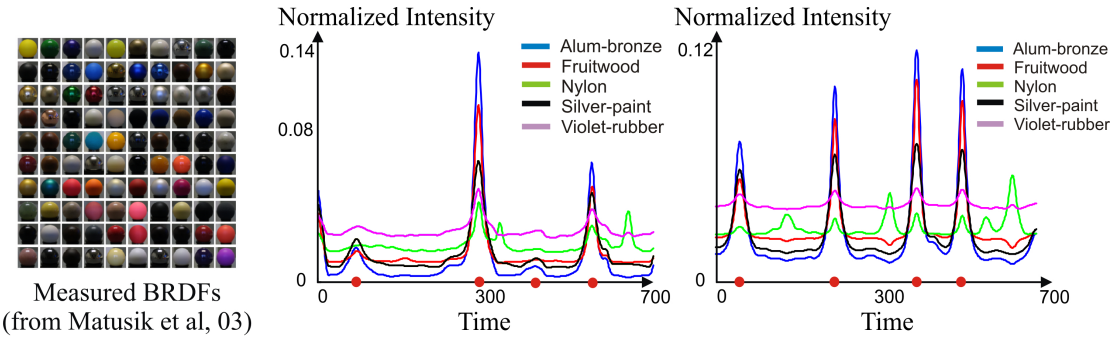



Figure 2.4: **Profiles from real, measured BRDFs:** Matusik et al ([57]) measured a large number of real world materials by using spheres, as shown on the left in a figure reproduced from their paper. Profiles were created from these measurements by simulating the path of a hand-waved light source and these are shown for two different surface normals. In each graph, the profiles for five materials share extrema locations (given by red dots) and these locations are different for the two normals.

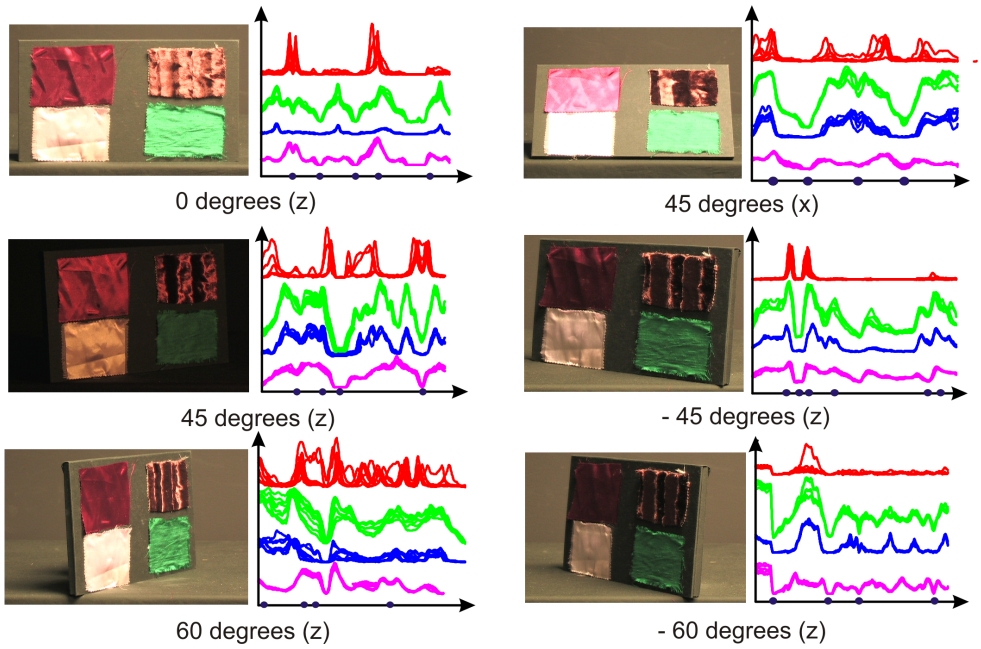



Figure 2.5: **Real profiles from different materials show the link between extrema and surface Normal:** Six experiments are conducted by placing textures at different orientations with respect to the camera. Note that the appearance of the different patches vary greatly, even though they are all flat on the board. Some profiles are shown from each of the textures in a color-coded fashion. Even though these materials are anisotropic (satin and velvet), their profiles still share extrema, which are marked by black dots on the x-axis.

In conclusion, we have presented four sets of experiments providing strong empirical evidence that shared extrema can be used as a material-invariant cue for a scene point's local surface normal. In the next section, an algorithm is presented that exploits this feature to produce iso-normal clusters.

### 2.1.3 Extrema in Linearly Separable BRDF Models

Here, we consider static scenes described by the class of linearly separable BRDFs ([62]), which include the commonly used lambertian, Oren-Nayar, Torrance-Sparrow and dichromatic models. If such a scene is illuminated by a smoothly moving distant light source, the observed radiance of a scene point at a time instant $t$ can be modeled as a dot product between "material terms" $\mathbf{M_i}$ (functions of material properties like diffuse and specular albedo $\rho$, roughness $\sigma$) and "geometry terms" $\mathbf{G_i}$ (functions of surface normal $\mathbf{n}$, viewing $\mathbf{v}$, and illumination directions $\mathbf{s}(t)$) [62]:

$$E(t) = \sum_{i}^{k} \mathbf{M_i}(\rho, \sigma)\mathbf{G_i}(\mathbf{n}, \mathbf{v}, \mathbf{s}(t)) \,. \tag{2.1}$$

The above linearly separable BRDF model represents a broad class of BRDFs (since no specific expressions for $\mathbf{M_i}$'s or $\mathbf{G_i}$'s are assumed) and many well known BRDFs used in computer vision (lambertian, Oren-Nayar, dichromatic) are special instances of this model ([16],[21],[82],[43]). The extrema of the appearance profile $E(t)$ are found by setting its first order derivative $E^{'}(t)$, with respect to time $t$, to zero:

$$E^{'}(t) = \sum_{i}^{k} \mathbf{M_i}(\rho, \sigma)\mathbf{G_i^{'}}(\mathbf{n}, \mathbf{v}, \mathbf{s}(t)) = 0 \tag{2.2}$$

One solution to this linear system occurs when $\forall \mathbf{i}\ \mathbf{G_i^{'}} = 0$, at a particular time instant $t$, irrespective of material terms $\mathbf{M_i}$ (which are assumed to be non-zero). These extrema are invariant to material properties, and we call them **Geometry-Extrema**. All other solutions to Eq. 2.2 are **Material-Extrema** since their time of occurrence depends on the material properties of the scene point $\mathbf{M_i}$'s. Two questions remain to be answered before we can achieve our goal: (a) If two scene points share all their Geometry-Extrema, does it follow that they are iso-normal? (b) How can we tell if the shared extrema between two scene points are Geometry-Extrema or Material-Extrema? We now show that the smooth, unstructured path of the light source addresses these issues.

#### Geometry-Extrema and Iso-Normal Clusters

Let two scene points have appearance profiles $E_1(t)$ and $E_2(t)$ and normals $\mathbf{n_1}$ and $\mathbf{n_2}$ respectively, such that $\mathbf{n_1} \neq \mathbf{n_2}$. Let a single shared Geometry-Extrema occur at some time instance $t$ in the two profiles. Then, from Eq. 2.2,

$$\forall \mathbf{i} \;\; \mathbf{G_i}^{'}(\mathbf{n_1}, \mathbf{s}(t)) = 0\,; \quad \mathbf{G_i}^{'}(\mathbf{n_2}, \mathbf{s}(t)) = 0\;. \tag{2.3}$$

Since in the above discussion the normals at the two scene points are fixed, we can rewrite Equation 2.3:

$$\forall \mathbf{i} \;\; \mathbf{G}^{'}_{\mathbf{i1}}(\mathbf{s}(t)) = 0\,; \quad \mathbf{G}^{'}_{\mathbf{i2}}(\mathbf{s}(t)) = 0\;. \tag{2.4}$$

The terms $\mathbf{G}^{'}_{\mathbf{i1}}$ and $\mathbf{G}^{'}_{\mathbf{i2}}$ are functions on the hemisphere of directions. The two equalities in Equation 2.4 only hold when $\mathbf{s}(t)$ is a *root* of both these functions. Let us assume that the set of common roots to both $\mathbf{G}^{'}$ terms is finite. Since the light source is being hand-waved, the light source positions $\mathbf{s}(t)$ are being generated randomly at every time step, on the hemisphere of directions. The probability that such a randomly generated $\mathbf{s}(t)$ intersects the finite set of common roots is low. Even if such a rare extrema occurs a few times, it would be very unlikely to occur frequently unless the two scene points have the same surface normal. Therefore, if two profiles share many Geometry-Extrema, it is likely that the corresponding scene points are iso-normal. The longer the profiles become, the greater the number of shared Geometry-Extrema will be and therefore the likelihood of iso-normality increases.

### Increasing Shared Geometry-Extrema

The above discussion suggests that we should group together those profiles that share all their Geometry-Extrema to get iso-normal clusters. However, this requires the ability to distinguish between Material-Extrema and Geometry-Extrema in a profile. We avoid this difficult problem and instead show that profiles created by hand-waving a light source are dominated by shared Geometry-Extrema while having negligibly few shared Material-Extrema.

Consider a light source passing directly over a scene-point's normal at time $t$. This would give maxima in foreshortening term $f(\mathbf{n}, t) = \mathbf{n}.\mathbf{s}(t)$. Note that every geometry term in the linear separable model has to have a foreshortening term in it. Therefore the $\mathbf{G}$ terms in our appearance profile model can be written as:

$$\mathbf{G_i}(\mathbf{n}, t) = \mathbf{g}(\mathbf{n}, t).f(\mathbf{n}, t) \tag{2.5}$$

where $f$ is the foreshortening of that particular point. Since we are only observing a single scene point, we will remove the normal variable $\mathbf{n}$ from the equations. Taking the derivative, with respect to $t$, we obtain,

$$\mathbf{G}^{'}(t) = \mathbf{g}^{'}(t).f(t) + \mathbf{g}(t).f^{'}(t) \tag{2.6}$$

Consider a situation where a foreshortening extrema occurs at some time $t_i$, and therefore, $f^{'}(t_i) = 0$. If we assume that the BRDF term $g(t)$ stays constant for a small time interval around $t_i$, then we get a Geometry-Extremum at $t_i$,

$$\mathbf{G}^{'}(t_i) = 0.f(t_i) + \mathbf{g}(t_i).0 = 0 \tag{2.7}$$

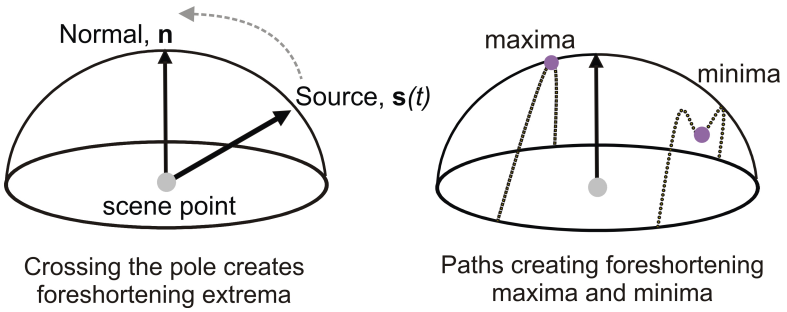



Figure 2.6: **Increasing foreshortening extrema increases geometry-extrema:** Paths of the light source that go directly over the scene point's normal, such as on the left, create a foreshortening maxima in the profile. In fact, any curved path, such as on the right, creates foreshortening maxima, even if they do not cross the pole. If such curved paths contain small changes in direction, as on the right hemisphere, these create foreshortening minima. We demonstrate in the paper how increasing foreshortening extrema results in increased geometry-extrema in the profile.

Figure 2.6 shows the types of light source paths that create foreshortening maxima and minima, and therefore create Geometry-Extrema. Note that curved paths which do not go over the pole of the hemisphere also create foreshortening maxima. In addition, curved paths with changes in direction, as on the right of Figure 2.6 create foreshortening minima. Since real scenes contain several different surface normals, an unstructured, random motion of the light source (as opposed to a structured one) will eventually create paths such as in Figure 2.6 at every scene point. In practice, we observe that these types of paths generate enough Geometry-Extrema to create accurate clusters.

### Reducing Shared Material-Extrema

While hand-waving a light-source produces many shared Geometry-Extrema, it is critical that Material-Extrema are *not* shared across profiles. Consider a profile at scene point $P$ which has a Material-Extrema at time $t$,

$$E'_P(t) = \sum_i^k \mathbf{M_i}(\rho, \sigma)\mathbf{G'_i}(\mathbf{n}, \mathbf{v}, \mathbf{s}(t)) = \tilde{\mathbf{M}}_\mathbf{P}.\tilde{\mathbf{G}}'_\mathbf{P} = 0. \tag{2.8}$$

Here $\tilde{\mathbf{M}}_\mathbf{P}$ is a vector of $\mathbf{M_i}$s, and $\tilde{\mathbf{G}}'_\mathbf{P}$ is a vector of $\mathbf{G'_i}$s. If the profile at $P$ shares this Material-Extrema with another profile at $Q$ at time $t$, then,

$$E'_Q(t) = \tilde{\mathbf{M}}_\mathbf{Q}.\tilde{\mathbf{G}}'_\mathbf{Q} = 0. \tag{2.9}$$

There are four cases in which coincident Material-Extrema can occur, which are expressed in terms of properties that are shared or not shared between scene points $P$ and $Q$, as shown in Figure 2.7. We explain each case below:

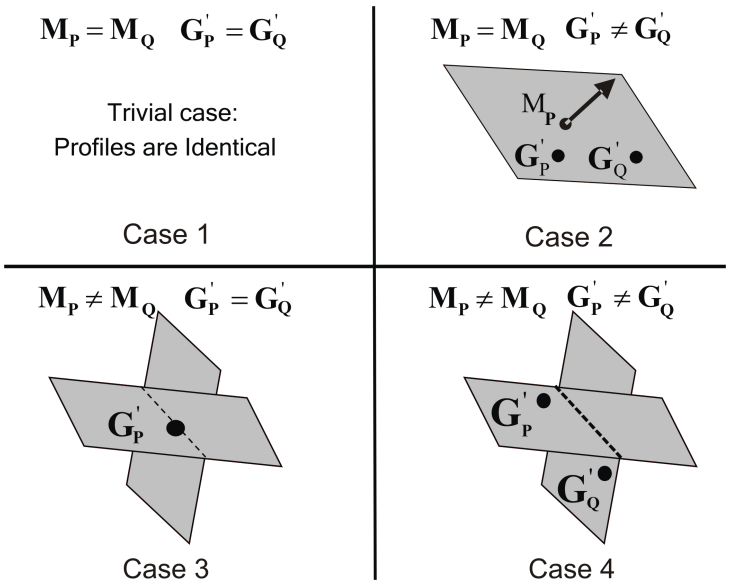



Figure 2.7: **Random hand-waving decreases material-extrema:** Coincident material extrema can be represented by the following four cases, where certain the geometry terms $G$s lie on specific hyperplanes defined by $M$s. By hand-waving a light source we ensure that the $G$s are randomly generated, and therefore the likelihood that these scenarios occur many times is low. Therefore the appearance profiles created do not have large numbers of coincident material extrema.

**Case 1:** Consider the situation where $P$ and $Q$ have identical material and geometry properties. This is the trivial case, where profiles from both scene points share all extrema and shared material extrema does not confuse isonormality and instead supports it.

**Case 2:** In this scenario, the scene points have the same material terms, $\tilde{\mathbf{M}}_\mathbf{P} = \tilde{\mathbf{M}}_\mathbf{Q} = \tilde{\mathbf{M}}$, but different geometry derivative terms, $\tilde{\mathbf{G}}^{'}_\mathbf{P}$ and $\tilde{\mathbf{G}}^{'}_\mathbf{Q}$. Coincident Material-Extrema occur at time instance $t_j$ when

$$\tilde{\mathbf{M}}.\tilde{\mathbf{G}}^{'}_\mathbf{P}(\mathbf{s}(t_j)) = 0; \quad \tilde{\mathbf{M}}.\tilde{\mathbf{G}}^{'}_\mathbf{Q}(\mathbf{s}(t_j)) = 0 \tag{2.10}$$

**Case 3:** If $P$ and $Q$ have the same geometry derivative terms, $\tilde{\mathbf{G}}_\mathbf{P}^{'} = \tilde{\mathbf{G}}_\mathbf{Q}^{'} = \tilde{\mathbf{G}}^{'}$, but different material terms $\tilde{\mathbf{M}}_\mathbf{P}$ and $\tilde{\mathbf{M}}_\mathbf{Q}$, then coincident Material-Extrema occur at time instance $t_j$ when

$$\tilde{\mathbf{M}}_\mathbf{P}.\tilde{\mathbf{G}}^{'}(\mathbf{s}(t_j)) = 0; \quad \tilde{\mathbf{M}}_\mathbf{Q}.\tilde{\mathbf{G}}^{'}(\mathbf{s}(t_j)) = 0 \tag{2.11}$$

**Case 4:** In the case where the scene points have different geometry terms ($\tilde{\mathbf{G}}^{'}_\mathbf{P}(\mathbf{s}(t_j))$ and $\tilde{\mathbf{G}}^{'}_\mathbf{Q}(\mathbf{s}(t_j))$) and different material terms ($\tilde{\mathbf{M}}_\mathbf{P}$ and $\tilde{\mathbf{M}}_\mathbf{Q}$), coincident Material-Extrema occur when:

$$\tilde{\mathbf{M}}_\mathbf{P}.\tilde{\mathbf{G}}^{'}_\mathbf{P}(\mathbf{s}(\mathbf{t_j})) = 0; \quad \tilde{\mathbf{M}}_\mathbf{Q}.\tilde{\mathbf{G}}^{'}_\mathbf{Q}(\mathbf{s}(\mathbf{t_j})) = 0 \tag{2.12}$$

In the above four cases, for Material-Extrema to occur, the geometry terms $\tilde{\mathbf{G}}'_{\mathbf{P}}(\mathbf{s}(t_j))$ and $\tilde{\mathbf{G}}'_{\mathbf{Q}}(\mathbf{s}(t_j))$ must occur on two specific hyper-planes in $n$ dimensions, whose normals are defined by $\tilde{\mathbf{M}}_{\mathbf{P}}$ and $\tilde{\mathbf{M}}_{\mathbf{Q}}$. Since we are hand-waving the light source along an unstructured path, the light source positions $\mathbf{s}(t_j)$ are generated randomly. The likelihood that two randomly generated geometry terms occur on two specific planes *at the same time* is small. However, even if this unlikely event occurs a few times, it is less likely to occur frequently. Therefore, shared Material-Extrema between the two points $P$ and $Q$ are rare.

### 2.1.4 Algorithm to Create Iso-normal Clusters

Our algorithm is extremely simple to implement and is summarized in Table 1. It has four steps, **1)** Detect brightness extrema, **2)** transform the appearance profiles (Figure 2.8) and **3)** use a common similarity metric to **4)** cluster the scene with any unsupervised clustering technique from machine learning. We write it out in pseudocode below.

**Step 1 (Input):**

While acquiring frames by randomly waving a light,

Detect intensity extrema at each pixel and
store their occurrences in time.
(No need to store whole image sequence)

end

**Step 2 (Transformation):**

Construct a feature vector from each scene point's profile by piece wise linear interpolation of its extrema stored in Step 1 (Figure 2.8).

**Step 3 (Metric):**

Compute distance metric between (normalized) feature vectors $\vec{a}$ and $\vec{b}$ using dot-product: Distance = 1 - $\vec{a}^t\vec{b}$.

**Step 4 (Output):**

Cluster the normals based on the metric in Step 3.

**Transformation applied to profiles:** From the discussion in the previous sections, it may appear obvious that extrema locations in a profile should be directly used for clustering. However, it is not clear how to compare profiles with different numbers of extrema. We propose a transformation of the appearance profiles that involves linearly interpolating between extrema locations, to create a new profile made of line segments, as shown in Figure 2.8. The slope of each line segment is the sign of the profile's derivative in that segment: it is either $+1$ or $-1$. There is no difficulty in comparing profiles with different numbers of extrema, since all the transformed profiles have the same length.

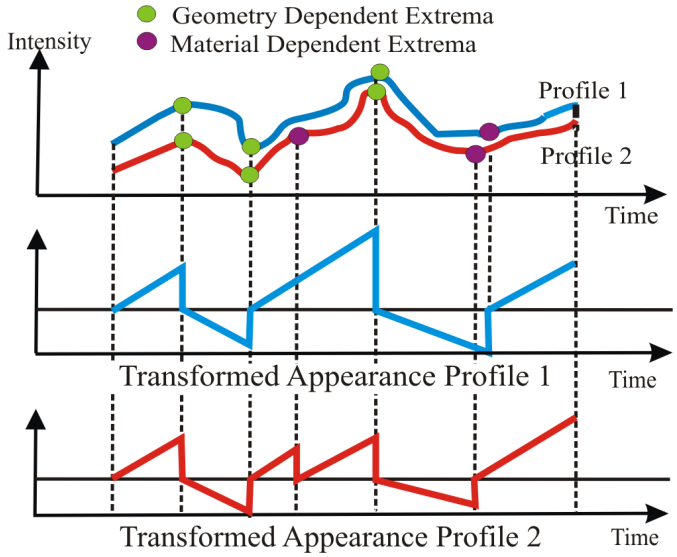



Figure 2.8: **Transformation applied to appearance profiles:** This illustration shows the effect of transformation on two hypothetical appearance profiles. Consider the 'segments' between extrema. The slope of transformed profile is the sign of the first derivative of a segment. Therefore two segments that have positive first derivative (monotonically increasing), get the same positive slope of 1. Note that in segments where there are no material dependent extrema, the transformed values are identical.

**Distance Metric between Profiles:** To cluster the transformed profiles we need to specify a distance metric. We recommend the "dot-product" metric which has been shown to accurately match extrema locations of profiles ([84]). Mathematically, if $A$ and $B$ are the transformed appearance profiles of two scene points, the "dot-product" metric is simply $1 - a^T\ b$, where $a$ and $b$ are the unit vectors obtained by normalizing the profiles $A$ and $B$.

**Bounding error due to Material-extrema:** Here, we show the dot-product distance between *transformed* iso-normal profiles becomes zero as their length becomes very large. On the other hand, transformed profiles from different normals will have many more unshared extrema compared to shared extrema, and therefore their dot-product distance will not go to zero. Consider two appearance profiles A and B from scene points that share the same surface normal in Figure 2.9. Consider two consecutive geometry-extrema located at frames $T_1$ and $T_2$. We say the intensity values between $T_1$ and $T_2$ in the appearance profile A are part of a *segment*. Let $i$ be an index for values in this segment. Therefore, $i$ varies between 1 and $T_2 - T_1 + 1$. Let $p$ be the frame location of the $i_p^{th}$ value in the segment. Now, according the transformation described before, we linearly interpolate between extrema locations with a slope whose magnitude is fixed, and whose sign is the sign of the gradient of the profile. In this discussion, we fix the magnitude of the slope to be 1, but any constant could be used. Then, the value of the transformed profile of A at $p$ is:

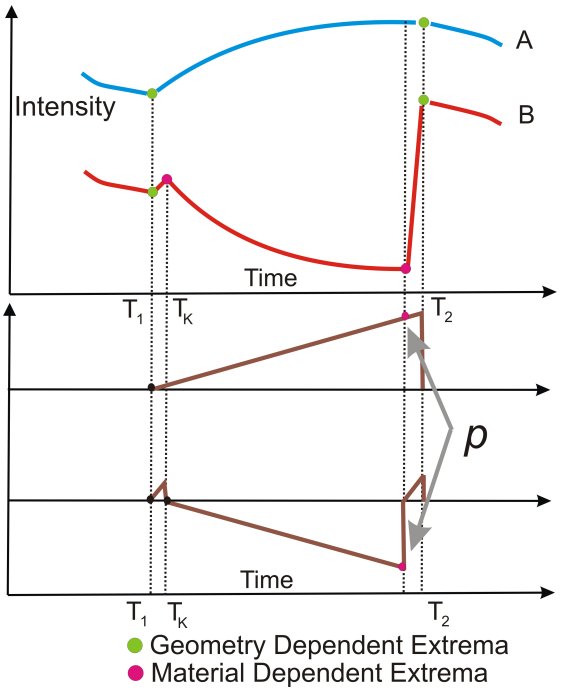



Figure 2.9: **Transformation bounds error due to material-extrema:** In the figure above, the red profile has a material-extrema at location $p$. Because of the material extrema, the intensities in the red and blue profiles vary drastically. This error can be arbitrarily large because we make no assumptions about the BRDF of the scene. We propose a transformation that only takes into account the location of the extrema, not the intensity value at that extrema location. Therefore, after transformation the error is constant, irrespective of the intensity value variation between the blue and red profiles.

$$A_{transf}(p) = i_p \tag{2.13}$$

Now let us say B has a material dependent extrema at location $T_k$. Let $j$ be the index of values in the profile B, for values in the segment between $T_k$ and $T_2$. The index $j$ varies from 1 to $T_2 - T_k + 1$. Since B has material-dependent extrema between $T_1$ and $T_2$ the absolute value of the transformed B profile at $p$ becomes less than the absolute value of the transformed A profile at $p$ (see Figure 2.9). This is because the transformed B profile starts at $T_k$, while the transformed A profile starts at $T_1$, and $T_K > T_1$. If $j_p$ is the index of the $p^{th}$ value of the transformed profile B, then:

$$B_{transf}(p) = -j_p \tag{2.14}$$

and therefore for the Figure 2.9,

$$B_{transf}(q) < A_{transf}(q) \tag{2.15}$$

Note that the example we have shown has profile $A$ is monotonically increasing within the segment. In the case where $A$ is decreasing in the segment,

the signs for the equations above change. Therefore, we extend for any point $q$, between $T_1$ and $T_2$:

$$\|B_{transf}(q)\| < \|A_{transf}(q)\| \tag{2.16}$$

If we normalize the profiles, then the only error between $A$ and $B$ comes from the segments such as in Figure 2.9, with material-dependent extrema. Let the set of locations in the segment be $P$. If the error in the values at locations in $P$ are removed, then the new appearance profile so obtained is identical to $A$, and hence the dot-product distance becomes 0. This would mean the actual dot-product of the unit profiles is 1. Let the error due to material-extrema for each location $i \in P$ in $B$ be $e_i$. For those values:

$$B_{transf}(i) = A_{transf}(i) + e_i, \quad i \in P \tag{2.17}$$

For any other location, $i \notin P$. For segments without material dependent extrema, the error is zero and therefore:

$$B_{transf}(i) = A_{transf}(i), \quad i \notin P \tag{2.18}$$

Let the length of the profiles $A$ and $B$ be $d$. The dot-product between the two unit profiles:

$$F = \frac{\sum_{k=1}^{d} A_{transf}(k) B_{transf}(k)}{\left( \sum_{k=1}^{d} (A_{transf}(k))^2 \sum_{k=1}^{d} (B_{transf}(k))^2 \right)^{\frac{1}{2}}} \tag{2.19}$$

From Equation 2.16,

$$F = \frac{\sum_{k=1}^{d} A_{transf}(k) B_{transf}(k)}{\left( \sum_{k=1}^{d} (A_{transf}(k))^2 \sum_{k=1}^{d} (B_{transf}(k))^2 \right)^{\frac{1}{2}}} > \frac{\sum_{k=1}^{d} A_{transf}(k) B_{transf}(k)}{\left( \sum_{k=1}^{d} (A_{transf}(k))^2 \sum_{k=1}^{d} (A_{transf}(k))^2 \right)^{\frac{1}{2}}} \tag{2.20}$$

and from Equations 2.17 and 2.18, this becomes

$$F > \frac{\sum_{k \notin P} (A_{transf}(k))^2 + \sum_{k \in P} A_{transf}(k) B_{transf}(k)}{\left( \sum_{k=1}^{d} (A_{transf}(k))^2 \sum_{k=1}^{d} (A_{transf}(k))^2 \right)^{\frac{1}{2}}} \tag{2.21}$$

Adding and subtracting a term in the numerator:

$$F > \frac{\sum_{k \notin P} (A_{transf}(k))^2 + \sum_{k \in P} (A_{transf}(k))^2 - \sum_{k \in P} (A_{transf}(k))^2 + \sum_{k \in P} A_{transf}(k) B_{transf}(k)}{\sum_{k=1}^{d} (A_{transf}(k))^2} \tag{2.22}$$

Since the sum of the first two terms is the denominator, we can simplify further,

$$F > 1 + \frac{-\sum_{k \in P} (A_{transf}(k))^2 + \sum_{k \in P} A_{transf}(k) B_{transf}(k)}{\sum_{k=1}^{d} (A_{transf}(k))^2} \tag{2.23}$$

and from Equation 2.16

$$F > 1 + \frac{-\sum_{k \in P} (A_{transf}(k))^2 - \sum_{k \in P} A_{transf}(k)^2}{\sum_{k=1}^{d} (A_{transf}(k))^2} \tag{2.24}$$

Since $F$ is the expression for the dot-product, and we use the distance metric specified by $1 - F$, then we can an expression for, $D$, the distance between profiles:

$$D = 1 - F < \frac{\sum_{k \in P} (A_{transf}(k))^2 + \sum_{k \in P} A_{transf}(k)^2}{\sum_{k=1}^{d} (A_{transf}(k))^2} \tag{2.25}$$

or,

$$D < \frac{2 * \sum_{k \in P} (A_{transf}(k))^2}{\sum_{k=1}^{d} (A_{transf}(k))^2} \tag{2.26}$$

We want the dot-product distance between to be zero when $A$ and $B$ are generated by the same normal. Recall that we previously introduced randomly waving a light source as a way to create profiles with lots of geometry-extrema and few material-extrema. Since profiles will have far more segments without material-extrema than with. Therefore, for a very long profile:

$$\sum_{k=1}^{d}(A_{transf}(k))^2 >> \sum_{k\in P} A_{transf}(k)^2 \tag{2.27}$$

Therefore $D \Rightarrow 0$ as the profiles become long. We can explain this result more intuitively by recalling Equation 2.14 that relates the transformed values of the profile to indices within a segment. Therefore Equation 2.26 can be written in terms of these indices:

$$D < \frac{2 * \sum_{k\in P}(i_k)^2}{\sum_{k=1}^{d}(i_k)^2} \tag{2.28}$$

Note that we are summing the squares of the indices in each segment. In the numerator we do this for segments with material-extrema, and in the denominator we do this for segments without material-extrema. Now, the sum of squares of $n$ positive numbers depends on the $n^3$. To make our expression concise, we say the sum of squares is given by a function, $S(n^3)$. Since the number of indices in each segment depends on the length of the segment (see Figure 2.9), the terms in the numerator and denominator depend on the lengths, $L$, of the segments:

$$D < \frac{2 * \sum_{k\in P} S((L_k)^3)}{\sum_{k=1}^{d} S((L_k)^3)} \tag{2.29}$$

If profiles are created by the unstructured method discussed previously, then the lengths of segments with material extrema are much smaller than those without material-extrema. This is the reason why the distance between profiles $A$ and $B$ from the same normal goes to zero. If the profiles are from different normals, then there is no such ratio between material-extrema segments and geometry-extrema segments. In this case the distance $D$ does not go to zero. With long profiles, our distance metric therefore can be used with any machine learning algorithm to create iso-normal clusters of scenes.

Finally, one of the parameters that needs to be decided by our algorithm is the number of clusters, $k$. Calculating the number of clusters automatically is an unsolved problem in machine learning. It may be possible to use domain specific knowledge about appearance profiles to calculate the right $k$ for our clustering algorithm, but that would require some knowledge of scene properties. Instead we advocate a well-known and simple method to decide $k$ called hierarchical agglomerative clustering, which involves merging clusters. First, over-cluster the scene with a large $k$. In successive steps, clusters are merged

Figure 2.10: **Setup:** Our acquisition setup with a Canon XL2 video camera, a 60 watt light attached to a wand. In real experiments the camera and light source are further away to satisfy orthographic assumptions.

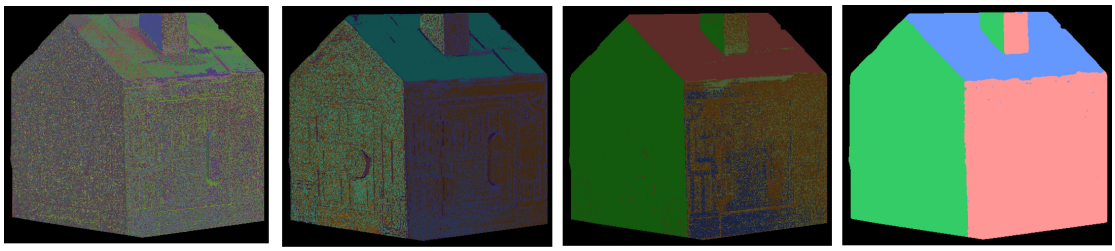



Figure 2.11: **Choosing the number of clusters:** Here the number of clusters, $k$, are varied for a particular scene. In each case the clusters created are still iso-normal. Merging the clusters at each step allows us to create bigger iso-normal clusters. The merging is stopped when the distance between cluster centers is larger than a user-defined threshold.

if their distance is less than some user-defined threshold. However, there are many cases, such as smoothly moving cast shadows and curved surfaces, for which there are an infinite number of valid clusters in the scene. Clustering simply gives a piecewise approximation, and the best choice of a $k$ for the algorithm is hard to obtain in such cases. In these scenarios we suggest the user input a reasonably large value of $k$. Although this over-clusters the scene, it makes sure that the sub-clusters produced are consistently iso-normal.

### 2.1.5 Experiments with Real Scenes

We will now demonstrate our algorithm using a wide range of real indoor and outdoor scenes with complex scene structure and material properties. Our setup consists of a Canon XL2 digital video camera observing a static scene as shown in Figure 4.13. As discussed before, number of clusters is decided using a simple merging technique, For example, in Figure 2.1.5 we cluster a painted house model where the number of clusters, $k$, was automatically selected when the distance between clusters became greater than a user-defined threshold (which was $0.5$ in this experiment).

Our algorithm was first tested on piecewise planar scenes consisting of real textures from the CURET database ([13]). The CURET patches are arranged in a scene and light source is waved, creating 200 video frames. Note the boxed regions at the top of Figure 2.12. These textures are artificial grass and real straw, whose appearances are quite complex; for example, artificial grass has many specularities and is also rough. The second row shows steel wool and real wool, which are 3D textures with uneven height from the patch surface. Our algorithm clustered all these textures properly, even though their material properties were very different. In Figure 2.13, our algorithm clusters anisotropic materials, such as velvet, satin, shiny paper and fur, implying that our method works even in some cases that are not described well by our illumination model. Results are also shown for non-planar objects which contain an infinite number of normals. In these cases, our method creates a piecewise planar approximation of the continuous curved surface.

In Figure 2.14 we show more complex planar scenes, containing occlusion, cast shadowing and inter-reflection. In these regions, our method may over-cluster the scene, but note that the smaller clusters are still geometrically consistent. Clustering was also demonstrated on some everyday, indoor scenes such as the chair and table shown in Figure 2.15. Even though these are non-lambertian scenes with materials such as wood, plastic, metal and smooth tile, our algorithm creates meaningful clusters. In Figure 2.16, clustering results obtained for outdoor images of a scene collected from the WILD database ([63]) are shown. Note that this scene does not satisfy many of our assumptions. For example, it is illuminated by the sun and sky instead of a randomly-moving point light source. There is also significant depth in the scene (see [63]), violating the orthographic assumption. A good result is still obtained because the diverse and random weather illumination (sunny, cloudy, fog, mist) creates appearance profiles with enough intensity variation to produce valid clusters.

We believe iso-normal clusters will enable a variety of applications in vision and graphics. One such application is transferring texture in videos. The challenge here is to transfer appearance that is consistent under varying illumination. Profiles within a cluster share the same intensity extrema and therefore the corresponding scene points 'light up' and 'go dark' together. Transferring profiles within a cluster creates new pixels whose brightness varies *consistently*, as shown in Figure 2.17. The complex appearance effects of the materials are preserved through the length of this video sequence.

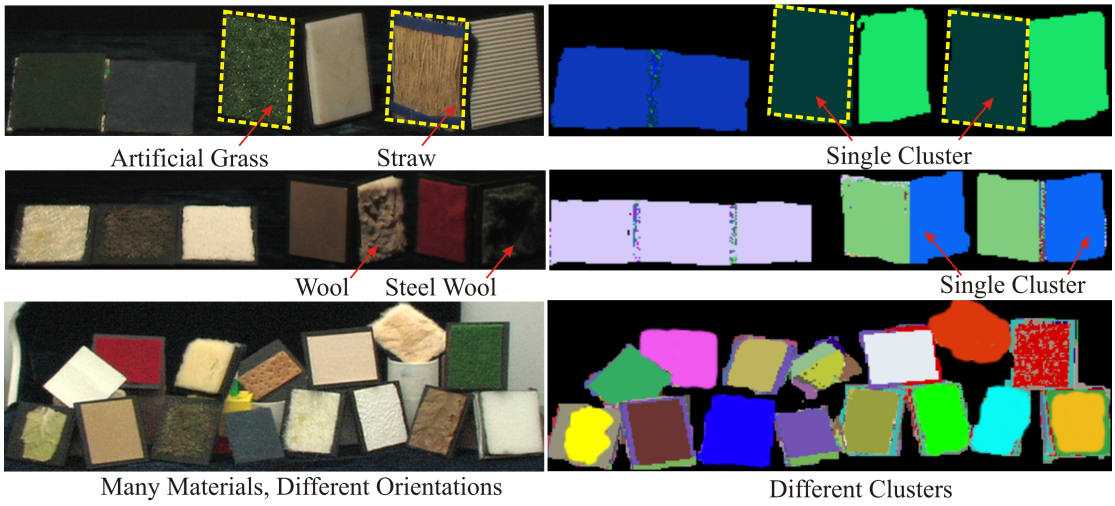



Figure 2.12: **Results obtained when our algorithm is used to cluster materials in the CURET database.** Image sequences of real CURET textures were obtained by waving a light source (We did not use the still images distributed by Columbia University). Notice the top row containing materials such as artificial grass and straw and the middle row with examples of real wool and steel wool. Despite significant appearance differences, these samples cluster together accurately because they share the same surface normal.

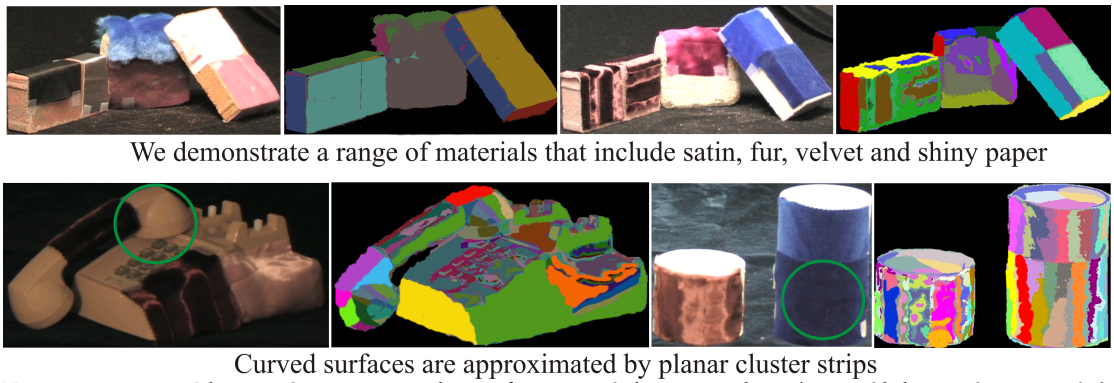



Figure 2.13: **Clustering curved surfaces with complex (possibly anisotropic) materials** When anisotropic BRDFs are present in the scenes, our method still produces meaningful clusters. Furthermore, for curved surfaces, our method produces a piecewise planar approximation.

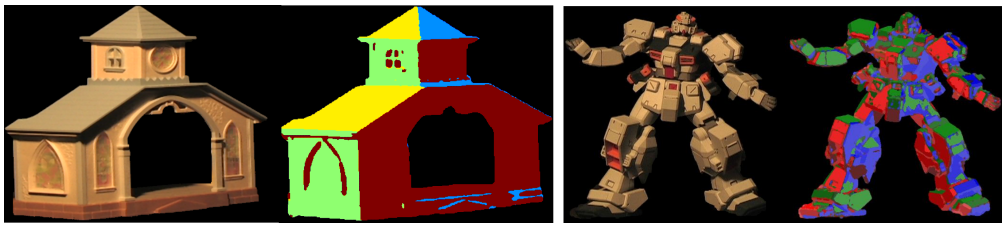


Figure 2.14: **Clustering surfaces with cast shadows.** When complex effects such as cast shadows and inter-reflections are present in the scenes, our method works for simpler scenes such as on the left. For more complex scenes, such as on the right, our method may fail to group all pixels in the scene that have the same normal. Instead, the algorithm simply over clusters the scene into smaller iso-normal clusters.

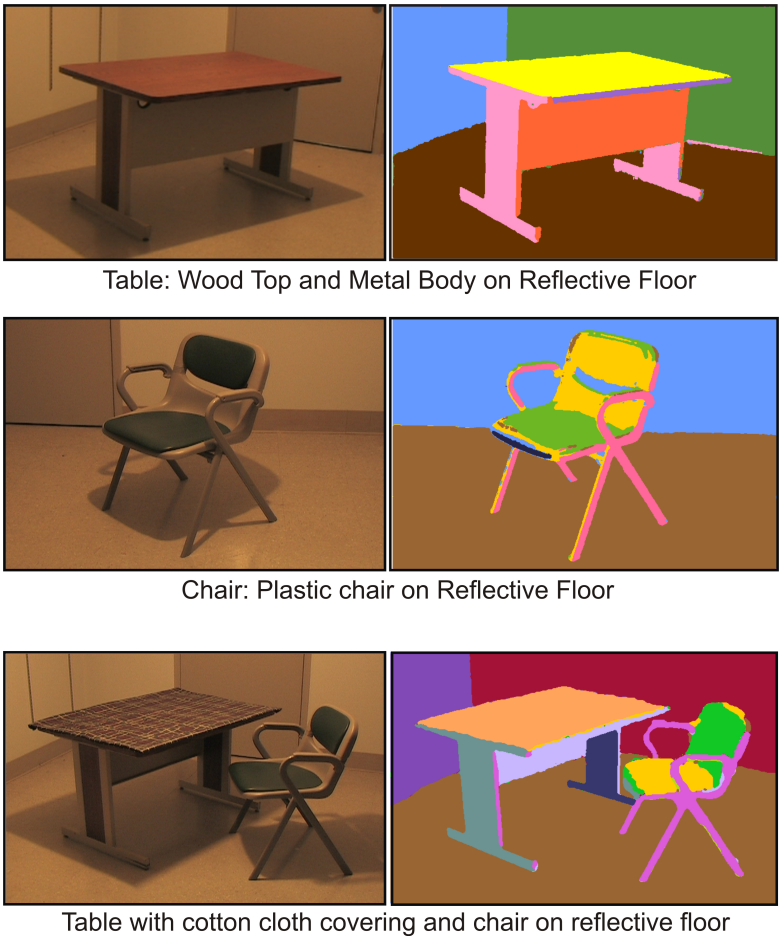
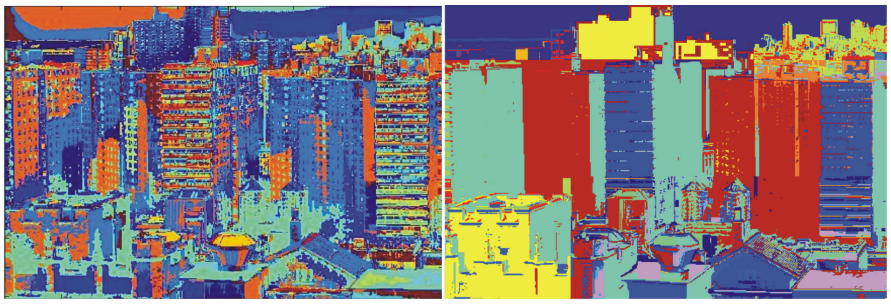


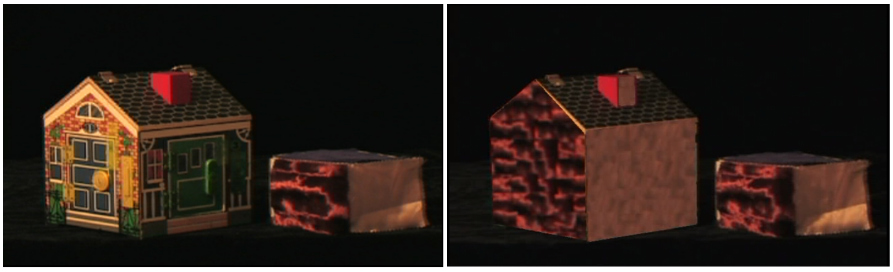

Figure 2.15: **Clustering indoor scenes.** The following are three indoor scenes containing non-lambertian objects, such as the metal table with a wood top, a metal door, reflective floor tile, plastic chairs and a texture cotton cloth placed on the table. In spite of all of these our clustering algorithm does well. Note: some of the clusters have been merged by the user.

Outdoor images under varying illumination

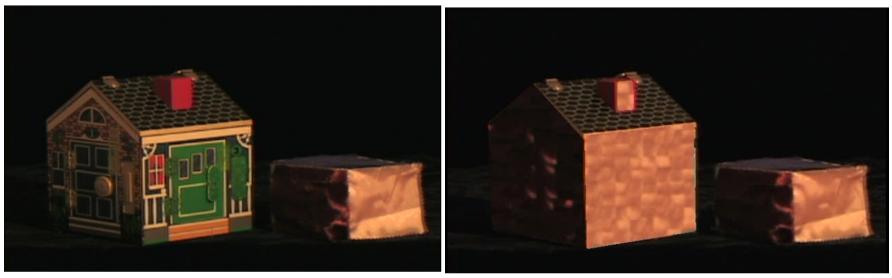


Using raw profiles Our Method

Figure 2.16: **Clustering WILD database:** Note the complex appearance effects that occur in this data set. Our transformation of the appearance profile and the dot-product distance metric does significantly better than using Euclidean distance metric on raw profiles. In both cases, k-means was used to cluster appearance profiles. Note: some sub-clusters were merged by the user.



Lighting direction that shows velvet effects



Lighting direction that shows satin effects

Figure 2.17: **Texture transfer** of complex materials (such as velvet and satin) between similar surface normals in a scene. A patch of the original scene is chosen by the user and a simple repetitive texture synthesis method is used to transfer this patch onto other areas of the scene with the same surface normal. Note the consistency in geometry and lighting in the transferred regions. Please see video at [47] for many more lighting variations.

## 2.2 Depth Cues from Uncalibrated Near Point Lighting

While in the previous section we modeled the motion of a distant source (like the sun), here we consider a stationary scene illuminated by a point source that is moved approximately along a line or in a plane. We observe the brightness profile at each pixel and demonstrate how to obtain three novel cues: plane-scene intersections, depth ordering and mirror symmetries. Our cues are obtained with respect to the light source path and not the camera. In practice, we extend our results to non-lambertian results by applying heuristics to remove the effect of sharp specularities and glossiness. We show results on a variety of indoor and outdoor scenes.

### 2.2.1 Maxima in Near Point Appearance Profiles

Let a static scene be illuminated by an isotropic point light source moving in a straight line (see Figure 2.18) at constant speed. At a line position $d = i$, the source $S$ is nearest to scene points lying on a plane perpendicular to the light source path. In this section, we show that maxima will occur in the brightness profiles of all of these scene points, owing to a minimum in the inverse-square fall-off from the point light source. By simply detecting brightness maxima occurring at every light source position (or frame), we obtain plane-scene intersections. Our results appear to be created by intersecting the scene with a plane that is translating along the source path. This is similar in spirit to structured light striping where a sheet light is swept across the scene.

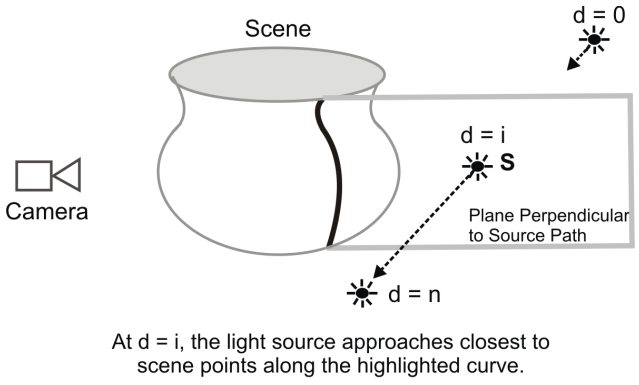



Figure 2.18: **Detecting a plane-scene intersection using brightness maxima:** Consider a light source moving in a line in front of a scene. At any line position d = i, the scene points that are closest to the light source path will display a maxima in their profiles. We use this maxima to create plane-scene intersections, similar to structured light line striping.

Let us assume, without loss of generality, that the light source has unit intensity. If the BRDF is given by $B$, the foreshortening by $f$, the incident light

angles by $\theta_s$ and $\phi_s$, the viewing angles given by $\theta_v$ and $\phi_v$, and the distance between the light source and the scene point at line position $d$ by $R(d)$, then the brightness profile $E(d)$ at a scene point is:

$$E(d) = \frac{B(\theta_s(d), \phi_s(d), \theta_v, \phi_v).f(d)}{(R(d))^2} = \frac{F(d)}{(R(d))^2} \tag{2.30}$$

where the $F$ term contains both BRDF and foreshortening. Taking the derivative with respect to position, $d$, gives us an expression for when the maxima of $E(d)$ should occur,

$$E^{'}(d) = \frac{(R(d))^2\ F^{'}(d) - 2\ R^{'}(d)R(d)\ F(d)}{(R(d))^4} = 0 \tag{2.31}$$

### 2.2.2 Plane-Scene Intersections for a Lambertian Scene

Without loss of generality, consider a scene point located at the origin, $P = (0, 0, 0)$, and let the light source move along a line parallel to the z-axis, and let its 3D position be $\vec{S}(d) = (D, 0, d)$ where $D$ is its closest distance to the origin. The distance from the scene point to the light source is $R(d) = (D^2 + d^2)^{0.5}$. For a scene point with albedo $\rho$ and surface normal $\vec{\mathbf{n}} = (n_x, n_y, n_z)$ we write:

$$F(d) = \frac{\rho\ \vec{\mathbf{n}}.(\vec{S}(d) - \vec{P})}{R(d)} \tag{2.32}$$

An iso-planar maxima occurs if $F^{'}(d)$ is zero when the light source reaches the closest distance to the scene point. Consider the expressions for $F^{'}(d)$ and $R^{'}(d)$,

$$F^{'}(d) = \frac{\rho R(d).(n_z) - \rho R^{'}(d)(Dn_x + n_z d)}{(R(d))^2} \tag{2.33}$$

$$R^{'}(d) = \frac{d}{R(d)} \tag{2.34}$$

Putting these into $E^{'}(d)$,

$$E^{'}(d) = \frac{-\rho(2n_z d^2 + 3Dn_x d - D^2 n_z)}{(R(d))^5} \tag{2.35}$$

Setting $E^{'}(d) = 0$ gives us a quadratic equation in $d$ with two possible solutions. By checking the sign of the second derivative, $E^{''}(d)$, for all possible normals, we found that one of these solutions is always a maxima and is given by,

$$d_{maxima} = D\left(\frac{-\frac{3}{4} + \sqrt{(\frac{3}{4})^2 + \frac{1}{2}(\frac{n_z}{n_x})^2}}{\frac{n_z}{n_x}}\right) \tag{2.36}$$

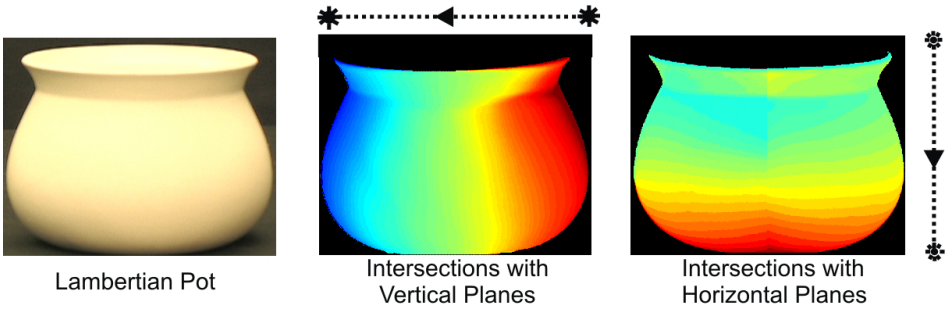



Figure 2.19: **Plane-scene intersections on a lambertian pot:** In this figure we show results of our method applied to a Lambertian clay pot. We obtain both horizontal and vertical plane intersections by moving a near point light source as shown by the marked paths. These results are similar to those obtained from structured light. The discontinuity in the result for horizontal planes is due to merging the intersections for the right and left halves of the pot. These were done separately to avoid blocking the camera view.

Ideally, iso-planar maxima should occur when $d = 0$, since that is when the light source is closest to the scene point and $R(0) = D$. Therefore, Equation 2.36 represents the error in the iso-planar maxima location. However, simulations show that this error remains bounded as $\frac{n_z}{n_x}$ is varied. For instance, $d_{maxima}$ is zero when $\frac{n_z}{n_x} \rightarrow 0$ and becomes $\frac{D}{\sqrt{2}}$ when $\frac{n_z}{n_x} \rightarrow \infty$. Although we will investigate this bound more thoroughly in future work, here we assume the error is negligible if the light source path is close to the scene point ($D$ is small).

In Figure 2.19 we show both horizontal and vertical plane-scene intersections for a lambertian pot, created by moving the light source first sideways and then upwards. We code the planes from blue to red, obtaining a continuum of color coded plane-scene intersections. For the second result, we merged two experiments for the left and right halves of the pot, to avoid blocking camera's view of the scene. Although these results appear similar to structured light images obtained by sweeping a plane over the scene, they were obtained by a user hand-waving a near point light.

**Shadows and Specularities:** Non-isoplanar brightness maxima are usually rare in lambertian scenes illuminated by a near light source. In fact, for a maxima to occur when $R^{'}(d) \neq 0$, a fortuitous occurrence of values for $R^{'}(d)$, $R(d)$, $F(d)$ and $F^{'}(d)$ would be required in Equation 2.31, which is less likely. However, scenes with sharp specularities still show non-isoplanar maxima. Fortunately, since specularities are characterized by a rapid rise in brightness, these are detected by thresholding the second derivative of the measured intensities. Shadows are handled by repeating the experiment with different parallel light source paths. Since a scene point shadowed in one experiment may be illuminated in another, we usually detect an iso-planar maxima.

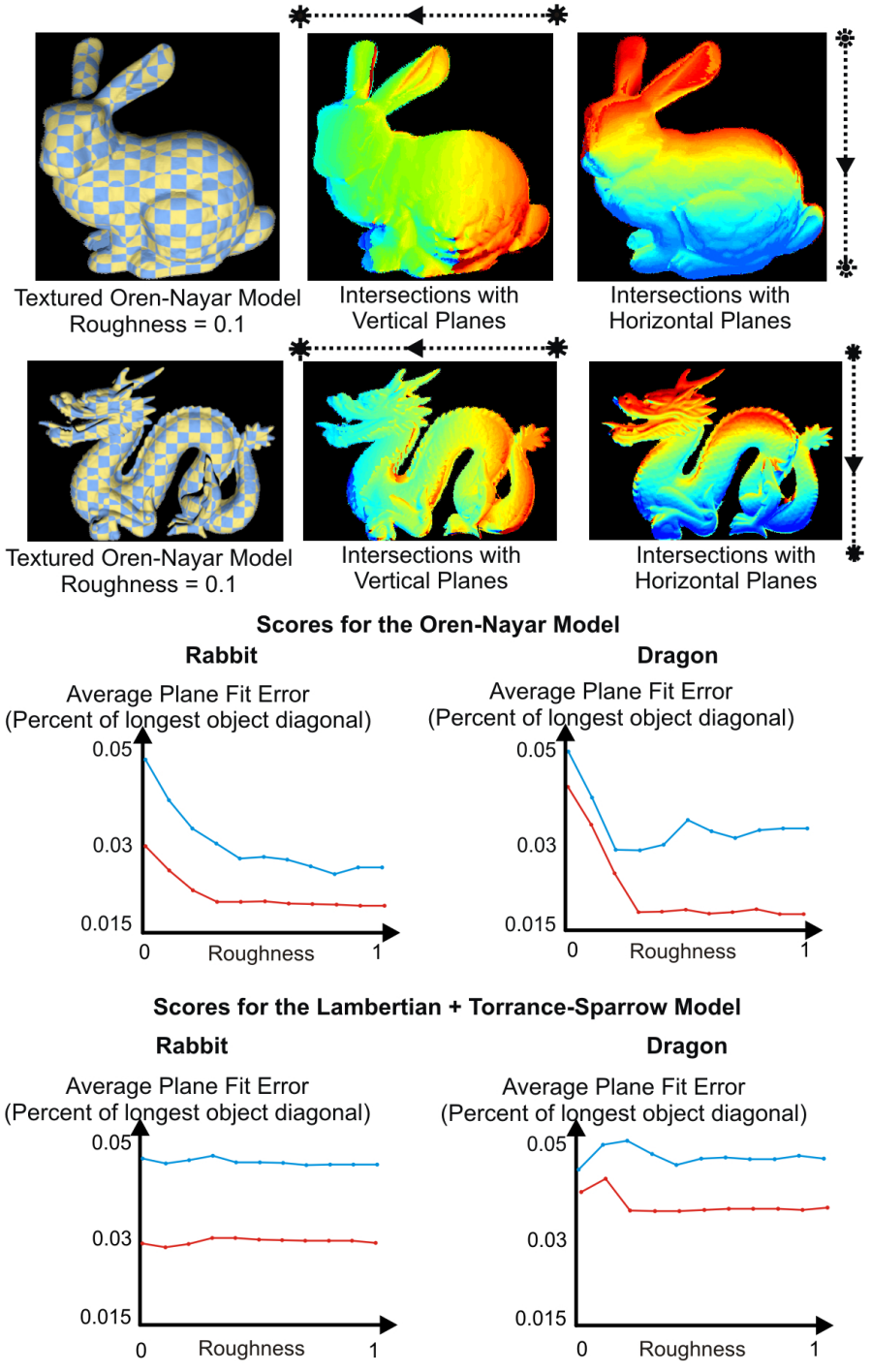



Figure 2.20: **Plane-scene intersections for rendered scenes:** We used a ray tracing software ([75]) to render two objects using both the Lambertian + Torrance-Sparrow and the Oren-Nayar models. For each set of images, we created plane-scene intersections, and we show two examples of these at the top of the figure. We fit a plane to each intersected region and measure the plane fit error. The average of these errors is plotted, as a percent of the longest distance contained in the object. These empirical results support the idea that the plane-intersection algorithm can be used with a variety of non-lambertian scenes.

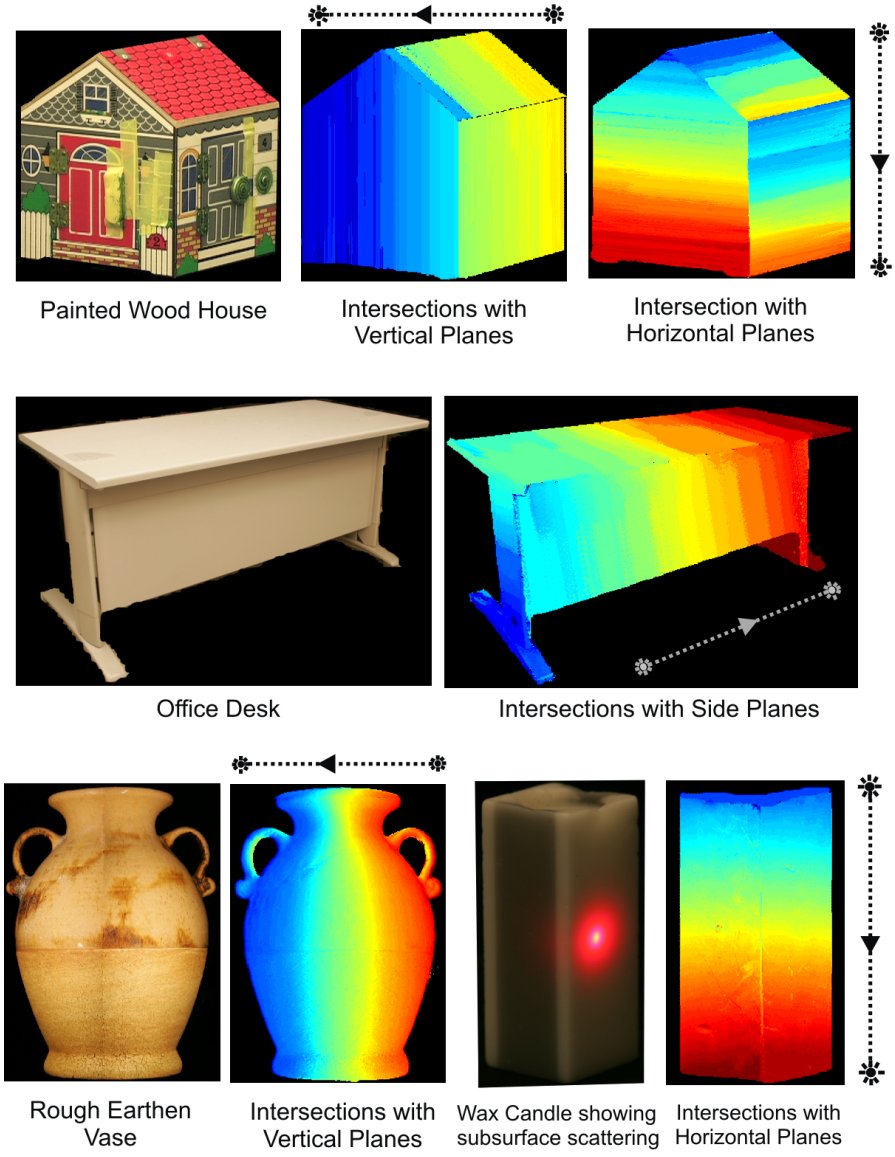



Figure 2.21: **Plane-scene intersections for real scenes:** At the top of the figure we show horizontal and vertical plane-scene intersections for a painted house, even though this scene demonstrates glossy specularities. We also show plane-scene intersections for an office desk made of metal and plastic. At the bottom we show the sub-surface scattering effects of a wax candle, by shining a point laser. Our method is able to create horizontal plane-scene intersections for this object with complex BSSRDF ([25]) appearance.

### 2.2.3 Experimental Results for Plane-Scene Intersections

In Figure 2.20, we show plane-scene intersection results for simulated scenes created using a ray-tracing tool ([75]). We rendered the bunny and dragon models by varying the parameters of the Oren-Nayar model (facet angle from $0$ to $\frac{\pi}{2}$) and of a Lambertian + Torrance-Sparrow model ($\sigma$ of facet distribution from $0$ to $1$). The light source was moved along the $x-$ and then $y-$ axes to create horizontal and vertical plane-scene intersections. Using ground-truth, we fit a plane to the 3D locations of scene points in the plane-scene intersections and plotted the sum-of-squared errors. The low errors indicate the robustness of our technique for non-lambertian BRDFs.

At the top of Figure 2.21, we show horizontal and vertical plane-scene intersections for a painted house model, which are detected despite glossy specularities in the scene. Our algorithm also produces good results for scenes with sharp specularities, such as the metal office desk in Figure 2.21 and for rough objects with cracks, such as an earthen pot in Figure 2.21 . We are also able to create plane-scene intersections for objects with sub-surface scattering (since BSSRDF ([25]) is smooth) such as a wax candle shown in Figure 2.21. The scattering effects are demonstrated using a laser pointer, and horizontal plane-scene intersections are shown.

### 2.2.4 Weaker Cues from Planar Light-source Motion

The cues in the following section can also be obtained from the smooth motion of a near point light source. However, unlike plane-scene intersections, they cannot be applied to scenes with non-homogeneous BRDF, and therefore we term them as weak cues.

#### Depth Ordering for Homogeneous Scenes

A scene with homogeneous BRDF is illuminated by a point source moving in a plane. Intuitively, scene points closer to the light source tend to be brighter than scene points further away. If we somehow manage to remove the effect of BRDF for any two scene points, then their appearance would depend only on their distances to the "base plane" containing the light source path. Although this is not possible generally, we will provide two heuristics to achieve good results for depth ordering: **a)** Move the light source over a large area of the base plane and **b)** Bring the base plane as close as possible to the scene. We support our method with strong empirical evidence using both simulations and real scenes.

#### Integrating the Brightness Profile

Consider a scene point $P$ illuminated by a light source moving in a plane. As before, we will assume the light source has unit intensity and let $F$ contain both the BRDF and foreshortening and $R(d)$ be the distance between the source and

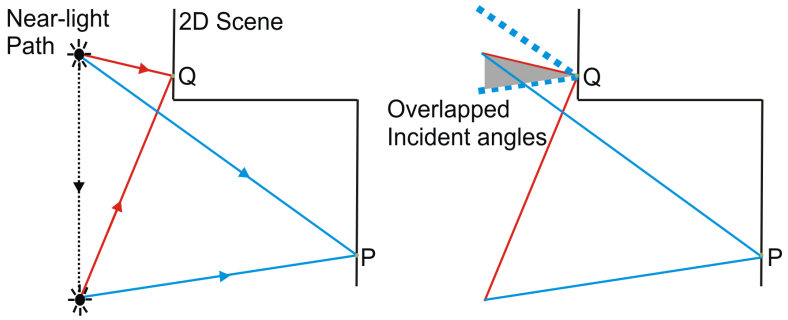



Figure 2.22: **Overlap of incident angles for a simple 2D scene:** A light source moving in a line illuminates a 2D scene with homogeneous BRDF. Let scene point P be at a greater depth than Q and let both have the same surface normal. Some of the incident angles at P are repeated in Q, and we show these *overlapped angles* in gray. Because of inverse-square fall-off, the measured intensities in the overlapped region will be higher in Q than in P. We extend this idea to create scene depth ordering for 3D scenes as well.

this scene point at position $d$. Let $S_p$ be the sum of the $P$'s intensities along the light source path from positions $d_1$ to $d_n$,

$$S_p = \sum_{d=d_1}^{d_n} \frac{F_p(d)}{(R_p(d))^2}. \tag{2.37}$$

Let there be another point $Q$ whose perpendicular distance to the plane containing the light source is less than that of $P$. At different light source positions, $P$ and $Q$ may observe identical incident angles. We term these as *overlapped* incident angles, and they are shown in gray in Figure 2.22. In this simple 2D scene, for an infinite path, all the incident angles at $P$ and $Q$ will overlap and, $S_p < S_q$.

This result is harder to demonstrate for real 3D scenes where $P$ and $Q$ can have different surface normals. To make the analysis easier, we assume that $R_p(d) > R_q(d)$ for all positions along the light source path (Figure2.23). We will show that this drawback is not severe since, in practice, we get good depth ordering results for scenes with a variety of geometries.

$S_p$ and $S_q$ consist of two components, one of which (denoted by $O$) comes from overlapped incident angles, as in the gray region in Figure 2.22. The other comes from non-overlapped or different incident angles (denoted by $N$). Since the order of the summation does not matter, we let the overlapped angles occur from $d_1$ to $d_i$, and the non-overlapped from $d_{i+1}$ to $d_n$. We separate the summation accordingly as:

$$S_p = \sum_{d=d_1}^{d_i} \frac{F_p(d)}{(R_p(d))^2} + \sum_{d=d_{i+1}}^{d_n} \frac{F_p(d)}{(R_p(d))^2} \tag{2.38}$$

which we write concisely as $S_p = O_p + N_p$. Similarly for scene point $Q$ we have $S_q = O_q + N_q$. The overlapped terms, $O_p$ and $O_q$, have the same incident

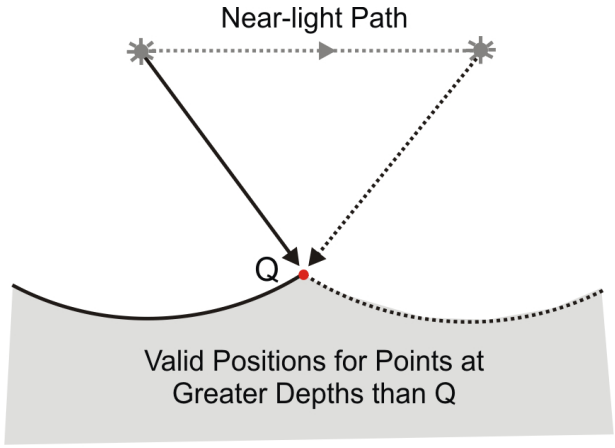



Figure 2.23: **Assumption for obtaining valid depth orderings:** A light source moving in a line illuminates a scene with homogeneous BRDF. Our depth ordering method works for scenes where any point at a greater depth than Q must be further away from the light source at every time instance. This restricts us to a non-planar region around Q as shown.

angles. Since $P$ is further away than $Q$, the inverse-square fall-off from the light source ensures that $O_p < O_q$. Therefore, $S_p < S_q$ if $N_p < N_q$. Consider a pair of the summation terms from $N_p$ and $N_q$. If the inequality holds for each of these pairs, then,

$$\frac{F_p(d)}{(R_p(d))^2} \leq \frac{F_q(d)}{(R_q(d))^2} \tag{2.39}$$

### Depth Ordering for Lambertian Scenes

When does Equation 2.39 hold? We answer this question by first investigating depth ordering for lambertian scenes. We divide the problem into two cases, based on whether the local geometry between the scene points $P$ and $Q$ is convex or concave.

**Convex neighborhood:** If the normals at $P$ and $Q$ are given by $\mathbf{n_p}$ and $\mathbf{n_q}$ and if the 3D position of the light source is $\mathbf{s}(d)$ then we can rewrite Equation 2.39 as,

$$\frac{\mathbf{n_p}.(\mathbf{s}(d) - \mathbf{P})}{\mathbf{n_q}.(\mathbf{s}(d) - \mathbf{Q})} \leq \left(\frac{R_p(d)}{R_q(d)}\right)^3 \tag{2.40}$$

For a convex object, a scene point's foreshortening to the light source is higher than one farther away. Therefore the LHS of Equation 2.40 is always less than one, and our method always works for pairs in a convex lambertian neighborhood. In Figure 2.24 we show simulations of depth orderings for convex objects of different curvatures demonstrating above 97% accuracy with small errors occur due to violations of the assumption in Figure 2.23. We also present

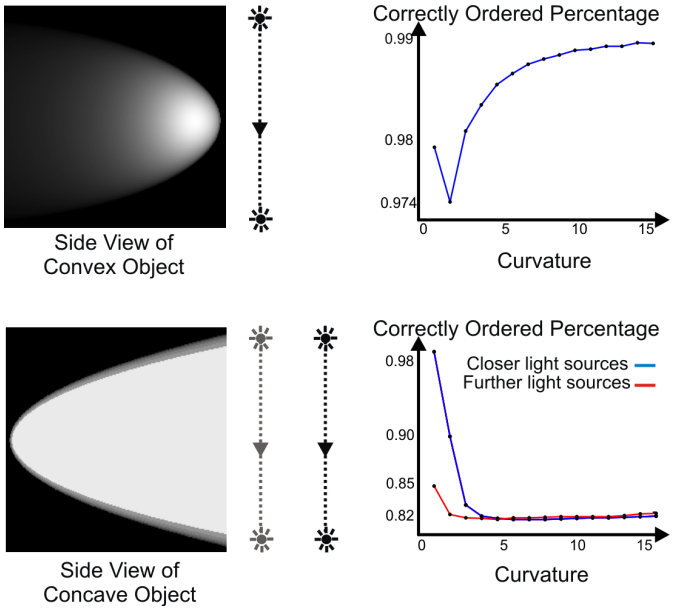



Figure 2.24: **Simulations of convex and concave Lambertian objects:** Our depth ordering method always works for convex lambertian objects. We show results with very low errors on renderings of a convex shape whose curvature we gradually increase. We also render concave objects at different curvatures. Although the depth orderings are much worse in this case, they stabilize and do not degrade with time. In addition, bringing the light source closer creates better orderings for shallow concave objects.

results on non-lambertian convex objects, such as corduroy and dull plastic (Figure 2.26).

**Concave neighborhood (shadows and interreflections):** The foreshortening of $P$ and $Q$ are now opposite to the convex case, and the LHS of Equation 2.40 is always greater than 1. In fact, the further the two points are, the larger the left hand side of the equation will be. Fortunately, our method still performs well thanks to the effect of shadows. We rewrite Equation 2.40 by adding a visibility term, $V(d)$,

$$V(d)\,\frac{\mathbf{n_p}.(\mathbf{s}(d)-\mathbf{P})}{\mathbf{n_q}.(\mathbf{s}(d)-\mathbf{Q})} \leq \left(\frac{R_p(d)}{R_q(d)}\right)^3 \tag{2.41}$$

Since $P$ is further away from the light source than $Q$, it is more often in shadow. Whenever this happens, $V(d) = 0$ and Equation 2.41 is true. Concave depth ordering has more errors than the convex case (Figure 2.24), the results are still reasonable (above 82%). In addition, the closer we bring the light source to the scene, the better the results get since the RHS of Equation 2.41 increases.

### Experimental Results

In Figure 2.25, we have rendered three scenes (buddha, bunny and dragon), using ray tracing ([75]). As before, we varied the roughness parameters for the

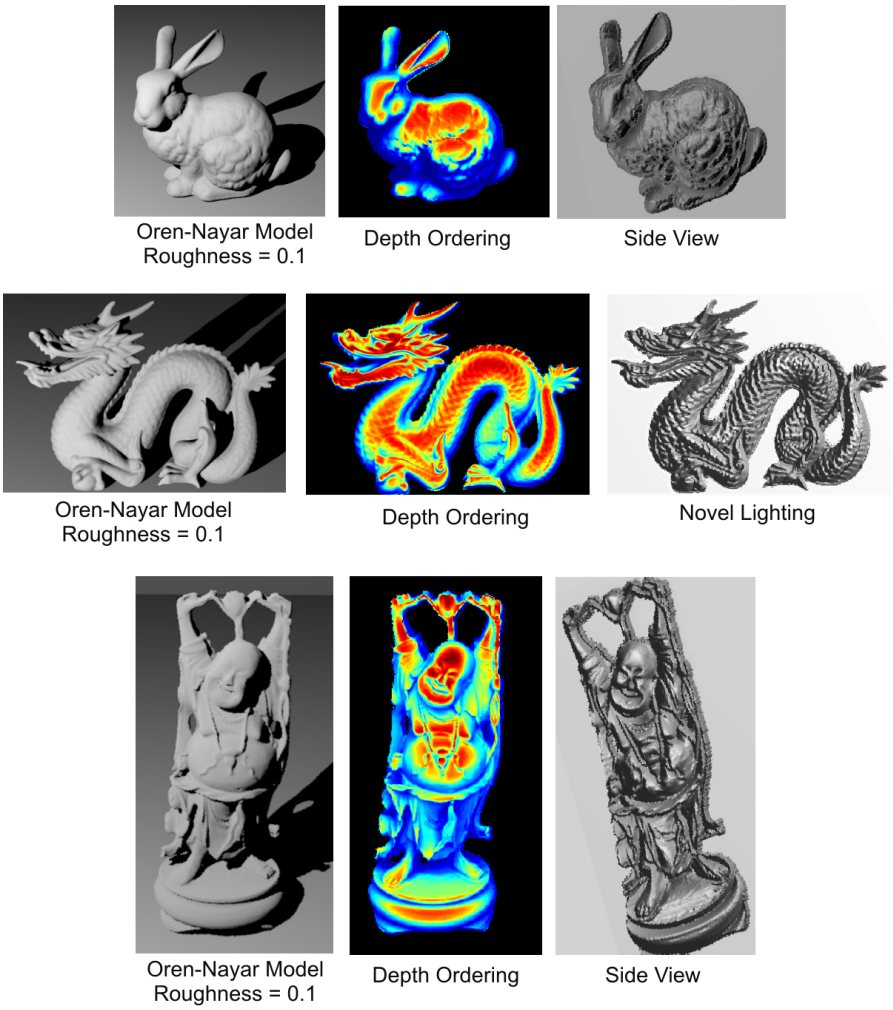



Figure 2.25: **Depth ordering for rendered scenes:** We used the PBRT software ([75]) to render three scenes using both the Lambertian + Torrance-Sparrow and Oren-Nayar models. We covered the space of roughness parameters for each of these models. Using ground truth depths for these scenes, we measured the percentage of correctly ordered scene points and we plot the results above. These results provide empirical evidence supporting the use of our method with diffuse non-lambertian BRDFs.

Oren-Nayar model (facet angle from $0$ to $\frac{\pi}{2}$) and the Lambertian + Torrance-Sparrow model ($\sigma$ of facet distribution from $0$ to $50$), getting $10$ sets of images, each with $170$ different light source locations. We compare the depth ordering with ground truth depth and show accuracy above $85\%$ for the Oren-Nayar model and above $75\%$ for the Lambertian + Torrance-Sparrow model. In Figure 2.27 we display the ordering for a polyresin statue containing shadows and interreflections as a "pseudo depth" in Maya. We have shown different views as well as its appearance under novel lighting. Finally, an advantage of our method is that these depth orderings are with respect to the light source and not the camera. We obtain orderings from different planes, without camera motion and avoiding the difficult problem of scene point correspondence. In Figure 2.28 we show ordering results for a wooden chair, for three planes.

### Mirror Symmetries in Homogeneous Scenes

In this section, we will investigate a way of finding mirror symmetry pairs for scenes with homogenous BRDFs. Although our method is simple, it is useful for objects that have the mirror symmetry in their shapes and produces very sparse pairs otherwise.

**Mirror Symmetry Pairs in Lambertian Scenes:** Consider a homogeneous Lambertian scene containing a mirror symmetry, viewed by an orthographic camera. Let the scene be illuminated by a point light moving in a plane parallel to the viewing direction. Reflecting the scene across such a plane does not change incident elevation angles and light source distances at scene points. Therefore, all the reflected pairs will have identical appearance over time and can be easily located. We show the result of matching identical profiles for a homogeneous Lambertian pot at the top of Figure 2.29. The pot has a reflection symmetry across a vertical plane passing through its center and perpendicular to the image plane. We wave a point light source in this plane and mark the matched scene points with the same color. Note that if the light source was moved in a different plane, different symmetry pairs would be obtained. However, these would most likely be sparse, since the reflection plane would not coincide with the mirror symmetry of the object.

**Shadows and Inter-reflections:** A scene with mirror symmetry has the same geometry across the plane of reflection. Therefore, geometric appearance effects such as shadows and inter-reflections will be identical between symmetry pairs, creating identical intensity profiles.

**Isotropic BRDFs:** In homogeneous non-lambertian scenes the effect of the incident azimuth angles must be considered. In general these will be different for the scene and its reflection. However, the absolute difference between the azimuth angles are the same, and therefore our method works for isotropic BRDFs. We find symmetry pairs by matching identical intensities for an office desk at the bottom of Figure 2.29, even though this scene contains non-lambertian materials such as metal and plastic.

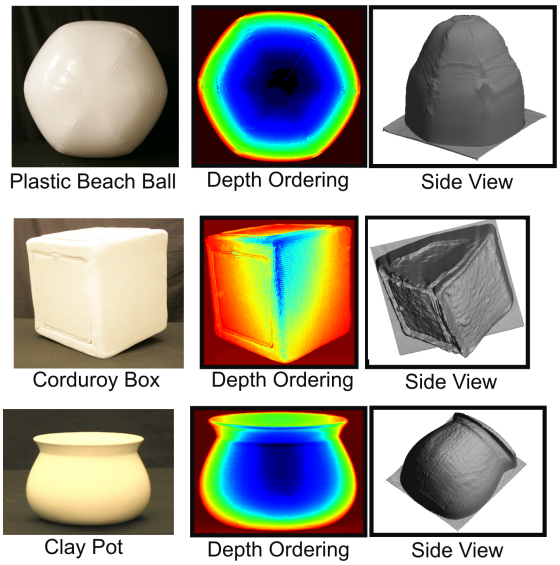



Figure 2.26: **Depth ordering results on real convex scenes:** We acquired images by waving a near point light in a plane. For each scenes we display the depth ordering obtained as a "pseudo depth", by plotting the ordering in 3D. The objects presented are made up of a variety of materials such as plastic and corduroy.

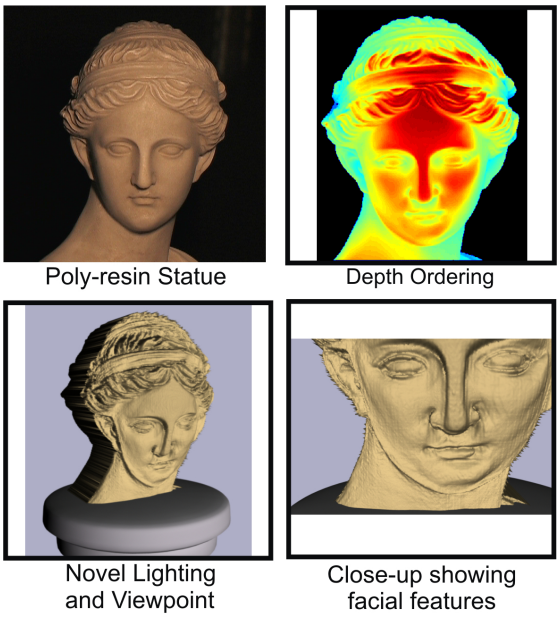



Figure 2.27: **Depth ordering results for a statue:** In the top row we show a polyresin statue and a 2D depth ordering image obtained from an uncalibrated near light source moving in a plane. We visualize the ordering as a "pseudo depth" in the bottom row, applying novel lighting, viewpoint and BRDF. Although this object has complex effects such as interreflections and self-shadowing, our depth ordering is accurate, as we show in the close-up. Please see [47] for many other view points.

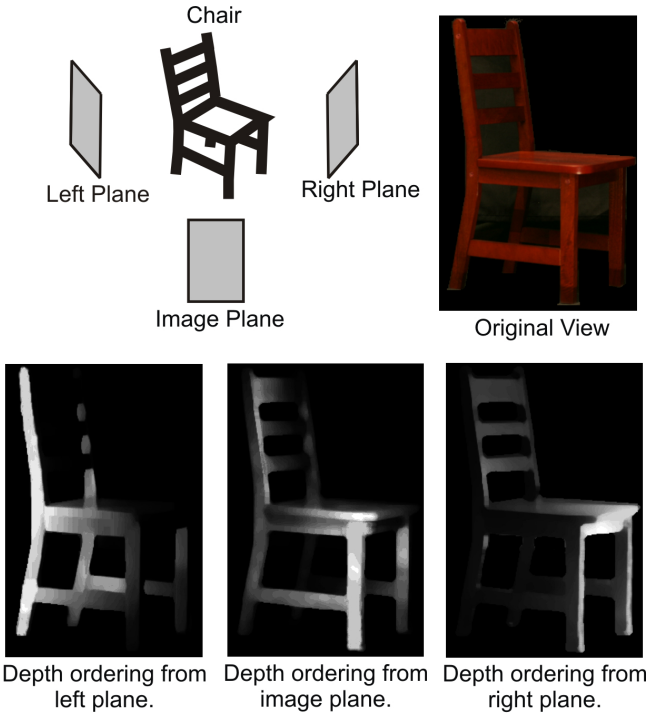



Figure 2.28: **Depth ordering from different viewpoints:** Here we show depth ordering of a wooden chair from the viewpoints of three different planes. Brighter scene points are closer than farther ones. The second depth ordering is from the camera point of view, and is similar to conventional depth results obtained from stereo algorithms. We also get the orderings from the left and right planes without moving the camera. This allows us to obtain novel visualizations of the scene, while avoiding the problem of scene point correspondence.

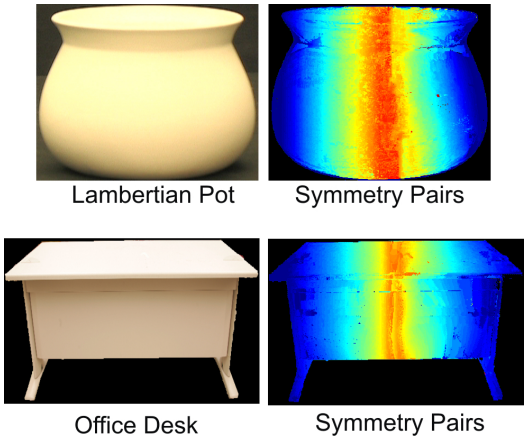



Figure 2.29: **Symmetry pairs in real scenes:** We show two real scenes, a lambertian pot and a office desk. A near point light source was hand-waved approximately in the plane of mirror reflection that is perpendicular to the image plane. We match identical profiles in the scene and mark them with the same color, giving us symmetric pairs in the scene.

The appearance profile at each point in a scene depends both on local properties, such as the BRDF and geometry, and on a variety of global factors such as cast shadows, inter-reflections and sub-surface scattering. In the last two sections, we show how the maxima and minima of the profile contain geometric cues and act as invariants to other effects. In this section, we focus on modeling the appearance profile, rendering new images.

We propose a general representation for the appearance profile. We derive an analytic approximation of the appearance profile using Taylor series, for relighting from a fixed viewpoint. This Taylor polynomial representation is non-linear and data-dependent which results in a compact low-dimensional representation of appearance with fewer terms than other methods like PCA and spherical harmonics. In addition, Taylor representations of appearance profiles can exploit localized appearance variation both in the angular and spatial domain by changing the center of expansion. Since we do not need regular sampling to compute a Taylor polynomial representation, we are able to collect input images in an uncalibrated setup, by simply waving a light-source around a scene. We demonstrate our method by relighting textures and scenes with several different appearance effects including sub-surface scattering.

## 2.3 Modeling the Appearance Profile

Radiometric reflectance models the interaction of light with scenes. Some models, such as the BRDF, explain local effects at a particular scene point and do not take into account interactions between different parts of the scene. Others such as the BTF and the BSSRDF are global models that explain the scene as a whole. Each of these explicitly model different physical properties of the scene including material roughness, occlusion and scattering. As a result, the more phenomena a model tries to describe, the more complex and higher-dimensional it becomes. For example, the BSSRDF, which models sub-surface scattering, has eight parameters which makes it difficult to fit the model to data ([40]).

Instead, we model general appearance variation, in particular under varying illumination, simply as a profile (curve) of viewing and lighting directions. The appearance profiles implicity model all the effects present in the scene, such as specularities, shadows, inter-reflections and sub-surface scattering. As we will show, the appearance profile is at most four dimensions. We will present two kinds of appearance profiles for relighting scenes and will assume fixed viewing direction. If we denote the measured intensity at a scene point by $I$ and the illumination direction in polar coordinates by ($\theta_s$, $\phi_s$), the appearance profile $A$ is written as,

$$I = A(\theta_s, \phi_s). \tag{2.42}$$

This appearance profile is defined at each scene point and it maps illumination directions to a single intensity. We will show relighting of fixed viewpoint BSSRDF and fixed view point BTF scenes using the above appearance profile. This analytic form of the appearance profile is similar to the concept of 'apparent BRDF' ([48]). Note, however, that unlike apparent BRDF which only

| Light Transport Function | Appearance profile |
|---|---|
| 2D Fixed Viewpoint BRDF | 2D $A(\theta_s, \phi_s)$ |
| 3D Isotropic BRDF | 3D $A(\theta_v, \theta_s, (\phi_v - \phi_s))$ |
| 4D Anisotropic BRDF | 4D $A(\theta_v, \phi_v, \theta_s, \phi_s)$ |
| 4D Fixed Viewpoint BTF$(x, y, \theta_s, \phi_s)$ | 2D $A(\theta_s, \phi_s)$ |
| 6D BTF$(x, y, \theta_v, \phi_v, \theta_s, \phi_s)$ | 4D $A(\theta_v, \theta_s, \phi_v, \phi_s)$ |
| 6D Fixed Viewpoint BSSRDF $(u, v, x, y, \theta_s, \phi_s)$ | 2D $A(\theta_s, \phi_s)$ |
| 6D Fixed View BSSRDF Zoom $(u, v, x, y, \theta_s, \phi_s)$ | 4D $A(x, y, \theta_s, \phi_s)$ |
| 8D BSSRDF$(u, v, x, y, \theta_v, \phi_v, \theta_s, \phi_s)$ | 4D $A(\theta_v, \theta_s, \phi_v, \phi_s)$ |

Table 2.1: **Appearance Functions:** In this table we show some common reflectance profiles in increasing order of dimensionality. Note that the corresponding appearance profiles all have dimensions lesser than or equal to 4D. We focus on those scenes with fixed viewpoints, such as fixed view BSSRDF and fixed view BTF and show that the appearance profile is able perform accurate relighting of these scenes. We also show results with fixed view BSSRDF textures that have smooth variation in space, virtually 'zooming' into the texture.

adds visibility to the BRDF, the appearance profile can implicitly model a much wider variety of effects including sub-surface scattering and spatial variation (See Table 1). It is able to do this since every scene point has its own specifically tailored appearance profile that explains the observed intensities. The power of the appearance profile comes from this specificity. The second appearance profile that we will present allows us to model scenes in which the appearance varies smoothly around location $(x, y)$ as

$$I = A(x, y, \theta_s, \phi_s). \tag{2.43}$$

In general, appearance profiles may not be continuous because of non-smooth effects such as sharp shadows and specularities. Consider scenes containing only smooth effects such as soft shadows and sub-surface scattering. The domain of such scenes is broad, and includes important areas for graphics such as homogeneous BSSRDF scenes and textures. In the rest of this section, we will show how to use Taylor polynomials to accurately model such appearance profiles. The major advantage of appearance profiles is that its dimensionality is at most 4. This low-dimensionality makes subsequent parameter estimation more feasible. Table 2.3 shows some common light transport functions of higher order that can be represented by profiles of equal or lower dimensionality.

### 2.3.1 Taylor Series of Appearance Profiles

We approximate the appearance profile using the Taylor series polynomial. The Taylor polynomial has many advantages as a tool for understanding appearance profiles. It is a single model for all the measured intensities, as opposed to local models such as splines. The coefficients of the polynomial are powers of the viewing and illumination directions, and therefore rendering novel images simply means creating new coefficients. Polynomial fitting is robust to noise in a least squares sense. In addition, polynomial calculations are fast in hardware

and rendering is easy. Finally, operations such as differentiation and integration that are hard on analytic reflectance profiles can be performed easily on their corresponding polynomial approximations.

Consider the appearance profiles $A$ from Equations 2.42 and 2.43. Since the Taylor expansion of this profile is multivariate, let the centers of expansion be $(\theta_{s_c}, \phi_{s_c})$ and $(x_c, y_c, \theta_{s_c}, \phi_{s_c})$. Then, using the power notation for derivatives of [108], their Taylor series approximations are

$$A(\theta_s, \phi_s) = \sum_{j=0}^{p} \left\{ \frac{1}{j!} \left[ (\theta_s - \theta_{s_c}) \frac{\delta}{\delta \theta_s} + (\phi_s - \phi_{s_c}) \frac{\delta}{\delta \phi_s} \right]^j A(\theta_{s_c}, \phi_{s_c}) \right\}, \tag{2.44}$$

$$\begin{aligned} A(x, y, \theta_s, \phi_s) = \sum_{j=0}^{p} \Bigg\{ \frac{1}{j!} \Bigg[ & (x - x_c) \frac{\delta}{\delta x} + (y_v - y_c) \frac{\delta}{\delta y} \\ & + (\theta_s - \theta_{s_c}) \frac{\delta}{\delta \theta_s} + (\phi_s - \phi_{s_c}) \frac{\delta}{\delta \phi_s} \Bigg]^j A(x_c, y_c, \theta_{s_c}, \phi_{s_c}) \Bigg\}. \end{aligned} \tag{2.45}$$

Such an expansion of $A$ produces an order $p$ Taylor series. Each additional $j^{th}$ order produces derivatives of $A$ up to order $j$. Re-arranging and expanding the above equations so that all terms with the same derivative order of $A$ are grouped together allows us to write the Taylor polynomials in the following notationally simplified form:

$$A = D_0 + \sum_{i}^{n} C_i . D_i \, . \tag{2.46}$$

where the $C_i$s are the power terms and the $D_i$s are the partial derivatives of $A$ evaluated at the center of expansion. For example, in Equation 2.44,

$$C_i = K . (\theta_s - \theta_{s_c})^a . (\phi_s - \phi_{s_c})^b, \;\; 0 \leq a, b \leq p, \;\; K \in \mathcal{R} \tag{2.47}$$

$$D_i = \frac{\delta^e}{\delta^e \theta_s} . \frac{\delta^f}{\delta^f \phi_s} A(\theta_{s_c}, \phi_{s_c}), \;\; 0 \leq e, f \leq p. \tag{2.48}$$

For a profile of $m$ variables, an order $p$ Taylor polynomial will have $n = \binom{m+p}{m}$ terms. For instance, the fourth order Taylor polynomial of the 4D appearance profile $A$ will have 70 terms. Given $k$ samples of $A$, where $k > n$, at known illumination and viewing directions (known $C_i$s), estimating the $D_i$s becomes simply solving a linear system of equations,

$$\begin{pmatrix} 1 \; C_{11} \; C_{12} \; ... \; C_{1n} \\ . \\ . \\ . \\ 1 \; C_{k1} \; C_{k2} \; ... \; C_{kn} \end{pmatrix} \begin{pmatrix} D_0 \\ . \\ . \\ . \\ D_n \end{pmatrix} = \begin{pmatrix} A_1 \\ . \\ . \\ . \\ A_k \end{pmatrix} \tag{2.49}$$

Two properties are essential to accurately approximate a profile with a Taylor polynomial: boundedness and continuity (Pg 94. [20]). Boundedness is important since by Taylor's theorem, the residual error, $R_n$, of a $n$ term Taylor

series expansion of a bounded profile goes to zero as $n$ goes to infinity. Appearance profiles are all bounded, since the radiance from a scene onto the image plane can never be infinity (conservation of energy), and therefore is always smaller than some large positive number. Not all appearance profiles satisfy the second, harder condition of continuity, for instance due to sharp shadows and point specularities. Therefore we only apply the Taylor polynomial representation to smooth appearance profiles.

What are the advantages of a polynomial model for appearance? Consider scenes containing effects such as soft cast shadows, glossy specularities and sub-surface scattering. The variation in appearance at the scene point is smooth, which means that higher order derivatives of the appearance profile vanish. Therefore, a finite Taylor polynomial exists that accurately models the appearance at each scene point ([20]). In addition, varying the center of expansions allows localized approximations of the appearance profile. Therefore, Taylor polynomials can result in compact representations for a variety of scenes, which we will now first verify with analytic models.

For example, consider the appearance profile of isolated convex objects whose surface reflectance can be modeled with isotropic BRDFs. Taylor polynomial approximations of the appearance profile of such objects can be analytically computed using specific BRDF models. Figure 2.30 shows rendered spheres of three common BRDFs, Phong, Torrance-Sparrow and Oren-Nayar, along with their Taylor approximations. By increasing the complexity of the models, we show in the middle row of Figure 2.30, that the number of terms needed in the Taylor polynomial grows linearly. Therefore, we can safely assume that the number of terms required will not be infinity for other BRDF models commonly used in vision and graphics. From the bottom row of Figure 2.30 it is clear that the normal direction is a good candidate for the center of expansion of the Taylor polynomial for general appearance profiles.

We just showed that Taylor polynomial representation works well for simple BRDFs. Real scenes, however, contain various smoothly varying appearance effects that are still challenging to model. How well do Taylor polynomials capture smooth appearance effects? To answer this question we isolated these effects in experiments summarized in Figure 2.31. We have shown images from four different videos, each of which captures a specific appearance effect under varying illumination. In particular we have investigated soft shadows, glossy specularities, sub-surface scattering and inter-reflections. These effects are smooth effects since the intensities in the images vary smoothly, without the kind of drastic change in appearance displayed by scenes with point specularities or sharp shadows. For the reported experiments, we used Taylor polynomials of orders less than 10. These results clearly demonstrate that our Taylor polynomial representation can be applied widely.

Consider an $n$ term Taylor polynomial that accurately represents an appearance profile for each scene point. Let us construct a set of basis images whose $i^{th}$ basis image contains all the $D_i$ terms for every scene point. From the definition of the Taylor series, the derivatives of the appearance profile evaluated at the

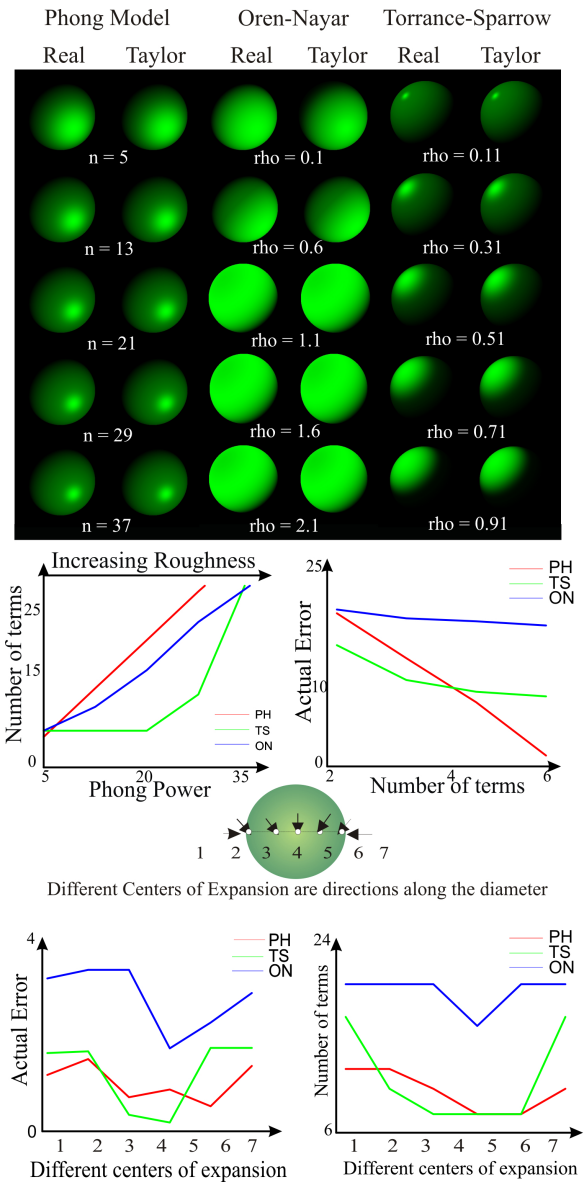



Figure 2.30: **Taylor polynomials of common BRDFs**: In the first two columns the Phong model's metallic parameter, $n$, is varied from top to bottom. The center two columns show Oren-Nayar model with different illumination directions and roughness parameter $\rho$. Finally, in the last two columns we render spheres using the Torrance-Sparrow model and compare with the images rendered using the Taylor polynomial representation. The right-top two graphs explore the variation of model parameter values corresponding to increased roughness and increased metallicity compared to the number of terms needed for 1% error between the original analytic form and its Taylor polynomial representation. The results show that the number of terms needed for a good Taylor approximation does not increase exponentially with increase in model complexity. In the right-bottom two graphs above, we explore how the error between the analytic model and center of expansion in the Taylor Series varies. In both cases the error is least around the surface normal direction of the scene point.

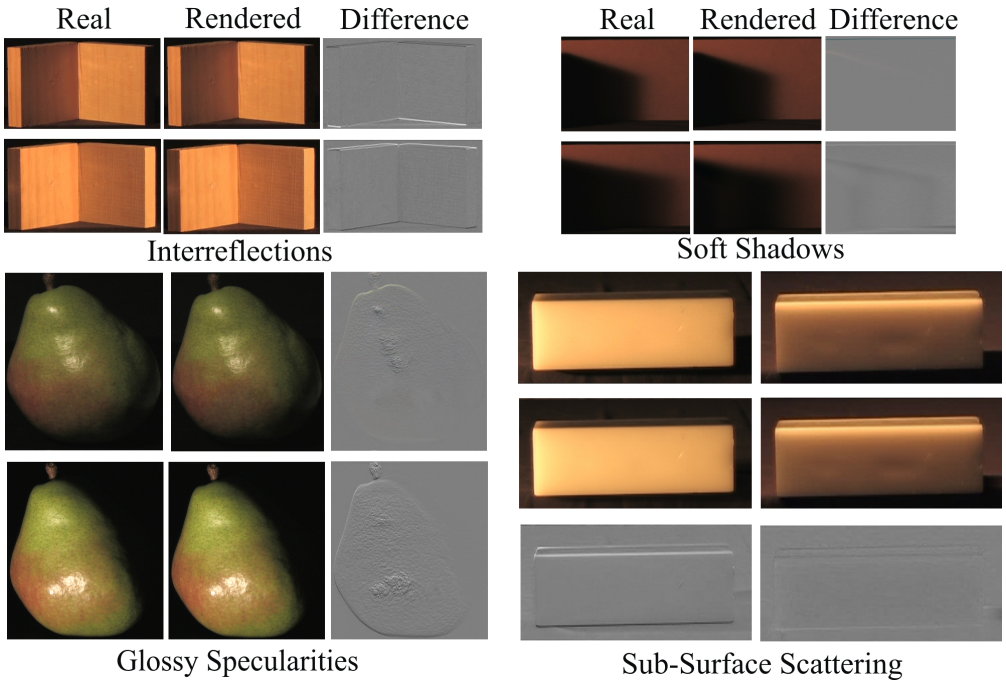



Figure 2.31: **Taylor representations of isolated effects**: In each of the four rows, we show frames that were generated by Taylor polynomial representations of smooth appearance effects such as inter-reflections, sub-surface scattering, soft shadows and glossy specularities. The results show that the Taylor polynomial representation can capture these global effects and is applicable to general scenes containing all these effects.

center of expansion form the basis for the scene appearance space. We propose using these non-linear data-dependent bases for representing the appearance profile instead of other general linear bases such as PCA and spherical harmonics. Due to its non-linearity, these derivative bases can better capture the overall appearance variation in lower order terms. Unlike PCA, the coefficients of this basis are the powers of the viewing and illumination directions, and therefore have physical meanings. This provides us intuitive means for rendering image from novel viewpoints and under novel illumination conditions. Compared to spherical harmonics, our basis is data-dependent, since the appearance profile derivatives are specific to the given data. This allows us to have comparable accuracy to spherical harmonics with far fewer terms. We show these comparisons in Figure 2.32.

### 2.3.2 Taylor Series of Texture Appearance

Textures can be described by a 6D profile known as the Bidirectional Texture profile (BTF) ([12]). Many textures are roughly planar and tend to have smooth appearance variation. Therefore local approximation methods such as the 2D appearance profile described in Equation 2.42 work extremely well. In this section we describe different applications of our Taylor polynomial representation

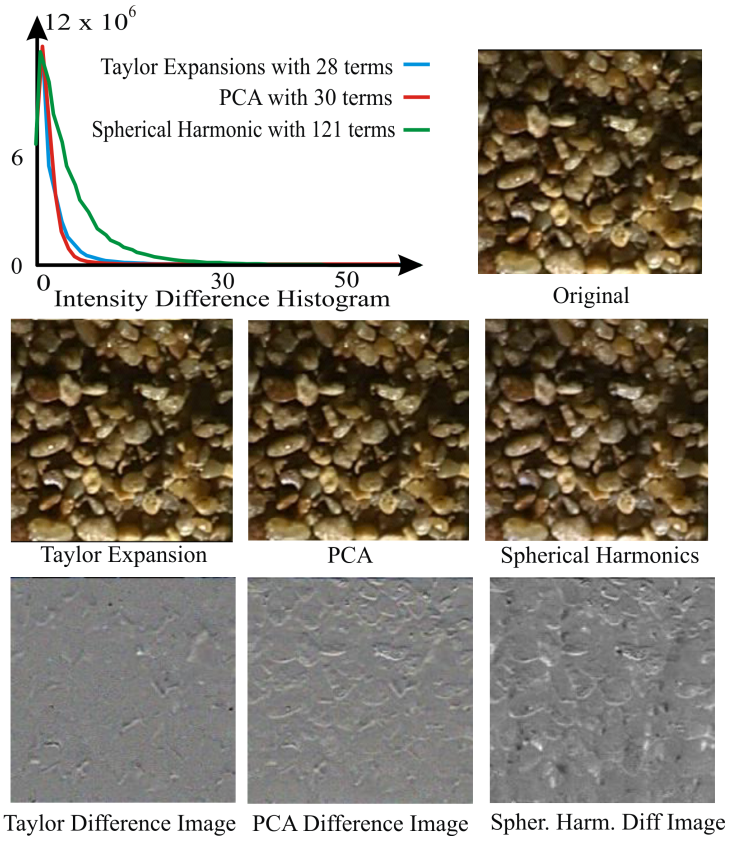



Figure 2.32: **Comparing taylor polynomial accuracy with other techniques**: We compare the effectiveness of the Taylor polynomial model representation with two other techniques, PCA and spherical harmonics. We use the gravel texture example and selected a light source position between grazing and normal directions, since such an image displays both significant shadows and specularities. Note that our method performs better than spherical harmonics with far fewer terms, and compares slightly better than PCA.

for texture relighting, synthesis, measurement and rendering. We used textures measured by Koudelka et al. ([48]) in these experiments.

**Texture Relighting:** Consider capturing the appearance variation of a texture sample under varying illumination with an orthographic camera from a fixed viewpoint. In this case the BTF reduces to a 4D profile and the appearance profile at every location becomes a 2D profile of lighting directions. In Figure 2.33 we present three different textures illuminated from various lighting directions. The synthetic images are rendered using illumination directions that are not used to calculate the coefficients of the Taylor polynomial. The error between the synthetic image and the actual image is shown. For all of these examples, 98% of the pixels were within 20 gray scale values of the original video. We are able to model high frequency effects such as shadows and glossy specularities that give rise to dramatically different texture appearances.

**Appearance Consistent Texture Synthesis:** Consider the derivative basis images explained in the previous section. Each basis is an array of derivatives

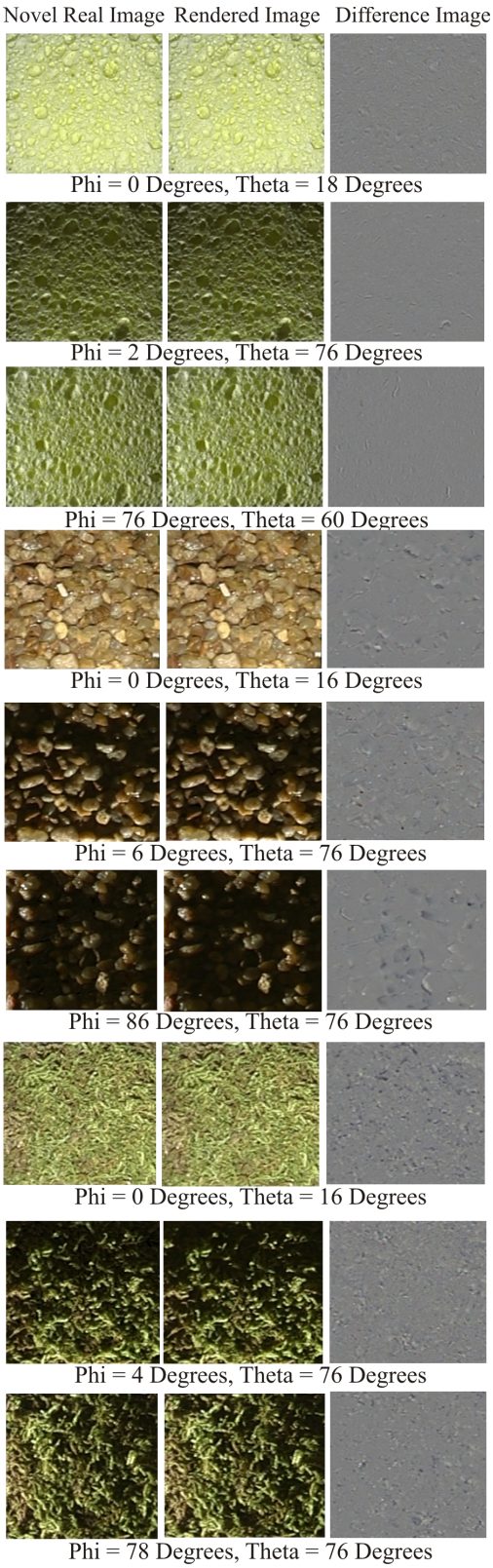



Figure 2.33: **Texture relighting**: Some examples of relighted textures from a database of real textures. ([48]) The second and third rows represent two different grazing angles. Note the drastic change in appearance in the textures. 98% of the pixels in the rendered images were within 20 gray scale values of the original frames.

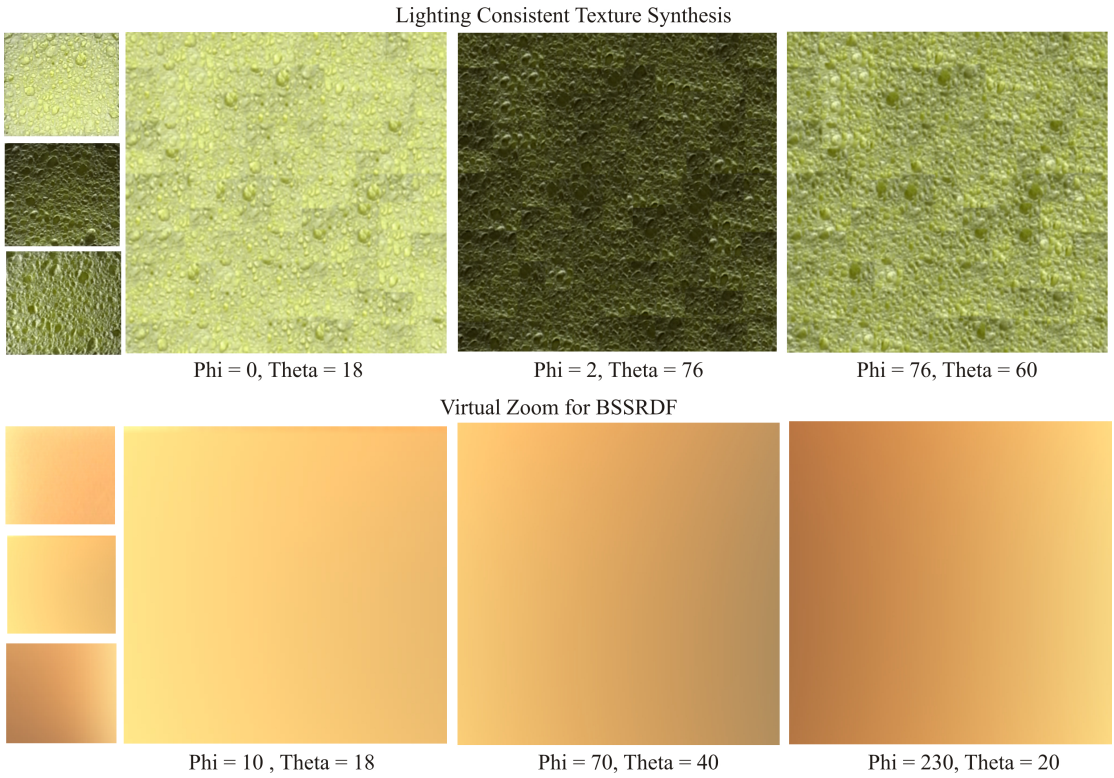



Figure 2.34: **Texture synthesis**: By applying the texture synthesis work of [19] to the derivatives in the Taylor Polynomial, we are able to synthesize textures under different illuminations. In the top row, we show original 196x196 size images of a sponge texture, and the 453x453 size synthesized textures. We increased the frame rate of the original video from 15fps to 30fps for the synthesized texture. Furthermore, we can perform a 4D Taylor expansion using both illumination directions and spatial locations as variables. This allows to us effectively zoom in to textures that exhibit spatially smooth appearance variation such as sub-surface scattering.

of a specific order. If we apply texture synthesis algorithms on these elements, we can synthesize a novel set of basis images. Therefore, instead of creating one new synthesized texture at a particular illumination direction, we can create a whole space of textures whose incident lighting can be changed by varying the coefficients of the Taylor polynomials. We use the data-dependent texture synthesis algorithm of Efros and Leung ([19]). The relighting process now creates synthesized images of a user-defined size whose appearance under varying illumination is consistent with the source texture. On the left of Figure 2.34 we show a texture of sponge under different light source directions and also its corresponding larger, synthesized texture. Again, note that these images are not synthesized individually for different lighting conditions but are relighted after a single pass of texture synthesis.

**Virtual Zoom for BSSRDF:** In the case of BSSRDF textures, the appearance of the object changes smoothly across the spatial domain (the surface) as well

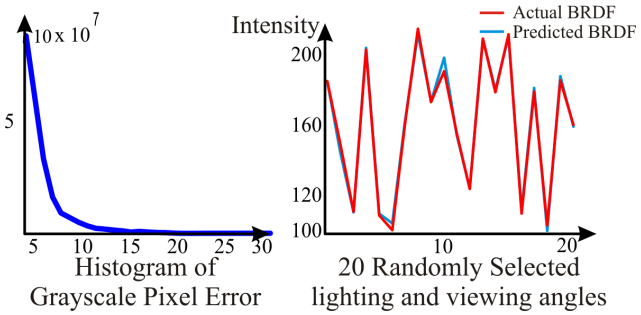



Figure 2.35: **Modeling BRDF**: Averaging the intensities across a BTF gives us the BRDF of that material. In this example we sample the BRDF of the 'sponge' texture from [48], and model it using a 4D Taylor polynomial of 35 terms. On the left, we show the intensity error histogram between the predicted values of the BRDF and the ground truth, showing that 98.4% of the pixels are within 20 gray scale values of ground truth. The figure on the right shows 20 randomly picked BRDF values in blue, plotted with their corresponding ground truth in red. These results show that the Taylor polynomial representation also accurately models BRDFs.

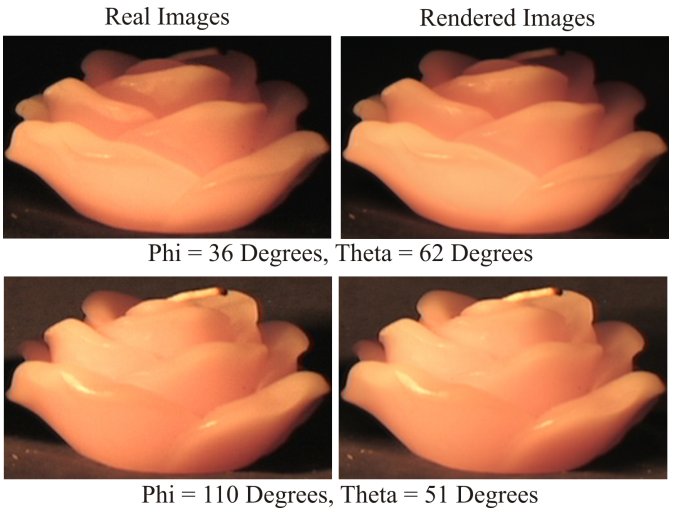



Figure 2.36: **Relighting homogeneous BSSRDF scenes**: In this example, a wax rose with BSSRDF effects is illuminated by a distance light source. Taylor polynomial representations of each scene point allows us to recreate the scene accurately, as shown by the center image. Sub-surface scattering is a smoothing process, and therefore any other effect in a homogeneous BSSRDF scene, such as shadows, are much easier to model with Taylor polynomials.

as across the angular domain (the incident light source directions). We exploit this smoothness by using the appearance profile described in Equation 2.43 to model the variation in appearance across location and illumination. Therefore we can now apply a 4D Taylor expansion of the BSSRDF texture to virtually zoom into the texture and expand the effects both in space and across light source directions. This allows us to create synthetically zoomed-in images as shown in Figure 2.34.

**BRDF Modeling:** Here we introduce a novel and simple method to model the BRDF of textures. The key idea is that the Taylor model does not need uniformly spaced measurements to calculate the coefficients. We use the texture database of Koudelka et al. ([48]), but we sample the database non-uniformly. We then model the BRDF as a 4D Taylor polynomial, and use it to predict novel values of the BRDF. The results show that the BRDF can be interpolated accurately at arbitrary viewing and illumination directions that are along the path of the hand-waved camera and light source.

### 2.3.3 Relighting Homogeneous BSSRDF Scenes

The Bi-directional Surface Scattering Distribution profile (BSSRDF) is an 8D profile that takes into account sub-surface scattering in the material ([67]). A variety of substances, such as marble, soap, wax, milk, and skin exhibit sub-surface scattering effects. Modeling these has proved hard in vision and graphics because of the high dimensionality of the profile. Measuring the BSSRDF accurately for rendering has required significant effort([40],[25]). From the previous sections, we have noted that fitting Taylor polynomials becomes more accurate in scenes with smoothly varying appearance. Translucent objects, in particular, can be modeled well, since the sub-surface scattering blurs sharp appearance effects. For example, highlights become glossy, attached shadows are smeared and sharp cast shadows are softened. In this paper, we only consider homogeneous scenes, making the problem easier since the distribution of scattering only depends on scene geometry. In Figure 2.36, we show rendered images of a wax object using the Taylor polynomial representation method. Although the BSSRDF is high-dimensional, we are able to accurately interpolate along the path of the light source to synthesize the appearance with finer angular resolution for the incident light source directions.

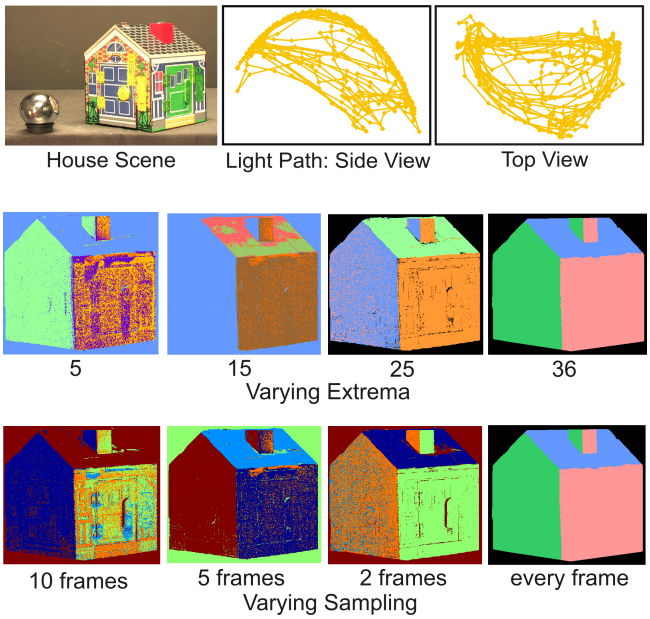



Figure 2.37: **Varying some aspects of our algorithm:** Here we show the variation in the quality of results as we change different parameters. The top row illustrates the scene, as well as two views of the light source path on the unit sphere. In the second row, the length of the video is changed to include progressively increasing numbers of extrema. As the numbers of extrema increase the resulting clusters become more iso-normally consistent. In the last row, we show how the results change as the sampling of the profiles changes. Note that iso-normal clusters occur even while sampling every fifth frame.

## 2.4 Chapter Summary and Discussion

In this chapter, we showed that, for a static scene, the smooth, continuous motion of a light-source produces a sequence of intensities at each pixel, which we termed an appearance profile. The locations *in time* of the intensity extrema (brightness maxima and minima) of this appearance profile can provide geometric cues, even when the material properties, lighting positions and camera calibration are all unknown. We also demonstrated a Taylor series model for the appearance profile for rendering.

**Technical summary:** In Section 2.1, we modeled the appearance profile intensities using the linearly separable BRDF model, which can represent a large group of materials. Setting the derivative of this model to zero determines the locations (in time) of brightness maxima and minima. For smoothly varying distant lighting, we derived a relationship between the intensity extrema locations and a scene point's surface normal, enabling iso-normal clustering of pixels. We also described an extension in Section 2.2 for point light sources with inverse-square intensity fall-off that move in a line. In this case, the intensity maxima locations obtain intersections of planes with the scene. Finally, as described in Section 2.3, the profile's Taylor series expansion allows interpolation and rendering of textures and other scenes.

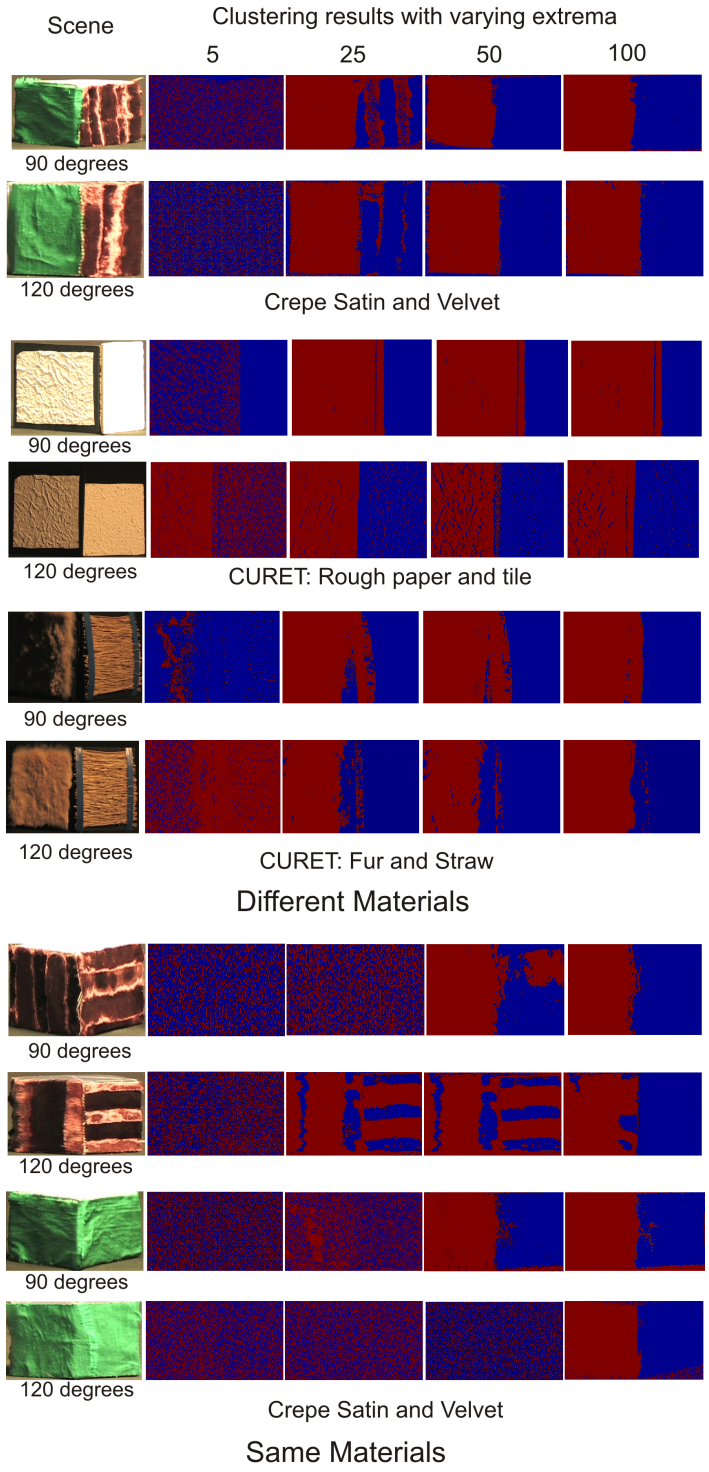



Figure 2.38: **Material vs. geometry variation:** We analyze how the clustering result changes as the number of shared extrema are increased, in a scene with two planes. In all cases, increasing the shared extrema creates better clustering. In the first six rows, drastically different materials allow easy disambiguation of the two planes. Fewer extrema are needed to disambiguate the two planes, irrespective of the angle between the planes. In contrast, in the last four rows, more extrema are needed as the angles between the plane increases since the materials are identical.

**Input and Output:** All these methods assume a static scene with a smoothly moving light source. For iso-normal clustering and depth cues, no other information about the scene is required. Taylor series modeling requires the light source location at each frame. The output for iso-normal clustering in Section 2.1 are groups of pixels with the same surface normal. The depth cue outputs in Section 2.2 are clusters of pixels that share a depth-related property (such as ordering in terms of perpendicular distance from a plane). The output for Taylor series modeling are the coefficients of the Taylor expansion.
**Applications:** The ease of use and accuracy of the clustering algorithm has motivated applications in vision and graphics. Sato et al. ([85]) obtain surface normal information directly from images taken under varying illumination, utilizing our extrema feature. A real-world application is for time-lapse video of outdoor scenes, where the sun moves smoothly: we show clustering for an outdoor scene in the WILD database ([63]). In addition, Sunkavalli et al.([97]) extends this idea by exploiting the relationship between profile extrema and light-source position to relight web-cam data.
**Alternative methods:** To our knowledge, no known single-view method exists to recover geometric properties of scenes with general, unknown BRDFs and unknown illumination. Healey's method ([31]) and Hertzmann and Seitz ([33]) are closest, but while the former deals with only Lambertian scenes, the later requires example objects. Many methods, could be applied to model the appearance profile, and our selection of Taylor series was to have control over the point of expansion, which could increase the quality of the rendered images.
**Implementation Issues and Discussion:** The clustering algorithm based on brightness extrema produces results for many real-world surfaces. Detecting the extrema (maxima and minima) locations in time does not require storing all the data and can be obtained instead by using a sliding window. Furthermore data collection in these methods is easy, requiring a user to just hand-wave the source. However analytically proving that our method will work for scenes with arbitrary materials and geometry is difficult. Here, we will both describe some of the iso-normal clustering algorithm's limitations (which hold broadly for depth cues and Taylor series modeling), as well as suggest certain heuristics supported by empirical evidence:

**1. Sampling Rate:** We also need to address the issue of sampling in our appearance profiles. We have used the continuity of the smoothly moving light source to capture extrema locations. However, in reality, incident intensity is measured through discrete frames of the camera and, therefore, we can only obtain a sampled version of the actual appearance profile. What is the minimum sampling rate of the appearance profile such that clustering still gets valid iso-normal clusters? Consider a scene illuminated by a light source waved by a user, as in Figure 2.3.3. In the last row the sampling of the profile is slowly increased, and the results of clustering at each step is shown. Although at very low sampling rate the clustering breaks down, it remains robust for all other sampling rates. This is not surprising once we recall that our intuition for using brightness extrema as an iso-normal feature was that foreshortening makes

scene points 'light up' or 'go dark'. If the path of our light source is smooth, then the foreshortening term, $\vec{n}.\vec{s}(t)$, is low frequency and, therefore, many samples are not required to capture its effect in the appearance profile. To calculate the exact minimum sampling frequency, the path of the light source as well as the scene geometry are needed, both of which are unknown to our algorithm. Since the foreshortening component has low frequency, we suggest collecting data using a video camera with frame rate above 30 fps. In practice, we have seen that this is sufficient to enable proper sampling of the profiles.

**2. Number of Extrema:** We have shown empirical evidence linking shared extrema in profiles and surface normal. This raises the question as to what would be the minimum number of extrema needed to properly cluster a scene, and the related number of frames required in the input video. This is difficult to calculate because the number of extrema needed depends both on the geometry of the scene and its material properties. For example, in the first six rows of Figure 2.38, planes consisting of very dissimilar materials are placed at different orientations. Cross-clustering between the two planes becomes less likely if the materials are different and larger angles between the planes allow easier disambiguation. In contrast, in the last four rows of Figure 2.38, more extrema are needed since the materials are identical. These conflicting factors of material and geometry properties make it difficult to say exactly how many extrema will be needed, especially since both these factors are unknown to our algorithm. Although we do not address this issue here, we propose a heuristic that takes advantage of the fact that the user can interactively create profiles by controlling the light source trajectory. Consider a situation where the user is aware of the approximate range of normals in the scene. The light source could be moved in a way that crosses directly over these surface normals (such that foreshortening is maximum) at different times, creating extrema that are picked up by the clustering algorithm. In a similar way, waving the light source over the normal of a region with difficult material properties could result in needing fewer frames to create iso-normal clusters.

**3. Orthographic Projection:** Another assumption that we have not relaxed is that of orthographic projection. If the scene has significant depth, then this is violated and therefore surfaces that have the same local normal, but are at great distances from each other, may cluster separately. However, in a manner similar to the case of cast shadows, the solution is to overcluster the scene. The important point to note here is that each of the separate clusters created are still iso-normally consistent.

# Chapter 3

# Occlusions: Dual Views from Masked Illumination

Of all the light rays emitted, scattered or reflected by a scene, only the light-field measured by a camera can be directly accessed. In addition, when a programmable source, such as a projector, illuminates the scene, the incident light-field can also be known and controlled. In this chapter, we introduce a third set of computable rays: the occluded rays of a non-programmable light source. This 'shadow-field' is captured by using an opaque mask to block rays from the light source. Therefore, our representation is purely geometric and not photometric, since incident light rays are detected by *not* measuring them. Ray reciprocity allows us to describe these rays as if they were viewed by a virtual camera, which we term a **shadow camera**. If the light-source is static while the occluded moves, then the shadow camera captures the ray geometry of the light-source. We will first demonstrate how this allows us to render new views of the scene, using Helmholtz reciprocity. In following sections, we will show applications of shadow cameras to remove unknown refractive and reflective distortions and for rendering real world illumination onto synthetic scenes.

## 3.1 Shadow Cameras

Although specific instances of shadow cameras have been used for scene reconstruction (as in [3]), we develop a view synthesis framework for image-based rendering. Historically, view synthesis involves either camera motion or multiple cameras, while scene relighting requires illumination control (as in a light stage [103]). Dual methods, in contrast, exploit Helmholtz reciprocity and treat light sources as cameras and vice-versa. Shadow cameras utilize reciprocity to allow flexible control of camera pixels, rearranging them in geometries determined by the relative source-mask motion. In this sense, we extend the dual photography technique ([94]) beyond programmable light sources.

The recovered light ray geometry depends on the mask shape, the light source type and the relative motion between the two. The shadow camera is located at either the mask or the source, depending on which is stationary. In this paper, we focus on the use of linear masks (approximated in practice by thin, rigid wires). Two such masks, perpendicular to each other and moving with uniform motion, produce two distinct intensity minima at each scene point. The times at which these minima occur represent the horizontal and vertical image coordinates of a shadow camera centered at the source.

If a distant light source is used, the shadow camera is **orthographic** in nature and if a near light source is used the shadow camera is **perspective**. Multiple pairs of such perpendicular masks allow the creation of a virtual shadow camera array. Varying the mask speed/angle controls the shadow camera's intrinsic parameters (skew and scaling). Depending on the type of viewing camera, it becomes possible to switch between orthographic and perspective views.

We can also create shadow cameras with multiple viewpoints. Traditional **cross-slit** cameras are created by imaging through two perpendicular thin slits separated by some distance. Similarly, a cross-slit shadow camera is created when the motion paths of the two linear masks do not intersect and are sepa-

rated. When compared to mosaicing cross-slits from many perspective views (as in [123]), our technique requires only a single reciprocal pair.

Traditional **push-broom** cameras are created by imaging through a translating slit. Similarly, we concatenate scene points shadowed by a single, translating linear mask to create a 'shadow-pushbroom' camera. Finally, we consider the case of a camera viewing an object reflected through an unknown mirror surface. If the scene is illuminated by a point source, the computed dual view removes any distortions in the non-single viewpoint catadioptric image.

The shadow-camera technique is simple, easy to implement and requires only a background calibration plane to estimate the relative source-mask motion. It is widely applicable since reliable shadow detection is possible for scenes with complex BRDFs. In addition, it is not necessary to store the complete video and instead, we can efficiently detect the moving shadow edge using only a window of a few frames. Finally, the shadow camera resolution depends on the edge detector quality and not on the shadow width.

### 3.1.1 Dual Views from General Light-sources

Let a scene be illuminated by a source located at $L = (L_x, L_y, L_z)$, and imaged by a pin-hole camera $C$ moving along on a plane $\Pi_1$ whose location is given by $(u, v)$ (Figure 3.1(I)). The moving camera samples the 4D light field at rays specified by their points of intersection, $(u, v, s, t)$, on $\Pi_1$ and a parallel plane $\Pi_2$. To render any new scene view we select light rays that describe the virtual camera's *caustic*, which is a curve in space that all the light rays must be tangent to ([98],[49],[28]).

For the virtual pin-hole camera $C^{'}$ in the figure, the caustic degenerates to a point in space (the camera center) which lies on a third plane $\Pi_3$ parallel to $\Pi_1$, at position $(u^{'}, v^{'})$. The key advantage of traditional camera-centric IBR is that a desired caustic can be created without knowing scene shape. The trade-off is that the entire light field has to be captured, requiring many images. This can be reduced by making scene BRDF assumptions ([96]) or by doing more work to find correspondences across fewer samples by statistical modeling ([122]).

Dual photography solves these issues by replacing the source at $(L_x, L_y, L_z)$ with a camera and placing a projector at the desired virtual camera location $(u^{'}, v^{'})$, as shown in Figure 3.1(II). If the camera image is given by $I(x, y)$ and a virtual image at the projector is denoted by $I^{'}(x^{'}, y^{'})$, then Helmholtz reciprocity relates these as $I^{'}(x^{'}, y^{'}) = I(x, y)$ where $(x, y) \leftrightarrow (x^{'}, y^{'})$ are corresponding projector-camera pixels. Surface normals, required in Helmholtz stereopsis, are not computed since the expression for pixel irradiance $I(x, y)$ contains both illumination and viewing foreshortening (see Appendix at [94]).

Dual photography's limitation is that, for a given projector position, it recovers a single *perspective* view of the scene. To create virtual views for other camera caustics, we have to imitate the traditional IBR setup in Figure 3.1(I), with a translating projector instead of a moving camera, capturing the virtual image at each projector location. Instead, we wish to more fully and efficiently exploit

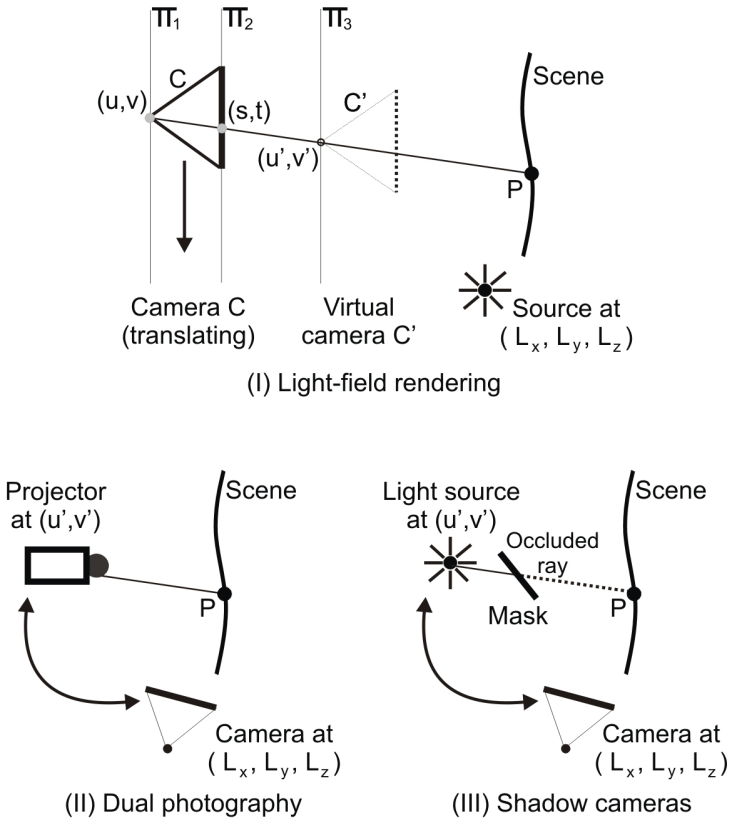



Figure 3.1: **Dual views for non-programmable illumination:** Traditional image-based rendering techniques (IBR) seek to capture the entire light field so that any desired virtual view can be rendered, as in (I). Dual photography (II) reduces the number of images required by placing a projector at the desired viewing location. However, for a single projector position, dual photography is restricted to a perspective view. Our method (III) recovers the dual view by using an illumination mask. Depending on the relative source-mask motion we create shadow cameras that are non-perspective and multi-viewpoint in nature.

Helmholtz reciprocity by creating dual views directly for non-perspective, non-programmable light-sources. The major challenge is finding correspondences between the real view and the dual view without the control a projector affords.

### Ray Correspondence through Shadows

Consider now a non-programmable point light source placed at $(u^{'}, v^{'})$, as in Figure 3.1(III). We move an opaque mask in front of the light source, creating a shadow that falls on the scene. A scene point $P$, located in the camera image $I$ at pixel $(x, y)$, displays a minima in its measured intensity at time $t$ if it is occluded by the mask. Let $R_P$ be the set of all time instances when such intensity minima occur at $P$. If, for every pair of scene points $P$ and $Q$, $R_P \cap R_Q = \emptyset$, then $R_P$ uniquely identifies a ray from the light source to $P$. Therefore each incident light ray at the scene is assigned a unique ID that corresponds to the locations in time when it was occluded.

If we know the caustic of the illuminant (defined similarly to a camera caustic) then we can map the identifier $R_P$ to a pixel on the virtual image $I^{'}$, $R_P \rightarrow (x^{'}, y^{'})$. The intensity at this virtual pixel is found by exploiting Helmholtz reciprocity such that $I^{'}(x^{'}, y^{'}) = I(x, y)$ for the correspondence $(x, y) \leftrightarrow (x^{'}, y^{'})$ between real camera pixels and the virtual camera pixels, *exactly* as in dual photography.

Finding the mapping $R_P$ can be done in two ways. The first is to design the mask motion such that the minima location gives the virtual image pixel directly. Since the pixel location has two degrees of freedom, we need at least two minima locations in time to do this. In Figure 3.2(I) we show a ray diagram of the shadow cast by a linear mask translating in front of a static point light source. Since the shadow hull of a line mask is a plane, the locations in time of two intersecting perpendicular plane shadow hulls specifies the horizontal and vertical coordinates of a virtual image pixel.

The second method to find $R_P$ generalizes beyond linear masks but requires a simple calibration step. The experiment must be repeated twice, first with the actual scene and the second time with a plane placed at the location where we wish the virtual image plane to be. If the plane is visible to the camera and the motion of the mask is identical in each experiment, then we can map each scene point to a coordinate on calibration plane.

The minima detection is independent of both BRDF variation and intensity-fall off and allows cast/attached shadow disambiguation. The virtual camera resolution depends on the shadow edge detector and not the shadow width and is theoretically only limited by the camera resolution. Our method also handles multiple sources since these would create many intensity minima scene points, each of which gives a different view point. Similarly, we can detect an 'minima interval' for area sources corresponding to a continuous set of viewpoints.

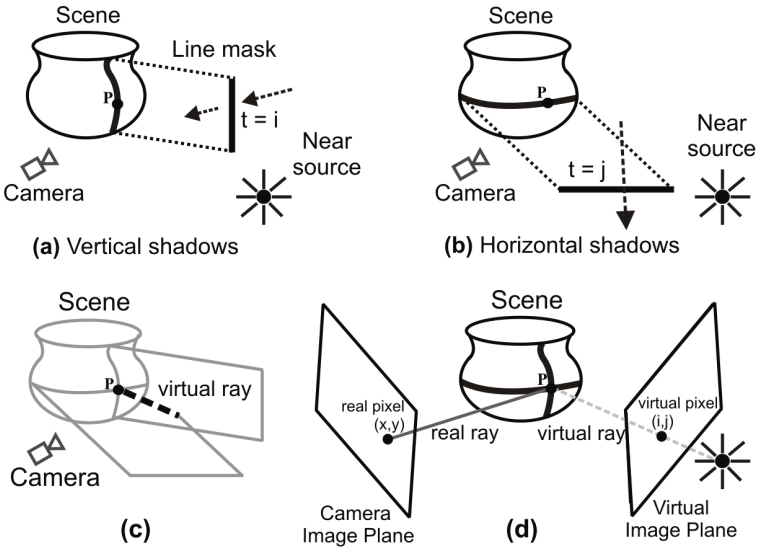
### Shadow Camera Resolution

In Figure 3.3 we show a mask moving from at speed $u$ units per second. The camera exposure time is $t$, and the mask moves a distance of $x = ut$. Since we only detect the shadow edge, $S$ determines the scene points shadowed during a single second on a plane at depth $D$ from the camera. Let $D$ be the greatest depth that exists in the scene. Without losing generality can assume the camera image plane is parallel to the plane at $D$. This can be relaxed by adding a skew term to the following equations. Using similar triangles, we can relate distance the shadow cross on the scene as:

$$S = \frac{Dx}{h} \tag{3.1}$$

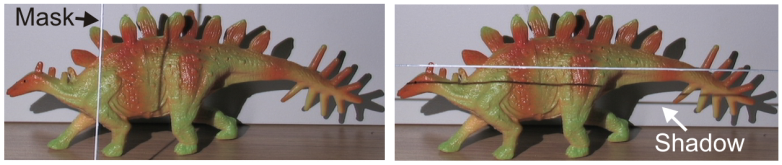
This is projected onto the camera image plane (of focal length $f$):

$$S = \frac{f\ D\ x}{h\ (D - f)} \tag{3.2}$$




(I) Ray diagrams of perspective shadow camera

(II) Non-convex plastic object




(III) Horizontal and vertical shadows from translating mask

(IV) Perspective dual view created by linear masks

Figure 3.2: **Virtual perspective view using linear masks:** In (I) we show a ray diagram for a perspective shadow camera. Using linear masks we obtain the input images, (III), which create a dual view in (IV). Note the foreshortening effects of looking down at the object (shortened legs, extended scales), that specularity locations do not change and that shadows in (II) are occluded in (IV).

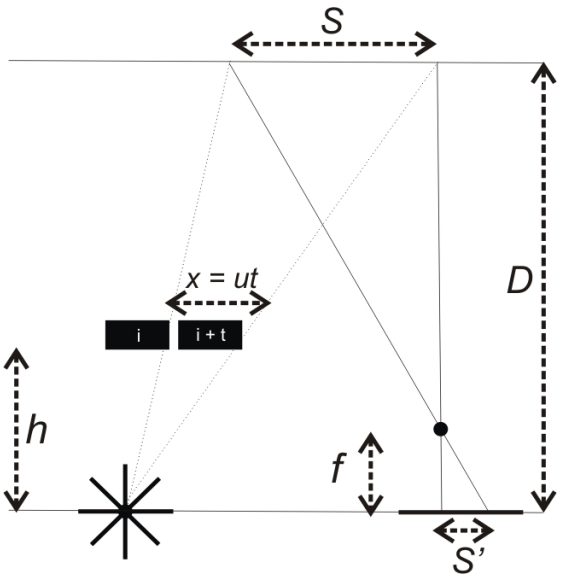



Figure 3.3: **Effective resolution of the dual view:** The resolution of the rendered image from the light-source point-of-view depends on the speed of the mask and the greatest depth in the scene. In this case we assume the camera image plane is parallel to a plane passing through the scene point at depth $D$, but this can be relaxed with an extra camera skew term.

If the width of a pixel (on the image plane) is given by $W$ and replacing $x$ by the mask speed and camera frame rate:

$$r = \frac{W\ h\ (D - f)}{f\ D\ u\ t} \tag{3.3}$$

where $r$ is the effective change in resolution in each dimension from the primal view to the dual view. Therefore if $r = 0.5$ the dual view is $\frac{1}{4}$ the resolution of the primal view. Note that $r$ is inversely proportional to the speed of the mask (faster masks mean lower resolution) and directly proportional to the camera frame-rate (faster frame rate cameras mean higher resolution). Of course, the detectable resolution of the shadow camera is bound by the actual resolution of the real camera.

### 3.1.2 Perspective Shadow Cameras

Our experiments were performed using a $12$ bit Canon XL2 video camera running at $30$fps. We selected a $1$mm diameter piano wire for the mask, held by a Manfrotto tripod that allows controlled height adjustment and our scenes were illuminated by a Lamina ceramics DK4 LED. We placed a plane behind the scene, to estimate the shadow locations and uniformly interpolated these to get proper sampling for the virtual image.

In Figure 3.2 (I) we show a ray diagram for a perspective shadow camera. I(a) and I(b) show two translating linear masks occluding P at times i and j respectively. The intersection of the shadows associated with the two masks is

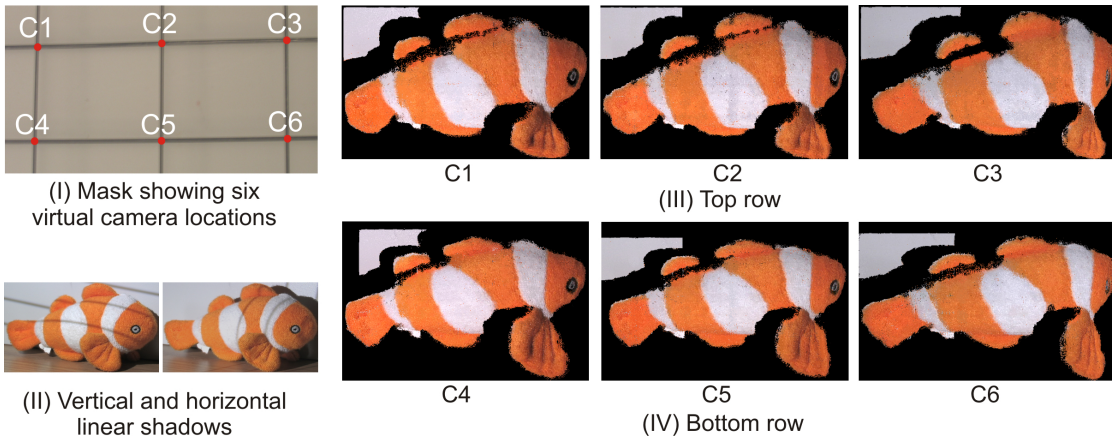



Figure 3.4: **Virtual camera array from multiple linear masks:** In (I) we show a grid mask with six intersections. We collect data using the horizontal and vertical masks separately, as shown in (II). We then place light sources at the six intersections of the mask, capturing six images. We swap these images using Helmholtz reciprocity and the minima locations of the mask shadows. The result is a virtual camera array, and in (III) and (IV) we show the two rows of the array. Note the images are stereo rectified both horizontal and vertically.

a virtual ray, shown at I(c). This shadow ray passes through the light source, and is associated with a pixel (i,j) in a virtual image at I(d). More formally, let the light-source location be denoted by $S$ and the moving vertical mask by $V(t)$ and the moving horizontal mask by $H(t)$. At some time $t$ we can construct either horizontal or vertical shadow planes $\Pi_v(t) = S \cup V(t)$ or $\Pi_h(t) = S \cup H(t)$. The virtual ray denoted in the figure is $R(i,j) = \Pi_v(i) \cap \Pi_h(j)$. This ray passes through the light source since $\forall t$, $\Pi_v(t) \cap \Pi_h(t) = S \cap (V(t) \cup H(t))$. Therefore any point $P$ shadowed by $\Pi_v(i)$ and $\Pi_h(j)$ lies on a ray $R(i,j)$ through the light-source center. The uniqueness of this mapping follows from the fact that we obtained it through shadowing - if there is another point $Q$ on $R(i,j)$ then it either occludes $P$ or is occluded by $P$ and we cannot detect shadows on it (or image it). Therefore, a one-to-one mapping between scene points and a location $S$ denotes a shadow perspective camera.

In Figure 3.2 (II) we show a non-convex plastic object. Using moving and opaque linear masks we obtain the input images, shown at (III). In Figure 3.2 (IV) we show a dual view for a non-convex plastic toy obtained using a set of two linear masks. Note the viewing foreshortening effects since the light source is higher than the camera position, such as shortening of the legs and the extension of the scales. In addition, the shadows of the dinosaur's tail are occluded in the dual image and vice-versa.

Multiple shadow cameras are possible by moving the source and performing experiments at the new light locations. We also demonstrate a different setup, where the mask is static and the light source moves. For example, in Figure 3.4 (I) we show a set of horizontal and vertical linear masks that are illuminated by a light source moving in a line. The shadows of the mask can be

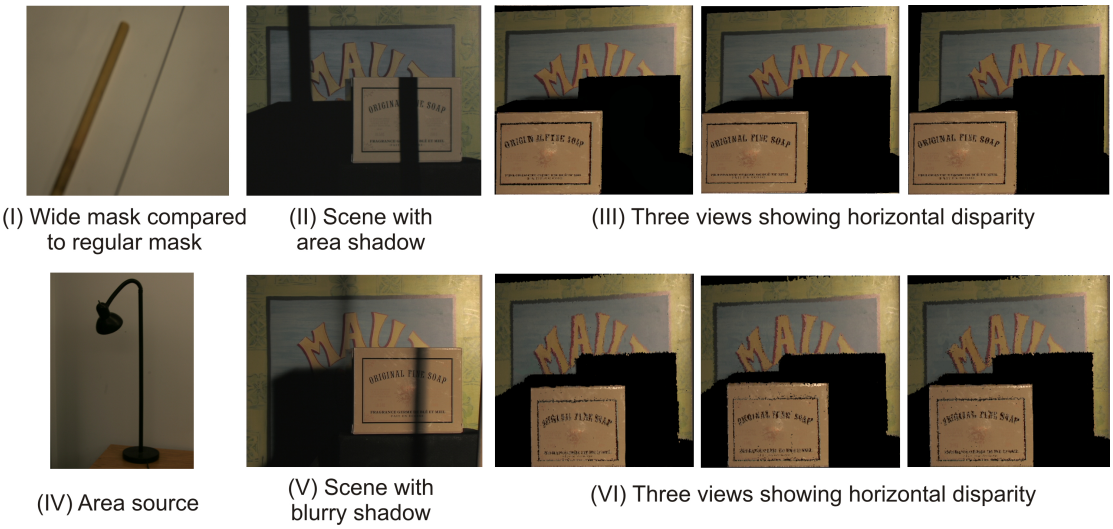
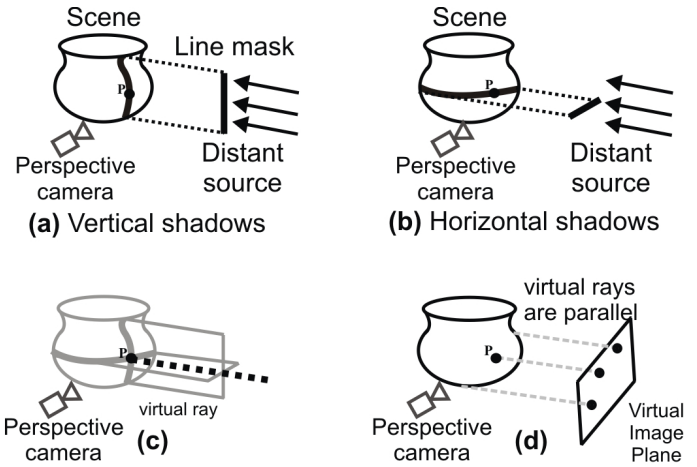



Figure 3.5: **Wide mask or area source enables multiple views:** In (I) we compare a wide mask with the regular mask we use. The shadow produces has significant area, as shown in (II). Each time instance when a scene point is shadowed gives rise to a virtual shadow plane, and selecting these for the horizontal and vertical shadows gives multiple views. In (III) we show three such views that demonstrate horizontal disparity. In (IV) we show an area source and the blurry shadow created by it (V). Similar to the wide mask case, we can create multiple views, shown in (VI).

seen in Figure 3.4 (II) and horizontal and vertical experiments are performed separately to enable easy minima detection. By recording the times at which the horizontal and vertical masks occlude every scene point, we can recover virtual shadow cameras at the intersection of the horizontal and vertical masks.

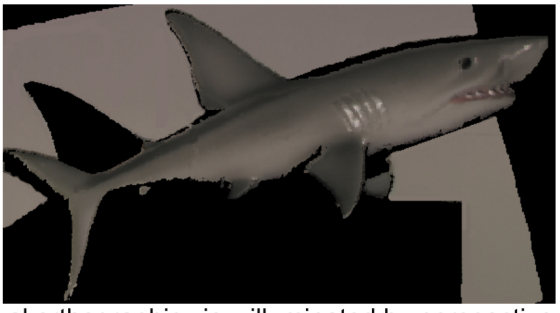
Extra steps are required when moving the source instead of the mask. Since the scene intensities vary, shadow detection becomes harder. We also need additional images taken with the light source placed at each of the mask's grid intersections. However, the advantage is that a single experiment produces multiple viewpoints: more precisely, if $n$ is the number of masks, we obtain an $n^2$ virtual array of shadow cameras. Six such images are shown in Figure 3.4 (III) and (IV), each of which are rectified horizontally and vertically since the wires in the mask array are perpendicular to each other.

Instead of moving the source, multiple shadow cameras are also possible by exploiting the ray geometry of wide shadows and area light sources. In the top row of Figure 3.5 we show two planes at different depths illuminated by a thick mask. By selecting different shifts of location in time for the detected minima in each plane, we can create a parallax effect and the dual images (shown in Figure 3.5 (III)). A similar set of images can be created by using the shadow of an area light source, shown in the second row of the figure. Again, different shifts in time for the detected minima provide new views of the scene.




(I) Orthographic shadow camera from distant light source

(II) Original perspective view with orthographic light-source



(III) Dual orthographic view illuminated by perspective source

(IV) Real orthographic view by co-locating source and camera

Figure 3.6: **Validating the dual view by perspective to orthographic conversion:** In (I) we show how the dual view due to a distant light source is orthographic. In (II) we show the perspective view of a plastic shark illuminated by a distant light source. Note the perspective effects since the shark's snout is close to the camera. In (III) we show the dual orthographic view. This compares reasonably with (IV) where we co-located the light source and the camera using a half-mirror (note the lack of shadows). The co-location does not move the light-source, explaining the difference in specularities between (III) and (IV).

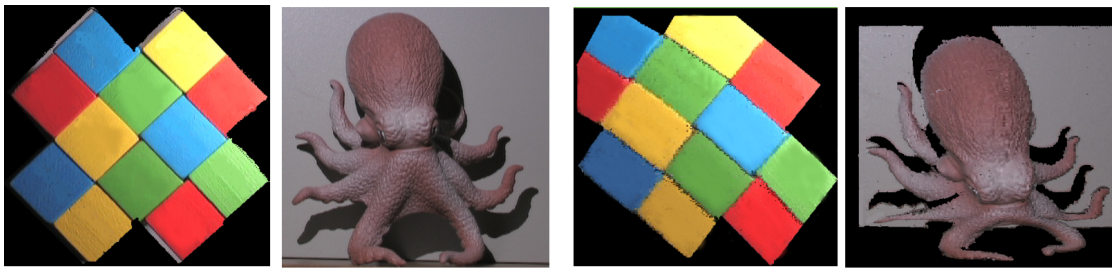



Figure 3.7: **Orthographic to perspective conversion:** In left column we show the original images viewed by an orthographic camera. In the right column, the dual views have foreshortening effects associated with perspective cameras.

### 3.1.3 Multi-view and Non-perspective Shadow Cameras

An indication of the potential of shadow cameras is that even for the limited case of linear masks and point light source discussed in this paper, we demonstrate many multi-view and non-perspective cameras. Figure 3.6 (I) shows that a ray diagram describing orthographic virtual view can be obtained from a perspective viewing camera by using a distant light source. In Figure 3.6 (II) we show a real view of plastic toy shark under distant (orthographic) illumination. Note the specularities on the shark and the shadows are correctly demonstrated in the dual view in Figure 3.6 (III). To validate the correctness of this view, we co-located the light-source and a camera using a half-mirror. Note the image in Figure 3.6 (IV) has no shadows. The shape of the shark is qualitatively similar in Figure 3.6 (III) and (IV). However, the appearance of the object is different since co-location is not the same as switching the source and camera positions. Despite this, the comparison demonstrates the correctness of the dual view.

Shadow cameras also allow us to demonstrate the opposite effect by switching from an orthographic view of the scene to a perspective views as shown in Figures 3.7. The perspective distortions are clear in the dual views, such as the change in angles of colored squares away from $90$ degrees and the foreshortening when looking down at the octopus. The other aspect of the switch is that the illumination also changes: for example, in the dual view of the toy octopus the shadows are smaller, due to the orthographic source.

A multi-perspective cross-slit image of the scene can be created if the two linear masks do not intersect at a point, and are instead shifted by some amount, as in Figure 3.8 (II). Mathematically, this is identical to having an image plane intersected by rays passing through two slits if the light source is moved between experiments. More formally (using the notation introduced previously), let the two light-source locations be denoted by $L_1$ and $L_2$ and the moving vertical mask by $V(t)$ and the moving horizontal mask by $H(t)$. At some time $t$ we can construct either horizontal or vertical shadow planes $\Pi_v(t) = L_1 \cup V(t)$ or $\Pi_h(t) = L_2 \cup H(t)$. Each of these is a fan of vertical and horizontal planes through $L_1$ and $L_2$ respectively. The intersection of a fan of planes is a line, and since $L_1$ and $L_2$ lie on the intersection of the two fans, the fan lines $f_1$ and

$f_2$ pass through $L_1$ and $L_2$. The virtual ray denoted in the figure is $R(i,j) = \Pi_v(i) \cap \Pi_h(j)$. Since the fan lines are contained in each plane, $R(i,j)$ passes through $f_1$ and $f_2$, which is the definition of a cross slit camera. In Figure 3.8 (III) we show such dual cross slit view: note that the octopus head appears 'unwrapped' while the colored squares are curved.

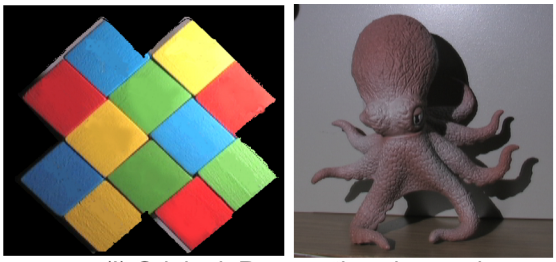
We also introduce a new shadow camera view inspired by the real pushbroom camera, which is usually created by imaging a translating slit. In our case we take the scene points corresponding to the shadow of a single translating linear mask and concatenate them together, as illustrated in Figure 3.8 (IV). The image, shown in Figure 3.8 (V), appears to be lit with light sources at both the camera and the light source positions. Note the double shadows of each of the tentacles of the octopus. We call this camera the 'shadow pushbroom' camera. In Figure 3.9 we show the shadow from a real-world light-fixture. Since the light is a tubular source, the shadow is also linear. We can obtain new shadow-pushbroom views of the object as shown in Figure 3.9 (IV).

Since shadow cameras are completely virtual, we can change their intrinsic parameters such as image skew or scaling. In Figure 3.10 we describe some of these perspective distortions applied to a scene. In Figure 3.10 (I) we show the dual view when the linear masks are perpendicular and move with uniform velocity, showing no distortions. We performed a third shadowing experiment with a slanted mask, which created virtual images with a non-zero pixel skew as shown at the left of Figure 3.10 (II). We also repeated the experiment with a horizontal mask moving slower than the vertical mask, resulting in higher horizontal sampling and a stretched image at the right of Figure 3.10 (II), which can be seen in the left dinosaur.

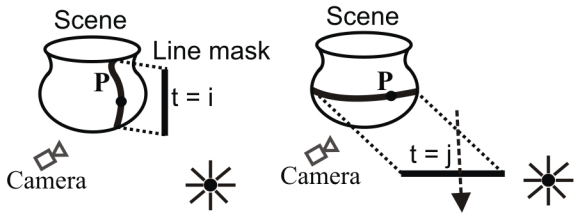
## 3.2 Unwarping Distortions: An Imaging Application

Various types of static optical elements exist whose material properties allow the bending of light. These include reflective surfaces, such as mirrors, as well as refractive solids such as glass and clear plastic. There are many situations where images are taken through these media: for example, catadioptric cameras ([98]) contain designed reflections that increase the camera field-of-view. However, unintended distortions are also possible due to lens imperfections and the presence of transparent occluders. Unwarping these effects without prior knowledge of the geometry of the optical elements is almost impossible.

If the light from the source is not affected by the optical elements that cause the distortion then shadow cameras can be used to render an warp-free image of scene. This assumption is broad and is true for both catadioptric cameras and lens distortions. In Figure 3.11 (II), objects are reflected off a spherical mirror and the straight lines of the colored squares appear curved. Since the light source is perspective, these distortions are removed in the dual views in Figure 3.11 (III). In Figure 3.12 we show a planar scene with writing viewed through thick glass objects. Note the heavy distortion, especially in the glass on the right. Figure 3.12(II) shows the undistorted view, with readable text.

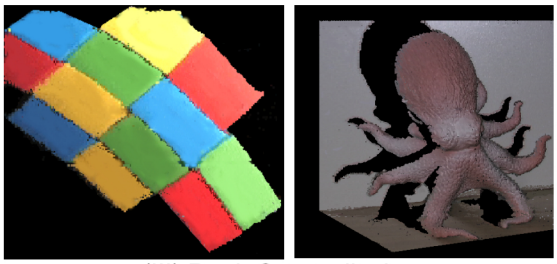


(I) Original: Perspective view and perspective light source

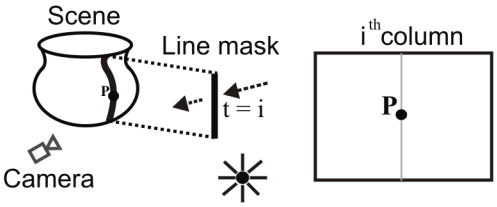



**(a)** Vertical shadows **(b)** Horizontal shadows

(II) Cross-slit: Motion vectors of masks do not intersect

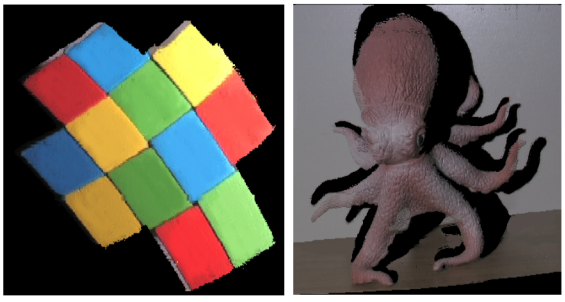


(III) Dual: Cross-slit view




**(a)** Vertical shadows **(b)** Virtual image

(IV) 'Shadow pushbroom': Concatenate single mask's shadows



(V) Dual: 'Shadow pushbroom' view

Figure 3.8: **Non-perspective and multi-view cameras:** In (II) we show a cross-slit view created when the motion vectors of the linear masks do not intersect and the light source is moved between experiments. We demonstrate the distortion effects in (III). In (IV) we create a pushbroom-like view using the shadow from a single moving mask. Note the octopus in (V) appears illuminated by two light sources and each tentacle has two shadows.


Figure 3.9: **Line shadows from an indoor light fixtures:** In (I) we show an image of a ceiling indoor light-fixture. The tubelights in the fixture act as line illuminants, producing a line-shadow (as in (II)). We use this to produce shadow pushbroom views of the scene shown in (IV).

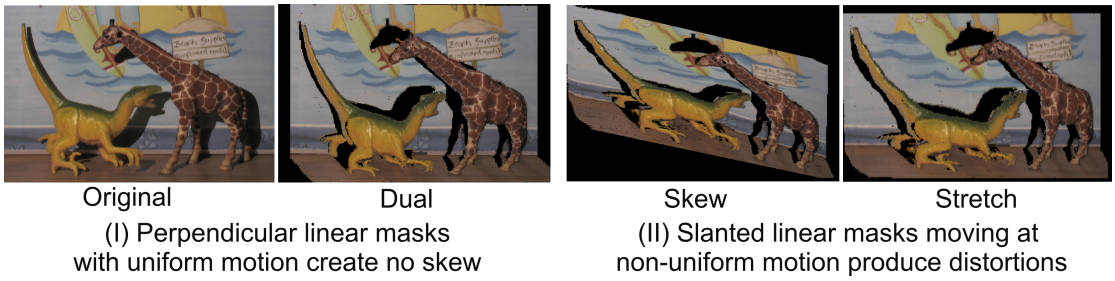



Figure 3.10: **Controlling intrinsic parameters of shadow cameras:** Since shadow cameras are virtual, we have control over the locations of the pixels in space. In (I) we show a dual view created by two linear masks moving with uniform motion at 90 degrees to each other. Varying the speed of these two perpendicular masks or adding a third linear mask that is not perpendicular to either of the first two results in skewed and stretched images as seen in (II).

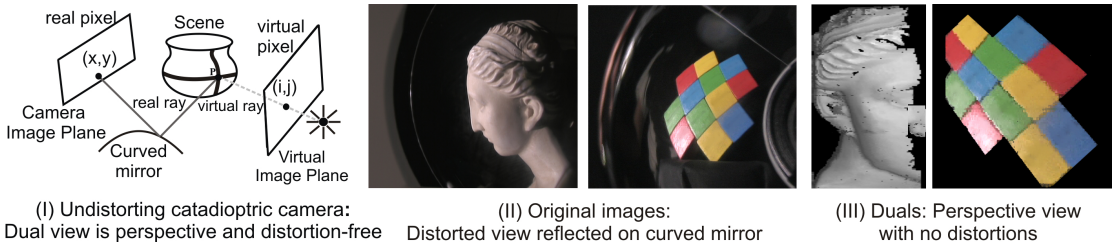
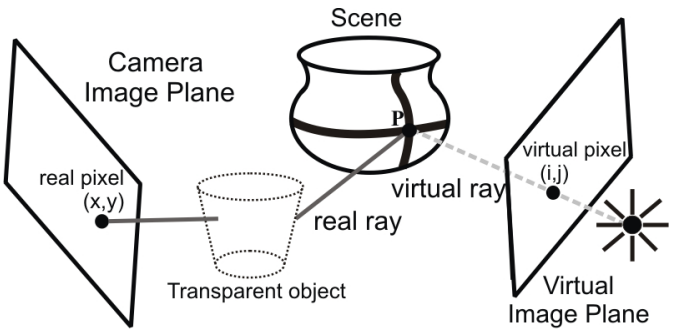
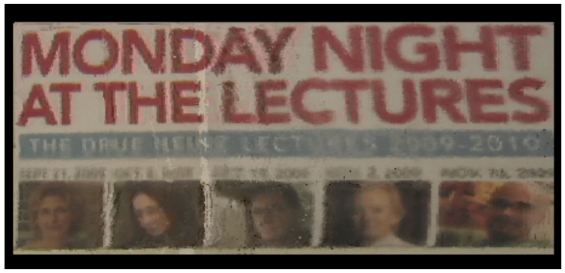



Figure 3.11: **Unwarping catadioptric distortions:** In (I) we show how the dual view from a perspective light source avoids catadioptric distortions. In (III) we show unwarped results: note the colored squares' lines are straight. The non-shadow holes are due to stereo occlusion.




(I) Undistorting transparent object refractions:
Dual view is perspective and distortion-free


(II) Original image showing distortions



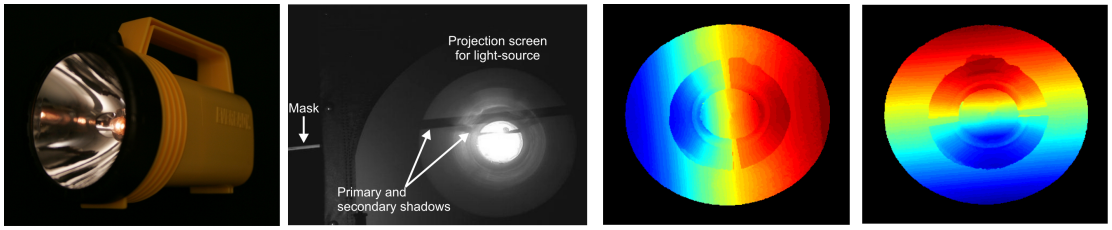

(III) Dual view with readable text

Figure 3.12: **Unwarping refractive distortions:** In (I) we show how the dual view of an object looking through refractive elements can be warp-free. This occurs when the light-source illuminates the object without being occluded by the refractive elements. In (II) we show a planar poster through two thick glass objects. (III) shows the dual view, which has undistorted, readable text.

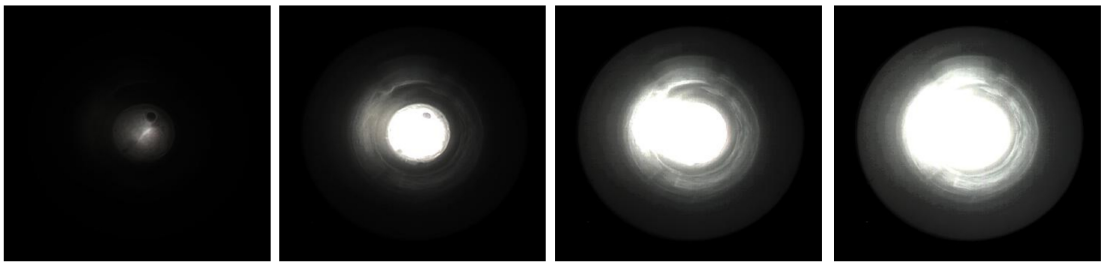
## 3.3 Light-source Light-fields: A Rendering Application

Unlike cameras, light-source appear in a variety of shapes and sizes, from point LEDs and lava lamps to car-headlights and water caustics. An important problem in vision and graphics to capture digital models of these types of illumination in-situ, for use in illuminating rendered scenes. Previous work had recognized that a single environment map, which assumed distant-lighting, was a terrible approximation for real-world illumination. [1] created a special goniometer, which essentially allowed the capture of environment maps at different locations in the scene. While this could model complex illumination for rendering, interpolating or extrapolating from the measurements was difficult since it requires obtaining correspondences across the environment maps. Furthermore, collecting the data requires many samples across the scene.

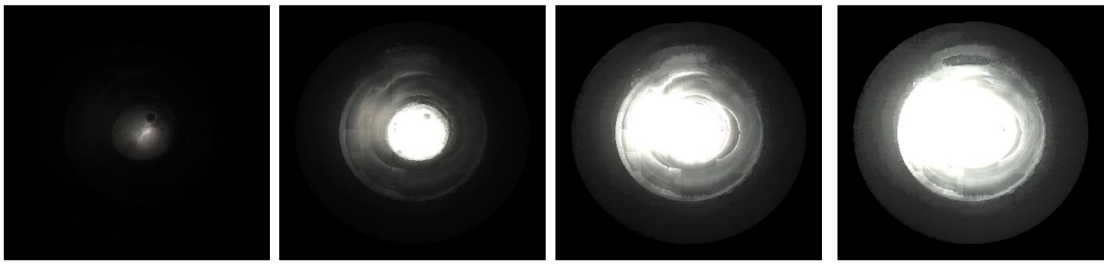
In contrast to the above effort, [24] modeled illumination as a light-field through two planes. A basis-filter was raster scanned over the first plane, and the dot-product of the illumination with the filter was measured by a camera on the second plane. To model complex illumination, $k$ basis filters are needed, making the number of scans $mnk$, where $m$ and $n$ are the dimensions of the screen. Furthermore, since the images must be normalized the method does not account for intensity fall-off. We correct both these limitations with a shadow camera experiment performed on a known scene as in Figure 3.13. We perform two experiments, each involving two 1D linear scans. Therefore the number of scans are $2m + 2n << mnk$. Furthermore, we do not smooth the illumination and recover high-frequency information when we render a new screen location in Figure 3.13 (IV). This is because our 'filter' is in fact a linear mask which has a high-spatial frequency itself. Finally, since we obtain correspondence between the two planes, measured by a camera, we can model the intensity fall-off along each ray given the distance between the planes. This allows us to extrapolate the light-field anywhere in the 3D space, and therefore the digital model captured can be used in any rendered scene.



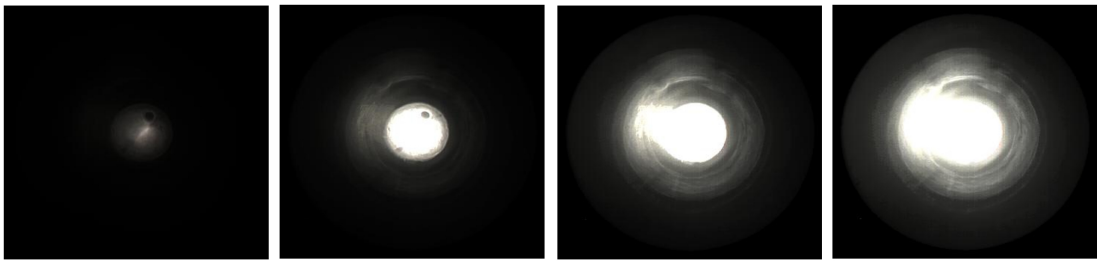

(I) Torch light and setup with screen (II) Horizontal and vertical correspondences



(III) Real HDR slices for first location of screen



(IV) Rendered HDR slices for a new screen position between (III) and (V)



(V) Real slices for a second location of the screen

Figure 3.13: **Light-field of a light-source:** In (I) we show our setup, which consists of a screen back-lit by a light-source of unknown shape. In this case, we use a torch. Note the complex radial patterns due to the reflective mirrors in the torch case. We move two linear masks, as shown in (I) horizontally and vertically, occluding the light-source. This casts a shadow on the screen. The horizontal and vertical minima locations in time for each pixel are shown as images in (II). In (III) and (V) we show HDR slices of the image of the torch, for two separate positions of the screen. Given the shadow correspondences, we capture the light-field of the light-source, allowing us to render how the torch's caustic would look at a virtual screen position between the two real positions. Note how the high-frequency information in the caustic is faithfully rendered.

## 3.4 Chapter Summary and Discussion

In this chapter we demonstrated a reciprocal approach by creating novel views from the relative motion between non-programmable sources and occluding masks. These views can be created for scenes with complex reflectance, without reconstruction and with no assumptions about light-source type.

**Technical summary:** Our method involves smooth motion between a light-source and an opaque line mask such as a thin wire. The mask shadow is easily detected as an intensity minima at each pixel. A minima location, say at some time instance, defines a shadow hull determined by the mask location at that time instance. For example, the shadow hull due to a thin wire mask is a shadow plane through the mask. Multiple minima detected at a scene point correspond to intersections of these shadow hulls. For example, minima created by two, different positions of the line mask define a line at the intersection of the two shadow planes. If the intersection of shadow hulls is a line (as in this case), then the line is a light-ray from the source to the scene point. Detecting two minima at each pixel, uniquely identifies all incident rays and defines the image coordinates of a virtual, dual scene view.

**Input and Output:** The input are images of a static scene taken either under smoothly moving illumination and fixed masks or smoothly moving masks and fixed illumination. The end result are mask shadows that fall onto the scene and are detected as intensity minima. The output is a dual view of the scene. The geometry of this dual view depends on whether the light-source or the mask is static. If the light-source is static, then this dual view is how the scene appears from the point-of-view of the light-source.

**Applications:** Our first application unwarps images that have distortions due to refraction (such as imaging through thick glass or lens distortions) or due to reflection (such as catadioptric systems or other camera-mirror setups). This is done with no knowledge of the shape of the optical elements involved, and no calibration object in the scene. The second application uses two calibration planes to capture the light-field of a light-source. This is essentially a method to capture a digital model of any complex light-source for use in rendering.

**Alternative methods:** Our approach shares aspects with Dual Photography ([94]) and extends it beyond programmable light-sources (such as projectors) to all types of illumination. However, we are unable to recover the whole light-transport matrix and therefore are restricted to obtaining dual views.

**Implementation Issues and Discussion:** As in the previous chapter, detecting the minima at pixels does not require storing the whole input data, but, instead, a sliding window can be used. Shadow detection determines the resolution and quality of the dual views, and good shadow detection is possible only if the mask moves slowly relative to the camera frame-rate (see Section 3.1.1). We now discuss future work in this area:

**1. Recovering off-diagonal information in the light-transport matrix:** In [94] patterns sent to a programmable light source are used to obtain the light transport matrix between a camera and the source. One (time intensive) way to achieve this is to switch on each pixel of the projector individual and record the

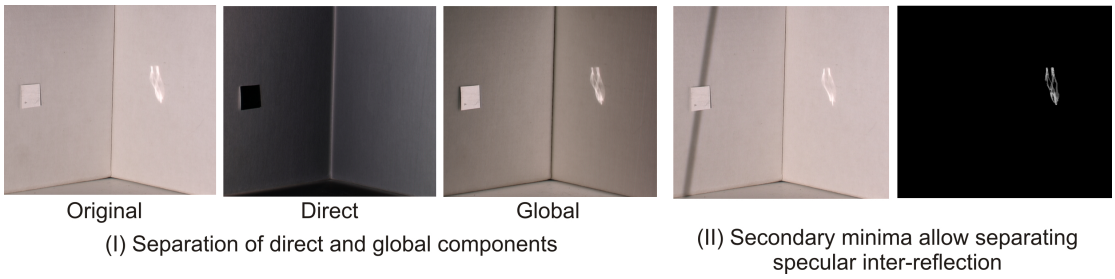



Figure 3.14: **Separating the specular interreflection from the global component:** By using the method of [65] we can obtain the global illumination from a scene, as shown in (I). However, looking at secondary minima (that occur when a scene point is not directly shadowed) as in (II), allows us to separate the specular interreflection (caustic) from the global component, which in this case is a one bounce reflection.

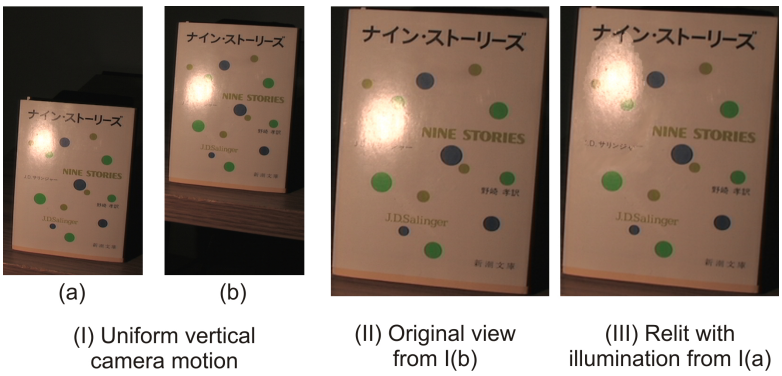



Figure 3.15: **Relighting cameras:** In (I) we show the first and last images of a camera moving vertically. In (III), using the EPI images of this object, we relight the image at I(b) with pixels from the image taken at I(a). From Helmholtz reciprocity, this is equivalent to moving the light source. Note the position of the specularity in (III) has changed from its original location in (II).

image. When a single pixel of the projector is on, we obtain the direct component of the illuminated scene point and its global contribution onto the whole scene. On the other hand, using a linear mask allow us to recover the diagonal values of the light-transport matrix (after separation with the method of []) as well as the correspondences between pixels in the virtual and real images (using the shadow camera method) for non-programmable sources. In this section, we discuss the possibility of recovering the off-diagonal information light transport matrix for scenes illuminated by non-programmable sources, just like [94] recovers it for projectors.

Consider a point mask that is raster scanning the illumination from a non-programmable source. If we subtract each of these images from an image taken with no masking, then this is exactly the image created by lighting up each ray individually. Therefore, we can get the full light-transport matrix from such a scan. However, performing such a scan is difficult and time consuming. Can

we instead obtain the same information with linear scans? In Figure 3.14 we show a simple V-groove scene, with a specular square on one facet. Note that after applying the method of [65], we can obtain the global and direct components. In any profile from the scene, a minima occurs when the mask occludes the direct illumination. However, there is another minima that may occur when the shadow occludes points that reflect indirect illumination as shown in Figure 3.14 (II). We can detect this and remove it, creating an image with just the specular interreflection. The disadvantage of this is that we have an ambiguity of where the reflecting scene point lies (along the linear shadow). In addition, the camera must have high-dynamic range to detect all the secondary minima.

**2. Relighting Cameras:** The inverse of our technique could relight scenes with static light sources and moving cameras. For example, a translating camera can generate epipolar plane images ([2]) with dense correspondences. We can relight any image in the sequence by replacing its pixels with corresponding pixels from other frames. From Helmholtz reciprocity, such images are identical to those created by moving the light source. We term this dual method as *relighting cameras* and show a planar example in Figure 3.15, where we change a specularity location with a translating camera. The challenge here will be finding correspondences for non-linear camera motions.

# Chapter 4

# Strobing: Active Vision from Fast Illumination Dithering

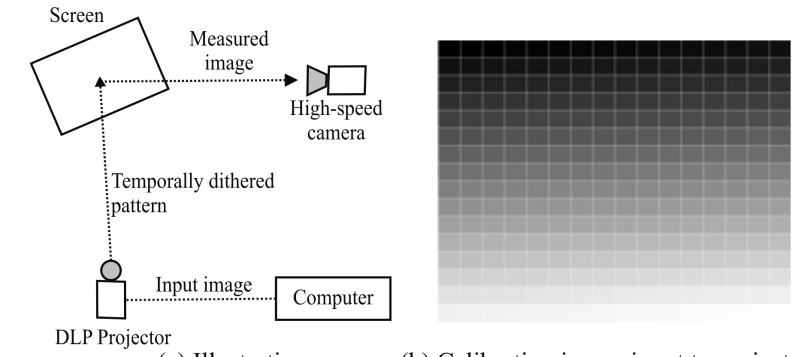
Dynamic scenes pose challenges to active lighting algorithms. Identifying the incident ray at each scene point is doubly hard, since now we must track scene points as they move across the field-of-view. To avoid this problem many active lighting methods use a programmable source to identify the incident ray at each scene point by using a spatio-temporal pattern that is varied much faster than the object moves. This transforms the problem back to the static case, since the scene is essentially stationary as the light varies. However, most projectors are limited to a maximum of 60Hz for changing the projected pattern. Many dynamic scenes move much faster than this, and camera-domain methods have an advantage since high-speed cameras of a 1000Hz and more and easily available. In this section, instead of refreshing the programmed pattern sent to a source, we investigate the temporal fluctuation or strobing of illumination. For many light-sources, such fluctuation is an inherent characteristic and usually occurs at a much faster rate than any sort of electronic control over the source. For example, all fluorescent lights flicker at 120Hz. However, all lighting plugged into the electric systems flicker at the same frequency, and this is not suitable for most active vision applications. Instead, we investigate the strobing due to the DMD chip present in DLP projectors for active lighting of dynamic scenes. While the strobing cannot be directly controlled, we can place multiple unsynchronized sources in the scene, and we demonstrate speed-ups for popular active vision techniques.

## 4.1 DLP Projectors

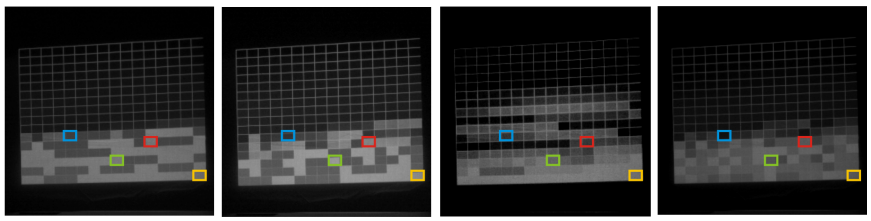
Recently, Digital Light Processing (DLP) technology (http://www.dlp.com) has enabled mass production of low cost projectors with high quality. The key component of a DLP projector is the Digital Micromirror Device (DMD). Each mirror in the DMD is $14 \times 14$ microns and can switch between two orientations, $+12^o$ and $-12^o$ [17]. In one orientation, incident light is reflected by the mirror toward the outside scene and in the other, light is reflected onto a black surface within the projector. These mirrors can switch between orientations in a few microseconds, enabling high precision control of illumination. As a result, the DMD device has found applications in areas ranging widely from microscopy to chemistry to holographic displays [17].

## 4.2 Temporal Dithering of DLP Illumination

In order to project a desired intensity value, the DLP projector emits a series of light pulses of different time intervals [17]. A sensor aggregates the pulses of light over the duration of its integration time (say, 1/30s in a video camera) to capture the final gray-valued brightness. This Pulse-Width modulation (PWM) by the projector is unique for every input intensity and can be termed as "temporal dithering" of the illumination. As we shall show, this dithering allows us to encode scene illumination in novel ways to achieve significant speedup in the performance of virtually any active vision technique.

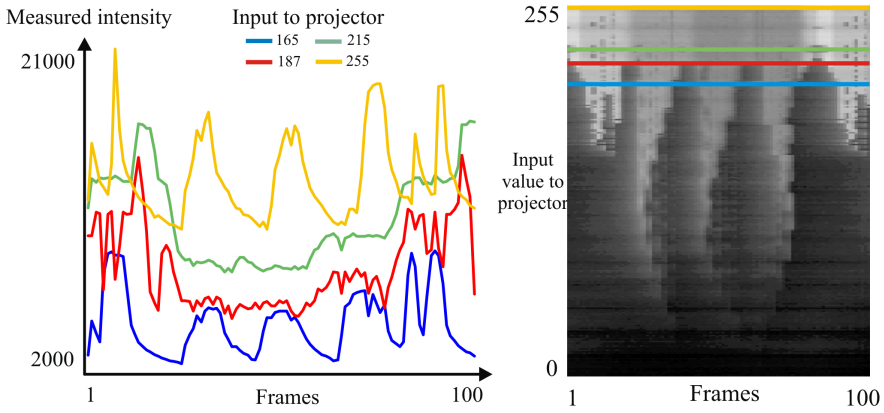



(a) Illustration (b) Calibration image input to projector



Frame 5 Frame 27 Frame 85 Frame 91

(c) Images measured by a high-speed camera




(d) Observed temporal dithering (e) Projector codes

Figure 4.1: **Reverse engineering a DLP Projector:** (a) A DLP projector converts the input intensity received into a stream of light pulses that is then projected onto a screen. A high speed camera viewing the screen aggregates the brightness over the duration of its integration time. (b) A calibration image composed of $5 \times 5$ pixel blocks each with a different intensity from 0 to 255 is input to the projector. (c) The camera records the projector output at 10 kHz. In (d) we show gray-valued intensities measured over time by the high speed camera for 4 example intensities input to the projector. Notice the significant variations in the plots. In (e), the temporal dithering for all 256 projector input intensities is collated into an image. This temporal dithering is repeatable and can be used to encode illumination in a novel way, enabling fast active vision.

But how do we capture this high speed dithering? The exposure time (1/30s) of a video camera is too long to observe the temporal illumination dithering clearly. One possibility is to precisely synchronize the camera with a DLP projector and to expose the camera only for the duration of a single projected light pulse (a few microseconds). Raskar et al [78] and Cotting et al [10] use this technique to embed illumination patterns in the scene that cannot be observed with the naked eye. The focus of these works is on intelligent office applications with 30-60Hz performance requirements.

In contrast, our work focuses on exploiting the temporal dithering for fast active vision. For this, we use a novel combination of a high speed camera and an off-the-shelf DLP projector. Figure 4.1 illustrates the dithering of an 8-bit InFocus IN38 DLP projector as observed by a Photron PCI-1024 high speed camera. A calibration image composed of $5 \times 5$ pixel blocks each with a different intensity value from 0 to 255 is input to the projector. Each intensity at a pixel $C$ in this calibration image is projected onto a flat screen using a unique temporal dithering $D_C(t)$, over discrete time frames $t$. The high speed camera observes the projected images at 10 kHz. Notice the significant variation in the images recorded. The plot in Figure 1(d) shows the patterns emitted by the projector for 4 input brightnesses (165, 187, 215, 255), as measured over 100 camera frames. The temporal ditherings corresponding to all the 256 input intensities in the calibration image are collated into a photograph for better visualization of this principle. The temporal dithering is stable and repeatable but varies for each projector-camera system.

### 4.2.1 Implications of Temporal Dithering

The high speed illumination modulation of a DLP projector can be exploited to speed up a series of well-known active vision problems, making them applicable to dynamic scenes. For each of these problems, we select a simple existing algorithm to demonstrate our technique, although more sophisticated algorithms may be used to achieve further speed up:

**(a)** The unique encoding of intensities allows us to obtain camera-projector pixel correspondences allowing 3D reconstruction at high speeds.

**(b)** By multiplexing illumination from three projectors, we compute the surface normals using photometric stereo [34] at high speeds.

**(c)** We de-multiplex illumination [90] from multiple projectors to capture the appearances of a dynamic scene from different lighting directions.

**(d)** We demonstrate the ability to project high frequency complementary patterns to separate the direct and global components [65] in a dynamic scene.

**(e)** We discuss motion blurring of an object illuminated by a DLP projector and captured by a low frame rate camera (30-60 Hz). The temporal dithering preserves higher frequencies in the motion-blurred image. This is similar to the work of Raskar et al [77] who demonstrate that fast camera shutter modulation during image acquisition preserves higher frequencies.

In methods (a)-(d), the projector receives a single image as input via a com-

puter, whereas the high speed camera acquires a sequence of frames. The effective speedup achieved depends on the task at hand and the quality of the result desired given the signal-to-noise ratio in the captured images. In addition, the intensity variation due to dithering can be observed reliably even with camera frame rates as low as 300 fps enabling applications with slower performance requirements. Unlike previous work, our techniques do not require any projector-camera synchronization, hardware modification or re-programming of the DMD device, or the knowledge of proprietary dithering coding schemes.

### 4.2.2 Reliability of Codes

If we look at any particular pixel over time at a given frame rate, as in Figure 4.2, we can see that it varies periodically. This is the temporal dithering or code that the projector uses to represent that intensity and is created by the binary switching of the micromirrors in the DMD chip. Of course, depending on the camera frame rate, that dithering is integrated to produce a smooth profile that is not binary. Figure 4.2 also investigates the consistency of the pattern for intensity A at frame rate F. We can see the standard deviation of the repeated values is low, which means that the temporal dither that we measure is stable.

From Figure 4.1, we can tell that different pixel intensities display different temporal dithering. Although each pixel has a unique encoding, we will now show that there are a few clusters of codes that vary similarly. In Figure4.3, we show confusion matrices for all $256$ patterns for three frame rates. We have picked three metrics, euclidean, dot-product and KL-divergence. Although most of the energy in the matrices is along the diagonal, there are many off diagonal elements, implying that clusters exist in the pattern. For the dot-product matrices, we have created binary matrices by specifying a cut-off threshold. These clearly show the different clusters, and we have selected profiles from each of these clusters to show their similarity.

### 4.2.3 Designing New Patterns for Fast Active Vision

Consider an active vision system for a dynamic scene, as in Figure 4.5. An object in the scene moves with speed $v$ and the frame rate of the camera is $f$. Usually an algorithm specifies $n$, the minimum number of images required to extract some scene property: for example classical photometric stereo requires three images under different distant lighting. Since we would like the object to appear to be static, the error $e$ is the distance the object has moved:

$$e = v.\left(\frac{n}{f}\right) \tag{4.1}$$

To get effective performance we would ideally like $e$ to be zero or at least sub-pixel. Loosely, this implies that the illumination vary fast enough that the object 'appears static'. This is challenging since we cannot control the object motion $v$ and usually the frame rate $f$ of the camera is fixed. Therefore the only

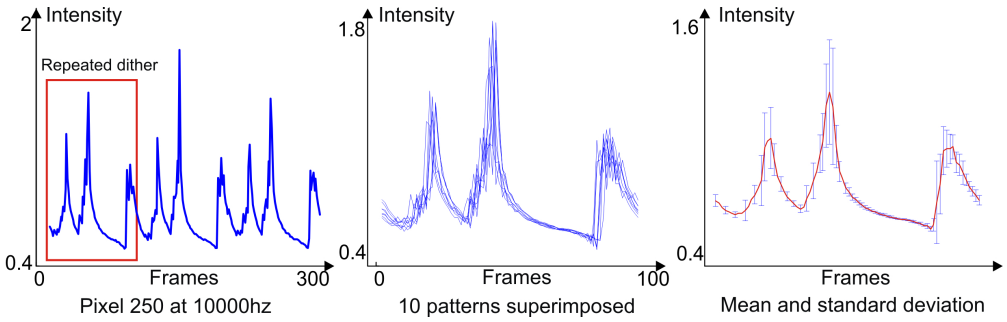



Figure 4.2: **How reliable is the dithering code?:** We performed an experiment for particular value of the projected intensity (100) and show that the DMD chip repeats the code over time, as on the left graph. In the center graph we show 10 patterns from the code superimposed. Note that although there is some variation, the shape of the patterns appears similar. Finally, in the right graph we show the mean and standard deviation of the intensity at each point in time. Note that the error is never large enough to 'flip a bit' or change any extrema (peak or trough) of the code.

way we can reduce the error is by requiring as few images $n$ as possible in our active vision algorithm.

However, most active vision algorithms typically require correspondence to disambiguate what illumination is incident on a scene point. For example, in the structured light pattern in Figure A, it is important to tell what part of the object belongs to which light stripe. Since the light stripe modulates over time, the more frames we have (larger $n$) the more robust matching we obtain and therefore there is a tradeoff in choosing $n$ in Equation 4.2.3. Figure 4.4 shows how the confusion matrices change with different number of frames with the matrices becoming less diagonal as the number of frames becomes low.

## 4.3 Applications

In the rest of this chapter, we will present different applications for the temporal dithering by speeding up popular active vision techniques from vision and graphics. We choose the number of frames to obtain the fast possible results in each of these applications.

### 4.3.1 Structured Light Projection

Structured light-based triangulation has commonly been used for 3D reconstruction [110]. A known set of spatio-temporally varying intensity patterns is projected onto a scene and the reflected images are used to find the corresponding pixels between the projector and the camera. The corresponding pixels are then triangulated spatially (or by temporal analysis [11]) to obtain 3D structure. It is assumed that the scene motion is negligible while the patterns are

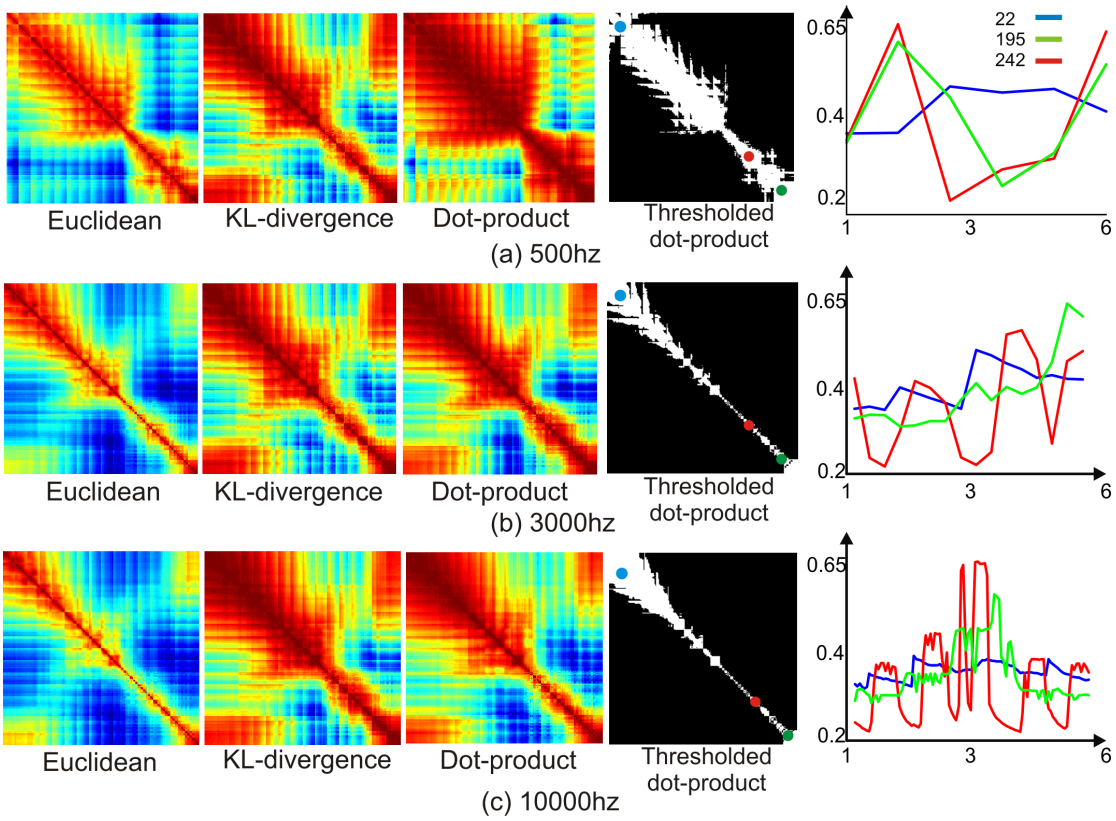



Figure 4.3: **Effect of frame-rate on confusion matrices for temporal dithering patterns:** In this figure we investigate the effect of frame-rate on the temporal dithering codes, which we plot in the right column for three values. The first three columns show diagonal matrices between all 255 temporal codes. The confusion in the matrices reduces as the frame-rate increase. This is expected since the original binary DMD codes are unique for most of the projected intensities (common codes may exist for some projector manufacturers). We show three distance metrics, with a thresholded dot-product as the fourth column.

projected. Since projectors have been operated at 30-60Hz, most implementations achieve slower than real-time performances. Fast illumination dithering in a DLP projector enables high speed reconstruction.

Our goal is to obtain correspondences between the projector and camera pixels at high speeds. Consider a high speed camera viewing a dynamic scene that is illuminated by the DLP projector. A *single* image composed of a set of horizontal lines of randomly chosen colors and intensities is input to the projector via a laptop. Let $I(t)$ be the vector of intensities observed, over a set of frames, at a scene point $P$. The normalized correlation between $I(t)$ and temporal dithering function $D_C(t)$ for each $C$ (Section 4.2) is computed to obtain the projector pixel $C$ corresponding to the image pixel $P$. But how do we synchronize the frames from the projector and the camera? One approach is to include a small planar patch in the scene where correspondence between the corners of the patch can be easily established (say, manually). This correspondence allows

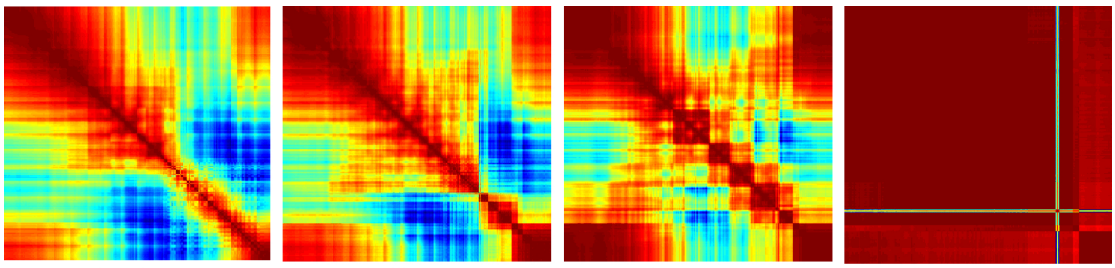



Figure 4.4: **Effect of code length on diagonality of confusion matrices:** In this figure we restrict ourselves to codes obtained at 10000Hz, which is faster than the speed of the DMD. We look at the confusion matrices by taking different lengths of the code. We note that only after 50 frames is the diagonality of the confusion matrix is strong. This implies a limit to the speed of the active vision results possible with our camera and set-up of around 200Hz. Faster results can be achieved with multiple unsynchronized projectors, and in this work we show a photometric stereo result at 300Hz.

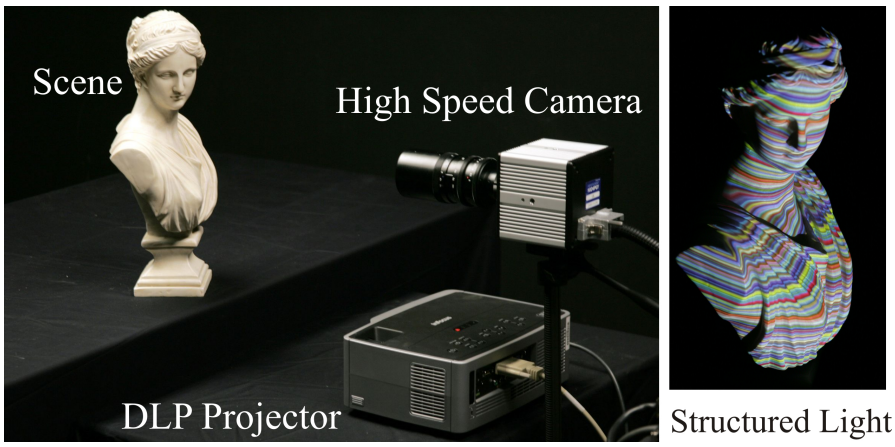
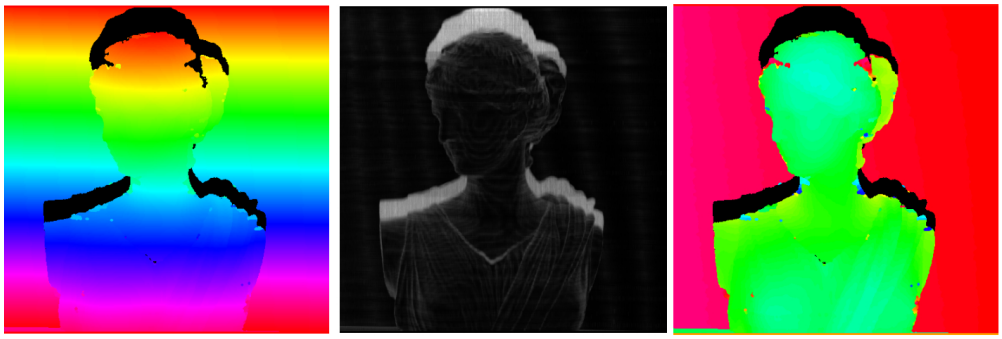


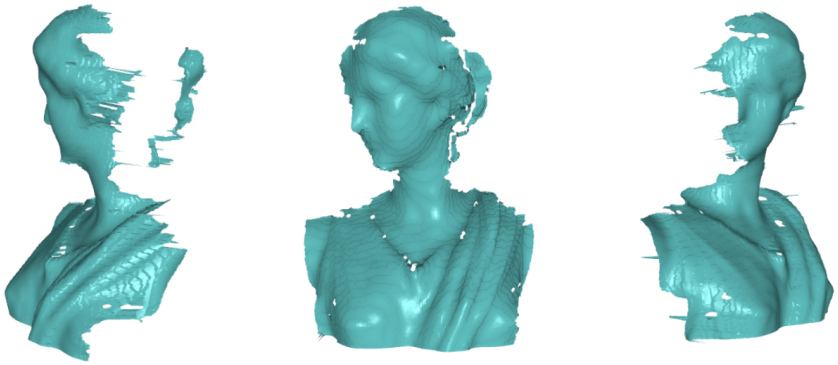

Figure 4.5: **Illumination and acquisition setup for structured light based 3D reconstruction:** The Photron high speed camera is placed vertically above the Infocus DLP projector. A vertical plane is placed behind the scene (statue) for calibration.

Frame 5555 Frame 5556 Frame 5562

(a) Input frames (3 of 25)



(b) Stripe correspondence (c) Matching error (d) Disparity map



(e) Recovered structure

Figure 4.6: **Results of 3D reconstruction using the DLP projector for a static bust:** (a) Three frames captured by the high speed camera illustrate the fast modulation of illumination incident on the scene. 20 continuous frames are used to match the intensity variation observed on the scene point against the normalized intensity variation observed on the vertical plane behind the object. (b) The best match finds correspondences between projector and camera pixels. The error map is shown in (c). The (d) disparity and (e) recovered shape of the object is shown from different viewpoints.

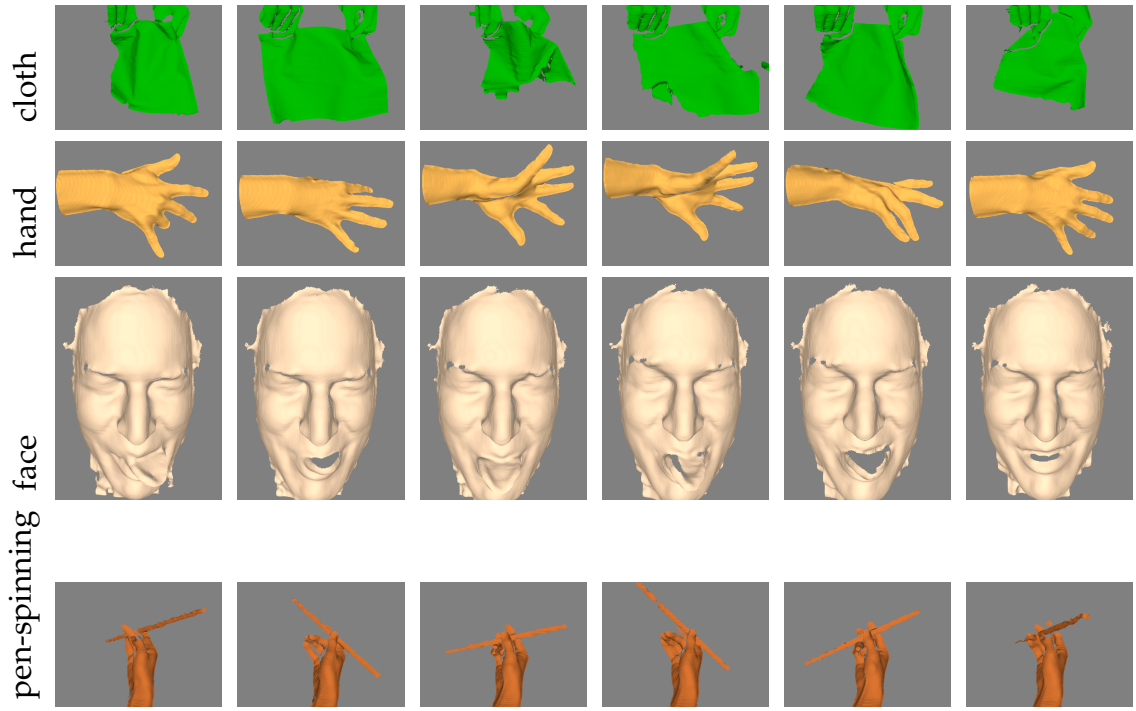



Figure 4.7: **Results of 3D reconstruction using the DLP projector for dynamic objects:** a waving cloth, a grasping hand, a face, and a hand spinning a pen. Each depth map is reconstructed using successive 25 frames in the high-speed images captured at 3000Hz.

us to synchronize the measured intensity vector with the temporal dithering.

We performed several experiments with both static and dynamic objects: a static statue, a waving cloth, a grasping hand, a face, and a hand spinning a pen. For convenience, the camera and the projector are placed in a fronto-parallel configuration with a vertical baseline (see Figure 4.5). The high speed camera is operated at 3kHz and the projector is reverse engineered at this speed as described in Section 4.2. A homogeneous vertical plane is used as the background behind the scene. The dithering $D_C(t)$ can be captured from the pixels on this plane and simultaneously with the object. Hence, in this setting, we simply correlate the normalized brightness vector $I(t)$ of a point on the object with the dithering vector $D_C(t)$ at every pixel $C$ on the background plane and no additional synchronization is needed.

Here, twenty frames were used to obtain correspondences, taking $20/3000 = 1/150s$. In general, the number of frames necessary depends on the desired matching accuracy and the SNR of the acquired images. By sliding the $20$ frame window across the full image sequence, 3D reconstruction can be obtained at rates as high as 3 kHz (speed of camera). However, in practice, the rate of reconstruction is lower considering the speed of the object's motion. Figure 4.5 shows the correspondences, the disparity and the reconstructions obtained for a classical bust. Figure 4.7 shows the reconstructions of other dynamic objects.

### 4.3.2 Illumination De-multiplexing for Appearance Capture

Acquiring scene appearance from multiple lighting directions is necessary for image-based relighting and photometric recovery of scene properties (BRDF and 3D structure). In most works, the scene is assumed to be static and the acquisition can take any where from a few seconds to several minutes. Using DLP illumination, we capture the appearances of a dynamic scene from multiple lighting directions. For this, we draw upon the idea of illumination de-multiplexing [90], where the images of the scene are simultaneously captured from multiple source directions and de-multiplexed in software to obtain the desired images under each lighting direction. This technique increases the signal-to-noise of the captured images while keeping their number unchanged.

The difference between Schechner et al. [90] and our technique is in the coding: they use binary Hadamard codes, whereas we rely on the temporal dithering of DLP illumination. The acquisition setup consists of three DLP projectors (Infocus IN38 and LP120, and Sony XGA DataProjector) that simultaneously illuminate the scene from different directions. Since we wish to illuminate the scene uniformly, a single constant brightness image is input to each of the projectors.

The three projectors differ in their brightness and contrast ratings and dithering behaviors. The captured intensity at time instant $t$ is written as a sum of irradiances due to the illuminations from all projectors ($k = 1 \ldots 3$):

$$I(t) = \sum_{k=1}^{3} D_k(t) E_k(t) \tag{4.2}$$

where, $D_k(t)$ is the dithering intensity of the projector $k$ at time $t$ and $E_k(t)$ is the irradiance due to the scene as if illuminated only from projector $k$ but with unit intensity. The intensities $D_k(t)$ can be obtained by observing a stationary mirror sphere placed in the scene. The observed irradiances $I(t)$ over time form a linear system which is solved to obtain the appearances $E_k(t)$ of the scene from each individual lighting direction. In practice, since the projectors are not synchronized when they illuminate the scene, the dithering intensities $D_k$ vary significantly over time, and hence the linear system is well-conditioned.

Figure 4.8 shows the results of applying the above approach to a scene with a falling wiry ball. Notice the 3 shadows of the ball and the mirror sphere that appear mixed in the multiplexed image $I(t)$. For robustness, we use 10 frames to solve the above linear system. Notice separation of the shadows in the demultiplexed images. As before, the effective rate of demultiplexing depends on the SNR in the high speed camera. We have thus far ignored color information, however, when the three DLP projectors emit intensities in different spectral bands, the de-multiplexing algorithm can be used to colorize the acquired high speed gray-scale video.

### 4.3.3 Colorizing High-speed Video by Demultiplexing

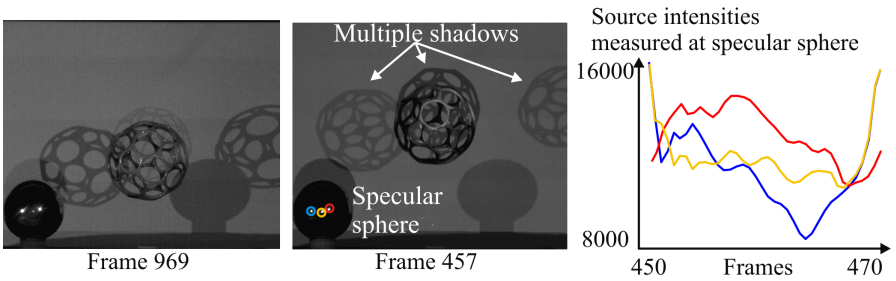
In the previous section we showed how choosing the right projector intensities can enable fast illumination demultiplexing for dynamic scenes. In this section, we show one application of demultiplexing to colorize a high speed video. In Figure 4.9 we show our setup in (a), where three DLP projectors are used to illuminate a scene and their centers of projection are kept close to minimize scene shadows. Each projector is fitted with a color filter, and the demultiplexed images obtained are the response of the scene to red, green and blue illumination. We can, therefore, create a color image of the scene as shown in the figure. Since we require 30 images for the demultiplexing, and the frame rate of the camera was 3000hz, the effective frame rate of the color video is 100hz.

Our Photron PCI-1024 camera produces black and white images, while a similar color high speed would cost almost double this. Therefore we believe our method is extremely practical and useful. One issue of our approach is that commercial DLP cameras have a color wheel inside them, that have red, green, blue and white filters turning at about 120hz. This violates our assumption that each projector corresponds to only one component of the spectrum. For example, for a projector with a external green filter, the response of the scene is different when the color wheel is at white, and when turns to red. Therefore, we removed the color wheel from the projector for these experiments.

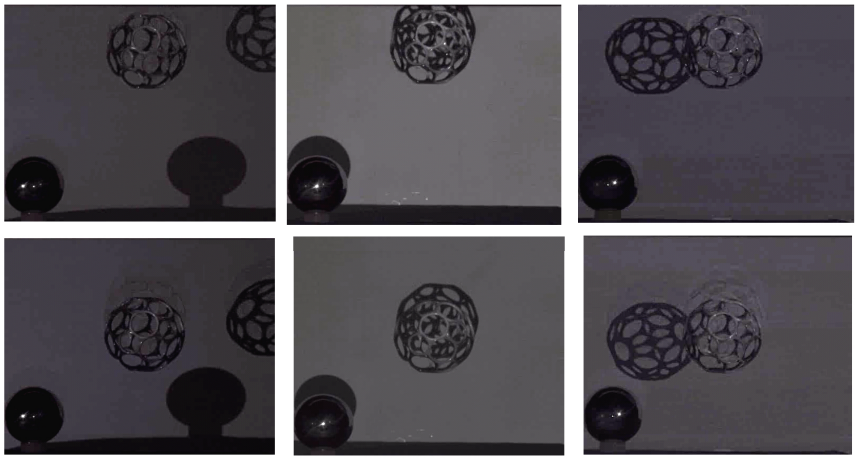
### 4.3.4 Illumination Multiplexing for Photometric Stereo

Photometric stereo is a widely used method to recover the surface normals and albedos of objects that are photographed under different lighting directions. There are many variants of this approach and we chose the classical one [112] for its simplicity. In this work, three images of a scene are used to obtain surface normals by assuming the lambertian reflectance model. We will extend this approach for fast moving scenes that are simultaneously illuminated from different directions.

The scene in our experiments consists of a steel sphere and a waving cloth (Figure 4.10) and illuminated by three DLP projectors simultaneously from different directions. The projectors and camera are far enough away from the scene to assume orthographic viewing and distant lighting. Since each projector must uniformly illuminate the scene, we provide a single constant brightness image as input to each projector (with different brightness values). The high speed camera records images at 3 kHz. The projectors are de-synchronized and hence, the "multiplexed illumination" results in significant variation in the observed intensities. The normalized intensities at a scene point are compared to those observed on the sphere. A matching length of 10 frames achieved robust results. A sliding window of 10 frames can be used to generate the normals up to a rate of 3 kHz. As before, the speed of the object determines the effective performance rate.




(a) Multiplexed Illumination




(b) De-multiplexed Illumination

Figure 4.8: Demultiplexing illumination from three projectors: Notice the mixing of the shadows in the multiplexed images of a falling wiry ball in (a). A mirror sphere is placed in the scene to measure the dithering intensities $D_k$ from the three projectors. (b) The results of demultiplexing for two frames in the video sequence. A total of 10 frames are used to demultiplex. Some ghosting artifacts are due to the noisy estimation of source intensities $D_k$. The variation in the contrast of the three demultiplexed images are due to the quality of the three projectors. Projector 2 (InFocus IN38) has the highest quality.

### 4.3.5 Complementary Patterns for Direct-Global Separation

The radiance of a scene point can be divided into two components - (a) the direct component $L_d$, due to the direct illumination from the light source and (b) the global component $L_g$ due to the illumination indirectly reaching the scene point from other locations in the scene [65]. The global component $L_g$ includes effects like interreflections, subsurface and volumetric scattering and translucency. Nayar et al [65] demonstrated that using high frequency illumination, it is possible to separate the two components and obtain novel visualizations of the components for the first time. A particular choice for high frequency illumination is a checker board pattern and its complement (with alternate bright and dark squares), both of which are projected sequentially for separation.

We exploit illumination dithering to obtain separation at video rates. However, in our setup, it is possible to input only one image to the DLP projec-

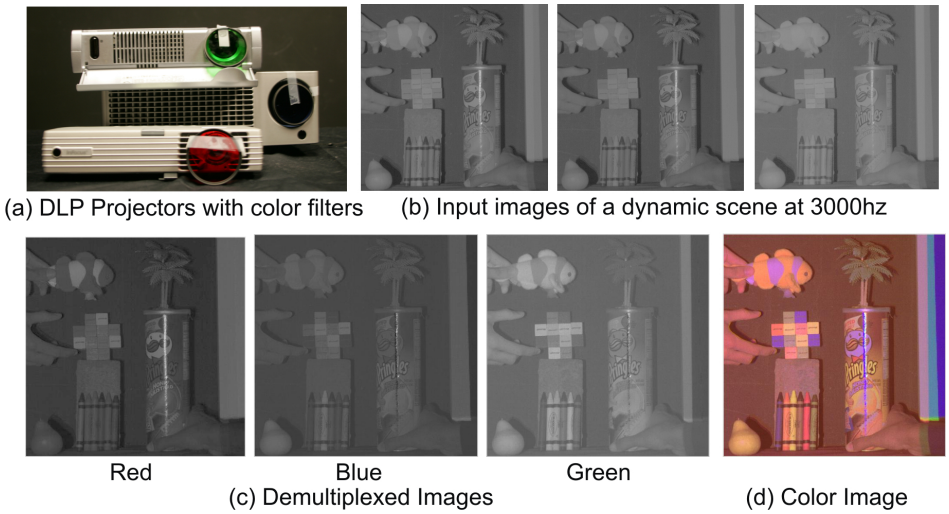



Figure 4.9: **Colorizing a high-speed video:** In (a) we show three projectors placed with their centers close to each other with red, green and blue filters. (b) shows a dynamic scene and in (c) we demultiplex the red, green and blue incident light. Note the calibration plane in (d) has a white strip where all the three colors overlap.

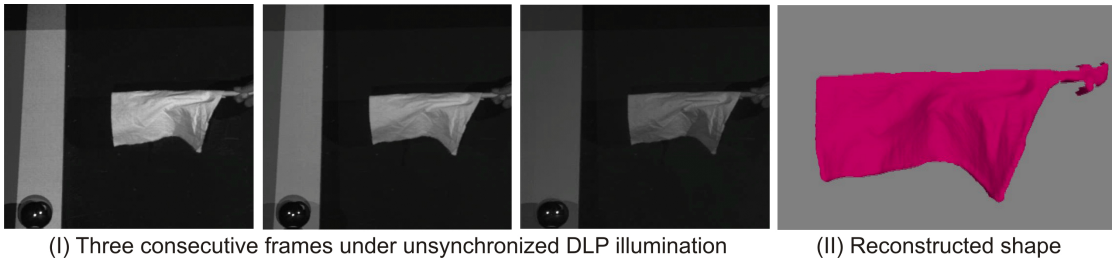



Figure 4.10: **Photometric stereo by example:** The scene in (I) consists of a waving cloth and a steel sphere showing the specularities from three DLP projectors. These simultaneously illuminate the scene and the camera operates at 3000Hz. The projectors and camera are far enough away from the scene to assume orthographic viewing and distant lighting. The surface normal at a point on the cloth is computed by photometric stereo with the light-source directions on the sphere. Since the projectors are not synchronized, the variation in multiplexed illumination from the 3 projectors is significant enough to obtain good estimation for surface normals. A matching length of 10 frames achieved robust results, which we demonstrate by integrating the normals to give depth in (II).

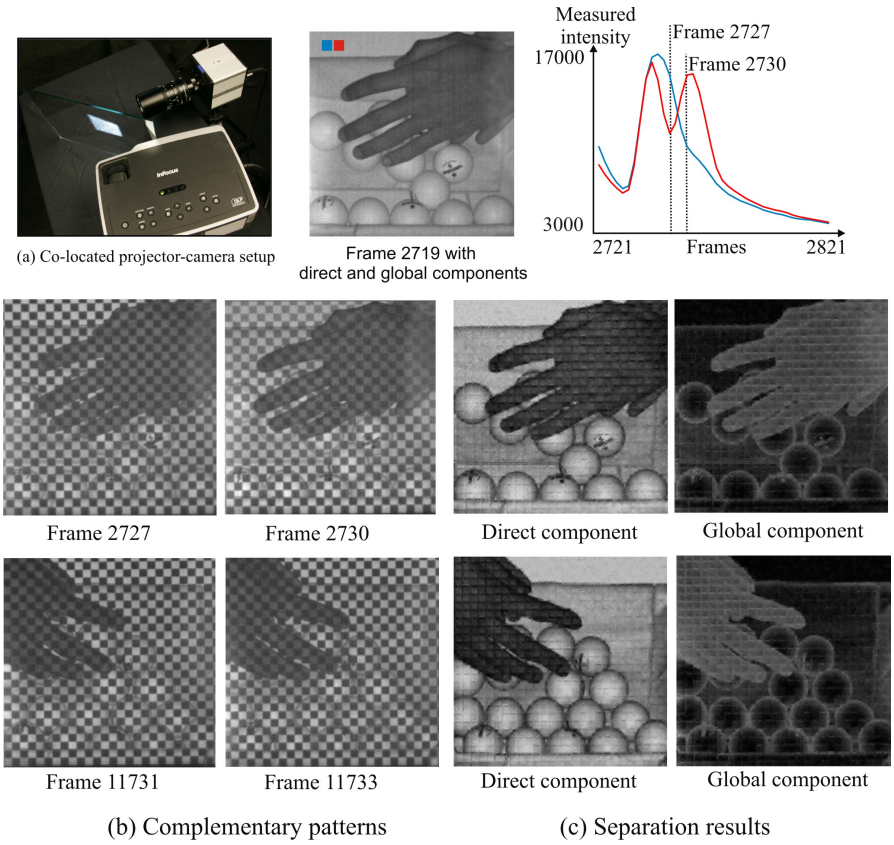



Figure 4.11: **Direct-Global Separation using DLP Dithering:** (a) The DLP projector and the camera are co-located using a beam splitter. A single checker pattern with two intensities 113 and 116 are input to the projector. The plot shows how the input intensities are dithered by the projector over time. Notice that at certain time instants, the patterns flip between bright and dark. Thus, the projector emits complementary checker patterns as in (b) that are used to separate the direct and global components (c). The flip occurs once in 1/100s.

tor in 1/60s and we have no control over the temporal dithering. So, how do we project complementary patterns much faster than 1/60s? We selected two specific input brightnesses 113 and 116 whose dithered patterns are shown in the plot of Figure 4.11. Notice how the two patterns "flip" from bright to dark and vice versa over time. Hence, a checker pattern with these two brightnesses are input to the projector. The dithering ensures that the two complementary patterns occur at high speeds. Let the observed temporally dithered values for input values 113 and 116 be $a$ and $b$, respectively, and the fraction of pixels that correspond to the value $a$ be $\alpha$ (0.5 in our experiments). The two captured images are [65]:

$$\begin{aligned} L^+(x,y) &= aL_d + [(1-\alpha)b + \alpha a]L_g \\ L^-(x,y) &= bL_d + [(1-\alpha)a + \alpha b]L_g\,. \end{aligned} \tag{4.3}$$

To solve the above equations, we need to know $a$ and $b$ in every frame. For

this, we place a white planar diffuse surface behind the scene of interest. For points on this plane, $L_g = 0$ and $L_d$ is a constant. This allows us to estimate $a$ and $b$ up to a single scale factor. Then, the above linear system can be solved at every pixel to obtain the separation. There is one additional complication in our setup beyond the method in [65]: it is hard to find out whether a scene point receives intensity $a$ or intensity $b$ from just the observed appearance of the scene. To address this problem, we co-locate the projector and the camera using a beam-splitter as shown in Figure 4.11. The pixels of the projector are automatically corresponded with those of the camera.

The scene in our experiment consists of a set of white ping-pong balls dropped from a hand. The ping-pong balls are mostly diffuse. Notice that the direct component for each ball looks like the shading on a sphere (with dark edges) and the indirect component includes the interreflections between the balls (notice the bright edges). For the hand, the direct component is only due to reflection by the oils near the skin surface and is dark. The indirect component includes the effect of subsurface scattering and dominates the intensity. The checker pattern "flips" once in approximately 1/100s and hence we achieve separation at 100Hz. Due to finite resolution of the camera and the narrow depth of field of the projector grid artifacts are seen in the results.

## 4.4 The DMD-Colorwheel Effect

Historically, imaging dynamic scenes proved to be challenging. Motion blur removes interesting detail, resulting in smeared images. Objects move in and out of focus while exhibiting appearance changes. Keeping the scene framed correctly on the object requires accurate control of the camera pose. In addition, image quality and object speed trade-off against one another. Today's high-speed cameras, although relatively expensive, have the frame rate and pixel resolution to address these issues for most applications.

However, there are still reasons to capture a single image of a fast moving scene. Summarizing a dynamic event is one application: for example, photographers may capture the aggressive posturing of two competitors in a sport. Another use is for experiments that do not justify the expense of a high speed camera, such as ballistics for a bullet passing through some material. Finally, high-speed photographs have aesthetic value and are used by artists to capture dynamic and complex scenes such as moving liquids and breaking glass ([60],[107]). In Figure 4.12 we show one of many photographs taken by Harold Edgerton ([44]) who used a high-speed strobe light to obtain many copies of moving objects. Figure 4.12(a) explains that the fast flickering of the strobe light creates multiple copies of the moving object. A similar effect is possible using the illumination of DLP (Digital Light Processing) projectors.

Every single-chip DLP projector contains two important components: a DMD (Digital Micromirror Device) device and a color wheel. The DMD chip modulates the projected light after it is reflected off an array of $10$ x $10$ micron mirrors ([17]). Any displayed intensity is made up of pulses of light created by these mirrors switching on and off. The MEMS mirrors can change their binary state within $10^{-6}$ of a second, resulting in crisp images with sharp contrast. Since a DMD device modulates light, it can only create binary images. In comparison, even the fastest LED strobes have a ramp-up time, creating grayscale values.

Projecting color images involves synchronized control between the DMD chip and the color wheel, which rotates at $120$Hz and is divided into red, green and blue filters. The 'rainbow effect' of the color wheel is well-known to display researchers who wish to remove or reduce it ([71],[45],[74],[102], [38], [68], [59]). Many researchers even remove the color wheel to increase the projector contrast in their experiments ([52], [5]). Instead of treating this effect as a problem that must be compensated for, our work demonstrates that DLP illumination can be exploited to photograph dynamic scenes.

In Figure 4.12 we illustrate how both these components create strobing effects for different classes of moving scenes. We term this the **DMD-Colorwheel effect**. For example, commercial cameras (which operate around $60$Hz) cannot detect the high frequency dithering of the DMD chip and, instead, the color-wheel effect will dominate for most dynamic scenes, as shown in Figure 4.12(b). Note that the region of zero intensity in between the color pulses are due to the mirrors on the DMD devices turned off.

For higher frame rate cameras, we can detect the dithered illumination within each color pulse, as in Figure 4.12(c). Therefore DLP illumination can produce

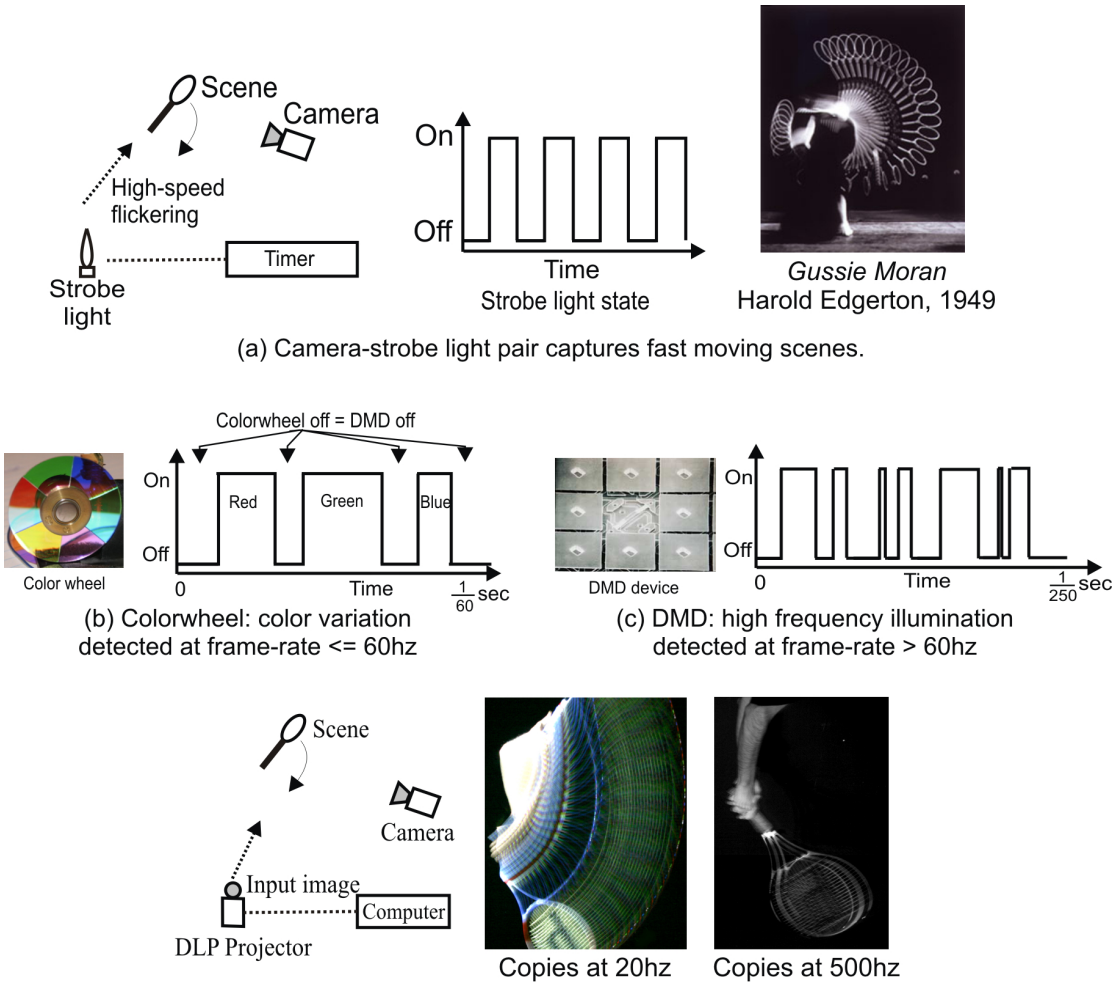



Figure 4.12: **Photographing fast moving scenes with varying illumination:** In (a) we show Edgerton's setup, which photographed moving scenes without motion blur using a strobe light. Strobe-light photography produces high-frequency object 'copies'. In (b) we show our setup with an unsynchronized DLP projector illuminating the scene. Both the projector's DMD and it's color wheel produce a similar strobe-like effect which we term the DMD-colorwheel effect, as shown in (d).

strobing effects for both real-time and high-speed scenes. This is illustrated in 4.12(d), where we image a tennis racket being swung. With a regular camera, the number of copies is large, and they are captured under different illumination (red, green and blue). In contrast, the copies in the photograph taken with a high-speed camera are fewer and grayscale.

DLP photographs offer two alternatives to the problem of deblurring images with deformable and articulated objects. First, a dynamic scene can be summarized by combining many DLP photographs. Second, videos can be created from DLP photographs that, although not deblurred, still give the perception of motion. For example, in some cases, the RGB channels in a DLP image can illustrate movement. We also discuss the frequencies present in DLP photographs, comparing them to images taken under both skylight and fluorescent illumination. Finally, we demonstrate that DLP illumination is programmable at each pixel. This is an advantage over similar camera aperture methods that can only control the shutter speed globally.

Talbot ([100]) created the first flash photography of dynamic scenes in 1851 using an electric spark. This technique was further improved on by Worthington ([113]), but was limited to scenes that did not cause much motion blur. In 1930 Edgerton ([18]) invented the first xenon flash tube and started creating truly high-speed strobe images as in Figure 4.12(a). Current LED strobe-lights have replaced the original xenon tube and can be computer controlled. As far as the authors are aware, DLP projectors have not been used widely for creating strobe-light photography.

## 4.5 DLP Photography

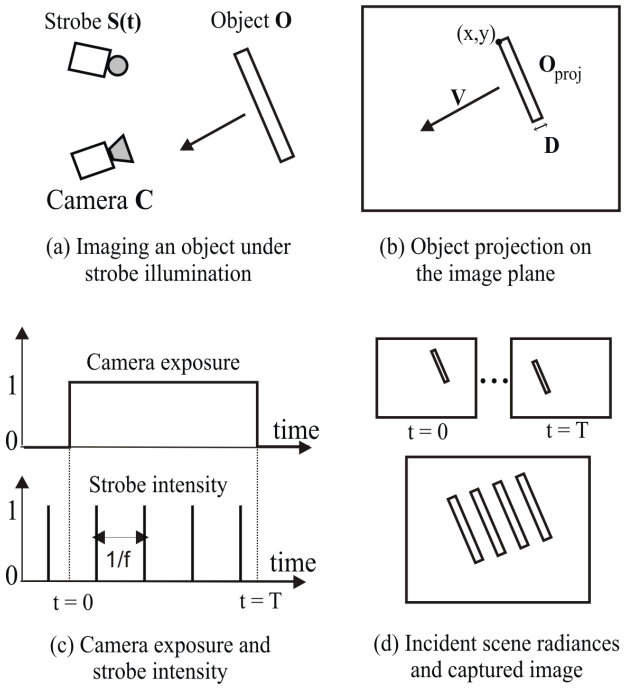
Consider a scene, as in Figure 4.13, consisting of a moving opaque object $O$ illuminated by a strobing distant light source $S(t)$ of frequency $\frac{1}{f}$. For the sake of simplicity, let us assume the object moves with uniform velocity in a plane with constant depth, and the optical flow of the projection of $O$ on the image, $O_{proj}$ is $\vec{V} = (u, v)$. The longest dimension of the object's image along $\vec{V}$ is $D$. The scene is imaged by a pin-hole camera $C$ whose exposure time is $T$. If $E(x, y, t)$ is the scene radiance incident at pixel $(x, y)$ at time $t$. then the measured image is:

$$I(x, y) = \int_{t=0}^{T} E(x, y, t)S(t)dt \tag{4.4}$$

Since $S(t)$ is the Dirac comb of frequency $\frac{1}{f}$, we can simplify the above integration into a summation. We further separate $E(x, y, t)$ into a sum of the background radiance and the object radiance:

$$I(x, y) = \Sigma_{o=0}^{\omega} O(x, y, t_o) + \Sigma_{b=0}^{\beta} B(x, y, t_b) \tag{4.5}$$

Here $t_o$ and $t_b$ are time indices for when the radiance is due to object and background respectively. Since we are interested in scenes containing fast moving objects, $\omega \ll \beta$. To prevent the background from dominating the mea-




(a) Imaging an object under strobe illumination

(b) Object projection on the image plane

(c) Camera exposure and strobe intensity

(d) Incident scene radiances and captured image

Figure 4.13: **Image formation model:** Consider a scene illuminated with a strobing light source, as in (a). Let an object move in a plane parallel to the image plane and at constant velocity, as in (b). Since the illumination is a dirac comb, the photograph can be modeled as a dot-product between a video of a moving object with the illumination, as in (c). The number of photographs is determined by the camera exposure, producing an image, (d).

sured intensity, our experiments are conducted in a dark room and therefore $B(x, y, t) = 0$. This is not a strict constraint and we note that with a camera of sufficiently high dynamic range, this would not be necessary. The image formation equation now becomes:

$$I(x, y) = \Sigma_{o=0}^{\omega} O(x, y, t_o) \tag{4.6}$$

To get an image containing many copies of the object, we would like to eliminate motion blur. Let the edge of the object be imaged at pixel $(x, y)$ as in Figure 4.13. To prevent motion blur, the optical flow of $O_{proj}$ must cover a distance $D$ in time $\frac{1}{f}$. Therefore the speed of the flow must be $Df$ and $\|\vec{V}\| = \sqrt{u^2 + v^2} = Df$. If the pin-hole camera has focal length $F$ and if the object moves in a plane at depth $Z$ then the actual speed of the object is $\frac{DZf}{F}$.

The DMD chip has a frequency of a $10^6$Hz, but the dithering in a commercial projector occurs at around $f = 10000$ Hz ([61]). If the ratio $\frac{Z}{F} = 100$, the longest dimension $D$ is $0.0001$ inch, then the actual speed of the object is $100$ feet per second. This is approximately the case for an air balloon bursting as showing in Figure 4.14. Under fluorescent lighting, viewed at $1000$fps, the balloon is smeared in a single frame, and this high-speed event is lost. However, when viewed under DLP illumination, the images at $1000$fps show copies of the edge

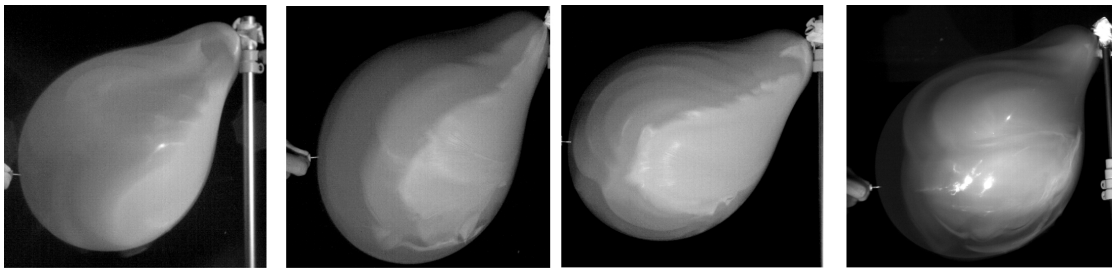



Figure 4.14: **DLP photographs of a bursting balloon:** An air balloon bursting can be captured fully using a 10000hz camera. In (a) we show what happens when the event is captured under fluorescent illumination, with a lesser rate of 3000hz. In one frame the event is lost in motion blur. In (b) we show images taken under 3000hz, but this time with DLP illumination. Notice the multiple copies of edge of balloon as it moves. We are able to capture images of this high-speed event, due to the temporal dithering of the DMD device in the projector. Similarly in (c) we use three DLP projectors, which are not synchronized. We are able to capture the balloon bursting at 500hz.

of the balloon. We used an Infocus In38 projector projecting a plain gray image of intensity $192$ of 3000 lumens, viewed by a Photron PCI-1024 high-speed camera. Our setup enables photography of an event occurring at $10$ times the frame rate of the viewing camera. Next, we use three projectors that are unsynchronized creating a higher strobing frequency and therefore obtaining a similar photograph at a lower frame rate of $500$Hz.

The color wheel has a frequency of around $120$Hz, and if we image a scene whose longest dimension $D$ is $0.001$ inch then the object speed must be at least $12$ feet per second. This is approximately the case for fast human movements. In Figure 4.15 we show pictures of a tabla (hand drum) being played, as well as a ballet dancer performing. Some of the copies appear at different colors, since they are illuminated when the color wheel turns the red, green or blue filters.

### 4.5.1 Separation of the Strobed Image Component

DLP photographs consists of two components. The first is due to the strobe effect of the light source and is created by the objects that move at a speed greater than or equal to $\frac{DZf}{F}$. We call this component the strobed component since it contains multiple object copies. The other parts of the image consist of pixels that have motion blur, which we call the non-strobed component. We wish to segment out the interesting high-frequency strobed component of the DLP photograph since these describe the motion.

To achieve this separation we make certain assumptions which may seem restrictive, but in practice, we obtain good results. First, we assume that the albedo of the object is constant and variation due to shading is negligible. This

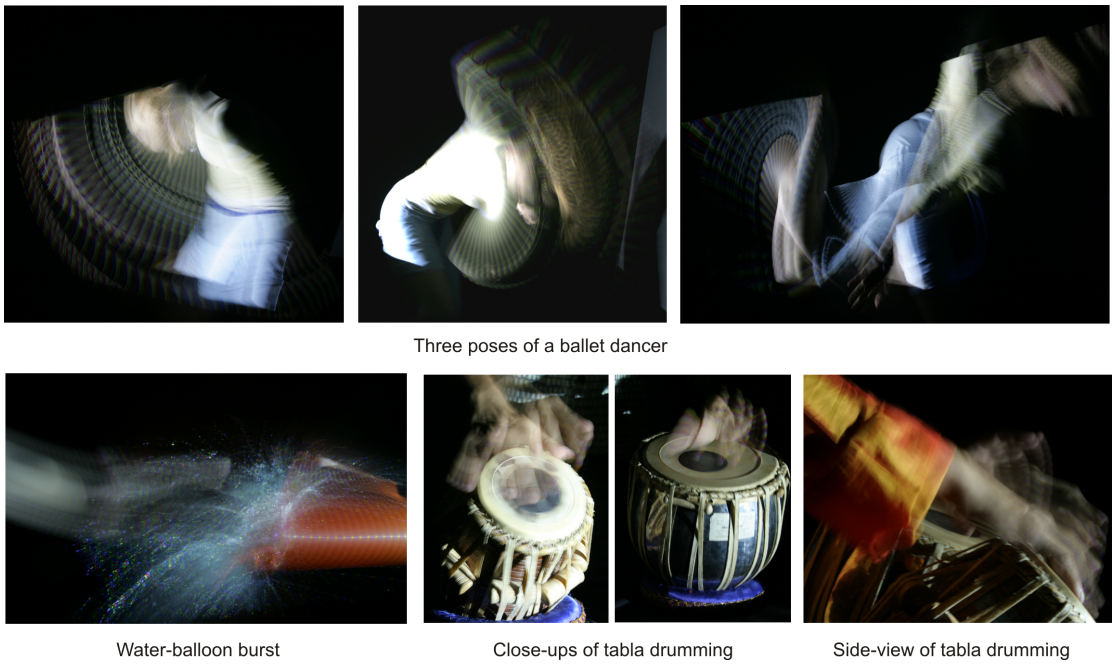



Figure 4.15: **Selected DLP photographs:** We photographed two artists, a ballet dancer and a tabla (hand drum) player, under DLP illumination. Both activities are 'real time' and the color wheel effect dominates the images. The camera exposure was 1 second.

is the same premise made in structured light, where scene points on the light stripe show an intensity maxima despite their different surface normals and BRDF. Second, we assume that every pixel is either strobed or non-strobed, and we wish to find the mask $\alpha(x, y) \in \{0, 1\}$ such that:

$$I(x, y) = \alpha(x, y)\ I_b(x, y) + (1 - \alpha(x, y))\ I_{nb}(x, y) \tag{4.7}$$

where $I_b$ and $I_{nb}$ are the blurred and non-blurred images respectively.

**Separation for DMD strobing**: A well-known method of blur identification ([80],[115],[81]) is to threshold the measured intensities. From Equation 4.6 we note that the strobed component would consist of a single scene radiance, whereas a non-strobed component would contain more. We use the mean of the measured intensities as a threshold. In Figure 4.16 we do this for a photograph of a tennis racket taken at 125Hz. Note that errors only happen when specularities occur since this violates our assumption of constant BRDF.

**Separation for color wheel strobing**: In this case, the regions of the image that are strobed have a dominant color (R, G or B). In contrast, the slower moving parts of the image have the normal distribution of RGB Therefore the color channel mean is less for strobed regions than non-strobed regions. We use this mean as a threshold to create a mask for separation, as seen in Figure 4.16.

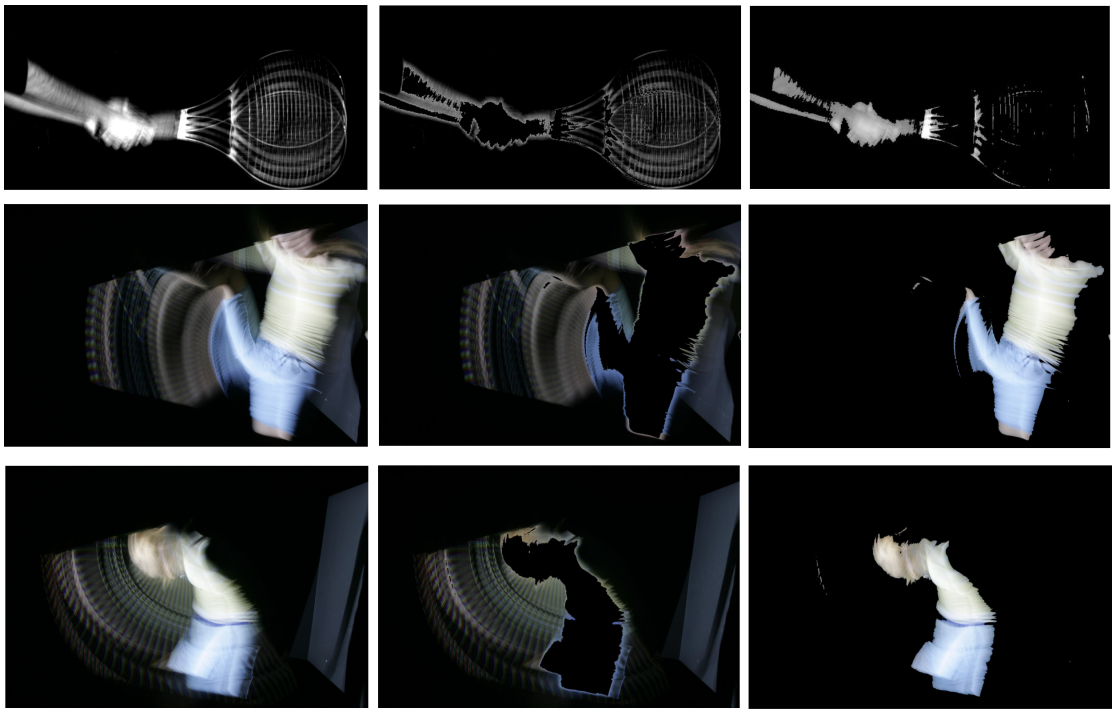



Figure 4.16: **Separating strobed and non-strobed components:** We use a simple appearance model to separate the strobed component of a scene: since the faster parts of the scene are imaged for a shorter time, their intensity is lower than other parts of the scene. This works especially well for the color wheel examples shown in the last two rows, where each strobed component is illuminated by either R, G or B light and has approximately a third of its original intensity.

### 4.5.2 Summarizing Fast Events

An image summarizing a video sequence can be created by stitching important frames together, as in shape-time photography ([22]). However, the object must move slowly since otherwise motion blur will render the final result difficult to interpret. DLP photographs summarize a short burst of action, since they contain multiple copies of moving objects. Applying a similar method as shape-time photography to a collection of DLP photographs creates a summary image for fast motion.

In Figure 4.17(a) we show images created by processing a volume of DLP photographs of a tabla player. We first separate the images into the strobed and non-strobed part. Except for the first image, the rest of the images are strobed. The top image is created by taking the intensity maxima of each pixel over all the photographs, which produces the effect of combining the different copies and gives a summary of the motions that occurred. In contrast, the bottom of Figure 4.17(a) is creating by masking the high intensity portions of each image and pasting them on top of each other. Instead of blending the outputs, these summaries enforce an order into the images.

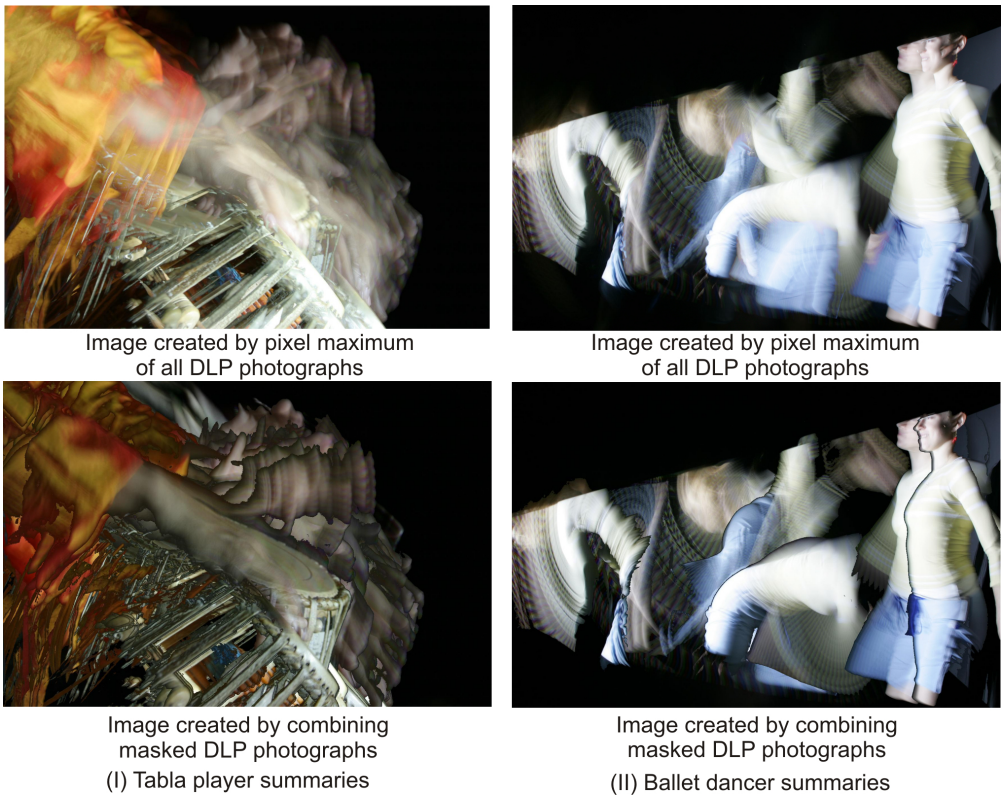



Figure 4.17: **Colorwheel motion summaries:** By combining DLP photographs in different ways we can summarize events. For both the tabla player, (a), and the ballet dancer, (b), we show summaries created by taking pixel maxima as well as by masking and superimposing the images. In the maxima case, no image ordering exists and all edges are blended. In the masking case, image order matters and edges exist between different stages of the action.

We can also create an impression of the scene's motion by blending the intensities of the different DLP photographs as in the top of Figure 4.15. Since each DLP photograph is already a summary of some part of the scene motion, we are able to compress a long and fast dynamic motion into a simple, pleasing summary. We show the mask-blending results for the ballet dancer in Figure 4.17(b). Since the scale of the scene is larger, the DLP effect is only observed in a frustum of illumination, which could be corrected by placing additional projectors (these may be too bright for the dancer, but may be fine for other objects). In this case, the order and number of the scene matters, since there is a lot of overlap in the original images, and these were chosen by the user.

While the previous examples showed images taken at 'real-time' rates (such as a human dancing or moving) with the colorwheel effect, now we look at strobing created by fast moving scenes with the DMD chip. In Figure 4.18 we show image summaries of a slingshot shooting a rock at high speeds. This summaries were created by applying the method of [22] to high-speed frames. The first result taken under fluorescent illumination with a 1000Hz camera, clearly shows high-frequency information. The same experiment taken with a 250Hz

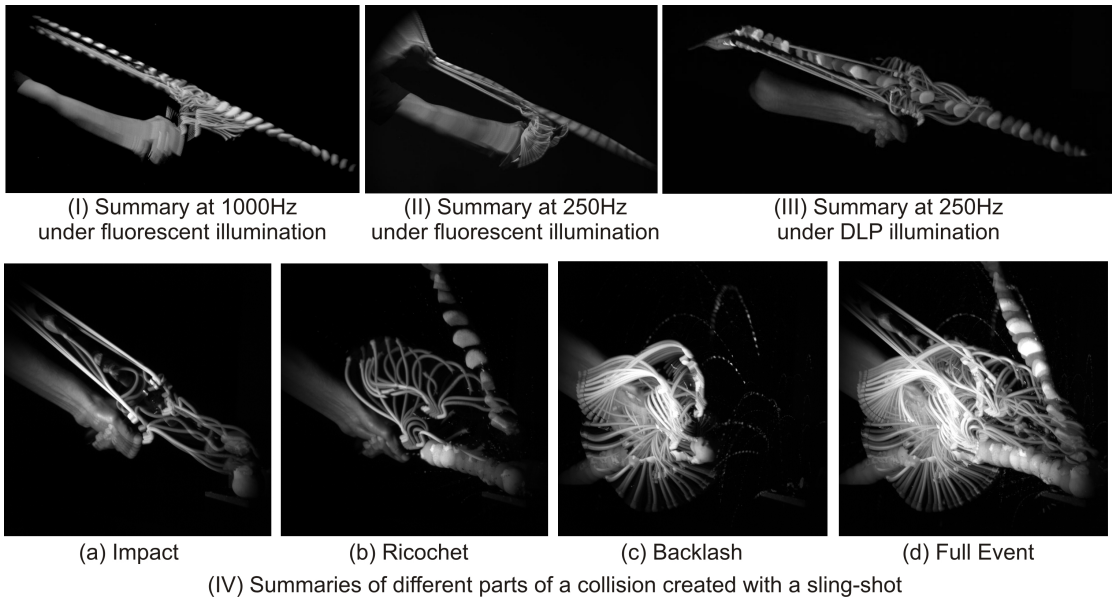



Figure 4.18: **DMD motion summaries:** Here we show experiments with a sling-shot, which moves fast enough that the strobing effects are due to the DMD dithering and not the colorwheel. In the top row we demonstrate how DLP illumination allows the image capture of dynamic scenes with a low frame rate camera. In the bottom row we show how applying the method of [22], allows the summary of different aspects of a fast moving event, again with a low-frame rate camera.

camera shows motion blur. However, replacing the fluorescent light with a DLP projector, creates an image on the right with the same high-frequency information obtained by a 1000Hz, but done with a 250Hz camera. Therefore, fast strobing of a DLP projector can enhance the effective frame rate of a camera by allowing to capture scenes moving faster than its frame rate. In the second row, we show a more interesting scene involving a collision of a slingshot rock and a stationary plastic pear. By applying [22]'s method to different subsets of the video, we can summarize different aspects of the event.

### 4.5.3 Creating the Illusion of Motion

Here we present three ways of processing a set of DLP photographs to produce a video that illustrates a scene's motion. We do not claim to deblur the scene or recover the motion in a quantifiable fashion. Instead, we believe these motion illustrations contain more information than the set of photographs by themselves and provide an easy way to visualize the event that occurred. The trade-off is that each of these techniques produces good results in certain broad classes of scenes, and may fail for others.

**Blending:** In Figure 4.19 (a) we show the first three frames of a motion illustration video for the ballet dancer. The frames in this video were created by differentially alpha blending the strobed and non-strobed components of DLP photographs taken at 1s exposure. The non-strobed component were blended

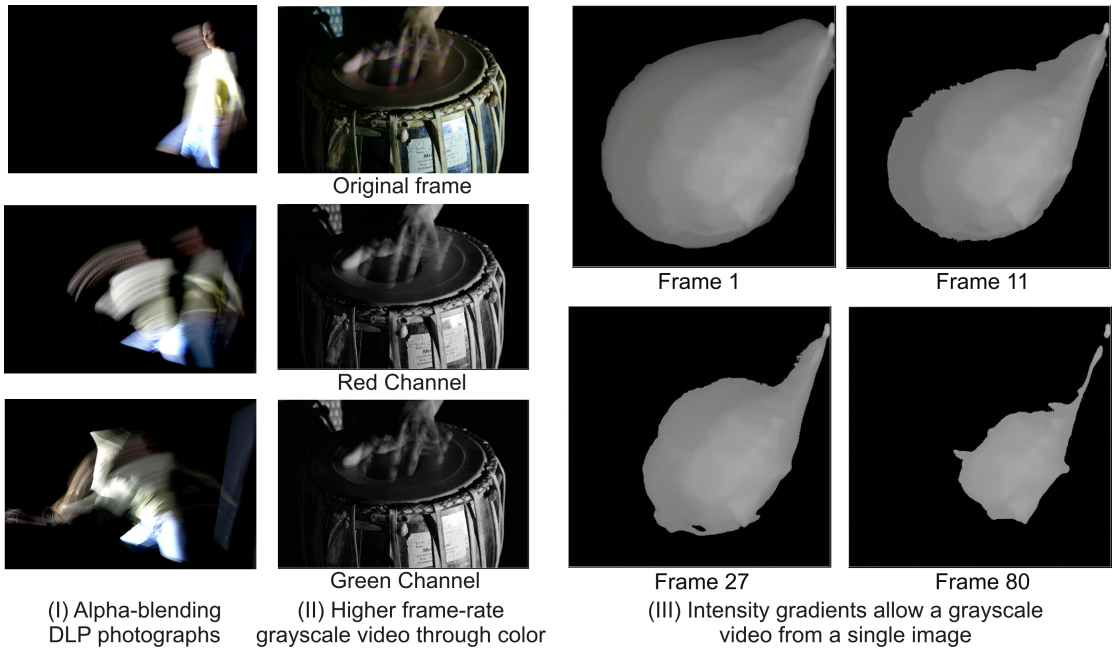



Figure 4.19: **Different types of motion illustrations:** We present three types of motion illustrations for DLP photographs. In (a), we differentially blend the strobed and non-strobed components. In (b) we triple the effective frame-rate by separating the RGB image components, exploiting the effect of the color wheel. Finally in (c), we apply intensity segmentation to convex objects (such as a balloon) to reconstruct the fast event.

slowly, at $10\%$ per frame. In contrast, the strobed component was blended quickly, at $50\%$ per frame. Since the strobed component naturally contains the fast moving parts of the scene, this gives the impression of motion.

**Color demultiplexing:** In Figure 4.19 (b) we show pictures of a tabla (hand drum) musician playing under DLP illumination. Due to the color-wheel effect, the different object copies are colored in a repeated series of red, green and blue. Each image can be demultiplexed into three grayscale images, tripling the frame-rate. Cycling the RGB copies gives the impression of motion only when the speed of the object is close to the frequency of the colorwheel ($120$Hz), resulting in fewer object copies and unlike the ballet photographs. This method works best when the objects in the scene are themselves close to grayscale: objects with significant red, green or blue components will imaged darkly or not at all in the complement illumination.

**Segmentation:** In Figure 4.19 (c) we create a video from an image from Figure 4.14 by thresholding the image intensities. Since the balloon parts that move first are replaced by the black background, these are least bright. Therefore the balloon shrinks from the outer edge inwards. This segmentation approach produces a believable result for a convex object, such as a balloon, since the image center is brighter than the outer edge.

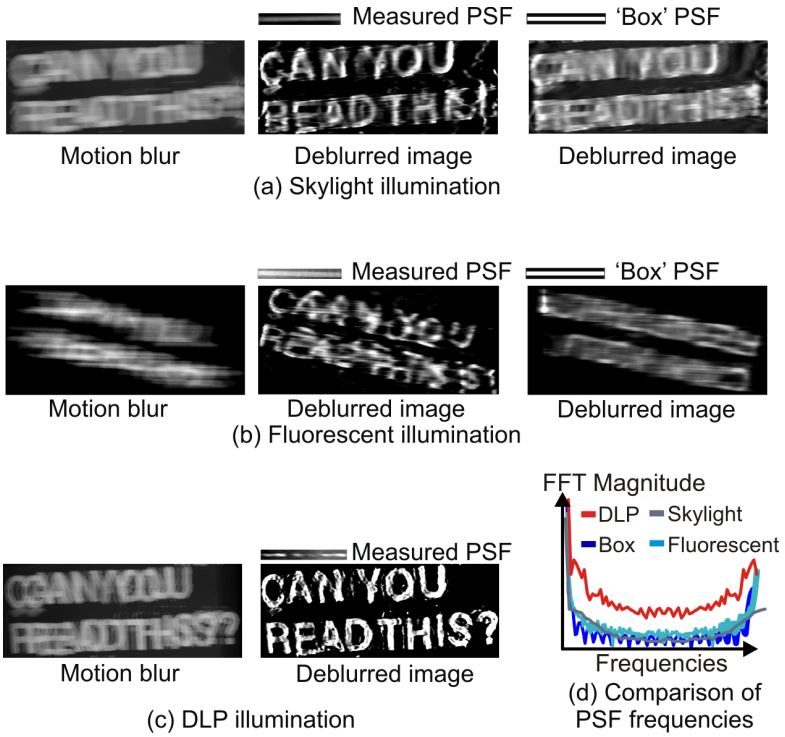



Figure 4.20: **DLP photographs contain higher frequencies compared to other types of illumination:** In (a) and (b) we show deconvolution results with skylight and fluorescent illumination. The blind deconvolution algorithm was given the intensity profile of a white dot as a starting point. The result for deblurring the same motion under DLP illumination (c) can be read easily. Analysis of frequencies in the recovered PSF shows DLP illumination preserves high frequency information.

### 4.5.4 Discussion: Towards Deblurring Photographs of Dynamic Scenes

In this section we conclude by analyzing the frequency space of images taken under DLP illumination, demonstrating the potential for deblurring scenes containing complex motion such as articulated and deformable objects. In Figure 4.20(a)-(c), we show images taken under DLP, skylight and fluorescent illumination. The object is a cardboard sheet translating from left to right with the PSF approximated by a small white dot placed on the sheet. We use this as a good starting point for blind deconvolution methods. For skylight and fluorescent light, we also tried the 'box' PSF which assumes constant incident illumination during exposure. Note that the best deconvolution occurs with the DLP photograph. In Figure 4.20(d) we show the frequencies of the PSFs. Note that the highest frequencies are due to the DLP illumination. Although previous work has either used camera apertures to create similar images ([77]) or shown some deblurring results ([61]), we are the first to analyze and compare the frequencies of DLP illumination to other types of lighting.

## 4.6 Chapter Summary and Discussion

In this chapter we demonstrated that illumination from DLP projectors shows flickering or dithering due to the effect of the DMD chip. This dithering occurs even when the projector projects a single slide, and is much faster (around $10^6$ Hz) than the slide refresh rate (around 60Hz). Therefore the dithering can be exploited to extend active vision techniques to dynamic scenes.

**Technical summary:** The Digital Micromirror Device (DMD) chip inside DLP projectors creates a perceived projected intensity through emitting binary patterns or pulses of light. These are integrated by human eye or by a low-frame rate camera. A higher frame rate camera can detect these pulses as a variation or flickering of the projected intensity. We have noticed that a frame-rate of at least 200Hz is required to detect the temporal dithering.

**Applications:** Most active lighting techniques require the scene to be static while the illumination varies. Our technique extends all these methods to dynamic scenes. We have shown this extension for structured light reconstruction, photometric stereo, illumination demultiplexing and separation of direct-indirect components.

**Input and Output:** The input to all our algorithms are images of a dynamic scene illuminated by one or more DLP projectors, viewed by a high-speed camera. Each projector projects only a single slide. The pattern projected depends on the technique: for example structured light techniques require a slide consisting of horizontal lines whose intensities are randomly selected. The photometric stereo techniques require blank slides of constant intensity. The output depends on the technique, and ranges from reconstructed shape to direct and global components of the scene.

**Alternative methods:** Human-computer interaction researchers exploited the DMD chip ([10]) to create imperceptible structured light for 'office of the future' settings. However, these algorithms ran at real-time rates. Our focus is on fast active-vision and we believe our technique is the only one offering as-is extension of the entire class of active lighting techniques to dynamic scenes.

**Implementation Issues and Discussion:** The selection of the pattern decides the frame-rate at which reconstruction can occur. This can easily be done using the calibration method of Figure 4.1. However, non-DMD factors such as projector heat, time-cycle, color-wheel start position etc seem to severely effect the calibration. Therefore, we suggest that the pattern be selected and used during or just previous to the experiment. Since we are dealing with active vision techniques, ambient light may cause issues with low dynamic range cameras. All our experiments were performed in the dark, but DLP projectors are bright enough to be detected in a variety of settings, such as underground, underwater and outdoor scenes at night. The most powerful DLP projectors for cinema can project over several meters, and therefore our techniques have application for a variety of long-range tasks. However, for applications requiring even longer ranges, passive techniques should be used instead.

**Camera frame-rates:** We note that the dithering can also be observed at lower frame rates and hence a lower cost camera may be used for slower reconstruc-

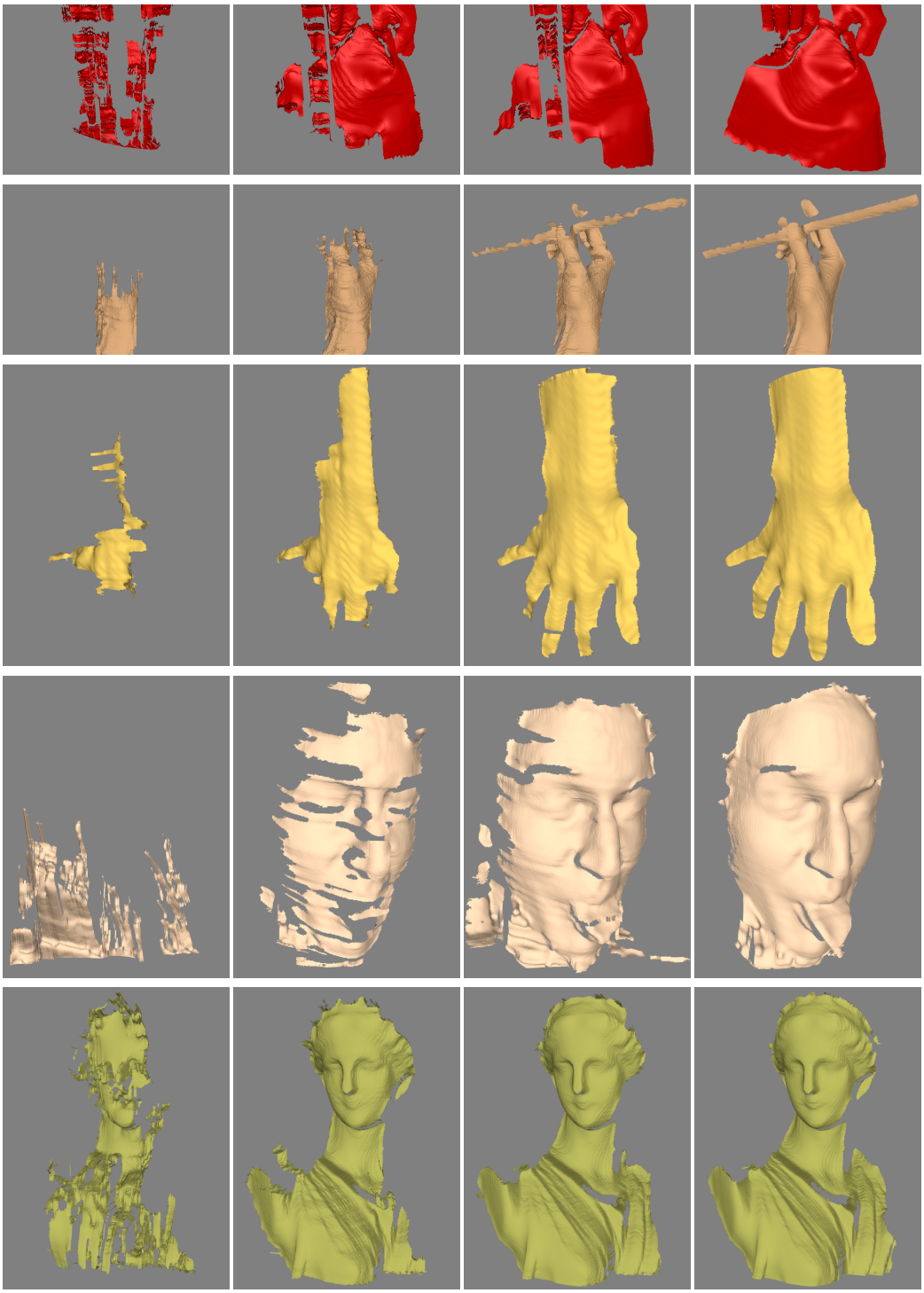



Figure 4.21: **Reconstructions obtained using videos captured at reduced frame rates.** Even at 500Hz, the quality of the reconstruction obtained remains acceptable indicating that temporal dithering can be exploited at this frame rate.

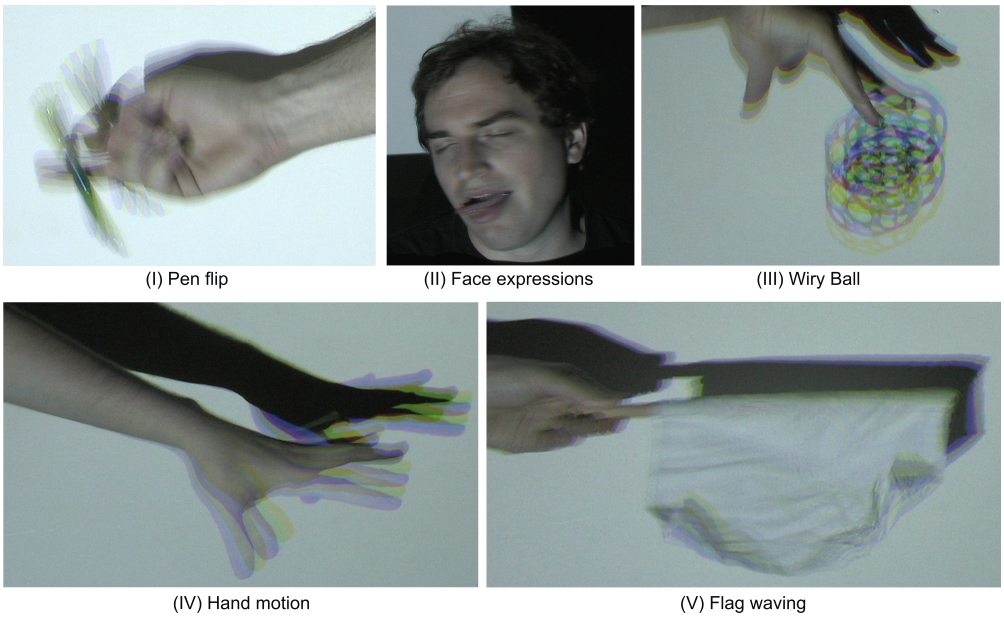



Figure 4.22: **Blurring when capturing fast moving scenes with a low frame-rate camera under DLP illumination:** Here we demonstrate why the reconstructions obtained require both a high-speed camera and a DLP projector. A 30Hz camera produces blur for all the scenes in this chapter, including the pen flip, face movement and hand-motion (reconstructed in Figure 4.7), flag waving (reconstructed in Figure 4.10) and wiry ball (demultiplexed in Figure 4.8).

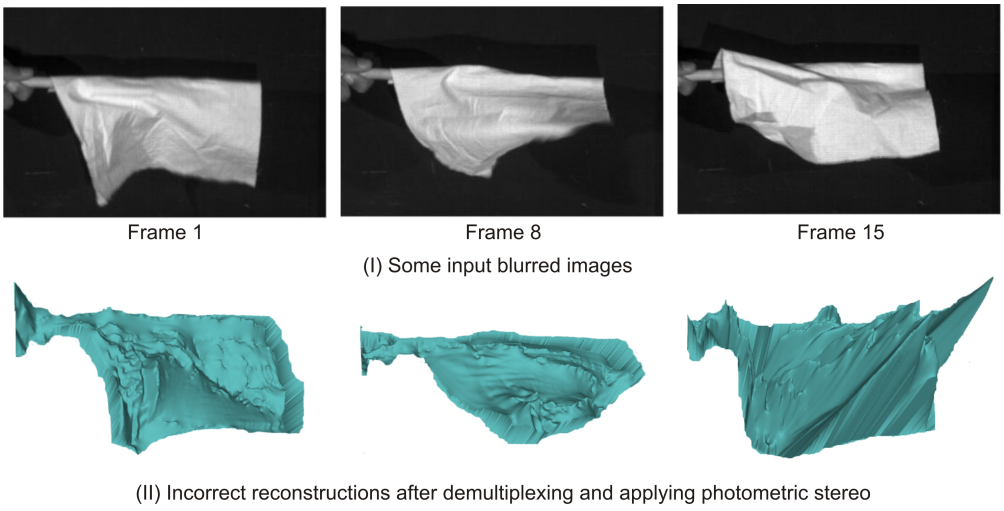



Figure 4.23: **Bad reconstruction at low frame-rate when applying photometric stereo to the flag example:** In (I) we show images of a cloth flag at 30Hz, created by integrating frames shown in Figure 4.10. Note the motion blur, especially at the edges of the cloth. The temporal dithering is almost destroyed by the low frame rate, and demultiplexing works badly, resulting in incorrect reconstructions as shown in (II).

tions. We repeated our experiments for the same scene at four lower frame rates — 1500Hz, 600Hz, 300Hz and 120Hz. Figure 4.21 shows the reconstructions obtained. The frame rate of 120Hz is too low to capture the required intensity variation and hence, the projector-camera pixel correspondences are unreliable. However, at 300Hz, the reconstruction quality is still acceptable indicating that the temporal dithering can be exploited even at this frame rate. This is extremely clear when we image the reconstructed scenes using a regular, 30Hz camera as in Figure 4.22. Note the blurring in the images makes impossible the application of methods that require triangulation, such as structured light based reconstruction. For pixel-based methods such as photometric stereo, the low frame-rate makes demultiplexing ill-conditioned and causes bad reconstruction, as demonstrated in Figure 4.23.

# Chapter 5

# Conclusions: Illuminating the Future

We began this thesis by discussing how recent advances in digital illumination have made active lighting techniques popular. Since these rely on finding incident light ray correspondences at scene points, they differ from traditional illumination-based approaches that model photometry. The problem of finding incident ray correspondence is analogous to discovering scene point correspondences in camera-based approaches, and is solved by projecting special patterns onto the scene. The key idea here is that spatio-temporal control can dominate photometry and allow easy detection of the incident rays.

We extend these ideas further to new situations. Rather than requiring programmable light-sources we instead exploited the motion, occlusion and strobing to obtain incident rays at a scene point. We demonstrated a variety of applications, from obtaining surface normal clusters and creating dual scene views to fast reconstruction of faces and summarizing dynamic events. In this chapter, we first discuss the continued influence that digital and configurable illumination will have on computational fields, especially computer vision. We then make predictions concerning the impact of the methods demonstrated in this thesis, with a focus on practical applications. Finally, we expand our ideas beyond modeling illumination by introducing the idea of ray geometry analogies.

## 5.1 The Confluence of Illumination and Computing

The impact of solid-state and MEMS technologies on the lighting industry has been tremendous. From being utilized in only a specialized set of applications, LEDs, LCDs and DMD are now widely used. The digital nature of these sources has made them easier to control by computer. Currently, computer vision algorithms have used these light-sources to program the direction, intensity and wavelength of emitted rays. However, in the future, all properties of light will be digitally manipulable by computer algorithms. Flexible materials may enable changing the light-source shape while mobile devices may allow controlling the light-source location. In addition, illuminating devices may connect to the local network and computers will communicate with them directly.

The implications for computer vision are even clearer when we compare the sheer number of light-sources versus cameras. Two factors ensure that there will always be more light-sources than cameras. Firstly, most cameras carry their own illumination (usually in the form of a flash) and the engines of growth in camera manufacture (such as cellphones and optical mice) contain either LCDs or LEDs. Secondly, humans require artificial illumination in all environments and, while a single camera can 'cover' a large area, intensity fall-off dictates that many light-sources be required to illuminate an identical space. Models of light-sources, therefore, should accompany any algorithm that wishes to interact with the real world.

## 5.2 New Directions for Active Lighting Applications

Here we discuss possible applications for the algorithms discussed in the thesis. Some of these are straightforward extensions or implementations of our methods while others may require further research:

**Hand-held Active Vision:** The methods in Chapter 2 suggest an interactive framework where humans wave a light-source in front of a scene to extract information. With the advent of hand-held devices, most of which have a light-source, great potential exists for algorithms that recover digital models of everyday objects. These would be shared by users across social networks. The challenges here involve a totally uncalibrated scenario, with low-intensity sources and low-quality cameras. In addition, the approaches in Chapter 4 relied on DLP projectors, and portable versions of these are already available. Extensions of our temporal dithering approach to a situation where the camera, scene and light-source are all moving should be simple. The goal here would be to enable fast and accurate optical flow using light-source strobing.

**Outdoor Scenes at Night:** Street lights and car headlamps are the dominant forms of illumination in the outdoors at night. Modeling these has implication for intelligent navigation and outdoor surveillance applications. The algorithms proposed in Chapter 3 suggest a one-time scan of static light-sources to recover their geometry. The fall-off along each ray could localize the observed pixels, especially when the illumination is changing over time and the scene has simple geometry (such as the flat surface of roads). Since street lights are mostly copies, this could done for one instance of the fixture.

**Tracking with Light Fixtures:** Most tracking algorithms in vision utilize illumination invariants. These tend to separate lighting information from images, and complement our techniques for modeling light-sources. We would analyze image data corresponding to illumination invariants (usually discarded by tracking algorithms) and extract information from it that could help the tracking problem further. For example, if a tracking algorithm is invariant to intensity fall-off, then the extent of fall-off over time could localize the object with respect to the light-source. This extra information can remove ambiguity due to occlusions or shadows where the object reemergence can be predicted.

**Relighting Indoor Scenes:** Images taken with indoor light-fixtures are interesting both due to the wide variety in illumination shape and the presence of complex near-field effects. A difficult problem is relighting such images, especially if they are obtained without calibration, such as from an internet collection. We believe that utilizing both a database of light-fixture geometries and by exploiting simple geometries in indoor scenes (such as walls) we can relight indoor scenes under arbitrary illumination. This has applications for visualizing architectural and indoor lighting designs.

**Deblurring Complex Motion:** The flutter-flash method explained in Chapter 4 suggests that fast encoding of illumination can be used for deblurring. The advantage of DLP dithering is that we can program the dithering individually for each pixel. This is important, since for articulated, deformable objects, the motion blur kernel is different at different pixels. We envision a system where multiple DLP projectors projecting random patterns illuminate a field-of-view. In the exposure time of the camera, at each pixel, many flutter-flashes will occur in different directions. Therefore, for every scene point, at least one direction has been 'tracked', preserving the high-frequency information.

**DLP Strobe Displays:** The interaction of DLP illumination with various objects that exhibit refraction and reflection allows an opportunity build novel displays. We can describe a family of DLP displays by varying one or more of the following factors: (1) Refractive element(s) (2) Location, color and reflectance of the projection screen (3) Output pattern. Using these displays, we expect to create new visual effects that blend colors and caustics, such as a DLP rainbow created by illuminating a prism. In addition, displays can be created when DLP illuminations interacts with objects showing periodic motion (such as a fan). These move fast enough to mix the colors from the color-wheel. For example, different speeds of the fan blend the primary colors from the DLP illumination to produce new colors.

**DLP Painting:** If the light from a projector is focused onto a white screen using a lens (or any refractive material) it produces caustics whose motion is controlled by the refractive element. Photographs of the screen correspond to the intersection of a 2D surface with the objects 5D caustic (space and incident angles). If this caustic is sufficiently complex, the image can be intricate and can have artistic value.

**DLP Videography:** The illumination sent out by a DLP projector is modulated temporally and spectrally. We can write the formation of the image $I(x, y)$ in this case as:

$$I(x, y) = \sum_{t=0}^{T} \lambda(x, y, t) D(x, y, t) \tag{5.1}$$

where $D$ encapsulates the temporally dithered scene radiance, $\lambda$ is a wavelength modulation term that represents the color wheel and $T$ is the exposure time. The captured image depends on the properties of the sensor or camera viewing the projected pattern. DLP videography is the multiplexing of content across wavelength or frame rate. A single slide from a DLP projector looks multicolored and static when viewed at $60$Hz with the naked eye. However, if viewed at higher frames rates (with a high-speed camera or shutter glasses) or through color filters, different pairs of $(T, D)$ or $(T, \lambda)$ can convey distinct content. While in Section 4.3.5 we showed an application of such a DLP video for separating direct and global components, the pattern was found manually. Here, the goal is to design the pattern given a task, automatically.

## 5.3 Beyond Illumination: Modeling Biological Sensors

In this thesis, we have discussed the similarity of light-sources and cameras from a ray geometric perspective and demonstrated the opportunity for novel results. Now, let us consider the *differences* between light-sources and cameras. In particular, unlike cameras, many types of illumination exist: light-sources are manufactured in different shapes and sizes, and advances in materials have enabled flexible, manipulable sources. Despite the ray geometric duality between light-sources and cameras, building a physical, analogous camera to an exotic light-source is difficult. For example, a neon sign has 3D curves of light, and building a dual camera with CCDs in a similar complex arrangement would require special electronics which may simply not exist.

In fact, building any type of complex camera is hard, including those that imitate biological sensors. For example, considerable effort has been required to build a single insect-like compound eye ([41]). However, such attempts have many disadvantages, such as high cost and reduced resolution, and most of these efforts are still in a very nascent stage of development. In contrast, the vast variety of illumination is ubiquitous and cheap, allowing even very complex light-source shapes to be easily available. By exploiting ray geometry, we can design light-sources that are duals of complex cameras, such as biological sensors. It then becomes possible to propose dual algorithms for complex cameras and sensors, *even though building these devices is currently impossible.*

We believe that finding ways to apply computer vision and graphics algorithms onto these camera analogies is a worthwhile endeavor. The vast variety in illumination shape, as well as in the types of biological sensors, give a sense of the power contained in exploiting the duality of light-sources and camera. From this large set we have, in this thesis, contributed to understanding a light-source's motion, geometry and strobing. We believe this is a significant step towards applying ray geometry to a wider range of sources and that it will have significant impact on computer vision, computer graphics and related fields.

# Bibliography


[1] I. Ashdown. Modelling complex 3D light sources. *SIGGRAPH*, 1993. 75

[2] R. Bolles, H. Baker, and D. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *IJCV*, 1987. 79

[3] J. Bouguet and P. Perona. 3d photography on your desk. *ICCV*, 1998. 8, 61

[4] Y. Caspi and M. Werman. Vertical parallax from moving shadows. *CVPR*, 2006. 8

[5] Q. Chen and T. Wada. A light modulation/demodulation method for real-time 3d imaging. *3D digital imaging and modeling*, 2005. 96

[6] S. Chen and Y. Li. Self-recalibration of a colour-encoded light system for automated 3-d measurements. *MeasureSciTech*, 2003. 8

[7] J. J. Clark. Active photometric stereo. *CVPR*, 1992. 7

[8] J. J. Clark. Photometric stereo with nearby planar distributed illuminants. *CRV*, 2006. 7

[9] E. Coleman and R. Jain. Obtaining 3-dimensional shape of textured and specular surfaces using four-source photometry. *ICGIP*, 1982. 7

[10] D. Cotting, M. Naef, M. Gross, and H. Fuchs. Embedding imperceptible patterns into projected images for simultaneous acquisition and display. *ISMAR*, 2004. 83, 107

[11] B. Curless and M. Levoy. Better optical triangulation through spacetime analysis. *ICCV*, 1995. 8, 85

[12] K. Dana, B. Ginnekan, S. Nayar, and J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 1999. 50

[13] K. J. Dana, B. V. Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real world surfaces. *CVPR*, 1997. 28

[14] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo : A unifying framework for depth from triangulation. *CVPR*, 2003. 8

[15] J. Davis, D. Nehab, R. Ramamoothi, and S. Rusinkiewicz. Spacetime stereo : A unifying framework for depth from triangulation. *CVPR*, 2003. 8

[16] J. DeYoung and A. Fournier. Properties of tabulated bidirectional reflectance distribution. *Graphics Interface*, 1997. 17

[17] D. Dudley, W. Duncan, and J. Slaughter. Emerging digital micromirror device (dmd) applications. *SPIE*, 2003. 8, 81, 96

[18] H. Edgerton. Strobe photography: A brief history. *Optical engineering*, 1984. 98

[19] A. A. Efros and T. Leung. Texture synthesis by non-parametric sampling. *ICCV*, 1999. 53

[20] W. Flemming. Functions of several variables. 1977. 47, 48

[21] A. Fournier. Separating reflection functions for linear radiosity. *Eurographics Workshop on Rendering*, 1995. 17

[22] W. Freeman and H. Zhang. Shape-time photography. *CVPR*, 2003. 102, 103, 104

[23] G. Garg, E-V. Talvala, M. Levoy, and H. P. A. Lensch. Symmetric photography: Exploiting data-sparseness in reflectance fields. *EGSR*, 2006. 8

[24] M. Goesele, X. Granier, W. Heidrich, and H. Seidel. Accurate light source acquisition and rendering. *SIGGRAPH*, 2003. 75

[25] M. Goesele, H. P. A. Lensch, J. Lang, C. Fuchs, and H. Seidel. Acquisition of translucent objects. *SIGGRAPH*, 2004. 36, 37, 55

[26] D. Goldman, B. Curless, A. Hertzmann, and S. Seitz. Shape and spatially-varying brdfs from photometric stereo. *ICCV*, 2005. 7

[27] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The lumigraph. *SIGGRAPH*, 1996. 8

[28] M. D. Grossberg and S. K. Nayar. A general imaging model and a method for finding its parameters. *ICCV*, 2001. 3, 62

[29] M. Gupta, Y. Tian, S. Narasimhan, and L. Zhang. De (focusing) on global light transport for active scene recovery. *CVPR*, 2009. 8

[30] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. *Cambridge University Press*, 2004. 7

[31] G. Healey and L. Z. Wang. Segmenting surface shape using colored illumination. *SCIA*, 1997. 58

[32] H. V. Helmholtz. Treatise on physiological optics. *Dover*, 1925. 8

[33] A. Hertzmann and S. Seitz. Shape and materials by example: a photometric stereo approach. *CVPR*, 2003. 58

[34] A. Hertzmann and S. M. Seitz. Shape and materials by example: A photometric stereo approach. *CVPR*, 2003. 7, 8, 83

[35] D. Huynh, R. Owens, and P. Hartmann. Calibrating a structured light stripe system: A novel approach. *IJCV*, 1999. 8

[36] Searchlight's in World War 2 (Wikipedia: The Free Encyclopaedia). http://en.wikipedia.org/wiki/searchlight#in_world_war_ii. 2009. 2

[37] Y. Iwahori, R. Woodham, H. Tanaka, and N. Ishii. Moving point light source photometric stereo. *IEICE*, 1994. 7

[38] T. Jarvenpaa. Measuring color breakup of stationary images in field-sequential-color displays. *Journal of the Society for Information Display*, 2005. 96

[39] C. Je, S. Lee, and R. Park. High-contrast color-stripe pattern for rapid structured-light range imaging. *ECCV*, 2004. 8

[40] W. H. Jensen, R. S. Marschner, M. Levoy, and P. Hanrahan. A practical model for subsurface light transport. *SIGGRAPH*, 2001. 45, 55

[41] K. Jeong, J. Kim, and L. Lee. Biologically inspired artificial compound eyes. *Science*, 2006. 115

[42] A. Jones, I. McDowall, H. Yamada, M. Bolas, and P. Debevec. Rendering for an interactive 360 degree light field display. In *SIGGRAPH*, 2007. 8

[43] J. Kautz and M. D. McCool. Interactive rendering with arbitrary brdfs using separable approximations. *Eurographics Workshop on Rendering*, 1999. 17

[44] G. Kayafas and E. Jussim. Stopping time: The photographs of harold edgerton. *Abrams*, 2000. 96

[45] S. Kima, T. Shibataa, T. Kawaia, and K. Ukaib. Ergonomic evaluation of a field-sequential colour projection system. *Displays*, 2007. 96

[46] G.J. Klinker, S.A. Shafer, and T. Kanade. A physical approach to color image understanding. *IJCV*, 1990. 7

[47] S. J. Koppal and S. G. Narasimhan. Novel depth cues video. *http://www.cs.cmu.edu/~koppal/depth_cues.mov*, 2007. 31, 43

[48] M. Koudelka, S. Magda, P. Belhumeur, and D. Kriegman. Acquisition, compression, and synthesis of bidirectional texture functions. *Texture*, 2003. 45, 51, 52, 54, 55

[49] M. Langer and S. Zucker. What is a light source? *CVPR*, 1997. 3, 62

[50] M. S. Langer, G. Dudek, and S. W. Zucker. Space occupancy using multiple shadowimages. *IROS*, 2005. 8

[51] D. Lanman, R. Raskar, A. Agrawal, and G. Taubin. Shield fields: modeling and capturing 3d occluders. *SIGGRAPH Asia*, 2008. 8

[52] J. Lee, S. Hudson, J. Summet, and P. Dietz. Moveable interactive projected displays using projector based tracking. *Symposium on User Interface Software and Technology*, 2005. 96

[53] M. Levoy and P. Hanrahan. Light field rendering. *SIGGRAPH*, 1996. 8

[54] S. Magda, T. Zickler, D. Kriegman, and P. Belhumeur. Beyond lambert: reconstructing surfaces with arbitrary brdfs. *ICCV*, 2001. 7

[55] S. Mallick, T. Zickler, D. Kriegman, and P. Belhumeur. Beyond lambert: Reconstructing specular surfaces using color. *ICCV*, 2005. 7

[56] S. Marschner, S. Westin, E. Lafortune, K. Torrance, and D. Greenberg. Image-based brdf measurement including human skin. *Eurographics Workshop on Rendering*, 1999. 7

[57] W. Matusik, H. Pfister, M. Brand, and L. McMillan. A data-driven reflectance model. *TOG*, 2003. 14, 16

[58] I. McDowall and M. Bolas. Fast light for display, sensing and control applications. In *IEEE VR Workshop on Emerging Display Technologies*, 2005. 8

[59] M. Mori, T. Hatada, K. Ishikawa, T. Saishoji, O. Wada, J. Nakamura, and N. Terashima. Mechanism of color breakup in field-sequencial-color projectors. *Journal of the Society for Information Display*, 1999. 96

[60] E. Muybridge, H. Edgerton, and J. Shaw. Time/motion. *Dewi Lewis Publishing*, 2004. 96

[61] S. G. Narasimhan, S. J. Koppal, and S. Yamazaki. Temporal dithering of illumination for fast active vision. *ECCV*, 2008. 99, 106

[62] S. G. Narasimhan, V. Ramesh, and S. K. Nayar. A class of photometric invariants: Separating material from shape and illumination. *ICCV*, 2003. 17

[63] S. G. Narasimhan, C. Wang, and S. K. Nayar. All the images of an outdoor scene. *ECCV*, 2002. 28, 58

[64] S. K. Nayar, V. Branzoi, and T. Boult. Programmable imaging using a digital micromirror array. *CVPR*, 2004. 8

[65] S.K. Nayar, G. Krishnan, M. D. Grossberg, and R. Raskar. Fast separation of direct and global components of a scene using high frequency illumination. *SIGGRAPH*, 2006. 8, 78, 79, 83, 92, 94, 95

[66] R. Ng. Fourier slice photography. *SIGGRAPH*, 2005. 8

[67] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis. Geometric considerations and nomenclature for reflectance. *Monograph 161, National Bureau of Standards (US)*, 1977. 55

[68] K. Ishikawa O. Wada, J. Nakamura and T. Hatada. Analysis of color breakup in field-sequential color projection system for large area displays. *Display Workshops*, 1999. 96

[69] Lighthouse of Alexandria (Wikipedia: The Free Encyclopaedia). http://en.wikipedia.org/wiki/lighthouse_of_alexandria. 2009. 2

[70] Theories of Light (Wikipedia: The Free Encyclopaedia). http://en.wikipedia.org/wiki/light#theories_about_light. 2009. 2

[71] M. Ogata, K. Ukai, and T. Kawai. Visual fatigue in congenital nystagmus caused by viewing images of color sequential projectors. *Display Technology*, 2005. 96

[72] T. Okabe and Y. Sato. Effects of image segmentation for approximating object appearance under near lighting. *ACCV*, 2006. 7

[73] M. Oren and S. K. Nayar. Generalization of the lambertian model and implications for machine vision. *IJCV*, 1995. 7

[74] O. Packer, L. Diller, J. Verweij, B. Lee, J. Pokorny, D. Williams, D. Dacey, and D. H. Brainard. Characterization and use of a digital light projector for vision research. *Vision Research*, 2001. 96

[75] M. Pharr and G. Humphreys. Physically based rendering: From theory to implementation. *Elsevier*, 2004. 13, 35, 37, 40, 41

[76] R. Ramamoorthi and P. Hanrahan. A signal-processing framework for inverse rendering. *SIGGRAPH*, 2001. 7

[77] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: Motion deblurring using fluttered shutter. *SIGGRAPH*, 2006. 8, 83, 106

[78] R. Raskar, K. Tan, R. Feris, J. Yu, and M. Turk. Non-photorealistic camera: Depth edge detection and stylized rendering using multi-flash imaging. *SIGGRAPH*, 2004. 83

[79] D. Raviv, Y. Pao, and K. Loparo. Reconstruction of three-dimensional surfaces from two-dimensional binary images. *Transactions on Robotics and Automation*, 1989. 8

[80] S. Reeves and R. Mersereau. Blur identification by the method of generalized cross-validation. *Transactions on image processing*, 1992. 101

[81] A. Rosenfeld and A. Kak. Digital image processing. *New York Academic*, 1982. 101

[82] S. Rusinkiewicz. A new change of variables for efficient brdf representation. *Eurographics Workshop on Rendering*, 1998. 17

[83] D. Kriegman S. Magda, T. Zickler and P. Belhumeur. Beyond lambert: Reconstructing surfaces with arbitrary brdfs. *ICCV*, 2001. 7

[84] G. Salton. Automatic text processing: The transformation, analysis, and retrieval of information by computer. 1989. 22

[85] I. Sato, T. Okabe, Q. Yu, and Y. Sato. Shape reconstruction based on similarity in radiance changes under varying illumination. *ICCV*, 2007. 58

[86] Y. Sato, H. Kitagawa, and H. Fujita. Shape measurements of curved objects using multiple slit-ray projections. *PAMI*, 1982. 8

[87] Y. Sato, M. D. Wheeler, and K. Ikeuchi. Object shape and reflectance modeling from observation. *SIGGRAPH*, 1997. 7

[88] S. Savarese, M. Andreetto, H. Rushmeier, and F. Bernardini. 3D reconstruction by shadow carving: Theory and practical evaluation. *IJCV*, 2005. 8

[89] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *CVPR*, 2003. 8

[90] Y. Y. Schechner, S. K. Nayar, and P. N. Belhumeur. A theory of multiplexed illumination. *ICCV*, 2003. 83, 90

[91] S. Seitz. The space of all stereo images. *ICCV*, 2001. 8

[92] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. *CVPR*, 2006. 7

[93] S. Seitz, Y. Matsushita, and K. Kutulakos. A theory of inverse light transport. *ICCV*, 2005. 8

[94] P. Sen, B. Chen, G. Garg, S. R. Marschner, M. Horowitz, M. Levoy, and H. P. A. Lensch. Dual photography. *SIGGRAPH*, 2005. 8, 61, 62, 77, 78

[95] A. Shashua. On photometric issues in 3D visual recognition from a single 2D image. *IJCV*, 1997. 7

[96] H. Shum, S. Chan, and S. B. Kang. Image-based rendering. *Springer*, 2007. 62

[97] K. Sunkavalli, W. Matusik, H. Pfister, and S. Rusinkiewicz. Factored time-lapse video. *Siggraph*, 2007. 58

[98] R. Swaminathan, M. Grossberg, and S. K. Nayar. Caustics of catadioptric cameras. *ICCV*, 2001. 62, 71

[99] O. Takahiro and S. Yoichi. Does a nearby point light source resolve the ambiguity of shape recovery in uncalibrated photometric stereo? *IPSJ*, 2007. 7

[100] W. H. F. Talbot. Note on instantaneous photographic images. *Philosophical magazine*, 1851. 98

[101] K.E. Torrance and E.M. Sparrow. Theory for off-specular reflection from roughened surfaces. *JOSA*, 1967. 7

[102] E. Umezawa, T. Shibata, T. Kawai, and K. Ukai. Ergonomic evaluation of the projector using color-sequential display system. *45th Annual Congress of the Japan Ergonomics Assoc.*, 2004. 96

[103] J. Unger, A. Wenger, T. Hawkins, A. Gardner, and P. Debevec. Capturing and rendering with incident light fields. *EGSR*, 2003. 13, 61

[104] V. Vaish, R. Szeliski, C. L. Zitnick, S. B. Kang, and Marc Levoy. Reconstructing occluded surfaces using synthetic apertures: Stereo, focus and robust measures. *CVPR*, 2006. 8

[105] D. Vaquero, R. Feris, M. Turk, and R. Raskar. Characterizing the shadow space of camera-light pairs. *CVPR*, 2008. 8

[106] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and Richard Baraniuk. An architecture for compressive imaging. *ICIP*, 2006. 8

[107] M. Waugh. Liquid sculptures. *www.liquidsculpture.com*, 2004. 96

[108] E. W. Weisstein. Taylor series. *MathWorld–A Wolfram Web Resource*, 2006. 47

[109] A. Wenger, A. Gardner, C. Tchou, J. Unger, T. Hawkins, and P. Debevec. Performance relighting and reflectance transformation with time-multiplexed illumination. *SIGGRAPH*, 2005. 8

[110] P. M. Will and K. S. Pennington. Grid coding: A preprocessing technique for robot and machine vision. *AI*, 2, 1971. 8, 85

[111] R. J. Woodham. Photometric stereo. *MIT AI Memo*, 1978. 7

[112] R. J. Woodham, Y. Iwahori, and R. A. Barman. Photometric stereo: Lambertian reflectance and light sources with unknown direction and strength. *UBC Technical Report TR9118*, 1991. 91

[113] A. M. Worthington. A study of splashes. *Longman green*, 1908. 98

[114] S. Yamazaki, S. Narasimhan, S. Bakar, and T. Kanade. Coplanar shadowgrams for acquiring visual hulls of intricate objects. *ICCV*, 2007. 8

[115] Y. You and M. Kaveh. A regularization approach to joint blur identification and image restoration. *Transactions on image processing*, 1996. 101

[116] M. Young, E. Beeson, J. Davis, S. Rusinkiewicz, and R. Ramamoorthi. Viewpoint-coded structured light. *CVPR*, 2007. 8

[117] J. Yu and L. McMillan. General linear cameras. *ECCV*, 2004. 8

[118] C. Zhang and T. Chen. A self-reconfigurable camera array. *EGSR*, 2004. 8

[119] L. Zhang, B. Curless, and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. *3DPVT*, 2002. 8

[120] L. Zhang and S. K. Nayar. Projection defocus analysis for scene capture and image display. *SIGGRAPH*, 2006. 8

[121] T. Zickler, P. Belhumeur, and D. J. Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. *ECCV*, 2002. 8

[122] C. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *SIGGRAPH*, 2004. 62

[123] A. Zomet, D. Feldman, and S. Peleg. Mosaicing new views: The crossed-slits projection. *PAMI*, 2003. 8, 62