Towards Casual Appearance Capture by Reflectance Symmetry

Wu Zhe

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Department of Electrical and Computer Engineering

National University of Singapore

2014

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

13 Aug 2014

Wu Zhe

13 August 2014

Acknowledgements

First of all, I would like to thank my PhD advisor, associate professor Tan Ping, without whose support and help, this thesis would not have been possible. I would also like to thank my collaborators and anonymous reviewers, who provided valuable comments and advice on my research projects and papers.

I also give my thanks to my parents, who has been supporting and encouraging me all the time.

Many people has generously helped me during my study in NUS, I am really grateful to their kindness, comments, suggestions and encouragement.

Finally, I would like to give special thanks to Kong Chao, who had accompanied me throughout many hard times.

Contents

Summary

The visual appearance of a real-world scene or an object depends on shape, surface reflectance properties, illumination, and the viewpoint from which we are looking. Given images of a scene or object under different lighting and viewing conditions, it is possible to reverse the image formation process so as to infer shape and/or reflectance, the key of which lies in an appropriate reflectance model. This thesis studies the problem of appearance capture, which consists of shape reconstruction and reflectance estimation, by utilizing several reflectance symmetry properties found in real-world materials.

First, the classical auto-calibration problem in photometric stereo is studied. Under unknown directional lighting, the uncalibrated Lambertian photometric stereo algorithm recovers the shape of a smooth surface up to the generalized bas-relief(GBR) ambiguity. As is shown in this thesis, the GBR ambiguity will destroy a low-rank structure of a 2D bidirectional reflectance distribution function(BRDF) slice, which is implied by half-vector symmetry. Thus, an algorithm can be proposed based on this theoretical result to solve the GBR ambiguity, hence, recover the shape.

Second, isotropy of BRDF induces 'iso-depth contours' in photometric stereo. An appearance capture method is proposed by combining these

iso-depth contours and sparse 3D points from multi-view stereo. Depth information is propagated along iso-depth contours to generate new 3D points. Through iteration of the process, an accurate 3D shape is reconstructed which on the other hand, enables accurate reflectance estimation. This method achieves high accuracy while maintaining generality.

Third, a handheld RGBD-M sensor, which is capable of materials sensing, is presented. An RGBD sensor is only able to capture color and depth information. By enhancing it with additional hardware and accompany it with suitable algorithms, this sensor is able to produce both shape and reflectance. Compared with traditional appearance capture system, this sensor features portability and usability, which enables non-professionals to acquire shape and reflectance in their daily life. To our knowledge, this sensor is the first one of its kind.

List of Figures

LIST OF FIGURES

Chapter 1

Introduction

The visual appearance of a real-world scene or an object is affected by several different factors: shape, illumination, surface reflectance properties and the viewpoint from which we are looking. Of all these factors, only shape and reflectance are intrinsic properties of the scene or object and cannot be easily altered. The aim of appearance capture is to recover either or both of the two properties.

Shape reconstruction has long been one of the most important research topics in the computer vision community. Based on the underlying principles, various vision-based shape reconstruction methods can be roughly divided into three categories: stereo, structure from motion and photometric stereo. Stereo methods [Seitz et al., 2006] reconstruct 3D point clouds by finding corresponding image points of two pre-calibrated cameras and doing triangulation with these points. A variation of stereo methods is structured light methods, in which case one of the two cameras is replaced with a projector. An important structured light system is Microsoft Kinect, which is a depth sensor of consumer level. The second category is structure from motion(SfM) [Hartley and Zisserman, 2003]. While SfM is also based on point correspondences of different images just like stereo, the relative poses between cameras, even the intrinsic parameters of cameras, are unknown. Thus, SfM reconstructs 3D point clouds and camera parameters simultaneously. Obviously, more information is required in SfM than in stereo. The third category of methods is photometric stereo [Woodham, 1980]. Being completely different from the previous two kinds of methods, neither does photometric stereo require image correspondences, since the camera is typically static, nor does it recover 3D points directly. Photometric stereo seeks to estimate the correct orientation of each surface point by analysing the intensity variations of its image projection under different lighting conditions. Given surface normals by photometric stereo methods, the 3D surface can be eventually reconstructed through 'integration' of the normal field. Compared with stereo and SfM, photometric stereo is able to recover fine details of a surface. Another advantage of photometric stereo is that it is strongly associated with surface reflectance. Thus it can be naturally combined with reflectance estimation.

Similar to shape reconstruction, reflectance estimation has also attracted much attention in the research community. The reflectance of a surface point is described by the so-called bidirectional reflectance distribution function(BRDF), which is a function of the incoming light direction and outgoing light direction in a local coordinate system. Instead of measuring this function directly using a complicated device called gonioreflectometer, vision methods seek to recover this function through images. Some methods assume a parametric model such as Lambertian, Ward [Ward, 1992] or Cook-Torrence model [Cook and Torrance, 1982], and fit the model to measured BRDF values. Another kind of methods approach this problem in a data-driven way by modelling an arbitrary BRDF as a linear combination of a set of real-world basis BRDFs. Recently some researchers explored controlled environment lighting in reflectance estimation [Tunwattanapong et al., 2013], which is proved to significantly reduce the number of images required by other methods. Given BRDFs estimated from a real-world scene, it is possible to render photo-realistic images for film production, gaming or virtual reality applications.

While it is possible to reconstruct 3D shape or to estimate reflectance alone, the two problems are not independent of each other. On the one hand, a reasonable reflectance model can help photometric stereo achieve high accuracy in recovering geometric details of a surface. On the other hand, a relatively accurate shape is a necessity for estimating reflectance. It is worth noting that most BRDF estimation methods assume shape is known a priori and typically, the shape is a simple planar surface. In view of the close relationship between shape and reflectance acquisition, it is a natural idea to acquire the two simultaneously, which is also desirable in applications.

In recent years, quite a few methods have been proposed for appearance capture and have produced plausible results. However, there are two problems overlooked by most of them, and will be covered in detail in this thesis.

The first problem is on choice of BRDF models. Many appearance capture methods, which combine photometric stereo and BRDF estimation, adopt a parametric BRDF model, and try to find a good combination of normals and BRDF parameters that fit input images well. While they can produce reasonably good results in some cases, their performance degrades as the real BRDF deviates from the assumed model. To achieve high accuracy while maintaining generality, reflectance symmetry is a good alternative assumption on BRDF. Compared with a parametric model, reflectance symmetry poses much weaker, yet effective constraints on BRDF. Besides, reflectance symmetry has been ubiquitously observed among real-world materials. Reflectance symmetry has seen quite a few applications in photometric stereo and achieved reasonably good results [Alldrin and Kriegman, 2007; Tan et al., 2011; Zickler et al., 2002]. Thus, its application in appearance capture is promising.

The second problem is on hardware setup and data acquisition. Traditionally, the main focus of appearance capture is accuracy. This is because its primary application is photo-realistic rendering in film production. To achieve highly accurate results, most appearance capture systems consisted of sophisticated hardware setups and involved tedious work in data acquisition. On the other hand, thanks to the fast evolution of 3D printing as well as BRDF fabrication, we can envision a demand for digital models in the consumer-level market in near future. Non-professionals are eager to capture shape and reflectance in their daily life. In addition to fabrication based on digital models, consumers could also use shape and reflectance as ‘keywords’ to search for similar products online. There is no doubt that the acquired digital model, especially the reflectance part, are better features than images [Gu and Liu, 2012; Shiradkar et al., 2014]. A key feature in the consumer-level market is that, instead of high accuracy, portability and usability become the new concern. Although the demand is urgent, none of the available portable 3D scanners on the market is capable of reflectance capture. A sensor capable of estimating both shape and reflectance will perfectly fill the gap between the market’s need and the research community.

This thesis focuses on three problems. In Chapter 3, the traditional auto-calibration problem in photometric stereo is approached. Classical methods in uncalibrated Lambertian photometric stereo recover the surface normals up to certain ambiguity. In Chapter 3, a novel constraint based on reflectance symmetry is proposed and proved to be able to fully resolve the ambiguity. A robust auto-calibration algorithm is also presented and validated by experiments.

In Chapter 4, a method for simultaneous capture of shape and reflectance is proposed. The symmetry of isotropy in BRDF enables identification of ‘iso-depth contours’. By iteratively propagating depth information of a sparse set of 3D points along these contours in multiple viewpoints, we can obtain an accurate shape model, which, in combination with input photometric images, can be used in BRDF estimation. This system features high accuracy without loss of generality.

In Chapter 5, a handheld RGBD-M sensor is presented. This sensor is capable of not only texture and depth sensing, but also material sensing based on a simplified BRDF model derived from reflectance symmetry. Built upon recent advances in camera tracking and dense scene reconstruction, it is both portable and easy to use for ordinary people. This sensor is the first one of its kind to our knowledge.

1. INTRODUCTION

Chapter 2

Background

Not only shape and reflectance, which are properties of objects or scenes, but also light sources and the imaging camera will affect how an image looks like. In this chapter, we review different models for them which are most related to this thesis. In addition, we present the image formation model which will also help the reader better appreciate the rest part of the thesis.

2.1 Shape Models

2.1.1 Height Map

The simplest shape model is the height map. A height map is a function defined on the image plane, assigning a scalar value to each pixel. This assigned value is a distance of the corresponding 3D point to a reference plane which is parallel to the image plane. Figure 2.1 illustrates this concept by showing the height map of a hemisphere. On the left side of Figure 2.1 is the shape of a hemisphere. In the middle is the color-coded height map, with red indicating larger values and blue referring to smaller values.

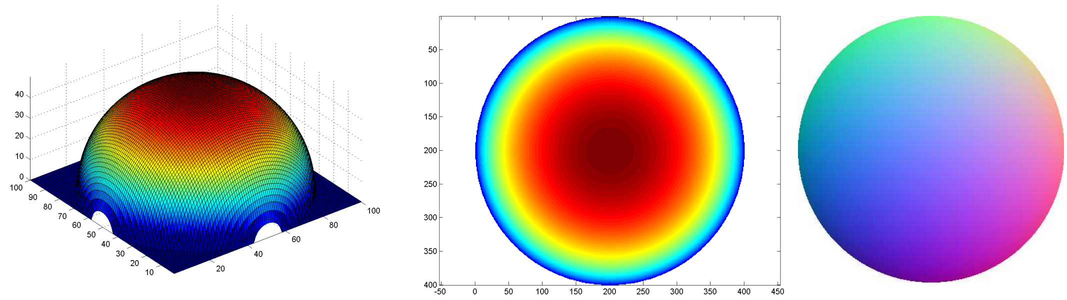

Figure 2.1: From left to right: a hemisphere, its height map, its normal map.

From the height map it is easy to calculate the partial derivatives as the height difference between two neighbouring pixels,

$$\begin{aligned}\frac{dh}{dx} &= \frac{h(x+1)-h(x)}{(x+1)-x} = h(x+1)-h(x), \\ \frac{dh}{dy} &= \frac{h(y+1)-h(y)}{(y+1)-y} = h(y+1)-h(y).\end{aligned} \tag{2.1.1}$$

Closely related to the first-order partial derivatives is the surface normal. The normal at a point on a smooth surface is calculated from the partial derivatives at the point

$$\mathbf{n} = \frac{(-\frac{dh}{dx}, -\frac{dh}{dy}, 1)^T}{\sqrt{(\frac{dh}{dx})^2 + (\frac{dh}{dy})^2 + 1}}. \tag{2.1.2}$$

The color-coded normal map for a hemisphere is shown on the right side of Figure 2.1. The RGB channels store the three components of a normal respectively in the following way

$$R = \frac{n_x+1}{2}, G = \frac{n_y+1}{2}, B = \frac{n_z+1}{2}. \tag{2.1.3}$$

Given the normal at every pixel on an image, it is possible to recover a smooth surface whose normals resemble, if not equal, the provided ones by [Horn, 1970].

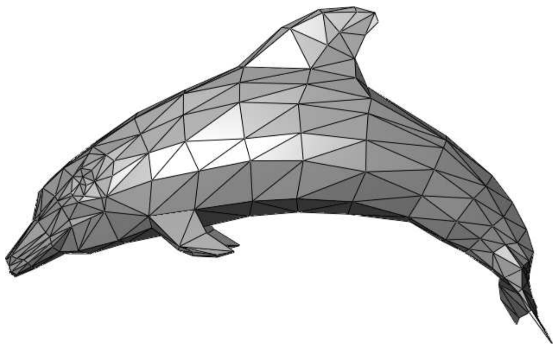

Figure 2.2: A triangle mesh from Wiki.

2.1.2 Triangle Mesh

Another shape model is the triangle mesh, which is widely used in computer graphics. A triangle mesh is a type of the polygon mesh and comprises a set of triangles that are connected by their common edges or corners. An example of a triangle mesh is shown in Figure 2.2. Each vertex in a triangle mesh is associated with a normal. The vertex normal is typically calculated by averaging the plane normals of its neighbouring triangles. The normal at a point within a triangle is usually interpolated from that of the three vertices of this triangle. A natural interpolation strategy in this case is barycentric interpolation. Assume the 3D point P is represented as $\alpha V_1 + \beta V_2 + \gamma V_3$, where $\alpha + \beta + \gamma = 1$, then the interpolated normal for the point P is

$$\mathbf{n}_P = \frac{\alpha \mathbf{n}_1 + \beta \mathbf{n}_2 + \gamma \mathbf{n}_3}{\sqrt{\alpha \mathbf{n}_1 + \beta \mathbf{n}_2 + \gamma \mathbf{n}_3}}, \tag{2.1.4}$$

where $\mathbf{n}_i$ are the normals at corresponding vertices.

Compared with the height map, the triangle mesh can be used to represent a much more complicated object or scene.

2.1.3 Volumetric Representation

The third shape model is a volumetric model used in [Curless and Levoy, 1996]. In this representation, 3D space is divided into small 3D voxels. Each voxel maintains a value. This value is a signed distance from the voxel center to an implicit surface. Thus, the 3D surface of an object is mathematically represented as an implicit function defined in 3D space. To explicitly represent the shape using a triangle mesh, the marching cube algorithm of [Lorensen and Cline, 1987] can applied on the volume to extract the surface of the object.

While such a model is not as straightforward as the height map or triangle mesh, it is easy to maintain and update when we are given multiple observations of a scene sequentially. Thus, we has seen its application extensively in dense scene reconstruction methods such as [Chen et al., 2013; Endres et al., 2012; Izadi et al., 2011].

2.2 Light Source Models

A light source emits light into a scene. An LED bulb and the sun are both light sources. In this section, we briefly introduce three models for light sources.

2.2.1 Point Light Source

A point light source is a 3D point and it emits light in every direction. For different surface points in the scene, the illumination conditions from the point light source are different in two aspects. First, the lighting directions are different. The lighting direction has to be calculated for each surface point separately. Second, the amount of energy received by two surface patches are different. As shown in Figure 2.3, even

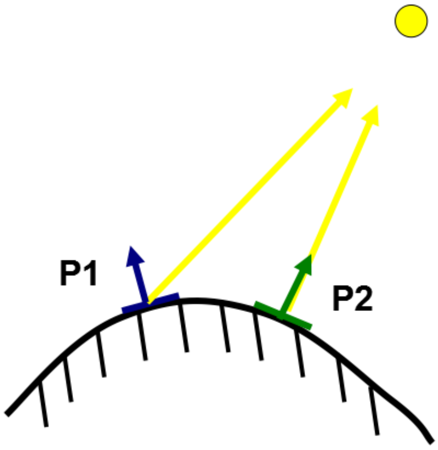

Figure 2.3: Illustration of point light source.

if the two surface patches have the same area, the received energy will be affected by their distance to the point light source and their orientations. It is quite intuitive that a patch receives more energy as it moves closer to the light source or turns its normal towards the light source,

$$E = \frac{E_0 \mathbf{n}^T \mathbf{s}}{d^2}, \tag{2.2.1}$$

where d is the distance between the surface patch and the light source, $\mathbf{n}$ is the normal, $\mathbf{s}$ is the lighting direction for this patch, E_0 is the amount of energy received when $d = 1$ and $\mathbf{n}$ is aligned with $\mathbf{s}$.

2.2.2 Directional Light Source

A directional light source can be obtained by moving the point light source to infinity. In this case, all points in the scene share the same lighting direction, and the energy received by two patches are only affected by their orientation. This is the simplest model for light source and is often used together with an orthographic camera model. In practice, a point light source can be regarded as a directional light source as long as its distance to the illuminated scene is much longer than the geometric size of the scene.

2.2.3 Environment Light Source

Environment light source can be regarded as a generalization of directional light source. A directional light source is a Dirac delta function on an unit sphere Ω

$$L_{\omega_0}(\omega) = L_0\delta(\omega - \omega_0) \tag{2.2.2}$$

where ω_0 is the lighting direction and

$$\int_{\Omega} \delta(\omega - \omega_0)\, \mathrm{d}\omega = 1. \tag{2.2.3}$$

If the the light comes not only from ω_0, but also from every other direction, the light source becomes an environment light source $L(\omega)$.

2.3 Reflectance Models

When an opaque material receives some light, apart from absorbing some of the energy, it will also reflect the light into space. This kind of phenomenon is accurately described by the bidirectional reflectance distribution function(BRDF). The strict definition of BRDF relies on a comprehensive understanding of radiometry, which can be found in [Palmer, 1999]. Here we simply explain the physical meaning of BRDF.

A BRDF is a function defined in a local coordinate system where the surface normal is aligned with z axis, which is shown in Figure 2.4. It is a function of the incoming

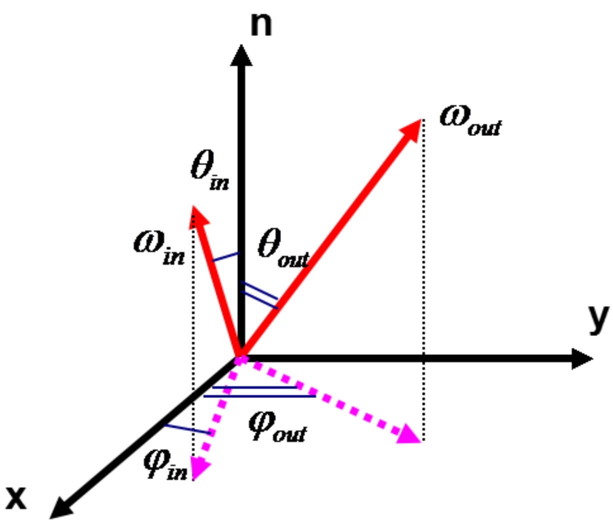

Figure 2.4: Definition of BRDF in a local coordinate system.

light direction ω_{in} and the outgoing light direction ω_{out}. Since both directions are vectors on a unit sphere in 3D space and can be parametrized using spherical coordinates, we have the following

$$f(\omega_{in}, \omega_{out}) = f(\theta_{in}, \phi_{in}, \theta_{out}, \phi_{out}). \tag{2.3.1}$$

Intuitively, the BRDF value is the amount of light it will reflect in the direction ω_{out} when it receives a unit amount of incoming light from direction ω_{in}. A physically valid BRDF has at least three properties, including positivity, conserving energy and Helmholtz reciprocity. The first two are obvious and the Helmholtz reciprocity means that the BRDF value does not change when swapping the incoming and outgoing light directions

$$f(\omega_{in}, \omega_{out}) = f(\omega_{out}, \omega_{in}). \tag{2.3.2}$$

The simplest BRDF model is Lambertian model, whose only parameter is the so-

Figure 2.5: A nearly lambertian object

called albedo ρ,

$$f(\omega_{in}, \omega_{out}) = \rho. \tag{2.3.3}$$

While this is an ideal model, many real-world materials' BRDF can be well approximated by Lambertian model. Figure 2.5 shows an image of a nearly Lambertian surface.

Besides diffuse materials which can be approximated by Lambertian model, specular highlight is often observed in other materials. Several parametric BRDF models have been proposed to model them.

Phong model [Phong, 1975] is an empirical(not physically valid) model for specular component of BRDF. It is a generalization of perfect mirror reflection.

Striking a beam of light onto the surface of a mirror, one could only expect to receive the light at a direction ω_{out} that is symmetric with the incoming light direction about the surface normal

$$\omega_R = 2\mathbf{n}^T\omega_{in}\mathbf{n} - \omega_{in}, \tag{2.3.4}$$

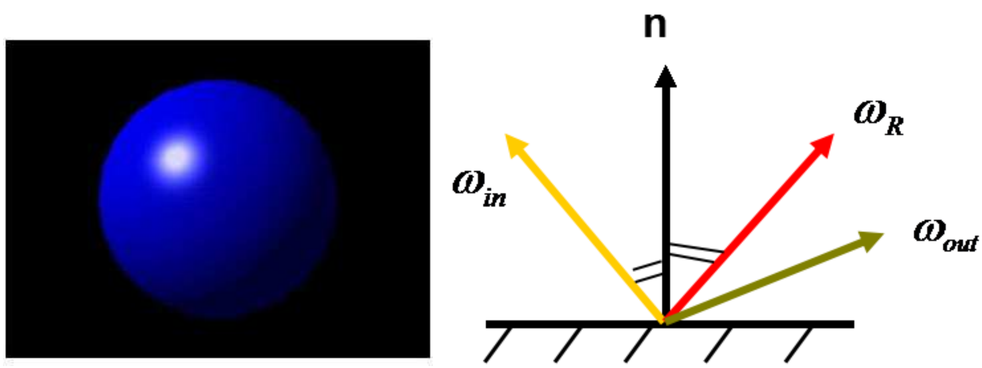

Figure 2.6: Left: A rendered sphere using Phong model; right: illustration of Phong model.

where ω_R is the direction one could see th reflective light. See Figure 2.6.

In Phong model, one could see the light not only in the specific direction ω_R. However, the BRDF value does have a relation with ω_R. Phong model assumes that the BRDF value gradually falls as the viewing direction ω_{out} deviates from ω_R in the following way

$$f(\omega_{in}, \omega_{out}) = (\omega_R^T \omega_{out})^\alpha, \tag{2.3.5}$$

where α is a shininess constant for the material. The larger α is, the more mirror-like the material is.

A major benefit of Phong model is efficiency in rendering applications. Thus, it has been an important model in computer graphics.

Cook-Torrance model [Cook and Torrance, 1982] is a physically valid BRDF model and is a suitable model for many real-world materials. This model is derived by analysing statistic properties of micro-facet distribution of a surface. It is expressed in the following manner,

$$f(\omega_{in}, \omega_{out}) = \frac{F(\omega_{in}, \mathbf{h}) G(\omega_{in}, \omega_{out}, \mathbf{h}) D(\mathbf{h})}{4(\mathbf{n}^T \omega_{in})(\mathbf{n}^T \omega_{out})}, \tag{2.3.6}$$

Figure 2.7: Renderings of 100 materials in MERL BRDF database

where $\mathbf{n}$ is the normal and $\mathbf{h}$ is the half vector

$$\mathbf{h} = \frac{\omega_{in} + \omega_{out}}{\|\omega_{in} + \omega_{out}\|}. \tag{2.3.7}$$

F, G and D of (2.3.6) are the Fresnel term, geometric term and distribution term respectively. Their exact definition can be found in [Cook and Torrance, 1982].

Apart from the traditional parametric BRDF models, another approach of modelling BRDF is the data-driven method. The basic idea is that although a BRDF lies in a very high dimensional space, the set of all real-world BRDFs form a low-dimensional manifold in this space. Thus, any BRDF can be approximated by a linear combination of a few other BRDFs which are close to it on the manifold

$$f = \sum_{i=1}^{k} f_i. \tag{2.3.8}$$

In this way, it is only necessary to obtain a sufficiently large database of BRDFs, and to

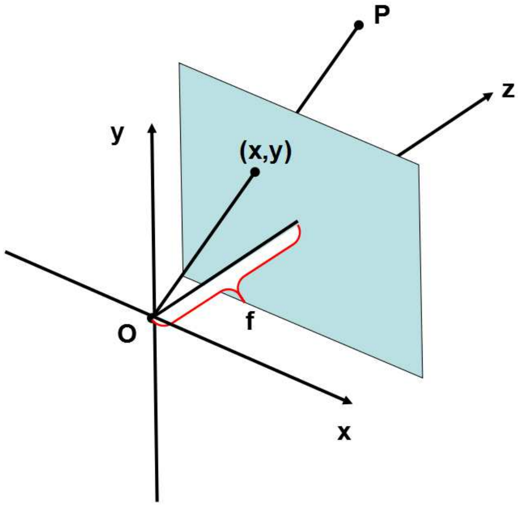

Figure 2.8: Illustration of the perspective camera model

estimate a new BRDF is simplified as estimating a sparse weight vector. [Romeiro and Zickler, 2010b] made use of this model to estimate BRDF and illumination simultaneously with the aid of the MERL BRDF database which contains a hundred real-world materials(see Figure 2.7).

Another trend in BRDF modeling is on exploiting symmetry properties of BRDFs. It has been observed that many real-world materials exhibit certain symmetry. For example, isotropy is a widely observed symmetry. These symmetries can greatly simplify a general BRDF model. Such a simplification not only reduces measurements required to acquire a BRDF, but also poses constraints on certain problem. In the following chapters, we will make extensive use of different symmetry properties found on real-world BRDFs in order to solve problems or build systems.

2.4 Camera Models

Many cameras can be modeled by the perspective camera model. In this model, the coordinate system is usually set up with the camera's optical centre being the origin O, and the z axis being parallel to the camera's optical axis. Thus, the image plane is parallel to the $x - y$ plane and the distance btween the two planes is the focal length f. As is shown in Figure 2.8, a 3D point P is imaged to the image plane in the following way

$$x = \frac{fX}{Z}, y = \frac{fY}{Z}. \tag{2.4.1}$$

This relationship can be elegantly expressed in a matrix form using homogeneous coordinates

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{2.4.2}$$

Typically, the topleft pixel in the image will be chosen as the origin for the 2D image plane. In this case, a translation will be involved and the resulting intrinsic matrix for the perspective camera is

$$K = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}. \tag{2.4.3}$$

A real-world camera is seldom a perfect perspective camera. There is usually slight lens distortion which are described by 5 coefficients. Nevertheless, all these parameters, including f, c_x, c_y, and distortion coefficients, can be calibrated using the method in [Zhang, 2000].

Another simpler camera model is the orthographic camera model. By this model, a 3D point is projected onto the image plane by the following transformation

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{2.4.4}$$

The orthographic camera model can be seen as the special case of the perspective camera model by moving the camera to infinity.

2.5 Image Formation

Given the above models, it is easy to express the pixel intensity at a pixel p

$$I(p, \omega_{out}) = \int_{\Omega} f_p(\omega_{in}, \omega_{out}) L(\omega_{in}) (\omega_{in}^T \mathbf{n}_p) \, \mathrm{d}\omega_{in}, \tag{2.5.1}$$

where Ω is the unit sphere, f_p is the BRDF at the 3D point corresponding to the pixel p, L is an environment light source, $\omega_{in}^T \mathbf{n}_p$ encodes the effect of angle between surface normal and the lighting direction.

This equation is called 'rendering equation' in computer graphics and lays the foundation for photo-realistic rendering. In this dissertation, the light source is either directional light source or point light source, thus for a specific pixel, the rendering equation is simply

$$I_p(\mathbf{v}) = L f_p(\mathbf{n}, \mathbf{v}, \mathbf{s})(\mathbf{n}^T \mathbf{s}), \tag{2.5.2}$$

where $\mathbf{n}$, $\mathbf{v}$, $\mathbf{s}$ are the surface normal, viewing direction and lighting direction respectively. L describes the strength of the light source's radiance.

Chapter 3

Calibrating Lambertian Photometric Stereo

Photometric stereo is an important research topic in computer vision. Unlike multi-view stereo or structure from motion(SfM), photometric stereo can provide direct access to surface normals, which not only are crucial in photo-realistic rendering, but also help improve 3D reconstruction accuracy [Nehab et al., 2005; Okatani and Deguchi, 2012]. Conventional photometric stereo algorithms(*e.g.*[Woodham, 1980]) require illumination conditions be known *a priori*. As a result, it becomes a necessity to calibrate illumination, which inevitably complicates the data capture process. Thus, auto-calibration algorithms are practically important.

Under unknown directional lighting, it is well known that surface normals of a Lambertian object can only be determined up to a linear ambiguity [Hayakawa, 1994]. Later it is shown [Belhumeur et al., 1999; Yuille and Snow, 1997] that this ambiguity can be reduced to the generalized bas-relief(GBR) ambiguity by enforcing the integrability constraint. In this chapter, we focus on further resolving this ambiguity.

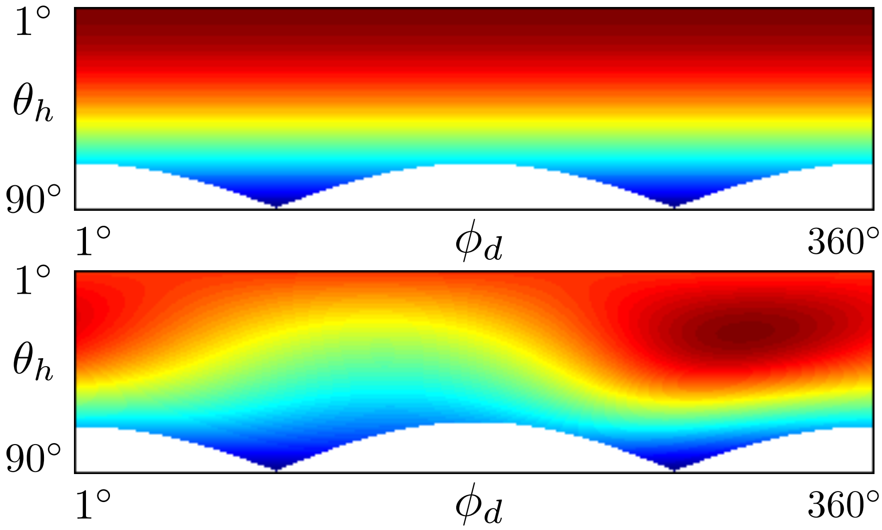

Figure 3.1: Synthesized 2D slices of a bivariate BRDF. Values are color-coded: red means larger BRDF values, blue indicates smaller values, and white regions are undefined for a BRDF. The horizontal and vertical axes are two angles ϕ_d and θ_h, which are defined in Section 3.3.1. The upper slice, which is obtained from ground truth normals and light direction, is constant along each row, while clearly this structure does not hold for the bottom slice estimated using GBR-distorted normals and light direction.

Reflectance of many real-world objects satisfies various symmetries, *e.g.*isotropy and reciprocity, which provides additional information to resolve the GBR ambiguity. As demonstrated in [Tan and Zickler, 2009; Tan et al., 2007], the GBR ambiguity can be solved using 'isotropic pairs' and 'reciprocal pairs' identified from one or two images. Surface points with specular spike [Drbohlav and Chaniler, 2005; Drbohlav and Šára, 2002] or diffuse maxima [Favaro and Papadhimitri, 2012] can also resolve the GBR ambiguity. However, these methods all require carefully identified special surface points, which are easily affected by image noise.

We solve the GBR ambiguity by a *holistic* analysis of *half-vector symmetry*, which suggests the BRDF value stays unchanged when rotating the incoming and outgoing light directions as a fixed pair around their bisector. This symmetry is closely related to the barycentric parameterization of isotropic BRDFs [Stark et al., 2005], and can be elegantly expressed in the halfway/difference parameterization [Rusinkiewicz, 1998]. Given the correct surface normals and light directions, we can obtain a 2D BRDF slice from each image of a curved isotropic surface. If the BRDF is half-vector symmetric, this 2D slice should form a special low-rank matrix when it is properly parameterized, as illustrated at the top of Figure 3.1. However, as we have observed and will present in subsequent sections, such a structure is generally destroyed when normals and light directions are distorted by a GBR transformation, as shown at the bottom of Figure 3.1. Restoring the special structure of 2D BRDF slices can resolve the GBR ambiguity. We propose a simple algorithm to find the solution based on this observation.

The **contribution** of the work presented in this chapter mainly lies in:

1) proposing half-vector symmetry as a novel cue to solve the GBR ambiguity;

2) proving that half-vector symmetry resolves the GBR ambiguity;

3) providing a simple auto-calibration algorithm based on a holistic analysis of this symmetry.

The rest of this chapter will be organized as follows. First we introduce necessary background knowledge and state the problem. After reviewing related works on the same topic, we show the main theoretical result and the derived auto-calibration method. Experimental results are then presented to consolidate our proposed method. Finally, we conclude this chapter with a discussion.

3.1 Background and Problem Statement

Photometric stereo seeks to recover surface orientations through varying lighting conditions. In a typical setup, a fixed orthogonal camera is looking at a static scene, which

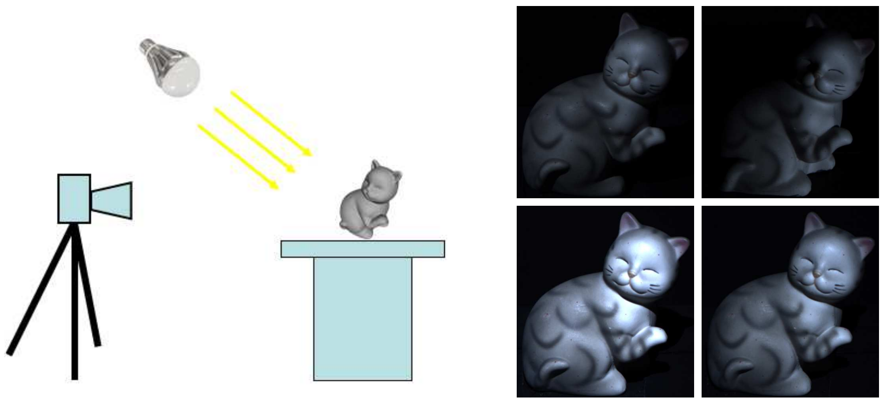

Figure 3.2: On the left is a typical setup for data capture in photometric stereo: a camera captures images of the cat model while the light source moves to different positions. On the right are four captured images.

is illuminated by a directional light source. The camera captures multiple images of the scene while light direction and intensity change with different images. See Figure 3.2 for an illustration.

Classical photometric stereo assumes Lambert's model for scene surface reflectances. When inter-reflection and shadow are ignored, the value i_{pf} of a pixel p on an image f can be expressed as

$$i_{pf} = \rho_p \mathbf{n}_p^\top s_f, \tag{3.1.1}$$

where $\mathbf{n}_p$ is the normal of the surface point projected to the pixel p. ρ_p is the albedo, which is the only parameter for a Lambertian BRDF. the norm $\|s_f\|$ and normalized vector $\mathbf{s}_f$ indicate the light intensity and direction respectively. Note that we use bold lowercase letters to indicate normalized vectors.

For all P pixels in an image in F images under different lighting conditions, Equation 3.1.1 can be written in a matrix form

$$\begin{pmatrix} I_{1,1} & I_{1,2} & \cdots & I_{1,F} \\ I_{2,1} & I_{2,2} & \cdots & I_{2,F} \\ \vdots & \vdots & \ddots & \vdots \\ I_{P,1} & I_{P,2} & \cdots & I_{P,F} \end{pmatrix} = \begin{pmatrix} N_{1,1} & N_{1,2} & \cdots & N_{1,P} \\ N_{2,1} & N_{2,2} & \cdots & N_{2,P} \\ N_{3,1} & N_{3,2} & \cdots & N_{3,P} \end{pmatrix}^{\top} \begin{pmatrix} S_{1,1} & S_{1,2} & \cdots & S_{1,F} \\ S_{2,1} & S_{2,2} & \cdots & S_{2,F} \\ S_{3,1} & S_{3,2} & \cdots & S_{3,F} \end{pmatrix} \tag{3.1.2}$$

or

$$I = N^{\top} S. \tag{3.1.3}$$

Each column of I is a vectorized image and each row is the intensity profile of a pixel under all different illumination conditions. Columns of N are surface normals multiplied by their corresponding albedos, and columns of S are light directions scaled by their intensities.

Given light intensities and directions for each image, it is easy to estimate N

$$N = (SS^{\top})^{-1} I^{\top}. \tag{3.1.4}$$

Surface normals and albedos can be easily calculated from N.

However, in uncalibrated photometric stereo, the matrix S, which encodes information of light intensities and directions, is unknown. Without introducing additional constraints, N and S can only be recovered up to a general linear transformation [Hayakawa, 1994] through Singular Value Decomposition(SVD).

$$I = N^{\top} S = (TN)^{\top} (T^{-\top} S) = \hat{N}^{\top} \hat{S}, \tag{3.1.5}$$

where T is an arbitrary invertible transformation.

By assuming the surface is smooth, it is possible to enforce the integrability constraint and reduce the ambiguity T to the so-called Generalized Bas-Relief ambiguity G [Belhumeur et al., 1999; Yuille and Snow, 1997]

$$I = N^\top S = (GN)^\top (G^{-\top} S) = \hat{N}^\top \hat{S}, \tag{3.1.6}$$

where $\hat{N} = GN$, $\hat{S} = G^{-\top} S$ and G has the following matrix form

$$\mathbf{G} = \begin{pmatrix} \lambda & 0 & \mu \\ 0 & \lambda & \nu \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.1.7}$$

As is shown in Equation 3.1.6, a normal $\mathbf{n}$ and a light direction $\mathbf{s}$ are distorted in the following way

$$\hat{\mathbf{n}} = \frac{G\mathbf{n}}{\|G\mathbf{n}\|}, \hat{\mathbf{s}} = \frac{G^{-\top}\mathbf{s}}{\|G^{-\top}\mathbf{s}\|}. \tag{3.1.8}$$

The physical meaning of the GBR ambiguity is that the same image can be explained as the results of various combinations of different shapes and illumination conditions.

The remaining problem, which is also the focus of this chapter, is to restore every $\mathbf{n}$ and each $\mathbf{s}$ from $\hat{\mathbf{n}}$ and $\hat{\mathbf{s}}$ respectively by recovering λ, μ and ν. Obviously, additional constraint is required to resove the GBR ambiguity.

3.2 Related Works

[Woodham, 1980] proposed the first photometric stereo method with directional lighting information known *a priori*. [Hayakawa, 1994] proved that under unknown directional illumination, normals of a Lambertian surface can only be recovered up to a linear transformation. This ambiguity is reduced to the GBR ambiguity by enforcing the integrability constraint [Belhumeur et al., 1999; Yuille and Snow, 1997].

Different methods have been proposed to resolve the GBR ambiguity. Besides those utilizing interreflection [Chandraker et al., 2005] or special lighting configurations [Zhou and Tan, 2010], the other methods can be divided into two categories.

The first category of methods resolves the ambiguity by analyzing reflectance properties. Earlier methods rely on specific reflectance models. [Georghiades, 2003] adopted the Torrance-Sparrow model [Torrance and Sparrow, 1967] to tackle the problem. In [Drbohlav and Chaniler, 2005; Drbohlav and Šára, 2002], the authors assumed specular-spike reflectance and showed that the ambiguity is solved by detected specular spots in images. More recent methods exploit general reflectance symmetries. [Tan and Zickler, 2009; Tan et al., 2007, 2011] exploited isotropy and reciprocity to recover GBR parameters from carefully identified 'isotropic pairs' and 'reciprocal pairs' in a single image.

In the second category, priors on surface albedos are exploited. [Alldrin et al., 2007] recovered the GBR parameters by assuming the true distribution of surface albedo has small entropy. [Shi et al., 2010] identified surface points with the same albedo but different normals to resolve the ambiguity. In a recent work [Favaro and Papadhimitri, 2012], the authors assumed smoothly varying surface albedos in order to locate the 'lambertian diffuse maxima', which are then used in a robust estimation framework to estimate the GBR paramters.

Another work worthnoting is [Papadhimitri and Favaro, 2013], which does not solve the GBR ambiguity directly. They have proved theoretically that the GBR ambiguity can be avoided by utilizing a perspective camera model instead of an orthogonal camera model, which has long been a common assumption in uncalibrated photometric stereo.

3.3 Theory on Half-Vector Symmetry and GBR

In this section, we will introduce half-vector symmetry of BRDFs and present the special low-rank matrix structure enforced by it. After that, we will examine how the GBR ambiguity destroys the structure of this matrix.

3.3.1 Half-Vector Symmetry

BRDF is a function of incoming and outgoing light directions $(\boldsymbol{\omega}_{in}, \boldsymbol{\omega}_{out})$ in a local coordinate system. [Rusinkiewicz, 1998] proposed to use four spherical coordinates $(\theta_h, \phi_h, \theta_d, \phi_d)$ to parameterize a BRDF as

$$f(\boldsymbol{\omega}_{in}, \boldsymbol{\omega}_{out}) = f(\theta_h, \phi_h, \theta_d, \phi_d). \tag{3.3.1}$$

Angles in Equation 3.3.1 are illustrated in Figure 3.3 and explained here. First of all, the half vector is defined as the bisector of lighting and viewing directions, *i.e.* $\mathbf{h} = \frac{\boldsymbol{\omega}_{in}+\boldsymbol{\omega}_{out}}{\|\boldsymbol{\omega}_{in}+\boldsymbol{\omega}_{out}\|}$. In a local coordinate system where the surface normal is aligned with the z-axis, θ_h and ϕ_h are the azimuthal and polar angles of $\mathbf{h}$ respectively. θ_h is called half angle. θ_d is named difference angle and is defined as the angle between $\mathbf{h}$ and $\boldsymbol{\omega}_{in}$. ϕ_d indicates the rotation angle of $\boldsymbol{\omega}_{in}$ and $\boldsymbol{\omega}_{out}$ as a pair around the half vector $\mathbf{h}$.

Common BRDF symmetries can be elegantly expressed using this parameterization. One of the various symmetries widely observed in real-world materials is isotropy, which means BRDF values stay unchanged as the lighting and viewing direc-

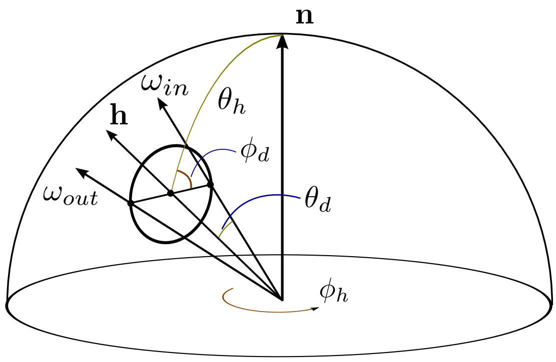

Figure 3.3: Halfway/difference parameterization of BRDF [Rusinkiewicz, 1998]

tions are rotated as a fixed pair around the normal. Thus, isotropy reduces the BRDF to a 3D function $f(\theta_h, \theta_d, \phi_d)$. Many isotropic materials also satisfy the half-vector symmetry, which suggests that BRDF values are invariant with rotation of lighting and viewing directions around the half vector. In this case, the BRDF does not depend on ϕ_d and is further reduced to a bivariate function $f(\theta_h, \theta_d)$.

This kind of bivariate BRDF model is reported in previous works. For example, [Stark et al., 2005] studied several traditional parametric reflectance models and showed that they are bivariate. [Shi et al., 2012] further used a biquadratic function to represent bivariate BRDFs. [Romeiro et al., 2008] evaluated the validity of such a representation on the MERL BRDF database [Matusik et al., 2003] and concluded that it can be used to represent most materials in the database to high accuracy. Besides, the same bivariate BRDF representation has already been adopted in calibrated photometric stereo [Alldrin et al., 2008] and reflectometry [Romeiro et al., 2008].

3.3.2 Structured 2D BRDF Slice

The pixel intensity of a general isotropic surface is calculated as $I = f(\theta_h, \theta_d, \phi_h)(\mathbf{n} \cdot \mathbf{s})$. We ignore the light intensity $\|s\|$ here for notation simplicity. Under the assumption of directional lighting, orthographic camera and homogeneous surface reflectance, pixel intensities in an image are determined as

$$I = f_{\theta_d}(\theta_h, \phi_d)(\mathbf{n} \cdot \mathbf{s}), \tag{3.3.2}$$

where $f_{\theta_d}(\theta_h, \phi_d) = f(\theta_h, \theta_d, \phi_d)$ is a 2D slice of the original BRDF.

For a curved surface with abundant normals, *e.g.* a sphere, when both the normals $\mathbf{n}$ and light direction $\mathbf{s}$ are known, we can estimate a 2D slice of the BRDF, namely $f_{\theta_d}(\theta_h, \phi_d)$, based on Equation 3.3.2. This BRDF slice can be arranged into a matrix form in the range $\theta_h \in [0, \frac{\pi}{2}]$, $\phi_d \in [0, 2\pi]$. For a bivariate BRDF, f does not depend on ϕ_d. Thus each row of the matrix is constant. Such a low-rank structure can be clearly seen at the top of Figure 3.1.

3.3.3 GBR-Distorted 2D BRDF Slice

When surface normals and light direction are distorted by a GBR transformation as in Equation 3.1.8, the 2D BRDF slice estimated from Equation 3.3.2 no longer has the low-rank structure. An example from synthetic data is shown at the bottom of Figure 3.1. This observation motivates us to resolve the GBR ambiguity by restoring the low-rank structure of BRDF slices.

In the special case that the lighting and viewing directions coincide, the low-rank property is preserved by the classic bas-relief ambiguity, *i.e.* $\mu = \nu = 0$ in Equation 3.1.7. In more general cases, however, a GBR transformation will destroy the low-rank structures of 2D BRDF slices. In fact, for a general bivariate BRDF, we are able to prove the following proposition.

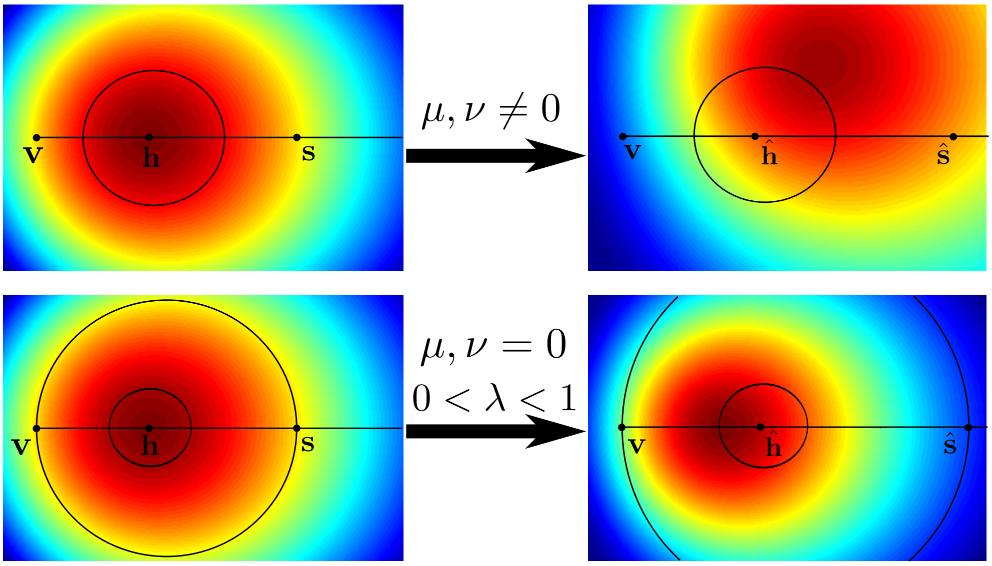

Figure 3.4: Projective plane with color-coded BRDF values. A black ellipse corresponds to a row in the BRDF slice of Figure 3.1. Top row shows the effect of a GBR transformation ($\mu, \nu \neq 0$); bottom row shows the case of a classic bas-relief transformation with $0 < \lambda < 1$. Before GBR transformation, the values on an ellipse are the same, while this is not true after GBR transformation.

Proposition 1. *Any GBR transformation other than the identity matrix cannot simultaneously perserve the special low-rank structure of bivariate BRDF slices estimated from two images whose light directions are not coplanar with the viewing direction.*

Here we give some intuitive explanations with illustrative figures. Please refer to appendix for a formal proof.

We consider the problem on the projective plane where a 3D unit vector (x, y, z) is represented by a point $(x/z, y/z)$. Viewing direction $\mathbf{v} = (0, 0, 1)^\top$, light direction $\mathbf{s}$, half vector $\mathbf{h}$ and every surface normal $\mathbf{n}$ can find their corresponding points on this plane. From an image of a curved surface, *e.g.* a sphere, we can observe sufficient number of points (surface normals) of this plane. Given the lighting and surface normal directions, a BRDF value can be estimated at each pixel of an image. By encoding these BRDF values into colors and mapping them to the projective plane according to the normal at each pixel, we obtain a 'BRDF map' shown in Figure 3.4, where red indicates larger values.

First of all, as marked by the black ellipse in Figure 3.4, points with the same θ_h form an ellipse around $\mathbf{h}$, with one of its symmetry axis being the line vs connecting $\mathbf{v}$ and $\mathbf{s}$. For a bivariate BRDF, the BRDF value should be constant along each ellipse, since θ_h is fixed for those points on the same ellipse and θ_d is fixed for all pixels in the same image. These ellipses correspond to rows in the matrix representation of the 2D BRDF slice in Figure 3.1.

Now we show how a GBR transformation can change the structure. The GBR transformation moves normals and the light direction in different ways, as shown in Equation 3.1.8. In fact, the transformed light direction $\hat{\mathbf{s}}$ will still lie on the line vs, according to the equation

$$\hat{\mathbf{s}} = \frac{G^{-T}\mathbf{s}}{\|G^{-T}\mathbf{s}\|} \simeq \begin{pmatrix} s_x \\ s_y \\ -\mu s_x - \nu s_y + \lambda s_z \end{pmatrix} \tag{3.3.3}$$

where the symbol $\simeq$ means equal up to a scale. So the transformed half vector $\hat{\mathbf{h}}$ will also stay on the line vs. On the other hand, a GBR transformation will translate all normals (along with the BRDF map associated them) by a displacement (μ, ν) and scale them by λ. Besides, BRDF values are also changed since they are estimated from pixel intensities and the GBR transformed shading $\hat{\mathbf{n}}^\top \hat{\mathbf{s}}$ using Equation 3.3.2.

In general, the different motions of normals and the light direction will make the BRDF value change along the transformed ellipse(consists of points forming the same half angle with $\hat{\mathbf{h}}$). As a result, the low-rank structure of the 2D BRDF slice in Figure 3.1 will be destroyed. For example, the top of Figure 3.4 shows the case of non-zero μ, ν. The transformed BRDF map has varying values along the ellipse around $\hat{\mathbf{h}}$. Similarly, at the bottom of Figure 3.4, we consider the case of $\mu = \nu = 0$ and $\lambda \neq 1$, *i.e.* the classic bas-relief ambiguity. $\lambda < 1$ is assumed here. In this case, the GBR transformation moves the light direction $\mathbf{s} = [s_x, s_y, s_z]^\top$ to $\hat{\mathbf{s}}$, which is even further from $\mathbf{v}$ (the origin). This can be seen from the following equation

$$\hat{\mathbf{s}} = \begin{pmatrix} \frac{1}{\lambda} & 0 & 0 \\ 0 & \frac{1}{\lambda} & \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} s_x \\ s_y \\ s_z \end{pmatrix} \simeq \begin{pmatrix} s_x/(\lambda s_z) \\ s_y/(\lambda s_z) \\ 1 \end{pmatrix}. \tag{3.3.4}$$

At the same time, normals and the associated BRDF map will be scaled and shrinked toward the origin $\mathbf{v}$. So the transformed BRDF value is no longer constant along the transformed ellipse. This again breaks the low-rank structure of the 2D BRDF slice.

3.4 Auto-Calibrating Photometric Stereo

In this section, we propose a new method for uncalibrated photometric stereo. We follow the dichromatic reflectance model [Sato and Ikeuchi, 1994] and assume that the reflectance of an object is the sum of a diffuse component and a *homogeneous* specular component: $f(\omega_{in}, \omega_{out}, x) = \rho(x) + f_s(\omega_{in}, \omega_{out})$, where x indicates a surface point, ρ is the diffuse albedo and f_s is the specular BRDF. Given multiple images taken under varying lightings and a fixed viewpoint, it is relatively easy to separate the diffuse and specular components using existing techniques [Sato and Ikeuchi, 1994; Tan and Ikeuchi, 2008]. From the diffuse images, we are able to recover surface normals and

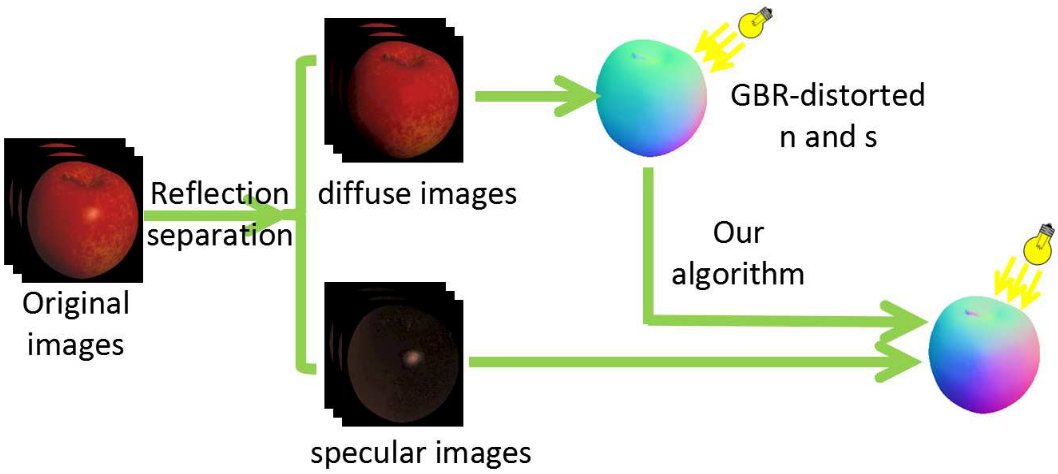

Figure 3.5: Pipeline for uncalibrated photometric streo. First the input images are separated into diffuse images and specular images. Surface normals and light directions can be recovered up to an unknown GBR transformation by decomposing the diffuse images. With the aid of specular images, our proposed algorithm can help resolve the GBR ambiguity and recover true normals as well as lightings.

light directions up to a GBR transformation by the uncalibrated photometric stereo method [Yuille and Snow, 1997]. See Figure 3.5 for the pipeline.

By assuming the specular BRDF is bivariate, it is guaranteed by our earlier discussion that surface normals are correctly recovered *iff* the low-rank structure in estimated specular BRDF slices is restored.

One natural idea of restoring the low-rank structure is to use the established TILT technique [Zhang et al., 2011], which recovers a low-rank pattern via domain transformation. In our case, however, both the position of each point (corresponding to a normal) on the 2D BRDF slice and its associated BRDF value are changed by a GBR transformation. Thus, TILT is not suitable for our problem and we need to resort to other solutions. We formulate a simple optimization algorithm to estimate the GBR parameters given normals and light directions up to a GBR ambiguity, together with a set of specular images.

We first define an objective function to measure how well the estimated 2D BRDF slice satisfies the special 'low-rank' constraint, which in our case means each row is constant. The most straightforward measure is defined as the sum of variances along each row(the axis of ϕ_d). It is obvious that the correct GBR parameters should correspond to the unique global minimum of the objective function.

However, there are some practical issues in adopting such a measure. Firstly, with a real image of limited resolution, some entries of the 2D BRDF slice are missing because the corresponding normals are not observed in the image. Different rows of the matrix have different numbers of observations. Thus we give higher weights to rows of a larger number of valid observations, since the variances calculated from those rows are more reliable. Secondly, the absolute variance level is biased by BRDF values. In other words, rows with larger BRDF values tend to have larger variances. So we divide the variance of each row by the square of its mean value for normalization. Thirdly, the BRDF values estimated from Equation 3.3.2 at large θ_h are usually noisy because the shading term $\mathbf{n}^\top\mathbf{s}$ tends to be small over those regions. So we only use the top 20 rows ($\theta_h = 1^\circ, 2^\circ, \ldots, 20^\circ$) when evaluating the variance. Here is the objective function we used,

$$\min_G \sum_{i=0}^{N-1} \sum_{\theta_h=1}^{20} \frac{k_{i,\theta_h}}{\sum\limits_{\theta=1}^{20} k_{i,\theta}} \frac{\mathrm{Var}\left[f_i(\theta_h, \phi_d)\right]}{\mathrm{Mean}^2\left[f_i(\theta_h, \phi_d)\right]}. \tag{3.4.1}$$

In this function, N is the number of input specular images and we can estimate a 2D BRDF slice f_i from each image. The integer k_{i,θ_h} is the number of valid observations in row θ_h of slice i. Note that the calculation of variance and mean are evaluated only on valid entries.

Empirically, we found it useful to add one more constraint to the objective function 3.4.1. We require the sum of intensities (BRDF value times $\mathbf{n}^\top \mathbf{s}$) in the top 20 rows makes up at least 5% of that of the whole 2D slice. If this condition is not satisfied in a BRDF slice, infinite large penalty value will replace the sum of variance for that slice. This simple practice helps exclude some degenerated solutions, *e.g.*a very small λ compressing the whole BRDF map to a single point.

The proposed objective function is difficult to optimize because both the position and value of the BRDF map depend on the GBR transformation. We simply adopt a coarse-to-fine search for optimization. The objective function is first evaluated at coarsely sampled grid points in the parameter space. We then search nearby the optimal grid point with higher sampling rate. Empirically, we find the objective function is smooth and this multi-resolution search can generate good results. As in [Alldrin et al., 2007], the search space is restricted to $-5 \leq \mu, \nu \leq 5, 0 < \lambda \leq 5$. The initial sampling step is set as 0.5. After each iteration, both search range and sampling step are reduced to $\frac{1}{5}$ of the coarser one. Three iterations suffices to find an accurate estimation of GBR parameters.

3.5 Experimental Validation

3.5.1 Experiments on Synthetic Datasets

We evaluated our method using images synthesized from the MERL BRDF database [Matusik et al., 2003]. For each material, we rendered four images of a sphere with light directions randomly sampled over the visible hemisphere. We transformed the known

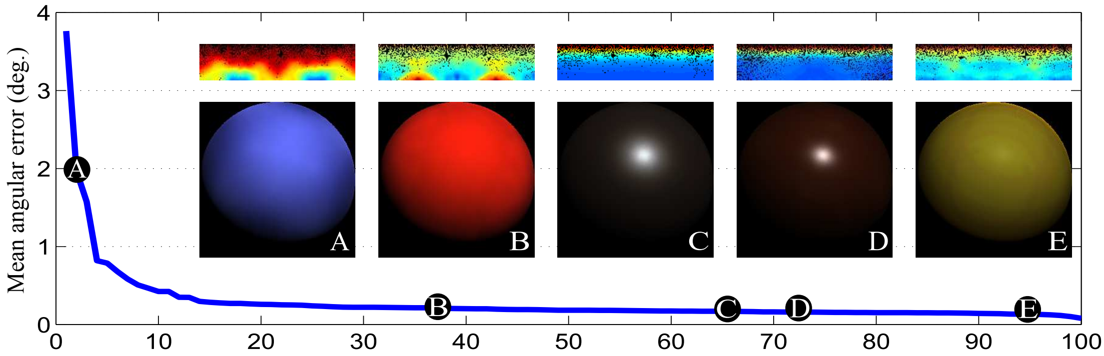

Figure 3.6: Error plot of results on MERL database. BRDFs are sorted according to error values. Five materials from the database are presented using a rendered sphere. One of the recovered BRDF slice(only the range $\theta_h \in [0°, 40°]$ is shown) is also shown above the rendered sphere.

normals and light directions using a GBR transformation whose parameters are randomly sampled in the search space. These four images and the GBR-distorted normals and lighting directions are fed into our algorithm for test. We compared the recovered normal directions with ground truth and recorded the mean angular error.

For each material in the database, we repeated the above process 10 times with different light directions and GBR parameters. A plot of the median(of the 10 obtained average errors) is shown in Figure 3.6 for all the BRDFs in the database. The results show that the proposed method is capable of recovering surface normals from the GBR ambiguity with high accuracy. Note that even some BRDFs are not strictly half-vector symmetric, the correct solutions still correspond to the global optimal of the objective function.

3.5.2 Experiments on Real Object Datasets

We have also evaluated our method on real data. Each dataset consists of a few images of an object with homogeneous specular reflection. As a first step, color information was used to separate each image into a diffuse image and a specular image [Sato and Ikeuchi, 1994]. To handle shadows, the algorithm in [Brand, 2002] was used to fill in missing values in shadow regions. After pre-processing, we applied the technique of uncalibrated Lambertian photometric stereo to obtain normals and light directions up to an unknown GBR tranformation [Yuille and Snow, 1997]. Finally, our coarse-to-fine optimization was performed using the specular images to fully recover surface normals.

To assess quality of the recovered normals, we compared our results with normals computed from calibrated photometric stereo. In the calibrated method, lighting information was known and the same set of diffuse images were used. As shown on the left of Figure 3.7, our algorithm successfully recovered normals of the apple surface and the recovered BRDF slice shows the expected low-rank structure, which is also observed in the result of calibrated photometric stereo. Figure 3.8 shows results on other datasets.

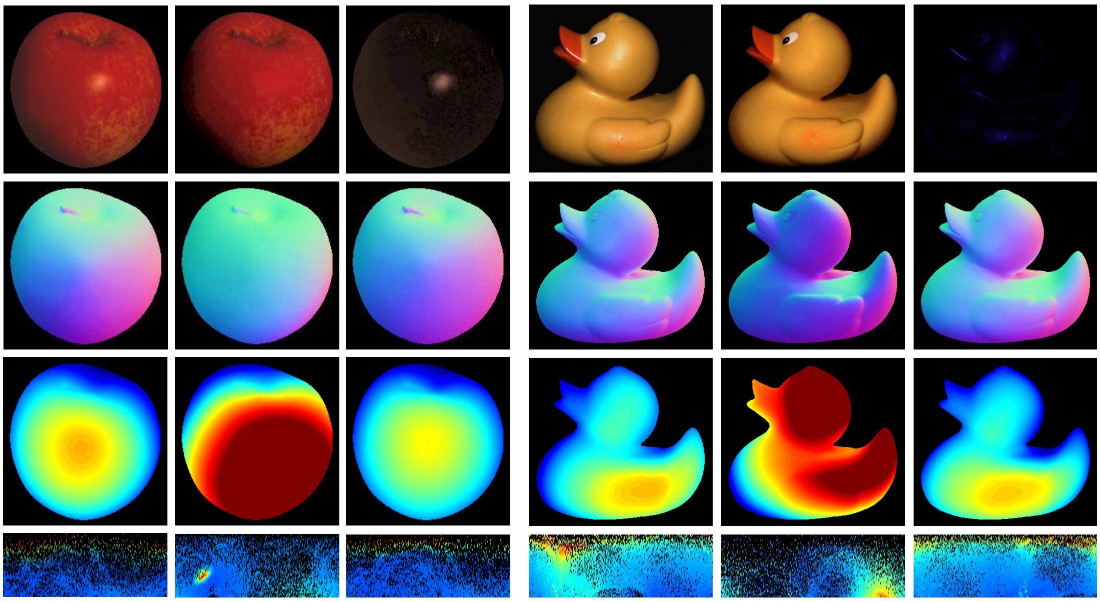

Figure 3.7: Results of the datasets *Apple*(left) and *Duck* (right). For each example, top row shows the input images, from left to right: an original image, separated diffuse image and specular image; Second row are normal maps encoded into colors $((R, G, B) = (\mathbf{n} + 1)/2)$, from left to right: calibrated photometric stereo, GBR distorted normals, our recovered normals; Third row shows the height map integrated from the normal field above it. The last row shows one of the recovered 2D BRDF slices using normals above it. Note that only rows $\theta_h \in [0°, \ldots, 40°]$ are shown.

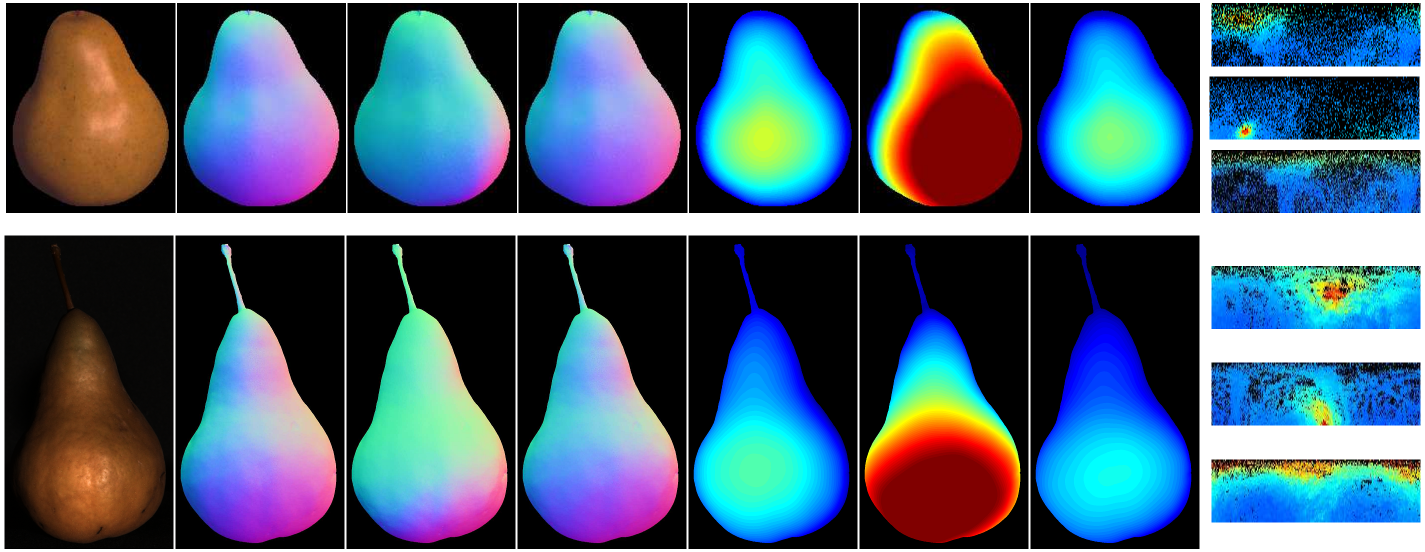

Figure 3.8: Results of datasets *Pear*(top) and *Pear2*(bottom). From left to right: an input image, normals from calibrated photometric stereo, GBR-distorted normals, our recovered normals, height map from calibrated photometric stereo, height map using GBR-distorted normals, our recovered height map, BRDF slices. In the last column, there are three BRDF slices for each dataset. From top to bottom, they are estimated using normals from calibrated photometric stero, GBR-distorted normals and our recovered normals respectively.

Method	Duck	Apple	Pear	Pear2
[Alldrin et al., 2007]	7.5(5.4)	9.0(3.9)	9.7(3.6)	23.8(9.4)
[Shi et al., 2010]	6.6(5.5)	8.9(3.6)	24.9(8.3)	23.7(10.3)
[Drbohlav and Chaniler, 2005]	7.7(4.4)	8.7(3.9)	4.6(2.4)	13.8(5.3)
[Tan et al., 2011]	7.3 (9.4)	9.8 (16.2)	N/A	N/A
[Favaro and Papadhimitri, 2012]	7.4(4.8)	**7.0**(2.9)	7.3(2.7)	**9.2**(4.9)
Ours	**5.7**(4.5)	7.8(3.1)	**4.4**(2.4)	11.7(4.9)

Table 3.1: Mean and standard deviation(in brackets) of angular error(deg) by different methods on four datasets. Normals obtained from calibrated photometric stereo are regarded as ground truth.

Comparison of our method with several other algorithms is shown in Table 3.1. Those [Alldrin et al., 2007; Favaro and Papadhimitri, 2012; Shi et al., 2010] utilizing only albedo information were run on diffuse images. As can be seen, our method achieves a similar performance as the diffuse maxima method [Favaro and Papadhimitri, 2012] and outperforms the other methods. Please notice that our method and [Favaro and Papadhimitri, 2012] exploit completely different sources of information. While [Favaro and Papadhimitri, 2012] is purely based on diffuse images and owns its robustness to a robust estimator, our method exploits information in specular reflection and its accuracy comes from a robust global structure. The dataset *Pear2* exhibits significantly higher errors than the others. This is due to imperfect diffuse/specular separation, which affects all methods.

It is worthnoting that in most examples, the BRDF slice recovered by our method shows an even better 'low-rank' pattern than that of the slice from calibrated photometric stereo. This difference suggests inaccuracy of recorded light directions. We carefully conducted another experiment to further validate our method. By painting a sphere with green paint whose BRDF is known, we prepared a dataset with known ground truths for both normals and the BRDF. After separating six images of the sphere

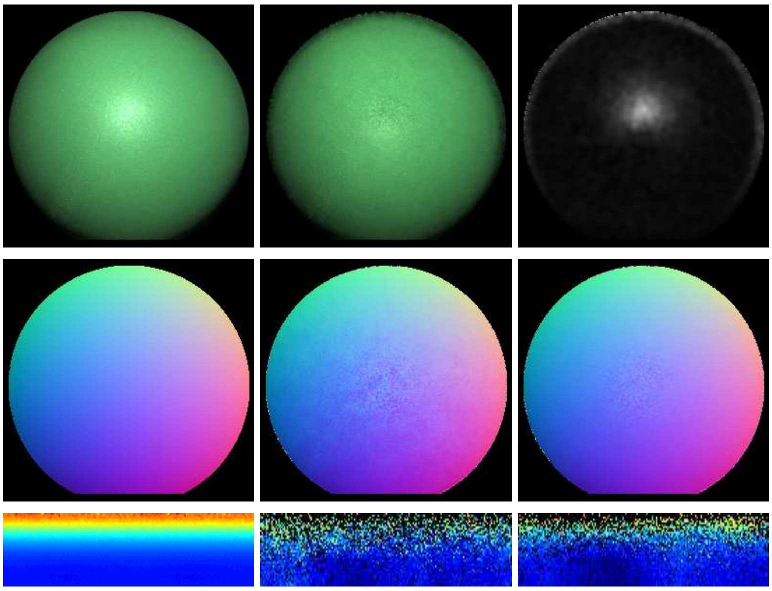

Figure 3.9: Results of the dataset *Sphere*. The first row shows the separation result of an image. The second row shows normals and the last row displays BRDF slices. From left to right are ground truth, results estimated from calibrated method, results produced by our method.

under directional lighting into diffuse and specular components, both the calibrated method and our method were run respectively. Since the BRDF of the sphere is homogeneous across its surface, we used the original images instead of separated specular images in our algorithm to solve the GBR ambiguity. We then estimated the BRDF slice from each image based on normals(and light directions) recovered from both methods respectively. Recovered normals and BRDF slices were compared to ground truths. As can be seen from Figure 3.9 and Table 3.2, our method performed slightly better than the calibrated method in this case.

Method	Normal error	BRDF error
Calibrated	4.03(2.87)	0.091
Ours	3.95(1.87)	0.071

Table 3.2: The first column shows mean and standard deviation(in brackets) of angular error(deg) of normals by different methods on the *Sphere* dataset; the second column shows the root-mean-square error of recovered BRDF slices.

3.6 Summary and Discussion

In this chapter, we have carefully examined the structure of a 2D BRDF slice estimated from a curved surface under directional illumination. We have shown that if the BRDF is bivariate, which is implied by isotropy and half-vector symmetry, the estimated BRDF slice will have a special low-rank structure and this structure is generally destroyed by GBR-distorted normals and light directions. Based on this observation, we have formulated a simple algorithm to automatically calibrate photometric stereo by restoring the structure. Our approach is distinguished from several previous works in that it seeks to recover a global structure instead of relying on a few critical surface points. This holistic approach makes our algorithm robust and accurate.

Limitations As implied by the algorithm, a major limitation to our method is that it requires sufficent normal variation to work. In fact, while information at all pixels with different normals helps achieve robustness, it excludes our method from dealing with extreme cases where only a few different normals are observed in an image. A possible solution is to use special light configurations [Zhou and Tan, 2010]. Alternatively, we might also fit parametric models to these limited observations before analyzing the structure of the BRDF slice.

Chapter 4

Capturing Appearance for Isotropic Materials

In this chapter, we present a method for appearance capture. After an introduction of the proposed method, a short review of related works will be presented. Since our method makes a general assumption on BRDF, we present this as a background before introducing the whole system. Then we look into individual components of the pipeline, shape reconstruction and reflectance estimation , in the following sections. Experimental results are then presented to demonstrate the effectiveness of the method. Finally, a discussion concludes this chapter.

4.1 Introduction

Appearance capture methods recover both 3D shape and surface reflectance of objects, allowing photo-realistic rendering of the captured objects under arbitrary lighting conditions from every possible viewpoint. This capture is an important problem with many applications in fields such as computer graphics and reverse engineering.

Simultaneous capture of both shape and reflectance is quite challenging since capture of either requires a good model of the other as a prerequisite. Most previous methods on appearance capture make use of parametric BRDF models. For example, the multi-view stereo method [Hernandez et al., 2008] assumes a Lambertian model for surface reflectance and [Goldman et al., 2005] adopts Ward's model. A major problem for these methods is that their performance degrades as the real objects' reflectance deviates from the assumed models. Therefore, such methods can only capture appearances of limited real-world objects.

In contrast to the above methods, we exploit reflectance symmetries to make our method work on more general objects. Specifically, we take advantage of isotropy and bilateral symmetry which are widely observed for BRDFs of real-world materials. According to [Alldrin and Kriegman, 2007], these two symmetry properties allow us to identify 'iso-depth contours', *i.e.*pixels with the same distance to the image plane, from a set of photometric images. In our proposed method, SfM [Hartley and Zisserman, 2003] is first applied to reconstruct a sparse set of 3D points using images of multiple viewpoints. Then the depth information of these 3D points are propagated to points on the same iso-depth contours which are collected from multiple viewpoints. Each pass of propagation generates additional 3D points, whose depth information can be further propagated. A surprisingly small number of 3D points (about two hundreds) can be propagated to reconstruct the complete 3D shape (about two million points). Once the shape is fixed, we use the same set of input images to infer the spatially varying reflectance. We assume the BRDF at each surface point is a linear combination of a few basis isotropic BRDFs which are represented by some discretized representation to handle general materials. The basis BRDFs and mixing weights at each point can be iteratively estimated by the alternating constraint least square method as in [Lawrence et al., 2006].

The proposed method not only works for general isotropic materials, but also requires simpler setup compared with other works [Ghosh et al., 2009; Holroyd et al., 2010; Tunwattanapong et al., 2013]. The system is a practical step towards an affordable solution for casual users to perform appearance capture.

4.2 Related Work

Image-based modeling. These methods reconstruct a 3D shape and a 'texture map' from images. Texture color at each surface point is decided according to its image projections. [Furukawa and Ponce, 2010; Lhuillier and Quan, 2005] are two recent representative methods. While this kind of methods produce plausible results in many cases, the implicit assumption of Lambertian BRDF is often insufficient to represent general non-Lambertian materials.

Shape scanning and reflectance fitting. To obtain precise 3D shape, either time-of-flight 3D laser scanners or triangulation based 3D laser scanners are used in [Davis et al., 2005; Levoy et al., 2000; Rusinkiewicz et al., 2002; Zhang et al., 2004]. Given a precise 3D reconstruction, parametric reflectance functions can be fitted at each surface point according to its image observations, as in [Lensch et al., 2003; Sato et al., 1997].A major issue for these methods is that they require precise registration between images and 3D shapes. Since different sensors(3D laser scanner and camera) are used for shape and reflectance capture respectively, this registration is difficult and often causes artifacts in misaligned regions. Some methods [Aliaga and Xu, 2008; **?**] handle this problem by using the same camera for shape reconstruction from structured-light system and reflectance estimation from photometric stereo, and this approach makes registration relatively simple. However, their problem is that they need to capture images under both structured-light and varying directional light at *each* viewpoint, which is tedious and requires a complicated setup.

Photometric appearance capture. The method proposed in this chapter belongs to photometric approaches that capture both shape and reflectance from the same set of images. Most of previous methods, *e.g.*[Goldman et al., 2005; Hernandez et al., 2008; Zhang et al., 2003], assume specific parametric BRDF models such as Lambert's model or Ward's model [Ward, 1992]. The performance of these methods will degrade when the real objects have different reflectances from the assumed model.

Some other methods employed a sophisticated hardware setup to achieve high quality results. [Ma et al., 2007] and [Ghosh et al., 2009] use a light stage where the intensity of each LED on the stage can be precisely controlled. [Holroyd et al., 2010] require specialized coaxial lights. [Tunwattanapong et al., 2013] build a metal arc with LEDs to produce spherical harmonics lighting. The requirement of expensive and complicated hardware has limited their wide application.

Recently, a few algorithms [Alldrin et al., 2008; Holroyd et al., 2008] are proposed for appearance capture by exploiting various reflectance symmetries that are valid for a broader class of objects. However, [Holroyd et al., 2008] require up to a thousand input images at *each* viewpoint and [Alldrin et al., 2008] relies on fragile optimization. [Tan et al., 2011] and [Chandraker et al., 2011] both recovered iso-contours of depth and gradient magnitude for isotropic surfaces. In their methods, additional user interactions or boundary conditions are required to recover the 3D shape.

The work closest to the proposed method in this chapter is [Alldrin et al., 2008]. Both methods are built upon reflectance symmetry embedded in 'isotropic pairs' in-

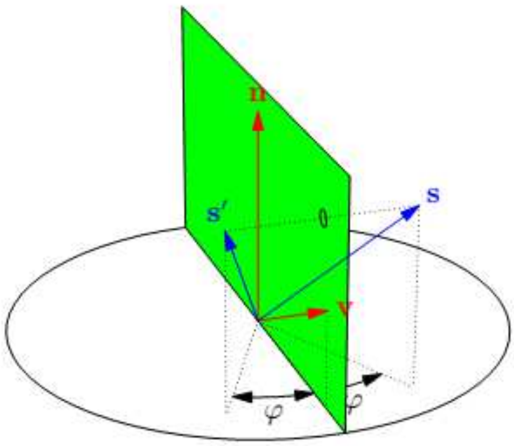

Figure 4.1: An illustration of bilateral symmetry from [Alldrin and Kriegman, 2007].

troduced in [Tan et al., 2007]. There are three key differences between the proposed method and [Alldrin et al., 2008]. First, a complete 3D shape can be obtained by the proposed method rather than a single-view normal map by [Alldrin et al., 2008]. Second, fragile optimization by alternatively adjusting shape and reflectance, which is used in [Alldrin et al., 2008], is avoided here by combining multi-view geometry and photometric cues. Third, the proposed method works with general tri-variant BRDFs while [Alldrin et al., 2008] assumed bi-variant BRDFs to constraint the problem.

BRDF acquisition. The proposed method is also related to BRDF acquisition methods such as [Aittala et al., 2013; Dong et al., 2010; Ren et al., 2011]. These methods are only applicable to near-flat surfaces where the surface normals are known beforehand. The proposed method can be considered as a generalization of these methods to non-planar surfaces.

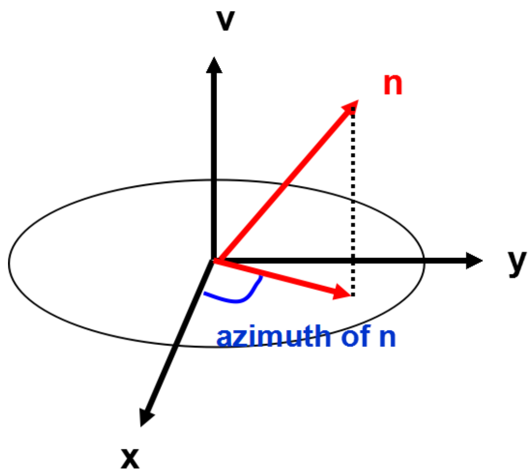

Figure 4.2: Definition of azimuth angle for a normal.

4.3 Background

Apart from isotropy and half-vector symmetry which we have seen in Chapter 3, there is another symmetry observed in many real-world BRDFs and referred to as bilateral symmetry.

Bilateral symmetry means that the BRDF is symmetric about the plane spanned by the surface normal $\mathbf{n}$ and viewing direction $\mathbf{v}$ with respect to the incident lighting direction $\mathbf{s}$. This symmetry is illustrated in Figure 4.1. Since the lighting directions $\mathbf{s}$ and $\mathbf{s}'$ are symmetric about the green plane, the BRDF values for the two configurations are the same by bilateral symmetry. A very interesting property of bilateral symmetry is that the angle between the normal $\mathbf{n}$ and the two lighting directions, $\mathbf{s}$ and $\mathbf{s}'$, are also the same. Thus, intensities observed at two pixels with this symmetric configurations will be the same

$$f(\mathbf{n}, \mathbf{v}, \mathbf{s})\mathbf{n}^T\mathbf{s} = f(\mathbf{n}, \mathbf{v}, \mathbf{s}')\mathbf{n}^T\mathbf{s}'. \tag{4.3.1}$$

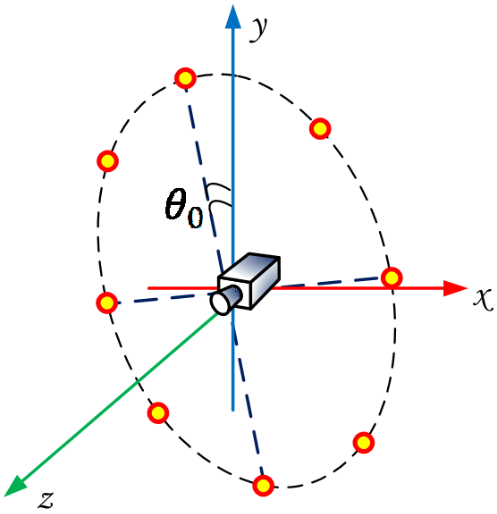

Figure 4.3: An illustration of a minimal lighting configuration

If we take the view direction $\mathbf{v}$ as the z axis and transform $\mathbf{n}$ and $\mathbf{s}$ into the new coordinate system, it can be easily seen that the symmetry plane spanned by $\mathbf{v}$ and $\mathbf{n}$ corresponds to the azimuth angle of $\mathbf{n}$ (See Figure 4.2), which is the angle between x axis and the projection of $\mathbf{n}$ in the $x - y$ plane.

Based on this observation, [Alldrin and Kriegman, 2007] proposed 'a minimal lighting configuration' for detecting this symmetry and hence the azimuth angle of $\mathbf{n}$ in the camera's local coordinate system. This configuration is a circle of light source positions parallel to the image plane of the camera and centred about the optical axis of the camera, which is shown in Figure 4.3. A typical intensity profile for a pixel under this lighting configuration is shown in Figure 4.6 (a). The vertical axis of the chart indicates pixel intensities, while the horizontal axis is the range of azimuth angles. The red symmetry axis of the intensity profile can be recovered by the naive approach described in [Alldrin and Kriegman, 2007], and it provides a good estimation of the normal's azimuth angle in the camera's local coordinate system.

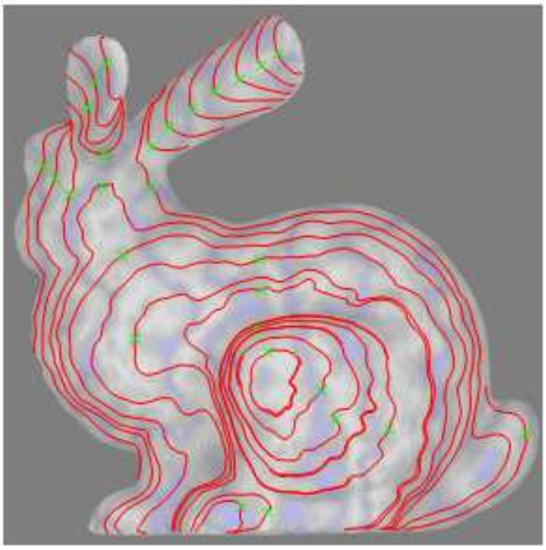

Figure 4.4: An example of iso-depth contours on the bunny model([Alldrin and Kriegman, 2007])

While bilateral symmetry is not implied by isotropy, nearly all physically valid isotropic materials in the real-world have this property. Thus, we do not distinguish the two properties and only use the term 'isotropy' to refer to both as in [Alldrin and Kriegman, 2007].

Given the azimuth estimation at every pixel in an image, it is trivial to identify iso-depth contours(see Figure 4.4) in the image following [Alldrin and Kriegman, 2007].

4.4 Overview

As an overview, we provide a block diagram of our system in Figure 4.5. First, we acquire images from multiple viewpoints under varying illumination. A robust algorithm is specially designed to identify iso-depth contours from these images for each viewpoint. Meanwhile, we apply the standard structure-from-motion technique on images from different viewpoints to reconstruct a sparse set of 3D points. A complete 3D shape is obtained by propagating the depths of these points along the dense iso-depth contours. This initial shape is further refined according to the method described in [Nehab et al., 2005]. Once the shape is fixed, we can estimate a set of basis isotropic BRDFs and their mixing weights at each surface point by the ACLS method [Lawrence et al., 2006] to model the surface reflectance.

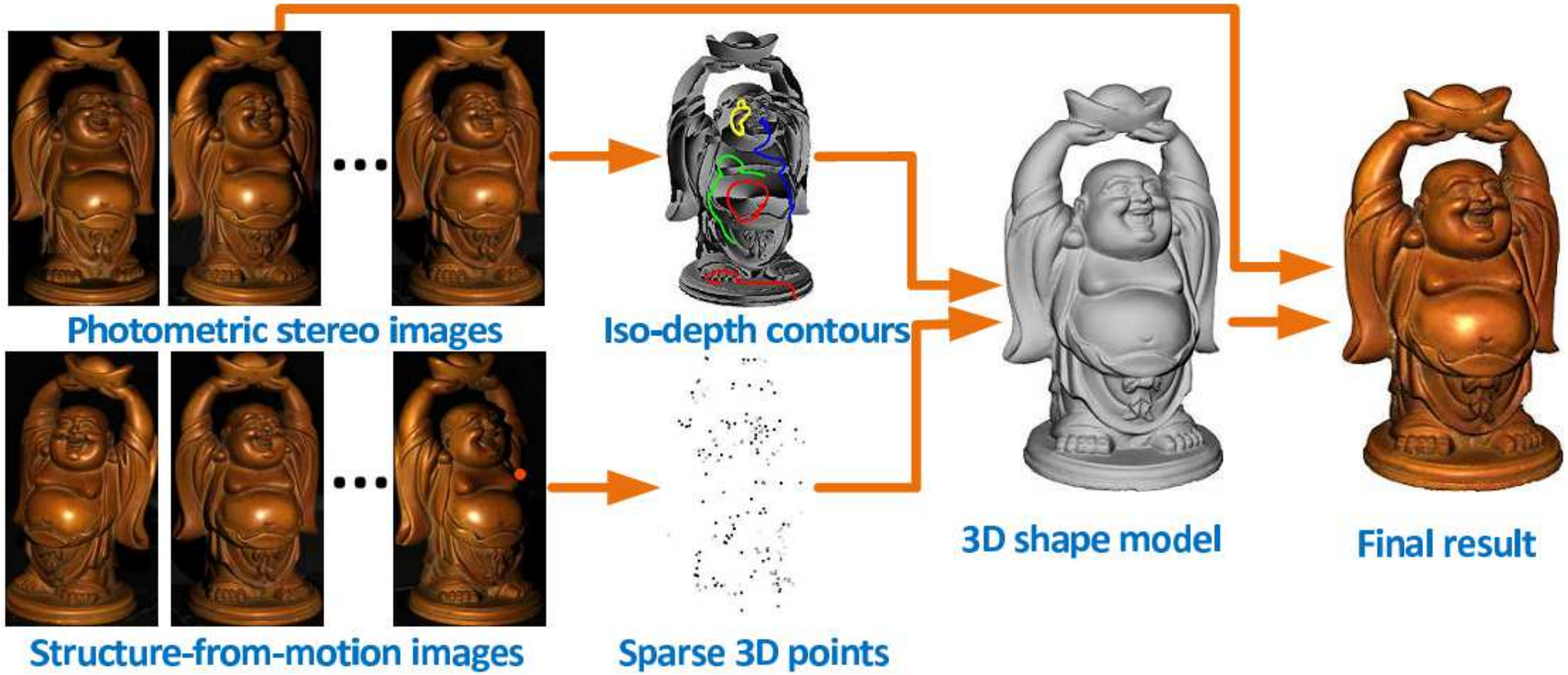

Figure 4.5: System pipeline. We recover iso-depth contours from photometric stereo images for each viewpoint and recover a sparse 3D point cloud by structure-from-motion. In the figure showing iso-depth contours, the gray intensity encodes the estimated azimuth angles, and the colored curves are iso-depth contours. We then propagate the depths of these 3D points along the iso-depth contours to recover the complete 3D shape. Once the shape is fixed, we estimate the spatially varying BRDF from the original input images.

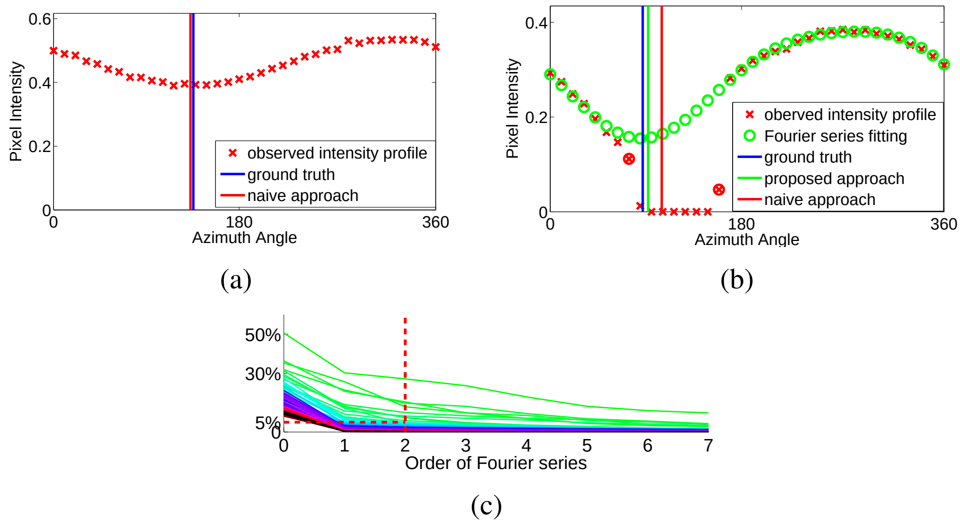

Figure 4.6: (a) The symmetry axis of intensity profiles tells the azimuth angle of a pixel's normal direction; (b) cast shadows can break this symmetry; (c) the intensity profile of most of isotropic BRDFs in [Matusik et al., 2003] can be well represented by a 2-order Fourier series.

4.5 Shape Reconstruction

4.5.1 Robust Iso-depth contour estimation

As mentioned in Section 4.3, [Alldrin and Kriegman, 2007] observed that isotropy allows almost trivial estimation of iso-depth contours. However, their work does not mention how to handle more general lighting configurations, and does not take into consideration of global illumination effects such as cast shadow and interreflection. Based on the original naïve approach, we propose two improvements to make iso-depth contour estimation more robust on real data.

Handheld Point Light Source In practice, it is more convenient to capture images with a handheld bulb, *i.e.*a point light source that does not lie precisely on a view-centred circle. So we compute spatially variant lighting directions at each pixel, and interpolate the desired observations from recorded pixel intensities.

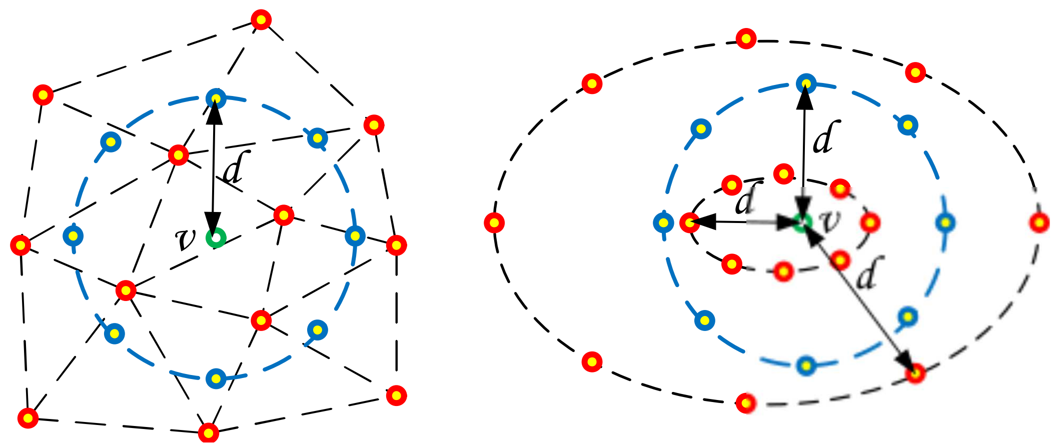

Figure 4.7: We compute a Delaunay triangulation of the original lighting directions (red dots) in the projective plane. The desired observations (blue dots) on a view-centred circle are generated by linear interpolation within these triangles. Left: the circle radius d is the mean distance between the red dots and the viewpoint $\mathbf{v}$. Right: the circle radius d is set as $(d_i + d_o)/2$. Here, d_i (or d_o) is the largest (or smallest) distance between $\mathbf{v}$ and the red dots on the inner (or outer) conic.

Specifically, we take the average depth of an object (computed from the reconstructed sparse 3D points in Section 4.5.2) to estimate an approximate 3D position of each pixel. We also calibrate the 3D positions of the light source (see the experiments section). The lighting directions at each pixel are then computed according to the 3D positions of that pixel and the light sources.

To allow flexible data capture, we interpolate observations under lighting directions lying on a view-centred circle, and compute the azimuth angle from these interpolated observations. We study this interpolation problem in the projective plane where a unit 3D direction (x, y, z) is represented by a 2D point $(x/z, y/z)$. As shown in the left of Figure 4.7, the original lighting directions at a pixel are represented by the red points. We compute a Delaunay triangulation of these points in the projective plane. The desired observations – those blue dots – on a view-centred circle are generated by linear interpolation within these triangles. The radius d of the blue circle is computed as the mean distance between the red dots and the viewpoint **v**.

Global Illumination Effects To improve accuracy, we need to identify cast shadows, which break the symmetry of pixel intensities. Figure 4.6 (b) shows an example pixel with cast shadow. (This pixel is marked in red in the input image of the 'Buddha' example in Figure 4.14.) The original intensity profile marked by red '$\times$' is asymmetric. Though we might use an intensity threshold to detect shadows, it is hard to identify penumbra this way. Two samples in the penumbra are marked with red '$\otimes$' in Figure 4.6 (b). As shown in Figure 4.6 (b), the azimuth angle estimated by the naïve method in [Alldrin and Kriegman, 2007] is far from the ground truth at this point. Points in the penumbra also cause problems in the reflectance estimation in Section 4.6. So we identify them as 'outliers' by fitting a parametric model to the observed intensity profiles. Consider a Lambertian point with surface normal $\mathbf{n} = (n_x, n_y, n_z)$ and albedo ρ. Its intensity should be $\rho r n_x \cos\theta + \rho r n_y \sin\theta - \rho z n_z$ when the lighting direction is $(r\cos\theta, r\sin\theta, -z)$. This motivates us to fit a truncated Fourier series

$$A_0 + \sum_k A_k \cos k\theta + \sum_k B_k \sin k\theta$$

to an intensity profile. We evaluate the fitting error on synthetic data generated according to the MERL BRDF database [Matusik et al., 2003]. For each BRDF in the database, we uniformly sample ninety normals along a longitude on the visible upper hemisphere, and render them under a light moving on a view-centered circle. Figure 4.6 (c) plots the normalized RMSE (root-mean-square error) of all materials with different orders of Fourier series. For most of materials, an intensity profile can be well represented by a second order (i.e. $1 \leq k \leq 2$) Fourier series with normalized RMSE less than 5%. So we always apply RANSAC to fit a second order Fourier series to each observed intensity profile, and estimate the azimuth angle according to the symmetry of the fitted curve. As shown by the green vertical line in Figure 4.6 (b), our estimated azimuth angle is closer to the ground truth. In fact, this fitting also makes our method less sensitive to specular inter-reflections, which are outliers above the fitted curve.

Tracing Contours Once an azimuth angle is computed at each pixel, we proceed to generate iso-depth contours. Starting from every pixel, we iteratively trace along the two directions perpendicular to the azimuth direction with a step of 0.1 pixel. Specifically, suppose the estimated azimuth angle is θ at a pixel $\mathbf{x}$. We trace along the two 2D directions $\mathbf{d}_+ = (cos(\theta + \pi/2), sin(\theta + \pi/2))$ and $\mathbf{d}_- = (cos(\theta - \pi/2), sin(\theta - \pi/2))$ to $\mathbf{x}_+ = \mathbf{x} + 0.1\mathbf{d}_+$ and $\mathbf{x}_- = \mathbf{x} + 0.1\mathbf{d}_-$. We then replace $\mathbf{d}_+$ and $\mathbf{d}_-$ according to the azimuth angles of $\mathbf{x}_+$ and $\mathbf{x}_-$ respectively and continue to trace. We stop tracing when the maximum number of iterations is reached (500 in our experiments). Pixels on one traced curve should have the same distance to the image plane. To avoid tracing across discontinuous surface points, we use the method described in the 'NPR camera' [Raskar et al., 2004] to identify discontinuities. Further, we define a confidence measure for these traced contours as the inverse of the maximum curvature along them. Intuitively, smoother contours with relatively small curvature are more reliable.

4.5.2 Multi-view depth propagation

A standard structure-from-motion algorithm such as [Lhuillier and Quan, 2005; Snavely et al., 2006] can reconstruct a set of sparse 3D points on the object. We capture experiment objects on a turntable with a checkboard pattern to ensure sufficient feature matching for textureless examples. Since structure-from-motion algorithms could be

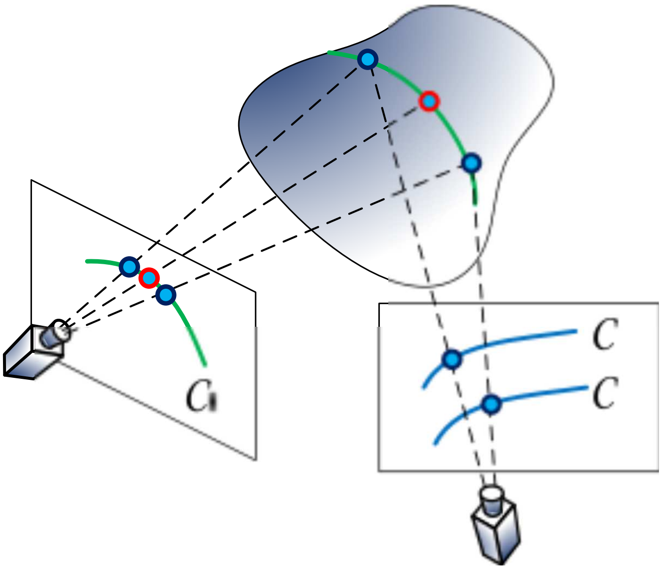

Figure 4.8: We propagate the depth of $\mathbf{x}$ to the iso-depth contour segment C_i that passes through its projection in the i-th view. This propagation generates new 3D points, *e.g.*$\mathbf{y}_1, \mathbf{y}_2$, whose depths in other images can also be propagated along their corresponding iso-depth contours C_{j1}, C_{j2}.

affected by moving highlights, we compute a median image at each viewpoint by taking the median intensity of each pixel and use these images for feature matching. Reconstructed 3D points are combined with the traced iso-depth contours to recover the complete 3D shape.

Depth Propagation As illustrated in Figure 4.8, given a reconstructed 3D point $\mathbf{x}$, we project it to all images where it is visible. Suppose an iso-depth contour C_i goes through its projection in the i-th image. We perform a depth propagation to assign the depth of $\mathbf{x}$ to all pixels on C_i. (If the depth of a pixel on C_i is already known, we keep it unchanged.) This propagation generates new 3D points, whose depths in other images can also be propagated. We begin with a sparse set of 3D points P reconstructed by structure-from-motion. Depth propagation with P in all images generates a large set of 3D points P'. We then replace P by P' and apply depth propagation iteratively. We

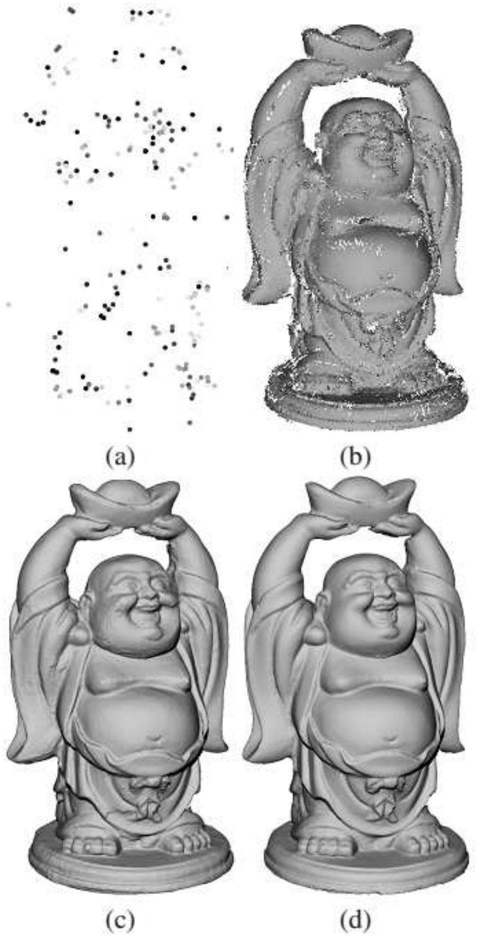

Figure 4.9: (a) initially reconstructed 3D points; (b) 3D points obtained by depth propagation; (c) initial shape after Poisson surface reconstruction; (d) final result after optimization.

keep iterating until P' is empty.

Direct application of the algorithm described above will generate poor results. There are a few important issues which must be addressed for robust 3D reconstruction.

Point Sorting We sort all points in P according to the confidence of their associated iso-depth contours. Note that if a point is visible in K different views, it is repeated K times in P and each repetition is associated with an iso-depth contour in one view. At each iteration, we only select half of the points in P of high confidence for depth propagation. We then remove those selected points, and insert P' into the sorted set P for the next iteration.

Visibility Check We should not propagate the depth of a 3D point in an image where it is invisible. However, the visibility information is missing for 3D points generated by propagation. So we apply a consistency check when propagating the depth of a 3D point $\mathbf{x}$ to a contour C. We check pixels on C one by one, starting from the projection of $\mathbf{x}$ to the two ends of C. If a pixel p fails the check, we truncate C at p, and only assign the depth of $\mathbf{x}$ to pixels on the truncated contour. If the updated contour is too short (less than 5 pixels in our implementation), we do not propagate.

To evaluate consistency at a pixel p, we assign it the depth of $\mathbf{x}$ to determine its 3D position. We then use the surface normal of $\mathbf{x}$ to select L ($L = 7$ in our implementation) most front parallel views where $\mathbf{x}$ is visible. We assume p is visible in all these L images and check the consistency of the azimuth angles at its projections. The azimuth angles at corresponding pixels in two different views uniquely decide a 3D normal direction [1]. If different combinations of these L views all lead to consistent 3D

[1]An azimuth angle in one view (with the camera center) decides a plane where the normal must lie in. Intersecting two such planes determines the 3D normal direction.

normals (the angle between any two normals is within T degrees), we consider p as consistent. Otherwise, we discard the view that is most different from the mean view angle and check consistency with the remaining $L - 1$ views iteratively. We consider p consistent, if it is consistent over at least 3 views. Otherwise, it is inconsistent. For each consistent 3D point, we set its normal as the mean of all consistent normals. In our implementation, we begin with $T = 3$, and relax it by 1.3 times whenever P' is empty until $T > 15$.

We note the number of consistent views for each 3D point when inserting it to the set P'. Points are first sorted by the number of consistent views in descending order. Those with the same number of consistent views are sorted by the confidence of contours.

Shape Optimization After depth propagation, we have a set of 3D points, each with a normal direction estimated. We apply the Poisson surface reconstruction [Kazhdan et al., 2006] to these points to obtain a triangulated surface. This surface is further optimized according to [Nehab et al., 2005] by fusing the 3D point positions and their normal directions.

Figure 4.9 shows the reconstructed shape at different stages. Shown in (a) are 3D points obtained from multi-view stereo. (b) is the 3D points (with normal directions) obtained by depth propagation. (c) shows the result after Poisson surface reconstruction. The final optimized shape is at Figure 4.9 (d). Note the face becomes clearly smoother after optimization.

4.6 Reflectance Capture

We assume the surface reflectance can be represented by a linear combination of several (K=2) basis isotropic BRDFs. Once the 3D shape is reconstructed, we follow [Lawrence et al., 2006] to estimate the basis BRDFs and their mixing weights at each point on the surface. We consider the general tri-variant isotropic BRDF, which is a function of θ_h, θ_d, ϕ as shown in Figure 4.10. We discretize θ_h, θ_d and ϕ into $90, 2$ and 5 bins respectively all in the interval $[0, \pi/2]$. Please refer to [Romeiro and Zickler, 2010a] for a justification of choosing this interval. Hence, a BRDF is represented as a 900×1 vector by concatenating its values at these bins.

We build an $N \times M$ observation matrix $\mathbf{V}$, and factorize it into a matrix of mixing weights $\mathbf{W}$ and a matrix of basis BRDFs $\mathbf{H}$ as,

$$\mathbf{V}_{N\times M} = \mathbf{W}_{N\times K}\mathbf{H}_{K\times M}.$$

$M = 900$ is the dimension of a BRDF. N is the number of 3D points. Each row of $\mathbf{V}$ represents the observed BRDF of a surface point. In constructing the matrix $\mathbf{V}$, we avoid pixels observed from slanted viewing directions (the angle between viewing direction and surface normal is larger than 40 degrees in our implementation), where a small shape reconstruction error can cause a big change in their projected image positions. $\mathbf{V}$ contains missing elements because of incomplete observation. We apply the Alternating Constrained Least Squares (ACLS) algorithm [Lawrence et al., 2006] to iteratively compute the rows of $\mathbf{W}$ and columns of $\mathbf{H}$.

To further improve reflectance capture accuracy, we first compute $\mathbf{H}$ from a subset of precisely reconstructed 3D points, whose reconstructed normals from different combinations of azimuth angles are consistent within 1.5 degrees. We then fix $\mathbf{H}$ and compute **W** at all surface points.

As an example, the recovered basis BRDFs and BRDF mixture weights for 'Buddha' are visualize in Figure 4.11. The red and green channels are the normalized mixture weight of the first and second basis BRDFs. Each basis BRDF is applied to render a sphere under frontal lighting and viewing directions.

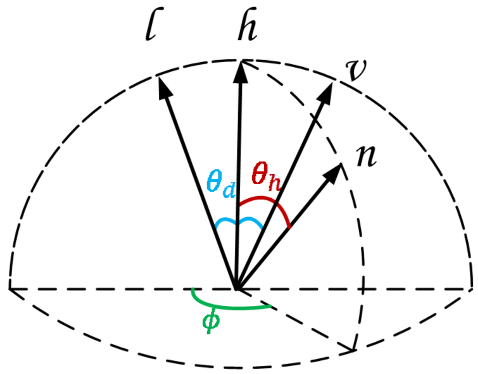

Figure 4.10: Definition of θ_h, θ_d and ϕ.

4.7 Experiment

We evaluated our algorithm on real data with two hardware setups. Both setups used a PointGrey Grasshopper camera, which captures linear images at 1200×900 resolution. The first setup used a handheld bulb as light source to ensure data capture flexibility. The second one used blinking LED lights synchronized with the video camera to speedup capture. We captured images viewpoint by viewpoint. This process can be speeded up by an automatic turntable. But we used a broken LP player to simplify the setup. After capturing images at one viewpoint, we manually rotated the LP player to capture the next viewpoint.

In our experiments, the 3D points obtained from the structure-from-motion algo-

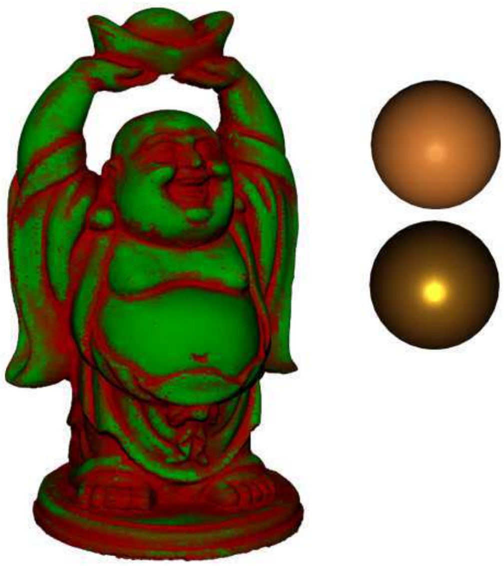

Figure 4.11: The normalized BRDF mixture weights are visualized in the different color channels. The corresponding basis BRDFs are used to render a sphere on the right.

rithm were often noisy. We only kept points with reprojection error less than 0.5 pixels. Typically, about 200 initial points were obtained for each example. Our system can also easily incorporate manual intervention in the form of matched feature points to handle textureless regions. To provide a 'ground truth' validation, all experimental objects were scanned using a Rexcan III industrial scanner, which is accurate to 10 microns. Our results were registered with the scanned shapes using the iterative closest point (ICP) algorithm [Besl and McKay, 1992].

4.7.1 A Handheld-Light System

Consisting of just a video camera and a handheld light source, this system is compact and portable. At each viewpoint, we moved a handheld bulb to capture a short video clip (about two minutes), and then uniformly sampled about 100 images with different

Figure 4.12: Results from the handheld-light system. First column: one of the input image; second column: the recovered shape rendered with uniform diffuse shading; third column: a rendering with the recovered reflectance model from the same viewpoint and lighting condition as the image in the first column; fourth column: the color-coded shape error in millimeters compared to laser-scanned ‘ground truth’.

lighting directions. The light source positions and intensities were recorded with calibration spheres. Examples are provided in Figure 4.12. The first column of Figure 4.12 shows a sample input image. The teapot example in the first row was captured from 10 viewpoints, which allow us to reconstruct part of its surface, while the cat example in the second row was fully reconstructed. To better visualize the recovered shapes, we render them with uniform diffuse shading in the second column. Most of the geometry details are successfully captured. The third column shows renderings according to the captured reflectance from the same viewpoint and lighting condition as the input image in the first column. To provide a quantitative evaluation on shape capture, we visualize the shape reconstruction errors (measured in millimeters) in the fourth column. The larger errors at the surface boundary are due to insufficient and slanted observations. Overall, the median (and mean) shape error is 0.53 (and 0.79) millimeters for the teapot, 0.62 (and 0.96) millimeters for the cat. The object diameters are 250mm and 140mm for teapot and cat respectively. It is worth noting that the rendering result

Figure 4.13: Our device consists of a video camera and two circles of LED lights.

for cat was a bit noisy because the brushed paint is not precisely isotropic.

4.7.2 A Ring-Light System

Setup To facilitate data capture, we built a simple device shown in Figure 4.13. 72 LEDs were uniformly distributed on two concentric circles of diameter 400 and 600 millimeters respectively. A video camera was mounted at the center of these circles, facing the direction perpendicular to the board [1]. The camera was synchronized with the LED lights such that at each video frame, there was only one light turned on. At each viewpoint, we captured 30 images with different lighting directions in 12 seconds (at 4fps). (Please refer to the supplementary file for a justification of the number of images per viewpoint.)

We pre-calibrated the intensities and positions of these LEDs. Since they are uniformly distributed and the circle radiuses are known, we only need to calibrate one parameter θ_0 to determine their positions. Here, θ_0 is the reference angle of the first

[1]The camera was mounted manually. It might not exactly sit on the circle center. Its direction might also be slightly off. We ignored these two factors as they introduce little errors according to our experiments.

LED light as shown in Figure 4.3. For more details of this calibration please refer to the supplementary files. We considered LEDs to be point light sources. Hence, at a general surface point, the local lighting directions will form two conics in the projective plane as illustrated on the right of Figure 4.7. When computing azimuth angles, we performed a Delaunay triangulation based interpolation as introduced in Section 4.5.1. We chose a circle with diameter $d = (d_i + d_o)/2$ to interpolate the required observations. Here, d_i (or d_o) is the largest (or smallest) distance between $\mathbf{v}$ and the original lighting directions – the red dots – in the inner (or outer) conic.

Results An example, an polished wooden 'Buddha', is provided in the first row of Figure 4.14. This example has focused and strong highlight. The object diameter is 120 mm. We captured it from 41 different viewpoints. This example contains many discontinuities at clothes folds and large concavities at the shoulder. These shape details were faithfully captured, as shown in the rendering in the second and third columns. The median (or mean) shape error was 0.36 (or 0.57) mm in this example. Most of the large shape errors appeared at concave carvings with strong inter-reflection.

Another four examples, 'Cup', 'Teapot2','Cup' and 'Frog', are also included in Figure 4.14. These examples cover a wide range of different material. The 'Cat' is covered by a brushed matte paint. The clay 'Teapot2' has soft and extended highlight. The rusted metal 'Cup' has quickly change reflectance over its surface. The painted 'Frog' also has significant spatial BRDF changes. The quantitative results is shown in Table 4.1. As can be seen from Table 4.1 and Figure 4.14, out method consistently performed well on all of them.

To further evaluate our results, we rendered all examples from the ringlight system under novel lighting and viewpoint and compared them with captured photographs in Figure 4.15. Note that these images were not used in our shape and reflectance capture

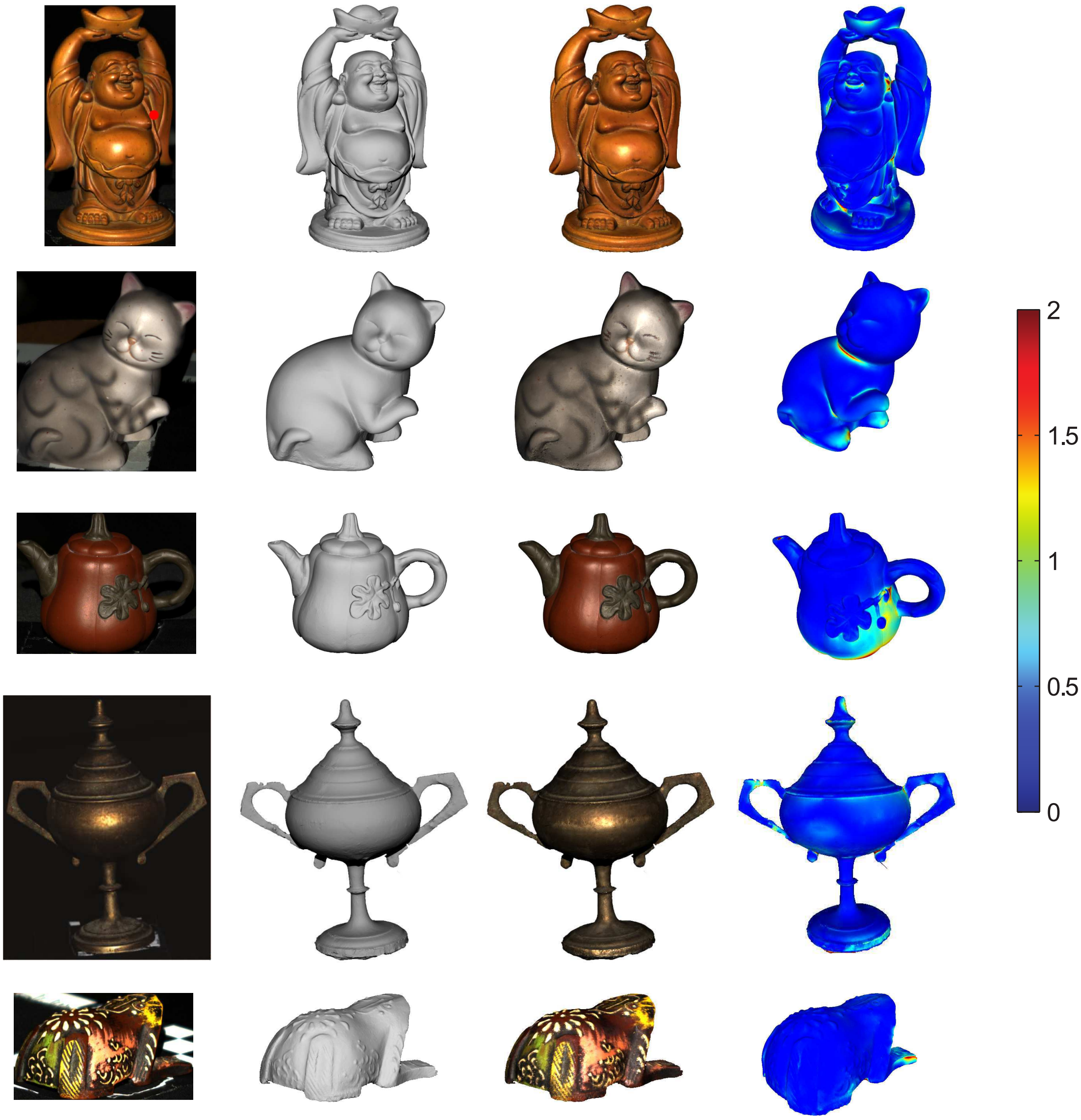

Figure 4.14: Five examples captured by the ringlight system. From left to right: an input image, shape model rendered with Lambertian BRDF, rerendering using the captured BRDF, color-coded shape error.

Example	Buddha	Cat	Teapot2	Cup	Frog
diameter(mm)	120	140	120	120)	90
No. of views	41	35	30	30)	34
median error(mm)	0.36	0.24	0.24	0.29	0.25
mean error(mm)	0.57	0.53	0.66	0.50	0.47

Table 4.1: Quantitative results for ringlight system on five different examples.

Figure 4.15: First column: a photograph; second column: rendering of the recovered example under the same viewing and lighting conditions as the first column; third column: rendering under environment lighting.

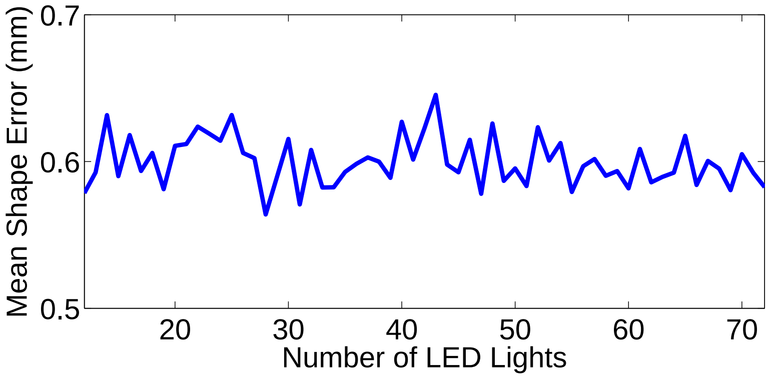

Figure 4.16: Mean shape error of the 'Buddha' example. This error does not change significantly with different number of LEDs.

system. It is obvious that the renderings look quite similar to the original images.

Number of images at each viewpoint

While the ringlight appearance capture system is able to produce high-accuracy results, the number of required input images is huge, given that 72 images are captured for each viewpoint. Thus, a natural question is whether this large number of input images for each viewpoint is an overkill. So we evaluated the accuracy of captured shape and BRDF with different number of input images from each viewpoint.

We first evaluated the shape accuracy on the 'Buddha' example. Figure 4.16 shows the mean shape reconstruction error (in millimeters) as a function of the number of LEDs in each viewpoint. We always chose equal number of uniformly distributed lights on both the outer and inner circles. Since our Fourier series fitting requires at least 5 LEDs from each viewpoint, we begin the plot from 10 lights (5 on each circle). We found the mean shape error did not change significantly for different number of LEDs.

We also evaluated the reflectance accuracy with different number of LEDs. We first measured the BRDF of a green paint after applying it to a sphere of known shape,

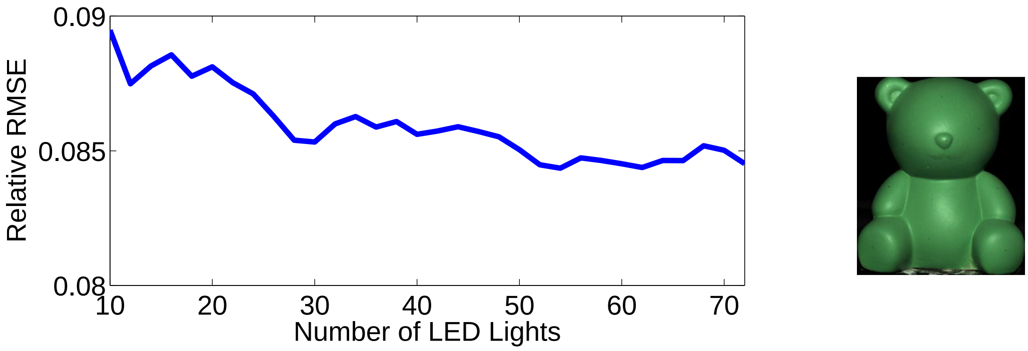

Figure 4.17: Relative RMSE BRDF error of a green paint. This error generally decreases with more LEDs.

and capturing the BRDF from images with calibrated directional lighting. We took this measurement as 'ground truth' and compared our result with it. This experiment is evaluated with another painted figurine shown on the right of Figure 4.17. The left of Figure 4.17 shows the relative root mean square error (RMSE) of our result with different number of LEDs from each viewpoint. We computed RMSE as the following

$$E_{rms} = \left(\sum_{\theta_h, \theta_d, \phi} \frac{(f(\theta_h, \theta_d, \phi) - \hat{f}(\theta_h, \theta_d, \phi))^2}{f(\theta_h, \theta_d, \phi)^2} \right)^{\frac{1}{2}}$$

Here, $f(\cdot), \hat{f}(\cdot)$ are the 'ground truth' and recovered BRDFs respectively. This error converges to about 8.5% when about 30 LEDs are used. So in our experiments, we always used 30 LEDs for the ring-light system.

4.7.3 Comparison with Existing Methods

We compared our results with those obtained from [Alldrin et al., 2008]. and [Hernandez et al., 2008]. We used the same code as the authors. The iterative shape and reflectance optimization in [Alldrin et al., 2008] is complicated and slow. It took over

Figure 4.18: Results according to [Alldrin et al., 2008].

40 hours to compute the results of *one* viewpoint with 72 input images at resolution of 200×350. Figure 4.18 shows the results from [Alldrin et al., 2008]. (a) is a color coded normal map where the x, y, z components of a normal direction are linearly encoded in the RGB channels, e.g. $(x + 1)/2 \rightarrow R$. Shown in (b) is a surface computed from the recovered normal map according to [Wu and Tang, 2006]. (c) is a rendering from novel lighting direction according to the estimated normal and reflectance. We can see clear artifacts in all these images. (d) is the color coded shape error (in millimeters). Notice the error range is from 0 to 5. The median (and mean) shape error is 2.38 (and 2.85) millimeters. The median (and mean) angular error of normal directions is 13.1 (and 17.6) degrees. Figure 4.19 shows the results from [Hernandez et al., 2008] which is designed for Lambertian surfaces, where most of the shape details are smoothed out.

4.7.4 Runtime Efficiency

Our implementation was not optimized for speed. We did all experiments on a computer with 24GB RAM and a 8-core 3.0GHz CPU. At each viewpoint, our matlab code computed azimuth angles in 1 minute, and traced iso-depth contours in 1.5 minutes. Depth propagation took 16 minutes (for 40 viewpoints), and the final shape optimiza-

Figure 4.19: Results according to [Hernandez et al., 2008].

tion took 1 minute. It took about 15 minutes to compute the basis BRDFs from 5,000 samples with ACLS. Our output mesh typically had about two million points with average spatial distance 0.095 millimeters. It took another 45 minutes to compute their BRDF mixing weights. Much of the involved process including azimuth angle computation, iso-depth contour tracing, and BRDF mixing weight computation can be easily parallelized.

4.8 Discussion

In this chapter, wee propose a method to capture both shape and reflectance of real objects with spatially variant isotropic reflectance. while our method is able to capture 3D shapes and reflectance to high accuracy, it has a few limitations. First, our method cannot model anisotropic material. It also cannot handle translucent objects and mirror surfaces. Second, although our method is robust to cast shadows and strong specular inter-reflections with Fourier series fitting, it suffers from diffuse inter-reflections. To resolve this problem, we could replace LEDs by projectors and apply the method in [Nayar et al., 2006] to separate inter-reflection. However, it would significantly complicate the hardware setup. Alternatively, we might iteratively estimate the shape and inter-reflection. Third, our method requires a few accurate initial 3D seed points. While this is available from multi-view stereo for surfaces with texture, it is hard locate corresponding pixels for textureless surfaces. In our experiments, we manually marked some corresponding pixels. A more practical solution is to use laser to project small dots onto the surface to facilitate correspondence finding. Last, our ring-light capture setup contains only two circles of LEDs. Hence, we only capture the BRDF of a point with two different θ_d values. (Note that θ_d is the angle between viewing and lighting directions as shown in Figure 4.10.) Hence, during reflectance capturing, we can only discretize θ_d to two levels, and cannot capture Fresnel effects faithfully. Note this limitation does not apply to the handheld setup. We could increase the number of circles of LED lights, or fit parametric Fresnel terms [Schlick, 1994] to solve this problem.

Chapter 5

Building a Handheld RGBD-M Sensor

5.1 Introduction

Just as stated in Chapter 4, simultaneous capture of both 3D shape and reflectance is an important research area and has a vast range of applications in industries. As a result, different methods, including the one described in Chapter 4, have been proposed in hope of achieving high accuracy of reconstructed shape and reflectance.

While these methods can produce high-quality results under controlled environment, most of them involve sophisticated hardware setup and/or complicated data capture process. For example, [Tunwattanapong et al., 2013] relied on a rotating arm of controlled LEDs for illumination while it typically took about $3 \sim 4$ hours to acquire enough images for the work described in Chapter 4. This complexity of data capture has prohibited any ordinary person who wishes to digitize interesting scenes and objects in his/her daily life from doing so. Another problem of current appearance capture methods is that the data acquisition systems are mostly static and therefore works only on small, movable objects. Thus, a handheld device, which can be used easily and acquire data rapidly, is highly desirable for appearance capture.

The device closest to the desired one is an RGBD sensor such as Microsoft Kinect or ASUS Xtion PRO LIVE, which is capable of both RGB image acquisition and depth sensing. By making use of solely depth images of an RGBD sensor, [Newcombe et al., 2011a] and [Izadi et al., 2011] reconstructed a dense 3D scene model in real time. A recent work [Zhou and Koltun, 2014] further estimated a color map by mapping the acquired RGB images onto the reconstructed geometry. However, the information of surface reflectance or material remains missing in the reconstructed scene model.

To make a handheld RGBD-M [1] sensor for appearance capture, we take one step further by enhancing the RGBD sensor with an additional infrared(IR) camera and a set of IR LED light sources. Without significantly sophisticating the original RGBD sensor or complicating data acquisition, the added camera acquires images under the added light sources. By processing these images together with reconstructed geometry using customized algorithms, material sensing becomes viable.

Viewing from another perspective, this RGBD-M sensor is a computational sensor in computational photography terminology [Raskar and Tumblin, 2009]. A computational sensor does not pass direct measurement, *i.e.*2D images, to the user. Instead, it attempts to understand and analyse a machine-readable representation of the scene using accompanied algorithms. After suitable data processing, the final output of our sensor is a color mesh and surface BRDFs, both of which constitute an essential representation of a scene.

This chapter focuses on building a handheld RGBD-M sensor for casual appearance capture, which, to our knowledge, is the first one of its kind. The structure of this chapter is as follows. We first present the hardware part of the sensor: setup and cali-

[1] 'M' stands for material.

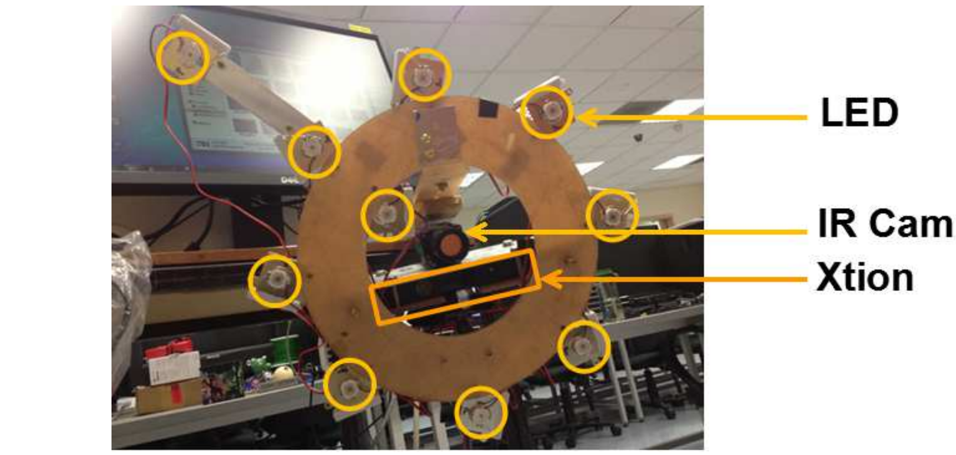

Figure 5.1: The RGBD-M sensor. LEDs are circled.

bration. After describing data acquisition, we switch to the software part and detail the data processing pipeline in the sensor, with special focus on two key algorithms. After that, the sensor is demonstrated to be capable of appearance capture with real-world examples and we summarize this chapter with a discussion of limitations.

5.2 Hardware Setup

Our RGBD-M sensor is built upon an off-the-shelf product, ASUS Xtion PRO LIVE. Apart from this commercial product, several IR LEDs and an IR camera are also used. The sensor is shown in Figure 5.1.

5.2.1 ASUS Xtion PRO LIVE

ASUS Xtion Pro LIVE(we will call it Xtion for brevity), which is similar to Kinect from Microsoft in functionality, consists of an RGB camera, an IR camera and an IR light projector. The RGB camera captures color images, while the IR camera and

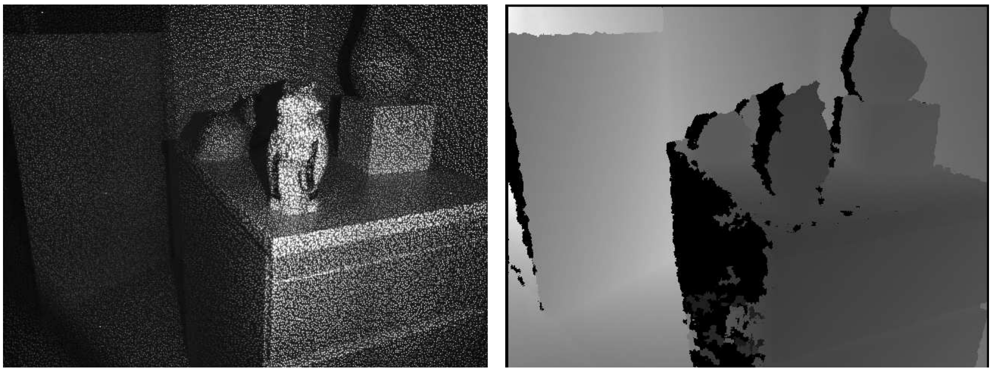

Figure 5.2: Left: an IR image captured by Xtion's IR camera, showing dotted pattern produced by Xtion's projector. Right: a depth image calculated from the IR image on the left. Both images are scaled for visualization.

projector are coordinated to produce depth images. The basic idea of Xtion's depth sensing is structured light. The IR projector projects a special pattern into the scene, while the IR camera on Xtion captures an image of the pattern and calculates a depth image by triangulation after decoding the pattern(Figure 5.2). For simplicity, we say that Xtion consists of a color camera and a depth camera. The depth camera's intrinsic parameters and poses refer to the corresponding parameters of Xtion's IR camera.

It is worth noting that Xtion is not the only choice for building up our sensor. Microsoft Kinect can also be used. However, Xtion is noticeably smaller in size than Kinect and Xtion does not require a dedicated external power supply as Kinect does. This feature makes it quite easy to integrate Xtion into the system.

5.2.2 IR Camera

Xtion is an RGBD sensor providing both RGB images and depth images. To enhance its capability and enable material sensing, we have added an additional IR camera on top of Xtion together with several IR LEDs as shown in Figure 5.1. We deliberately

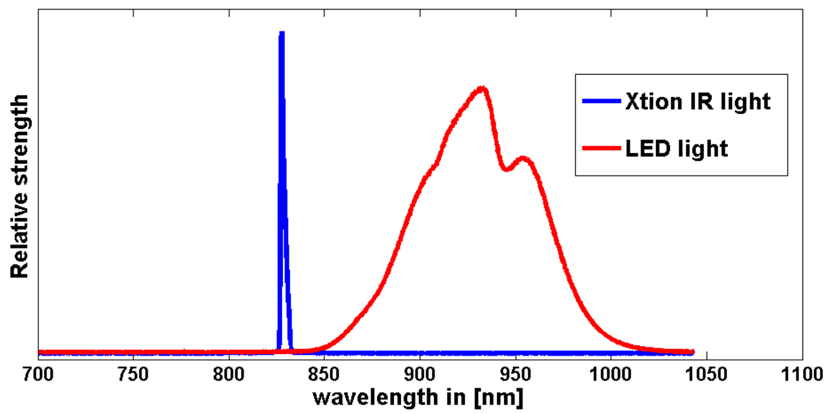

Figure 5.3: The spectra of Xtion's IR projector light and the added IR LED light

choose to work in the IR range to avoid intervention from indoor illumination in the visible light spectrum. The added IR LEDs will actively illuminate the scene, images of which will be acquired by the added IR camera.

There are some practical issues to be handled when building the sensor. First, the added IR camera, whose model is Manta G-145 NIR from Allied Vision Technologies, is sensitive not only in the near infrared spectrum, but also in the visible light spectrum. So undesirable visible light can ruin IR images. Another issue is that Xtion's IR projector also emits IR light which can be seen by the added IR camera. To circumvent these obstacles, we need to carefully choose a filter for the IR camera and select suitable LEDs. We have measured the power spectrum of Xtion's IR projector using a spectroscope. As can be seen in Figure 5.3, the emitted IR light by Xtion's projector has a wavelength around 835nm. Thus, we have adopted IR LEDs which emit light with wavelength around 940nm, to avoid intervention. Accordingly, a bandpass filter is attached to the IR camera's lens, allowing only light around 940nm to pass and blocking other IR light as well as visible light.

Both Xtion and the IR camera are rigidly fixed onto an Aluminium plate, with the IR camera on top of Xtion. A circular plastic plate with a radius of 16cm is connected to the Aluminium plate, with the IR camera at the center. Eight LEDs are evenly distributed on the edge of the circular plate, and another two LEDs are attached to the device differently, as shown in Figure 5.1.

The IR camera and LEDs are coordinated by a microcontroller(PIC16F877A) in such a way that the camera takes an image only when one of the LEDs is turned on. The image data are transmitted back to a desktop computer via an Ethernet cable.

5.3 Device Calibration

The RGBD-M sensor needs to be calibrated both geometrically and radiometrically. Geometric calibration includes estimation of the intrinsic parameters of different cameras, relative poses between cameras, and LED positions. Radiometric calibration involves vignetting calibration, estimation of camera response function, and the brightness of different LEDs. These calibration processes are described below.

5.3.1 Geometric Calibration

5.3.1.1 Intrinsics and Relative Poses

We first calibrate the depth camera of Xtion and the added IR camera. Specifically, we estimate intrinsic parameters of cameras and relative pose between them.

The intrinsic parameters we calibrate includes focal lengths and principal points. Radial and tangential distortions of lens are ignored because they are negligible in our

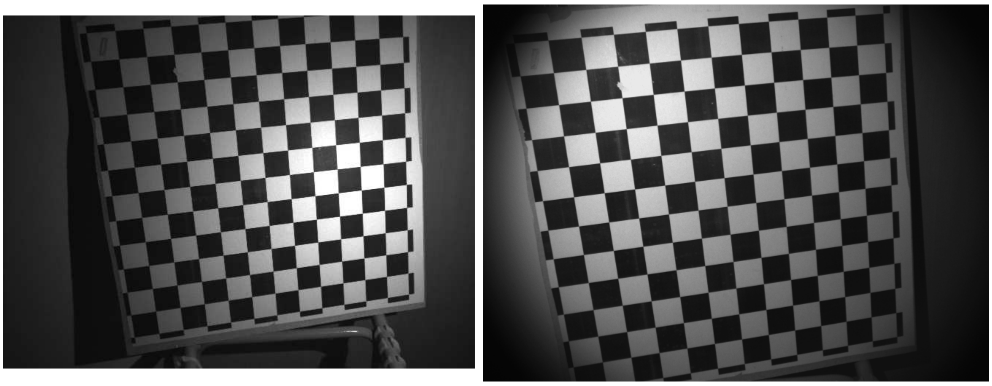

Figure 5.4: A pair of images captured simultaneously for camera calibration. Left: an IR image acquired by Xtion's depth camera with Xtion's IR projector blocked and the scene illuminated by a 835nm point light source; right: an image captured by the added IR camera under a 940nm light source.

situation. The camera intrinsic parameters are

$$K_d = \begin{pmatrix} f_d & 0 & c_{d,x} \\ 0 & f_d & c_{d,y} \\ 0 & 0 & 1 \end{pmatrix}, K_ir = \begin{pmatrix} f_{ir} & 0 & c_{ir,x} \\ 0 & f_{ir} & c_{ir,y} \\ 0 & 0 & 1 \end{pmatrix}, \tag{5.3.1}$$

where f_d and f_{ir} are focal lengths for Xtion's depth camera and the added IR camera respectively. $c_{d,x}$ and $c_{d,y}$ are the principal point for Xtion's depth camera, while $c_{ir,x}$ and $c_{ir,y}$ are the principal point for the added IR camera.

The relative pose between two cameras is a rigid transformation, consisting of a rotation matrix R and a translation vector T. Given a single 3D point and its coordinate X_d in the depth camera's local coordinate system, its coordinate in the IR camera's coordinate system can be calculated as

$$X_{ir} = RX_d + T. \tag{5.3.2}$$

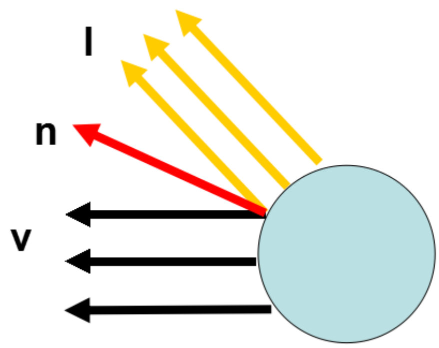

Figure 5.5: Estimating light direction l from a mirror sphere based on orthographic camera model and directional lighting model.

Calibration of the above parameters is an established problem in the computer vision community. We take advantage of [Bouguet, 2013] to solve this stereo camera calibration problem. [Bouguet, 2013] requires multiple pairs of images of a checkerboard pattern as input. To prepare input data, we place a planar checkerboard in front of the device, illuminating it with both 835nm and 940nm light while blocking the IR projector of Xtion. A pair of images are captured from the depth camera of Xtion and the added IR camera for every orientation of the board(see Figure 5.4).

The intrinsic parameters of the RGB camera of Xtion and its relative pose with regard to the depth camera can be calibrated in the same routine.

5.3.1.2 LED Positions

The ten LED bulbs' positions in the IR camera's local coordinate frame need to be determined for BRDF estimation.

Traditional lighting direction calibration assumes directional lighting and an orthographic camera. As illustrated in Figure 5.5, the camera captures an image of a mirror-like sphere. By identifying the specular point in the image and calculating its

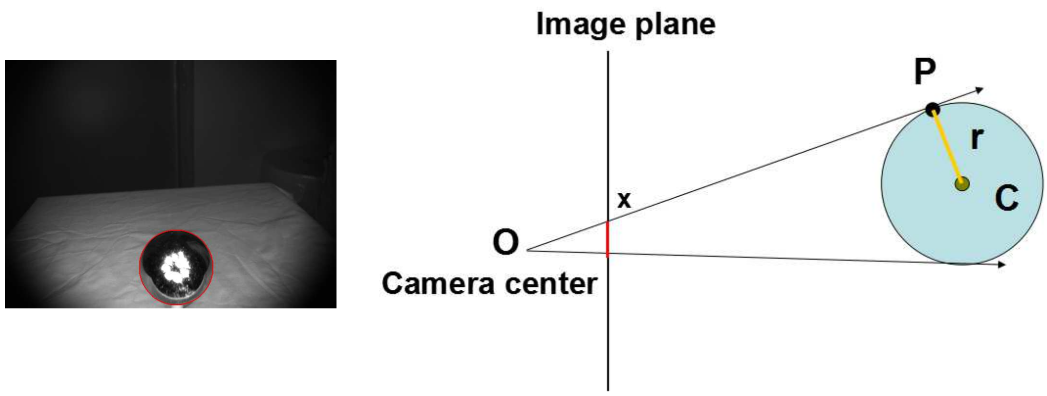

Figure 5.6: Left: an image of a mirror sphere with edge marked out; right: illustrative figure for sphere center estimation.

normal on the sphere, it is easy to infer the lighting direction with the assumption that the normal at the specular point bisects the angle between viewing direction and lighting direction

$$\mathbf{l} = 2(\mathbf{n}^T\mathbf{v})\mathbf{n} - \mathbf{v}. \tag{5.3.3}$$

In our case, however, we have adopted the point light source model and the perspective camera model. Thus, the above calibration method cannot be directly applied. To deal with the problem, we propose a new method for calibrating the positions of a set of point light sources.

We still make use of a mirror sphere for calibration. As a first step, we estimate the 3D position of the sphere center C.

Estimate Sphere Center

To estimate the sphere center, we first acquire an image taken under general IR illumination. Since the camera is a perspective camera, the sphere is projected to a region enclosed by an ellipse which can be easily marked out(see Figure 5.6). As can be seen from Figure 5.6, pixels on the ellipse correspond to 3D points where the view

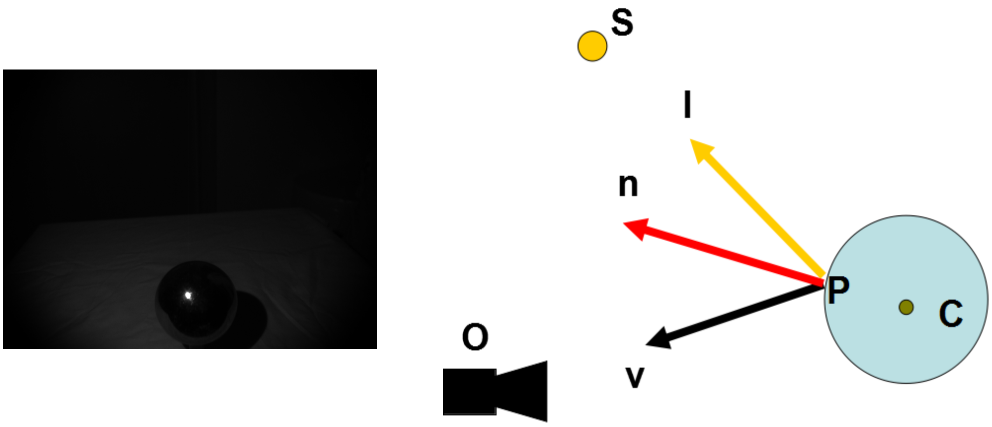

Figure 5.7: Left: an image of the mirror sphere with noticeable specular highlight; right: illustrative figure for explaining constraints on a light source

direction is tangent to the sphere. Assume a pixel on the ellipse has a homogeneous coordinate x, then the direction of the corresponding view direction is $v = K_{ir}^{-1}x$. Given the known radius r of the sphere, the following equation on distance from C to a line should holds

$$\|C - (vv^T)C\| = r, \tag{5.3.4}$$

where C is the sphere center to be estimated. By taking all k pixels on the ellipse into consideration, we can formulate the following optimization problem,

$$\min_C \sum_{i=1}^{k} (\|C - (v_i v_i^T)C\|^2 - r^2)^2. \tag{5.3.5}$$

By solving the above optimization problem, the sphere center C can be located.

Constraints on a Single Light Source

To obtain a constraint on one light source, we capture an image of the same sphere under this light source, as shown in Figure 5.7. Note that the surface normal at specular highlight point on the sphere still corresponds to the bisector of the angle between view direction and lighting direction. Thus, we locate the specular highlight and calculate its view direction as a first step. By intersecting the view direction and the known sphere in 3D space, the specular point P is known. The normal of P is easily computed as $\frac{P-C}{\|P-C\|}$. With known $\mathbf{n}$ and $\mathbf{v}$, the light direction $\mathbf{l}$ can be obtained directly from (5.3.3).

Till now, the light source S can be constraint to lie on a line in space

$$\mathbf{l} \times (S - P) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{5.3.6}$$

where P is the 3D position of the specular point and $\mathbf{l}$ is the local lighting direction for P.

By moving the sphere to different positions, we can have multiple constraints on the position of the LED

$$\begin{aligned} \mathbf{l}_1 \times S &= \mathbf{l}_1 \times P_1 \\ \mathbf{l}_2 \times S &= \mathbf{l}_2 \times P_2 \\ &\vdots \end{aligned} \tag{5.3.7}$$

Then S can be estimated by solving a linear system. The physical meaning is that the LED is the intersecting point of all the lines in space.

Constraints on All Light Sources

While we can solve the positions of all LEDs one by one following the above method, the result can be even more accurate when we optimize the positions of all LEDs simultaneously. Assume the coordinates of all LEDs are $\hat{S}_1, \hat{S}_2, \cdots$ in certain coordinate system, the only parameters to be estimated is a rigid transformation which

Figure 5.8: Some images used for estimating relative positions of all LEDs.

transforms $\hat{S}_i$ to the real position S_i in the IR camera's local coordinate system. Assume we have m LEDs and for each LED, we capture images of a mirror sphere at n different positions, we can define the following error function

$$f(R,T) = \sum_{i=1}^{n}\sum_{j=1}^{m} dist(R\hat{S}_j + T, l_{ij}, P_{ij}) \tag{5.3.8}$$

where $dist(S, l, P)$ is the squared distance from the light source S to a line passing the point P with a direction l

$$dist(S, l, P) = \|S - P - (ll^T)(S - P)\|^2. \tag{5.3.9}$$

This is a non-linear least squares problem and can be solve by Matlab function lsqnonlin.

The remaining problem is how to represent all LED positions in a single coordinate system. We simply solve the sub-problem by placing the LED plate on a checkerboard and take images of them from multiple viewpoints with a camera whose intrinsics are pre-calibrated(see Figure 5.8). Camera poses can be easily recovered given the size of squares in the checkerboard pattern. By marking out LEDs in all images, we can triangulate and solve the positions of all LEDs in the same coordinate system. These coordinates are exactly $\hat{S}_1, \hat{S}_2, \cdots$, which is what we need.

5.3.2 Radiometric Calibration

Experimental results show that the IR camera's pixel value is proportional to the irradiance going into the camera. However, images from the added IR camera suffer from noticeable 'vignetting effect'. In addition, the brightness of different LEDs are different. We briefly describe calibration procedures for them as follows.

5.3.2.1 Vignetting Effect

Photographed images generally exhibit a radial falloff of intensity from the center of the image. This is the so-called 'vignetting effect'. This effect has to be removed if the image is going to be used for material estimation. We have adopted the following model for vignetting

$$V(r) = 1 + \sum_{n=1}^{3} \beta_n r^{2n}, \tag{5.3.10}$$

where r is the distance of a pixel from the image center, $\beta_1, \beta_2, \beta_3$ are the three parameters to be estimated, and V is the vignetting value such that

$$\hat{I}(x, y) = I(x, y)V(r(x, y)), \tag{5.3.11}$$

where $I(x, y)$ is the expected pixel value at (x, y) while $\hat{I}(x, y)$ is the observed pixel value due to vignetting.

There are different methods on vignetting removal [Goldman and Chen, 2005; Kim and Pollefeys, 2008]. Here we adopt the method of [Kim and Pollefeys, 2008] due to its robustness. We illuminate a diffuse white board with a fixed point light source and capture images of this white board from different viewpoints. Since the white board is planar, the image of the board can be easily warped into another image by homography, which is estimated by locating the four pairs of corresponding corners in two images. The key observation is that for the same point on the white board, its pixel values in different images should be stay the same regardless of the view direction. Thus we have the following equation

$$\frac{\hat{I}_1(x_1, y_1)}{V(r(x_1, y_1))} = \frac{\hat{I}_1(x_2, y_2)}{V(r(x_2, y_2))}. \tag{5.3.12}$$

where $\hat{I}_1$ and $\hat{I}_2$ are two images, (x_1, y_1) and (x_2, y_2) are the projections of the same 3D point on the two images respectively. This equation can be viewed as a constraint on the vignetting function $V(r)$. More constraints can be added as even more points are involvecd. For details, please see [Kim and Pollefeys, 2008].

5.3.2.2 Relative Brightness of LEDs

The brightness of the ten LEDs are slightly different and need to be calibrated in order to normalize the captured images illuminated by different LEDs. The calibration process is quite simple. A diffuse white board is placed in front of the sensor and a set of IR images are captured under the illumination of different LEDs. After removing vignetting for each image, the observed value for a specific pixel p is

$$I_p = \rho \frac{(l_{i,p}^T n) L_i}{d_{i,p}^2}, \tag{5.3.13}$$

where ρ is the albedo across the white board, n is the surface normal of the white board, $d_{i,p}$ is the distance from the i^{th} LED to the point on the white board corresponding to the pixel p, $l_{i,p}$ is the light direction for the specific point under the i^{th} light source, L_i is the brightness of the i^{th} LED.

Since the LED positions are pre-calibrated, and the pose of the white board can be easily calibrated by marking out its four edges with known lengths, the brightness of an LED can be calculated based on the above equation

$$L_i = \frac{I_p d_{i,p}^2}{\rho(l_{i,p}^T n)}. \tag{5.3.14}$$

Thus, by assuming $\rho = 1$, we can calculate the brightness for each LED based on the pixel values at a single pixel. For robustness, the results from multiple pixels can be averaged in practice. After calibrating the brightness for each LED, the input IR images can be normalized by dividing the image by L of the corresponding LED.

5.4 Data Capture

Data capture is quite easy with the RGBD-M sensor. Since material sensing works in the IR range, no darkroom environment is required. The user only needs to aim the sensor to an object of interest and move around it. Typical data acquisition time is only 1 min for an object a desk.

Three data streams will be transferred to a computer simultaneously during the scanning stage: a depth image stream and a color image stream from Xtion, an IR image stream from the added IR camera.

An inevitable problem in data capture is synchronization. There are three cameras within the sensor, two of which are from Xtion and can be regarded as naturally synchronized. To synchronize the RGB camera of Xtion and the added IR camera, which run in different frame rates (30fps vs 22fps), we flash a beam of light into both cameras and assume the first IR image and RGB image(called 'head image') seeing the flash are taken simultaneously. Since each of the two streams of images is associated with a timestamp from its camera itself, we can subtract the timestamp value of the head image from the timestamp of each image. In this way, all images are aligned to the same timeline.

The problem for the above synchronization method is that the pair of head images are are not perfectly synchronized. However, the difference is guaranteed to be smaller than $\frac{1}{30}$sec. As long as the sensor moves not too fast, the effect of imperfect synchronization is negligible.

It is worth noting that perfect synchronization can be achieved by further modifying the hardware, which is beyond the scope of this chapter.

5.5 Data Processing

Data processing with suitable algorithms is an essential component for the RGBD-M sensor. As stated in the previous section, the RGBD-M sensor's hardware captures three streams of images. On the other hand, this sensor's software will transform input image streams into a compact scene representation with both shape and reflectance.

In this section, we will present the whole pipeline of data processing and detail two key algorithms in data processing.

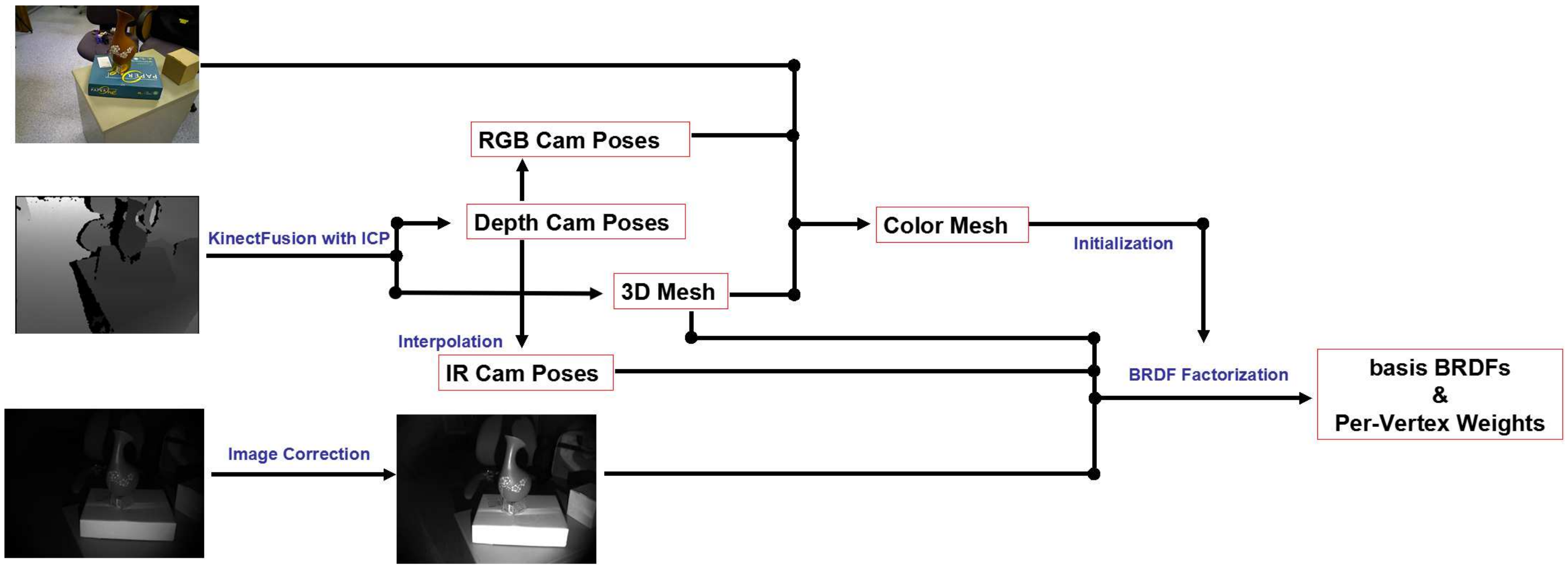

Figure 5.9: The computation pipeline for the RGBD-M sensor

5.5.1 Pipeline

Figure 5.9 shows how the sensor does computation to process input data and turn them into a compact representation of a scene or object.

As can be seen in the pipeline, we first run the customized algorithm 'KinectFusion with ICP', which is described in Section 5.5.2, on the depth stream to obtain both a 3D mesh model and poses for the depth camera. After obtaining the depth camera's poses, the RGB camera's poses can be easily calculated by applying a rigid transformation on the depth camera's poses. With the help of color images and the RGB camera's poses, a per-vertex color can be estimated for the 3D mesh.

On the material sensing side, the acquired IR images are first corrected to remove vignetting and normalized by the corresponding LED brightness. Since the IR camera is not synchronized to the depth camera in a frame-wise way, its camera poses for each IR image can only be obtained through interpolation. Specifically, the pose for one IR image is linearly interpolated from the poses of two depth images temporally close to it. The interpolation of the translational component is straightforward. For the rotational component, the interpolation is carried out on the quaternion representation of rotation. Given the camera poses for each IR image, 3D vertices on the model can be projected onto IR images and calculate BRDF values. As will be seen in Section 5.5.3, the BRDF of any vertex is modelled as a linear combination of several basis BRDFs. Though BRDF factorization, the reflectance model, including basis BRDFs and per-vertex weights, can be obtained. It is worth noting that the color information can be used for initializing BRDF factorization.

5.5.2 KinectFusion with ICP

KinectFusion, which has been presented in [Izadi et al., 2011; Newcombe et al., 2011a], is a method for camera tracking and scene reconstruction using only depth images. The method takes a sequence of depth images as input and reconstruct a 3D model as well as the camera pose for each input depth image. specifically, the method maintains a volumetric model of a scene. When a new depth image arrives, it will render a depth image from the model, based on its best guess of the pose of the incoming depth image. The new depth image is then aligned with the rendered depth image, giving an better estimation of its camera pose. After that, the volumetric model is updated by the newly aligned depth image. This process is repeated till all depth images are integrated into the volumetric model. Finally, a mesh model is extracted from the volumetric model.

Besides KinectFusion, there are other camera tracking methods available. However, those RGB-based methods, such as [Endres et al., 2012; Klein and Murray, 2007; Newcombe et al., 2011b], perform poorly when there is not much texture in the scene. By making use of solely depth images, KinectFusion is much more robust.

Despite its robustness, KinectFusion still cannot avoid the notorious problem of drifting, which means that small errors in frame-to-frame tracking could accumulate and result in a noticeably large tracking error for a long sequence.

Drifting can cause much trouble for our application. Since the depth camera of Xtion runs at a frame rate of around 30fps, a single minute's data capture will yield a depth sequence of almost 1800 frames. Simply running KinectFusion on this long sequence will produce inaccurate camera poses, which in turn, ruin the shape model.

To reduce drifting, we have adopted the following strategy. Instead of integrating every depth image into a single volumetric shape model, we partition the long sequence into multiple short sequences with a length of around 200 frames. For each short sequence, we run KinectFusion on it and obtain a mesh model which only represents part of the scene. Then these partial mesh models are aligned to each other by the iterative closest point(ICP) algorithm. At the same time, camera pose for each frame is transformed properly. To extract the final mesh model, Poisson surface reconstruction [Kazhdan et al., 2006] is used to merge all the aligned partial models.

By truncating a long sequence into small chunks, we have partially eliminated the possibility of accumulated tracking error. In fact, since KinectFusion is run only on a short sequence, the estimated camera poses and partial models are reasonably accurate. Alignment of partial models by ICP seldom introduces large error.

5.5.3 BRDF Estimation

Given the shape model, camera poses and LED positions, it is possible to estimate the BRDF values at each surface point by the following equation

$$f(\omega_{in}, \omega_{out}) = \frac{I}{\mathbf{n}^T \mathbf{l}} \frac{1}{d^2}, \tag{5.5.1}$$

where d is the distance from a vertex to an LED bulb. However, for each surface point, we do not have every combination of ω_{in} and ω_{out}. In order to facilitate BRDF estimation for every surface point, additional constraints are required.

First of all, we simplify the BRDF model by assuming isotropy and half-vector symmetry. As we have already discussed in Chapter 3, isotropy is ubiquitously observed in real-world materials and eliminates one variable of BRDF. Half-vector symmetry also significantly simplifies the BRDF model by eliminating another variable while still keeping the simplified BRDF a good approximation of the original one. The simplified BRDF for a surface point is mathematically expressed as

$$f(\omega_{in}, \omega_{out}) = \hat{f}(\theta_h, \theta_d), \tag{5.5.2}$$

where the definitions of θ_h and θ_d can be found in Figure 3.3.

The second assumption we make is that a real-world scene consists of only a few different materials. This is especially true for artificial scenes such as an indoor environment. Thus, the BRDF at each surface point can be expressed as a linear combination of several different basis BRDFs

$$H_{n\times m} = B_{n\times k} W_{k\times m}, \tag{5.5.3}$$

where each column of H is the BRDF of one surface point, each column of B is a basis BRDF, and each column of W is the non-negative weight vector for a surface point.

Estimating B and W from H is a standard non-negative matrix factorization(NMF) problem since both B and W are non-negative matrices. Given a reasonable initial estimation of k basis BRDFs, we can alternate between estimation of W and B till the Frobenius norm of $H - BW$ changes very little. Since the error is non-increasing at each iteration, B and W are guaranteed to converge to a local optimal.

However, the local optimal obtained by a simple NMF does not necessarily have to be a reasonable solution. Typically in our experiments, there is no observation for almost 90% of entries in H. These entries place no constraint on B and W. In order for B and W to converge to a reasonable solution, we follow [Lawrence et al., 2006] and enforce several different constraint on B and W.

Monotonicity For a BRDF $\hat{f}(\theta_h, \theta_d)$ with fixed θ_d, we force the BRDF value to be monotonically decreasing as θ_h goes down from 0 to $\frac{\pi}{4}$. This hard constraint can be implemented as a set of linear inequalities on entries of B.

Smoothness A BRDF is typically a smooth function. Thus, we add a smoothness term to the original objective function. The smoothness term is defined as the squared sum of all first order and second order derivatives of each basis BRDF $\hat{f}(\theta_h, \theta_d)$. Its physical meaning is that the value of every basis BRDF should change too fast as θ_h and θ_d vary.

Unity and Sparsity Unlike the above mentioned two constraints on basis BRDFs B, unity and sparsity are two related constraints imposed on each column of W. It is assumed that the BRDF at a surface point is a linear combination of some basis BRDFs. In most cases, this BRDF is dominated by a single basis BRDF. That means if w is a column of W, then only one entry of w is close to 1 and the rest are close to 0. This intuitive idea can be decomposed into a unity constraint and a sparsity constraint.

The unity constraint is that the sum of w is close to 1. In other words, the following term needs to be minimized

$$(1 - \sum_{i=1}^{k} w_i)^2. \tag{5.5.4}$$

The sparsity constraint is that each entry of w is close to 0 or close to 1. Mathematically, we minimize the following function

$$\sum_{i=1}^{k}(min((w_i - 0)^2, (w_i - 1)^2)) \tag{5.5.5}$$

Given the above hard and soft constraints, B and W will converge to a reasonable solution.

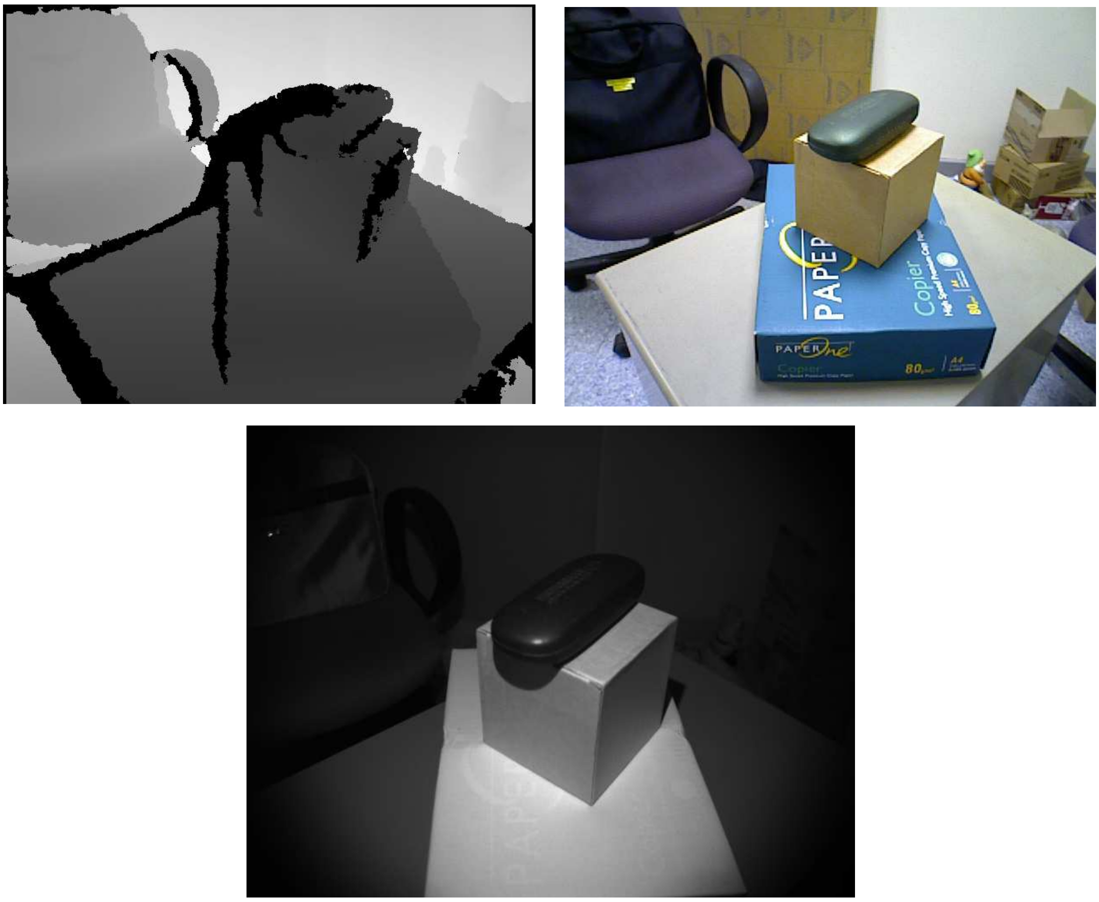

Figure 5.10: Three sample images captured by the RGBD-M sensor. Top row from left to right: depth image(scaled for better visualization) and RGB image; bottom row: IR image.

5.6 Experimental Demonstration

In this section, we demonstrate how the handheld RGBD-M sensor works by two examples and a comparison with KinectFusion.

Figure 5.11: Left: cropped mesh of eyeglass case and a box; right: mesh with color information.

5.6.1 First Example

In the first example, we try to capture the appearance of an eyeglass case together with a paper box underneath it. Figure 5.10 shows a set of representative input images of the scene. It can be easily observed from Figure 5.10 that the eyeglass case, which is made of plastic, exhibits some specular highlights while the paper box looks purely diffuse.

After scanning the scene with the RGBD-M sensor, the proposed method which is described in Section 5.5.2 is run on the input depth sequence to obtain a 3D mesh representation of the scene. Since we are only interested in the eyeglass case and the underlying box, so we manually remove the rest parts of the scene. The remaining mesh is shown in Figure 5.11.

With the estimated camera poses of the depth camera and its relative pose with regard to the RGB camera, we project each vertex of the mesh onto every RGB images.

the color information for each vertex is obtained in this manner. See Figure 5.11 for the result.

To estimate BRDFs for the model, vertices are projected onto IR images to collect BRDF values for the matrix H in (5.5.3). The factorization of H requires an initialization of basis BRDFs. Here we run the kmeans clustering algorithm based on the color at each vertex to classify these vertices into 2 groups. We assume vertices belonging to the same group share a common BRDF. Thus, one basis BRDF can be easily estimated based on all the vertices in a single group. In this experiment, 2 basis BRDFs are estimated.

After obtaining the observation matrix H and initializing basis BRDF matrix B, factorization is applied and produces results shown in Figure 5.12.

It can be shown that the material of the eyeglass case, which is made of plastic, is dominated by a specular BRDF, while the BRDF of the paper box is quite diffuse.

It is worth noting that the weights for the region that is on the top of the box and close to the eyeglass case, are clearly incorrect. Pixel values observed in this region are not only the result of direct illumination from the IR LED. Indirect lighting from the eyeglass case also contributes to its appearance. Since this interreflection in this concave region is not modelled by our data processing pipeline, such results can be expected.

Another artefact in the weight maps is the frontal corner of the box. Weights at this corner show inconsistency with its neighbouring regions. This is because the reconstructed mesh is not perfectly aligned with IR images due to small errors in camera tracking and the mesh itself. The negative effect of misalignment is exaggerated in regions where depth varies quickly, such as the corner of the box.

Since the RGBD-M sensor together with its accompanied algorithms recovers both

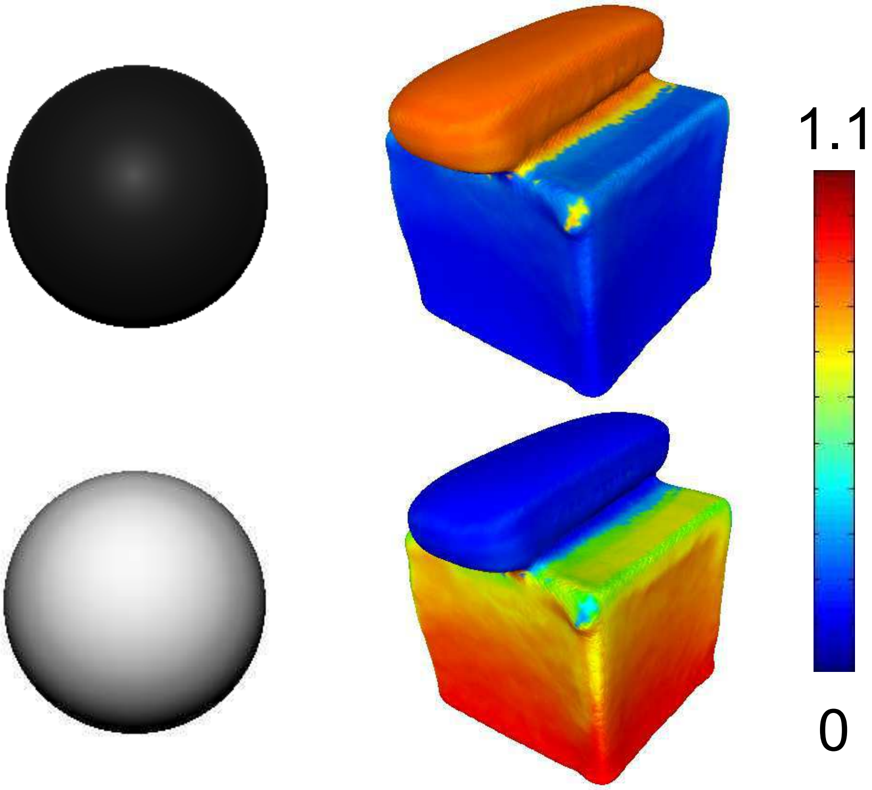

Figure 5.12: Left column: two estimate basis BRDFs rendered with a sphere under directional light; right column: weight maps for two basis BRDFs. Red indicates larger weights while blue indicates smaller values.

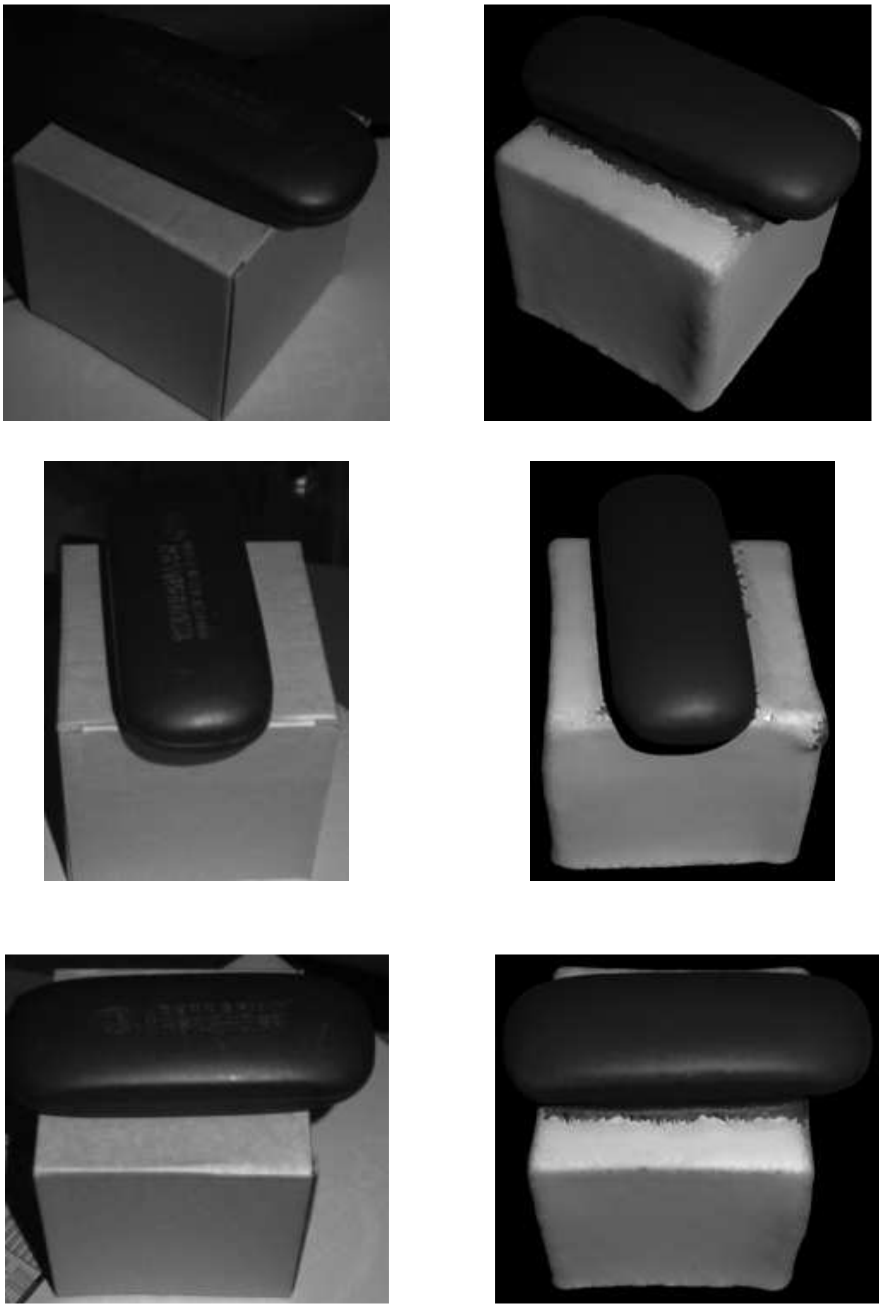

Figure 5.13: Left column: original IR images; right column: renderings of the mesh under the same light conditions as the original images. All images are cropped for visualization.

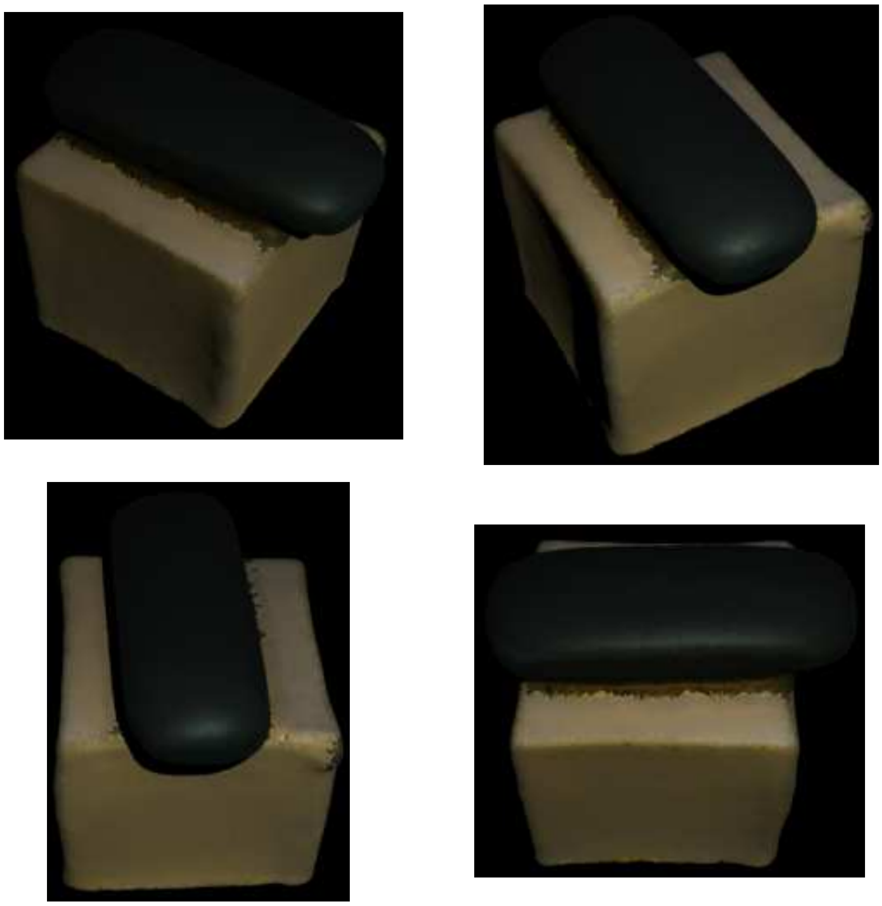

Figure 5.14: 4 renderings of the captured appearance model.

shape and BRDFs, we can produce renderings of the scene and compare them with the original input IR images. Figure 5.13 shows the comparison. As can be seen, the renderings look quite similar to the original images. The smooth specular highlight on the eyeglass case is faithfully reproduced. An issue for the rendering is that the carved words at the top side of the eyeglass case is not seen in the rendered image. This is because KinectFusion tends to smooth out details such as the carved words.

Per-vertex color can be added back to the mesh to render color images as shown in Figure 5.14.

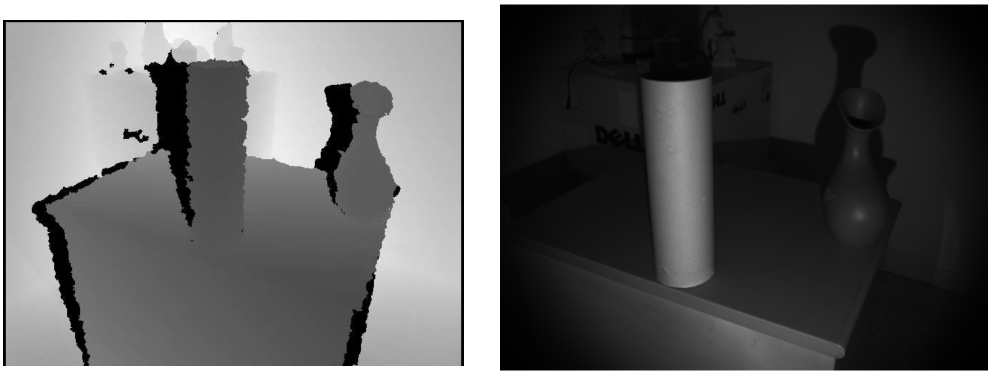

Figure 5.15: Left: a depth image(scaled for visualization); right: an IR image

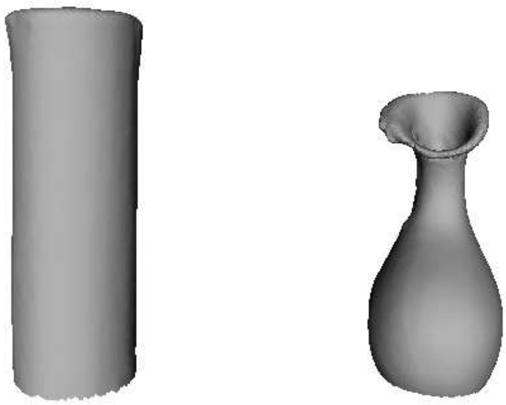

Figure 5.16: The reconstructed mesh with all background parts removed manually

5.6.2 Second Example

The second example involves a vase and a cylindrical pot. In this example, we scan one side of the scene and just as in the first example, we keep only regions of interest. However, we intentionally disable the RGB camera to demonstrate that the material sensing can still work without color images.

Figure 5.15 shows two sample images of the input streams. The reconstructed mesh is shown in Figure 5.16.

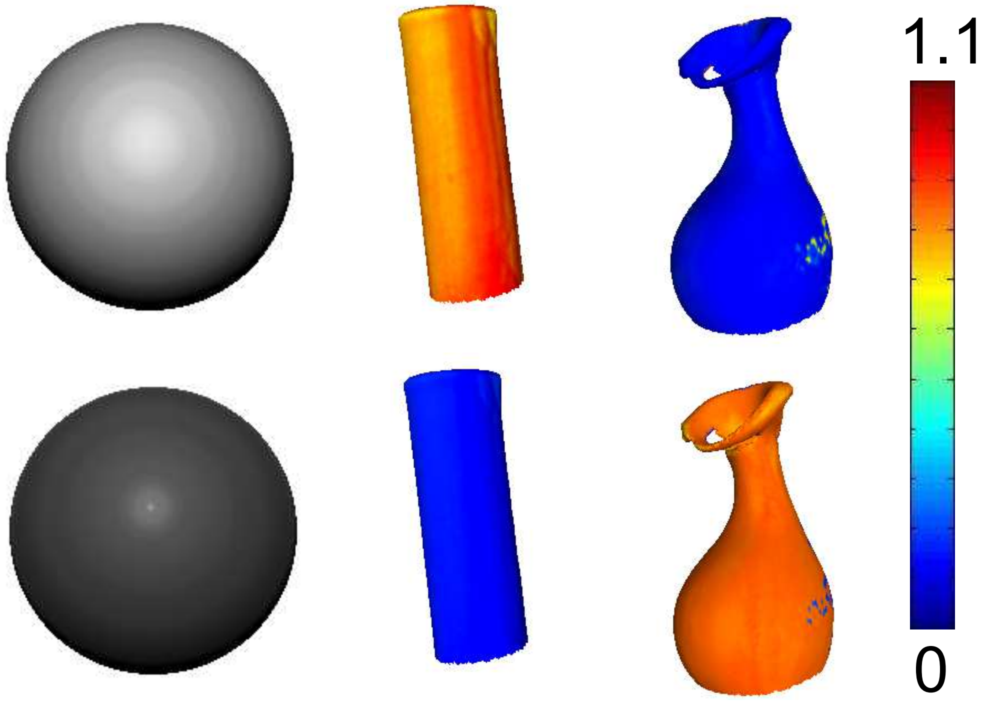

Figure 5.17: Top left visualizes the first basis BRDF by rendering a sphere with it; Top right visualizes weight at each vertex for the first basis BRDF. The second rows shows the results for the second basis BRDF.

To initialize basis BRDFs in the first example, color information is used for identification of vertices with similar materials. However, color is not the only prior for this task. For example, we can use the intensity profile of each vertex under all IR illuminations and group them by affinity propagation [Frey and Dueck, 2007]. This is another possible solution. Also in this example, by removing the background, the two objects with different materials are naturally separated. We utilize this shape prior by grouping together vertices of the same connected component of the mesh, to initialize basis BRDFs.

The final results of BRDF estimation is shown in Figure 5.17. It can be seen that basis BRDFs are successfully recovered. A noticeable artefact is on the right side of the vase where the weights are different from other parts of the vase. This is because this region of vase is made of a different materials from both basis BRDFs(See the last row of Figure 5.18).

We also render the mesh with BRDFs under original illumination and compare them with original IR images. Figure 5.18 shows the results. The renderings resemble corresponding IR images except the right side of the vase where there is a third material which is not modelled.

The above results demonstrates that the material sensing ability does not require color information as a necessity. In fact, given the captured BRDFs, it is quite easy for any user to modify the colors of the materials as they wish. See Figure 5.19 for illustration.

5.6.3 Comparison with KinectFusion

While the RGBD-M sensor relies on KinectFusion for camera tracking and 3D reconstruction, it provides more information that a Kinect sensor: BRDFs. To see this, a comparison is provided in the following.

A blue gymball, shown in Figure 5.20, is captured as scene of interest. By the sensor described in this chapter, shape, color and BRDF of this gymball are simultaneously obtained and thus, an image of this gymball can be rendered which faithfully reproduces the specular highlight of the gymball(left of Figure 5.21). However, by Kinect with only KinectFusion algorithm, only shape and color can be recovered. By making assumption of Lambertian surfaces, the rendered image of the gymball lacks that highlight, which violates our visual perception of the scene.

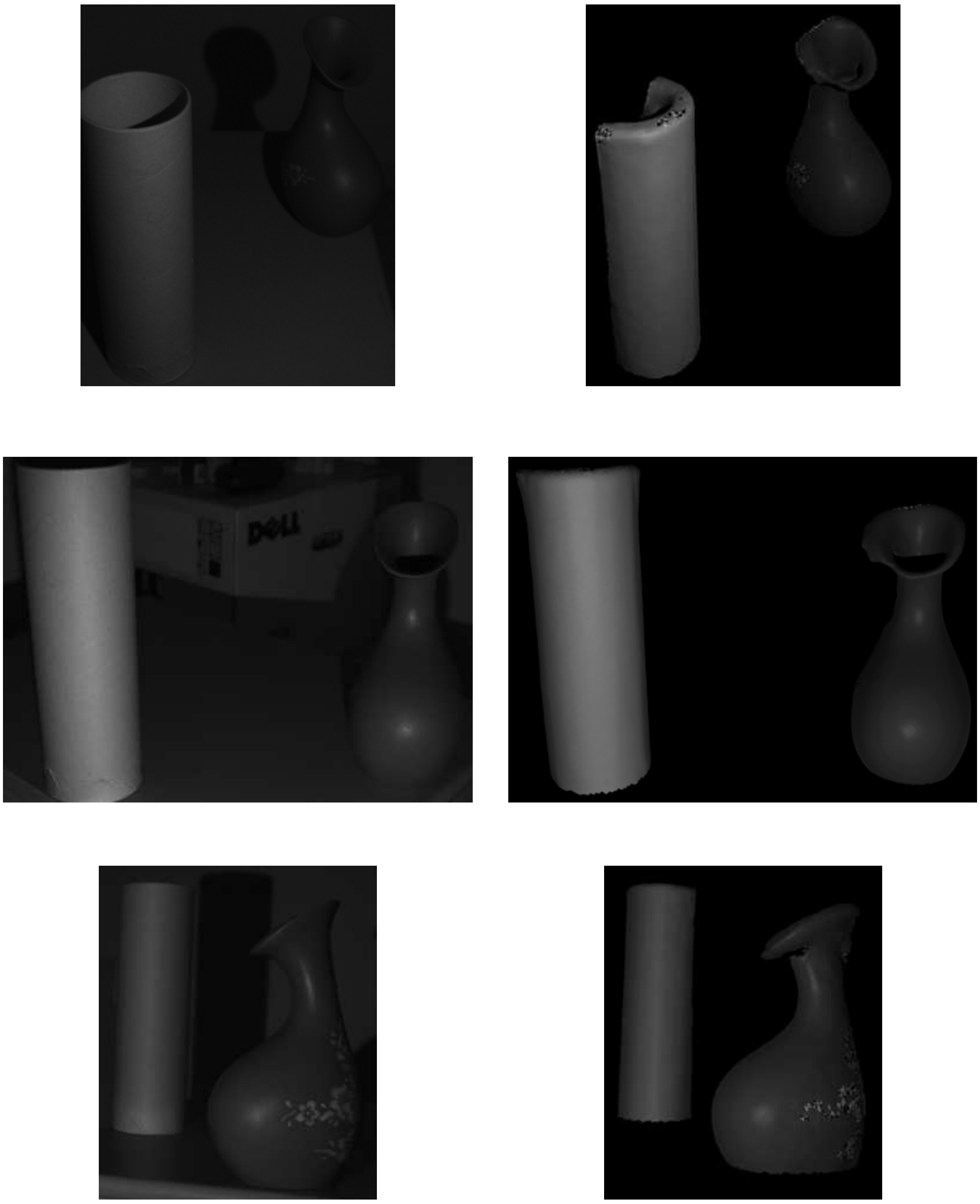

Figure 5.18: Left column: original images; right column: renderings of the mesh.

Figure 5.19: First row shows what the real vase and cylindrical pot look like; the second row shows the rendering results after assignment different colors to the basis BRDFs.

Figure 5.20: An RGB image of a blue gymball

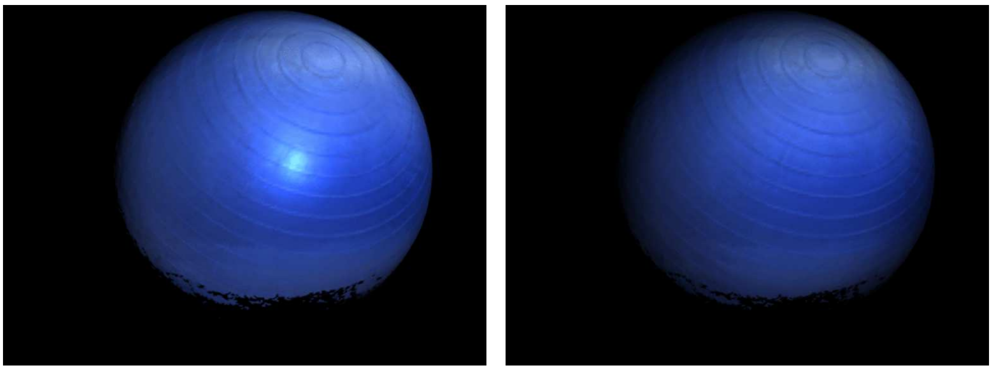

Figure 5.21: On the left is an image of the gymball rendered with the shape, color and BRDF information from the RGBD-M sensor, on the right is an image rendered with only shape and color information from KinectFusion.

5.7 Summary and Discussion

In the chapter, we have presented the handheld RGBD-M sensor we had built for capturing shape and material simultaneously in a **casual** way. Such a device is important in three different areas. First, when viewed as an appearance capture system, this sensor is much smaller in form factor and simpler in operation than other systems. Thus, it has the potential of enabling non-professionals to perform appearance capture in daily life.

Second, viewing from the aspect of scene reconstruction, this system has the advantage of material sensing in addition to recovering shape, while many other methods focus on shape and texture only.

Third, our device is also a computational sensor in computational photography terminology. By coordinating the hardware for data acquisition and software for customized computation, this 'camera' produces an essential representation of a scene or object, as opposed to a traditional camera capturing photos. This 'camera' provides much potential for image/video editing.

Given the similar ringlight configurations of last chapter and this chapter, it is a natural idea to use the method of last chapter with the sensor here. However, this is hardly achievable because of three issues. 1. To estimate azimuth angle for a view as in last chapter, we have to capture images of that view under different illumination while keeping camera fixed. In the case of a handheld device, the camera is constantly moving. Even if we purposely hold the device steadily, it is impossible to get per-pixel alignment between images due to trembling hands. This results in poor azimuth estimation, especially in regions where there is a sharp change in intensity. 2. The method described in last chapter requires highly accurate camera poses. This is achieved by placing a checkerboard beneath the object for localization. However, in this chapter, we try to deal with more general scenes and camera poses is given by KinectFusion. While KinectFusion is relatively robust, its accuracy is lower than what we would expect. The less accurate camera poses will severely ruin the shape results. 3. Depth propagation in last chapter relies on an underlying assumption: lighting direction is directional and the camera is orthographical. However, in the sensor setup, the LEDs will be close to the scene (0.5m 1.5m) and the IR camera has large field-of-view (to match the fov of ASUS camera). In this case, the estimated iso-depth contours are theoretically wrong and hardly a good approximation.

Limitations While we have demonstrated the sensor with real-world examples, it has some limitations. First, KinectFusion, which is the shape reconstruction method we rely on, tends to smooth out shape details as can be seen in the eyeglass case example. Second, since the sensor is constantly on the move, it is difficult to perform exposure stacking for high dynamic range imaging. Thus, over-exposure and under-exposure are inevitable. This will affect the accuracy of BRDF estimation. Third, small errors in camera pose estimation will cause misalignment between IR images and 3D mesh. In this case, the same vertex can be projected to a wrong image points. This is especially problematic in regions full of shape details or fast changing textures. Lastly, as in most appearance capture methods, inter-reflection is not modeled.

Since all the limitations are critical to high accuracy , further research can be done in this direction to solve them so as to improve results. We believe that the improved accuracy could lead to a leap from a consumer level sensor to a professional device.

Chapter 6

Conclusions

Shape reconstruction and reflectance estimation are quite important in computer vision and computer graphics. A suitable BRDF model is the key to both. In this thesis, we make extensive use of reflectance symmetries in capturing appearance of objects. In addition, we built a handheld device for non-professionals to perform appearance capture easily.

In Chapter 3, we had proposed a novel constraint based on a theoretical result about half-vector symmetry, to fully resolve the GBR ambiguity in uncalibrated Lambertian photometric stereo. the auto-calibration algorithm based on this result is proved to be accurate and robust. In Chapter 4, we had combined sparse 3D points, which are estimated by SfM, and iso-depth contours, which are estimated by assuming isotropic BRDFs, to reconstruct a 3D shape. This highly accurate shape enables accurate estimation of BRDFs in the second phase. Thanks to isotropy, such a system achieves high accuracy while maintaining generality. In Chapter 5, we had presented an RGBD-M sensor, which is capable of color sensing, depth sensing and most importantly, material sensing. This sensor is a combination of the dense tracking and mapping technique and BRDF estimation. Reflectance symmetries help constrain BRDFs to a reasonable solution. This sensor is the first one of its kind to our knowledge.

Despite the merits that have been mentioned in each chapter, there are a few issues unsolved and can serve as future work.

First, while isotropy, which is the underlying assumption of all three chapters, is a generally observed property of real-world materials, there are also interesting anisotropic materials. There have been methods on capturing appearance of anisotropic materials. However, the hardware setup is quite sophisticated. Developing a simpler system for anisotropic material acquisition is quite meaningful.

Second, none of the works in this thesis models inter-reflection. Inter-reflection is an annoying phenomenon in both photometric stereo and BRDF estimation. Current techniques can hardly handle it easily. However, recent advances in computational photography, such as [Wu et al., 2012], might shed some light on this problem. [Wu et al., 2012] has built an ultra-fast imaging system which can 'photograph' the transmission of light. Since it will take longer time for the light bounced multiple times to be received by the 'camera' than that bounced only once, it might be possible to only collect light directly returned from the surface. This new technique is a promising direction in solving inter-reflection.

Third, the geometry produced by KinectFusion lack details while the LEDs and IR camera on the sensor naturally make up a photometric stereo system which has the potential of refining the initial shape. A major problem for this direction is the correspondence problem. As we know, photometric stereo requires almost perfect alignment of pixels. However, due to small errors in camera poses and initial shape, it is not easy to establish good correspondence between pixels from different images. To solve this problem, a global optimization of camera poses and shape might be required.

Fourth, our portable sensor is not a real-time system. If material sensing can run in real-time, then it will be possible to involve the user in the capturing loop. The captured material will be visualized as the user scans a scene so that the user could decide whether some parts of the scene need to be scanned again. This will profoundly increase the strength of this sensor.

Appendix A: Proof of Proposition 1

Proof. Consider the problem on the projective plane. Apart from a BRDF value, each point is also associated with a shading value $\mathbf{n}^\top \mathbf{s}$ and an intensity value I which is the BRDF multiplied by the shading. For a bivariate BRDF, the BRDF map consists of a set of ellipses around $\mathbf{h}$ with a symmetry axis vs. It is clear the shading map is also symmetric about vs. So the intensity map is symmetric about vs and its maximum $\mathbf{m}$ must lie on vs.

When the GBR ambiguity is incurred, the transformed light direction $\hat{\mathbf{s}}$ will keep lying on the line vs based on equation 3.3.3, and so is $\hat{\mathbf{h}}$. However, the intensity maximum will experience a translation (μ, ν) and a scaling λ. If (μ, ν) is not parallel to vs, the intensity maximum will shift away from vs to $\mathbf{m}_1$. Now we consider the point $\mathbf{m}_2$, which lies on the other side of vs and is the mirror point of $\mathbf{m}_1$ across vs. Its intensity is smaller than that of $\mathbf{m}_1$ and their transformed shading values are equal: $\mathbf{m}_1^\top \hat{\mathbf{s}} = \mathbf{m}_2^\top \hat{\mathbf{s}}$. Thus, their transformed BRDF values are different. On the other hand, these two points have the same transformed half angle and thus they belong to the same row of the GBR distorted 2D BRDF slice. So the low-rank structure is broken.

Even if (μ, ν) is parallel to vs, the GBR transformation will still break the low-rank structure of the BRDF slice from another image whose lighting direction is not coplanar with $\mathbf{v}$ and $\mathbf{s}$.

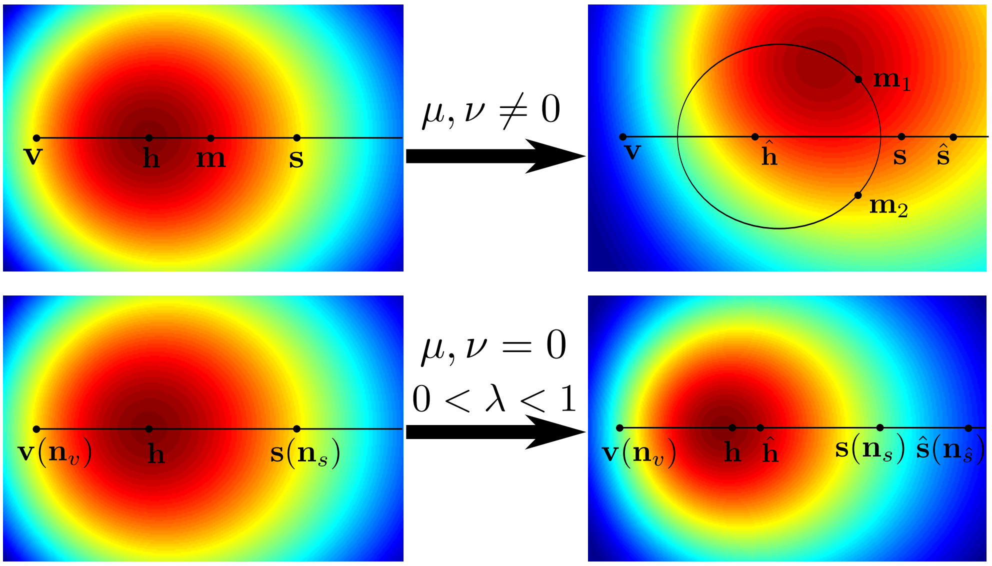

Figure 1: Projective plane with BRDF values. Top: BRDF transformed by a general GBR; bottom: BRDF transformed by GBR with $\mu, \nu = 0, 0 < \lambda < 1$.

In the following, we consider the case $\mu, \nu = 0$. We assume $\lambda > 0$ by ignoring the concave/convex ambiguity. We will prove by contradiction for the case $\lambda < 1$. The case of $\lambda > 1$ can be proved similarly. Assume the low-rank structure still hold in the GBR-transformed BRDF slice.

Since the normals $\mathbf{n}_v = \mathbf{v}$ and $\mathbf{n}_s = \mathbf{s}$ have the same half angle: $\theta_h = \theta_d$, they should have the same BRDF value

$$\frac{I(\mathbf{n}_v)}{\cos(2\theta_d)} = \frac{I(\mathbf{n}_s)}{\cos(0)}. \tag{.0.1}$$

A similar relation holds for the GBR transformed case

$$\frac{\hat{I}(\mathbf{n}_v)}{\cos(2\hat{\theta}_d)} = \frac{\hat{I}(\mathbf{n}_{\hat{s}})}{\cos(0)} \tag{.0.2}$$

where $\hat{I}$ is the transformed intensity map, $\mathbf{n}_{\hat{s}} = \hat{\mathbf{s}}$, and $\hat{\theta}_d$ is the transformed half angle for $\mathbf{n}_v$ and $\mathbf{n}_{\hat{s}}$. As shown in Section 3.3.3, $\hat{I}$ is the result of shrinking I toward $\mathbf{v}$ and $\hat{\mathrm{s}}$ is further from $\mathbf{v}$ than s, we have $I(\mathbf{n}_v) = \hat{I}(\mathbf{n}_v)$ and $\theta_d < \hat{\theta}_d$. Combining this with Equation (.0.1,.0.2), it can be derived that the relation $\hat{I}(\mathbf{n}_{\hat{s}}) > I(\mathbf{n}_s)$ should hold. On the other hand, $\mathbf{n}_{\hat{s}}$ is transformed from $\mathbf{n}_x$, which lies even further from $\mathbf{v}$: $\hat{I}(\mathbf{n}_{\hat{s}}) = I(\mathbf{n}_x)$. For most real-world BRDFs, Intensity decreases as the normal moves away from the fixed viewing direction $\mathbf{v}$ and lighting direction $\mathbf{s}$: $I(\mathbf{n}_x) < I(\mathbf{n}_s)$. thus, $\hat{I}(\mathbf{n}_{\hat{s}}) < I(\mathbf{n}_s)$. Contradiction.

$\square$

Bibliography

Miika Aittala, Tim Weyrich, and Jaakko Lehtinen. Practical svbrdf capture in the frequency domain. *ACM Trans. Graph.*, 32(4):110, 2013. 49

Daniel Aliaga and Yi Xu. Photogeometric structured light: A self-calibrating and multi-viewpoint framework for accurate 3d modeling. In *Proc. of CVPR*, 2008. 47

N. Alldrin, T. Zickler, and D. Kriegman. Photometric stereo with non-parametric and spatially-varying reflectance. In *Proc. CVPR*, pages 1–8. IEEE, 2008. xii, 29, 48, 49, 72, 73

Neil Alldrin and David Kriegman. Towards reconstructing surfaces with arbitrary isotropic reflectance : A stratified photometric stereo approach. In *Proc. of ICCV*, 2007. xii, 4, 46, 49, 51, 52, 55, 57

N.G. Alldrin, S.P. Mallick, and D.J. Kriegman. Resolving the generalized bas-relief ambiguity by entropy minimization. In *Proc. CVPR*, pages 1–7. IEEE, 2007. 27, 36, 41

P.N. Belhumeur, D.J. Kriegman, and A.L. Yuille. The bas-relief ambiguity. *International Journal of Computer Vision*, 35(1):33–44, 1999. 21, 26, 27

Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14:239–256, 1992. 65

Jean-Yves Bouguet. Camera calibration toolbox for matlab, 2013. 84

M. Brand. Incremental singular value decomposition of uncertain data with missing values. In *Proc. ECCV*, pages 707–720. Springer, 2002. 38

Manmohan Chandraker, Jiamin Bai, and Ravi Ramamoorthi. A theory of differential photometric stereo for unknown brdfs. In *Proc. of CVPR*, 2011. 48

M.K. Chandraker, F. Kahl, and D.J. Kriegman. Reflections on the generalized bas-relief ambiguity. In *Proc. CVPR*, volume 1, pages 788–795. IEEE, 2005. 27

Jiawen Chen, Dennis Bautembach, and Shahram Izadi. Scalable real-time volumetric surface reconstruction. *ACM Transactions on Graphics (TOG)*, 32(4):113, 2013. 10

Robert L Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics (TOG)*, 1(1):7–24, 1982. 2, 15, 16

Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312. ACM, 1996. 10

James Davis, Diego Nehab, Ravi Ramamoorthi, and Szymon Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(2), 2005. 47

Yue Dong, Jiaping Wang, Xin Tong, John Snyder, Yanxiang Lan, Moshe Ben-Ezra, and Baining Guo. Manifold bootstrapping for SVBRDF capture. *ACM Trans. Graph.*, 29(4), 2010. 49

O. Drbohlav and M. Chaniler. Can two specular pixels calibrate photometric stereo? In *Proc. ICCV*, volume 2, pages 1850–1857. IEEE, 2005. 22, 27, 41

O. Drbohlav and R. Šára. Specularities reduce ambiguity of uncalibrated photometric stereo. In *Proc. ECCV*, pages 644–645. Springer, 2002. 22, 27

Felix Endres, Jürgen Hess, Nikolas Engelhard, Jürgen Sturm, Daniel Cremers, and Wolfram Burgard. An evaluation of the rgb-d slam system. In *ICRA*, pages 1691–1696. IEEE, 2012. 10, 95

P. Favaro and T. Papadhimitri. A closed-form solution to uncalibrated photometric stereo via diffuse maxima. In *Proc. CVPR*, pages 821–828. IEEE, 2012. 22, 27, 41

Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007. 106

Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32:1362–1376, 2010. ISSN 0162-8828. 47

A.S. Georghiades. Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo. In *Proc. ICCV*, pages 816–823. IEEE, 2003. 27

Abhijeet Ghosh, Tongbo Chen, Pieter Peers, Cyrus A. Wilson, and Paul Debevec. Estimating specular roughness and anisotropy from second order spherical gradient illumination. *Computer Graphics Forum*, 28, 2009. 47, 48

Dan B Goldman and Jiun-Hung Chen. Vignette and exposure calibration and compensation. In *ICCV*, volume 1, pages 899–906. IEEE, 2005. 89

Dan B. Goldman, Brian Curless, Aaron Hertzmann, and Steven M. Seitz. Shape and spatially-varying brdfs from photometric stereo. In *Proc. of ICCV*, pages 341–348, 2005. 46, 48

Jinwei Gu and Chao Liu. Discriminative illumination: Per-pixel classification of raw materials based on optimal projections of spectral brdf. In *CVPR*, pages 797–804. IEEE, 2012. 4

Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003. ISBN 0521540518. 1, 46

H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *JOSA A*, 11(11):3079–3089, 1994. 21, 25, 26

Carlos Hernandez, George Vogiatzis, and Roberto Cipolla. Multiview photometric stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:548–554, 2008. xii, 46, 48, 72, 73, 74

Michael Holroyd, Jason Lawrence, Greg Humphreys, and Todd Zickler. A photometric approach for estimating normals and tangents. *ACM Trans. Graph.*, 27, 2008. 48

Michael Holroyd, Jason Lawrence, and Todd Zickler. A coaxial optical scanner for synchronous acquisition of 3d geometry and surface reflectance. *ACM Trans. Graph.*, 2010. 47, 48

Berthold KP Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. 1970. 8

Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011. 10, 78, 95

Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proc. of Eurographics Symposium on Geometry processing*, pages 61–70, 2006. 62, 96

Seon Joo Kim and Marc Pollefeys. Robust radiometric calibration and vignetting correction. *PAMI*, 30(4):562–576, 2008. 89, 90

Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *ISMAR*, pages 225–234. IEEE, 2007. 95

Jason Lawrence, Aner Ben-Artzi, Christopher DeCoro, Wojciech Matusik, Hanspeter Pfister, Ravi Ramamoorthi, and Szymon Rusinkiewicz. Inverse shade trees for non-parametric material representation and editing. *ACM Trans. Graph.*, 25:735–745, July 2006. 47, 53, 63, 97

Hendrik Lensch, Jan Kautz, Michael Goesele, Wolfgang Heidrich, and Hans-Peter Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Trans. Graph.*, 22:234–257, 2003. 47

Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, Jonathan Shade, and Duane Fulk. The digital michelangelo project: 3d scanning of large statues. In *Proc. of SIGGRAPH*, pages 131–144, 2000. 47

Maxime Lhuillier and Long Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27:418–433, 2005. 47, 58

William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *ACM Siggraph Computer Graphics*, volume 21, pages 163–169. ACM, 1987. 10

Wan-Chun Ma, Tim Hawkins, Pieter Peers, Charles-Felix Chabert, Malte Weiss, and Paul Debevec. Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Proc. of EGSR*, 2007. 48

Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Trans. Graph.*, 22:759–769, 2003. 29, 36, 55, 57

Shree K. Nayar, Gurunandan Krishnan, Michael D. Grossberg, and Ramesh Raskar. Fast separation of direct and global components of a scene using high frequency illumination. *ACM Trans. Graph.*, 25:935–944, 2006. 75

D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi. Efficiently combining positions and normals for precise 3d geometry. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 536–543. ACM, 2005. 21, 53, 62

Richard A Newcombe, Andrew J Davison, Shahram Izadi, Pushmeet Kohli, Otmar Hilliges, Jamie Shotton, David Molyneaux, Steve Hodges, David Kim, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, pages 127–136. IEEE, 2011a. 78, 95

Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *ICCV*, pages 2320–2327. IEEE, 2011b. 95

T. Okatani and K. Deguchi. Optimal integration of photometric and geometric surface measurements using inaccurate reflectance/illumination knowledge. In *Proc. CVPR*, pages 254–261. IEEE, 2012. 21

James M. Palmer. Radiometry and photometry faq, 1999. 12

Thoma Papadhimitri and Paolo Favaro. A new perspective on uncalibrated photometric stereo. In *Proc. CVPR*, pages 1474–1481. IEEE, 2013. 27

Bui Tuong Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975. 14

Ramesh Raskar and Jack Tumblin. *Computational Photography: Mastering New Techniques for Lenses, Lighting, and Sensors*. AK Peters, Ltd., 2009. 78

Ramesh Raskar, Kar-Han Tan, Rogerio Feris, Jingyi Yu, and Matthew Turk. Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. *ACM Trans. Graph.*, 23:679–688, August 2004. 58

Peiran Ren, Jiaping Wang, John Snyder, Xin Tong, and Baining Guo. Pocket reflectometry. *ACM Trans. Graph*, 30(4), 2011. 49

F. Romeiro, Y. Vasilyev, and T. Zickler. Passive reflectometry. In *Proc. ECCV*, pages 859–872. Springer, 2008. 29

Fabiano Romeiro and Todd Zickler. Inferring reflectance under real-world illumination. *Technical Report TR-10-10, Harvard School of Engineering and Applied Sciences*, 2010a. 63

Fabiano Romeiro and Todd Zickler. Blind reflectometry. In *Computer Vision–ECCV 2010*, pages 45–58. Springer, 2010b. 17

S. Rusinkiewicz. A new change of variables for efficient brdf representation. In *Eurographics Workshop on Rendering*, pages 11–22, 1998. xi, 23, 28, 29

Szymon Rusinkiewicz, Olaf Hall-Holt, and Marc Levoy. Real-time 3d model acquisition. *ACM Trans. Graph.*, 21, 2002. 47

Y. Sato and K. Ikeuchi. Temporal-color space analysis of reflection. *JOSA A*, 11(11): 2990–3002, 1994. 33, 38

Yoichi Sato, Mark D. Wheeler, and Katsushi Ikeuchi. Object shape and reflectance modeling from observation. In *Proc. of SIGGRAPH*, pages 379–387, 1997. 47

Christophe Schlick. An inexpensive BRDF model for physically-based rendering. *Computer Graphics Forum*, 13, 1994. 75

Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, volume 1, pages 519–528. IEEE, 2006. 1

B. Shi, Y. Matsushita, Y. Wei, C. Xu, and P. Tan. Self-calibrating photometric stereo. In *Proc. CVPR*, pages 1118–1125. IEEE, 2010. 27, 41

Boxin Shi, Ping Tan, Yasuyuki Matsushita, and Katsushi Ikeuchi. A biquadratic reflectance model for radiometric image analysis. In *Proc. CVPR*, pages 230–237. IEEE, 2012. 29

Rakesh Shiradkar, Li Shen, George Landon, Sim Heng Ong, and Ping Tan. A new perspective on material classification and ink identification. In *CVPR*. IEEE, 2014. 4

Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph.*, 25(3):835–846, 2006. 58

M.M. Stark, J. Arvo, and B. Smits. Barycentric parameterizations for isotropic brdfs. *IEEE Trans. on Visualization and Computer Graphics*, 11(2):126–138, 2005. 23, 29

P. Tan and T. Zickler. A projective framework for radiometric image analysis. In *Proc. CVPR*, pages 2977–2984. IEEE, 2009. 22, 27

P. Tan, S.P. Mallick, L. Quan, D.J. Kriegman, and T. Zickler. Isotropy, reciprocity and the generalized bas-relief ambiguity. In *Proc. CVPR*, pages 1–8. IEEE, 2007. 22, 27, 49

P. Tan, L. Quan, and T. Zickler. The geometry of reflectance symmetries. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(12):2506–2520, 2011. 4, 27, 41, 48

R.T. Tan and K. Ikeuchi. Separating reflection components of textured surfaces using a single image. *Digitally Archiving Cultural Objects*, pages 353–384, 2008. 33

K.E. Torrance and E.M. Sparrow. Theory for off-specular reflection from roughened surfaces. *JOSA*, 57(9):1105–1112, 1967. 27

Borom Tunwattanapong, Graham Fyffe, Paul Graham, Jay Busch, Xueming Yu, Abhijeet Ghosh, and Paul Debevec. Acquiring reflectance and shape from continuous spherical harmonic illumination. *ToG*, 32(4):109, 2013. 3, 47, 48, 77

Gregory J. Ward. Measuring and modeling anisotropic reflection. In *Proc. of SIGGRAPH*, pages 265–272, 1992. 2, 48

R.J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical engineering*, 19(1):191139–191139, 1980. 2, 21, 26

Di Wu, Gordon Wetzstein, Christopher Barsi, Thomas Willwacher, Matthew OToole, Nikhil Naik, Qionghai Dai, Kyros Kutulakos, and Ramesh Raskar. Frequency analysis of transient light transport with applications in bare sensor imaging. In *Computer Vision–ECCV 2012*, pages 542–555. Springer, 2012. 114

Tai-Pang Wu and Chi-Keung Tang. Visible surface reconstruction from normals with discontinuity consideration. In *Proc. of CVPR*, 2006. 73

A. Yuille and D. Snow. Shape and albedo from multiple images using integrability. In *Proc. CVPR*, pages 158–164. IEEE, 1997. 21, 26, 27, 34, 38

Li Zhang, Brian Curless, Aaron Hertzmann, and Steven M. Seitz. Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multi-view stereo. In *Proc. of ICCV*, 2003. 48

Li Zhang, Noah Snavely, Brian Curless, and Steven M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM Trans. Graph.*, 23:548–558, 2004. 47

Z. Zhang, X. Liang, A. Ganesh, and Y. Ma. Tilt: transform invariant low-rank textures. In *Proc. ACCV*, pages 314–328. Springer, 2011. 34

Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22:1330–1334, 2000. 18

Qian-Yi Zhou and Vladlen Koltun. Color map optimization for 3d reconstruction with consumer depth cameras. *ACM Transactions on Graphics (TOG)*, 2014. 78

Z. Zhou and P. Tan. Ring-light photometric stereo. In *Proc. ECCV*, pages 265–279. Springer, 2010. 27, 43

Todd E Zickler, Peter N Belhumeur, and David J Kriegman. Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. *International Journal of Computer Vision*, 49(2-3):215–227, 2002. 4

List of my Publications

Zhe Wu and Ping Tan. Calibrating photometric stereo by holistic reflectance symmetry analysis. In *CVPR*, pages 1498–1505. IEEE, 2013.

Zhenglong Zhou, Zhe Wu, and Ping Tan. Multi-view photometric stereo with spatially varying isotropic materials. In *CVPR*, pages 1482–1489. IEEE, 2013.