Multi-view Photometric Stereo with Spatially Varying Isotropic Materials

Zhenglong Zhou   Zhe Wu   Ping Tan
National University of Singapore

Abstract

We present a method to capture both 3D shape and spatially varying reflectance with a multi-view photometric stereo technique that works for general isotropic materials. Our data capture setup is simple, which consists of only a digital camera and a handheld light source. From a single viewpoint, we use a set of photometric stereo images to identify surface points with the same distance to the camera. We collect this information from multiple viewpoints and combine it with structure-from-motion to obtain a precise reconstruction of the complete 3D shape. The spatially varying isotropic bidirectional reflectance distribution function (BRDF) is captured by simultaneously inferring a set of basis BRDFs and their mixing weights at each surface point. According to our experiments, the captured shapes are accurate to 0.3 millimeters. The captured reflectance has relative root-mean-square error (RMSE) of 9%.

1. Introduction

Appearance capture methods recover both 3D shape and surface reflectance of objects, allowing photorealistic rendering of the captured objects from arbitrary viewpoints and lighting conditions. This capture is an important and challenging problem with many applications such as graphics and reverse engineering. Typically, appearance capture is performed with sophisticated hardware setups such as the light stage of Ghosh et al. [9] and the coaxial lights of Holroyd et al. [14]. Though these methods achieve highly accurate results, the data capture setup is expensive and complicated. We design a method with simple setup so that it can be used more widely. Our simplest setup only contains a digital camera and a handheld moving light source. Compared with [14], our method achieves lower but still useful accuracy (0.3 millimeters vs. 50 microns). This lightweight solution provides a practical step towards enabling casual users the ability to perform appearance capture.

The appearance of opaque objects is well represented by a bi-directional reflectance distribution function (BRDF). Most previous methods on simultaneous shape and BRDF capture, e.g. [10, 12], assumed specific parametric BRDF

models. Their performance degrades when the real objects have different reflectance from the assumed model.

We exploit reflectance symmetries to work on objects with general spatially varying isotropic BRDF. According to [2], isotropy allows us to identify 'iso-depth contours', i.e. pixels with the same distance to the image plane, from photometric stereo images. We collect iso-depth contours from multiple viewpoints to reconstruct the complete 3D shape. Specifically, we first apply structure-from-motion [11] to reconstruct a sparse set of 3D points. We then propagate the depths of these 3D points along iso-depth contours. Each propagation generates additional 3D points, whose depths can be further propagated. A surprisingly small number of 3D points (about two hundred) 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 3D discrete tables to handle general material. The basis BRDFs and mixing weights at each point are iteratively estimated by the ACLS method [16].

2. Related Work

Image-based modeling. These methods reconstruct a 3D shape and a 'texture map' to model objects from images. [19, 8] are two recent representative methods. Texture color at each surface point is decided according to its image projections. However, a texture map is often insufficient to represent general non-Lambertian materials.

Shape scanning and reflectance fitting. To obtain precise 3D shape, laser scanners and structured-light patterns were used in [18, 27, 36, 6]. Based on a precise 3D reconstruction, parametric reflectance functions can be fitted at each surface point according to the image observations, as in [28, 17].These methods require precise registration between images and 3D shapes. Since different sensors are used for shape and reflectance capture, this registration is difficult and often causes artifacts in misaligned regions. Some methods [23, 1] combine reflectance recovered from photometric stereo and shape recovered from structured-light, where registration is relatively simple. However, they

need to capture images under both structured-light and varying directional light at *each* viewpoint, which is tedious and requires a more complicated setup than ours.

Photometric appearance capture. Our method belongs to photometric approaches that capture both shape and reflectance from the same set of images. Most of previous methods, e.g. [35, 12, 10], assumed specific parametric BRDF models such as Lambert's or Ward's model [33]. The performance of these methods degrades when the real objects have different reflectance from the assumed model.

Some other methods employed a sophisticated hardware setup to achieve high quality results. Ma et al. [20] and Ghosh et al. [9] used a light stage where the intensity of each LED on the stage was precisely controlled. Holroyd et al. [14] required specialized coaxial lights. This requirement of expensive and complicated hardware limits their wide application. Recently, a few algorithms [3, 13] were proposed for appearance capture by exploiting various reflectance symmetries that are valid for a broader class of objects. However, [13] required up to a thousand input images at *each* viewpoint and [3] relied on fragile optimization. Tan et al. [32] and Chandraker et al. [5] both recovered iso-contours of depth and gradient magnitude for isotropic surfaces. Additional user interactions or boundary conditions are required to recover the 3D shape.

The work closest to our method is [3]. Both methods are built upon reflectance symmetry embedded in 'isotropic pairs' introduced in [31]. There are three key differences between our method and [3]. First, we reconstruct a complete 3D shape rather than a single-view normal map. Second, we combine multi-view geometry and photometric cues to avoid fragile iterative optimization of shape and reflectance. Third, our method works with general tri-variant isotropic BRDFs while [3] assumed bi-variant BRDFs to simplify the optimization.

BRDF acquisition. Our work is also related to BRDF acquisition methods such as [7, 25]. These methods are only applicable to near-flat surfaces where the surface normals are known beforehand. Our method can be considered as a generalization of these methods to non-planar surfaces.

3. Overview

We capture images from multiple viewpoints. At each viewpoint, we capture photometric stereo images with a moving light source. We design a robust algorithm to identify iso-depth contours from these images. Further, we apply structure-from-motion to images from different viewpoints to reconstruct a sparse set of 3D points. We then derive a complete 3D shape by propagating the depths of these points along the dense iso-depth contours. Once the shape is fixed, we estimate a set of basis isotropic BRDFs and their mixing weights at each surface point to model the surface reflectance.

4. Shape Reconstruction

4.1. Iso-depth contour estimation

Alldrin and Kriegman [2] observed that isotropy allows almost trivial estimation of iso-depth contours in the absence of global illumination effects such as shadows and inter-reflections. We propose an algorithm that is more robust in real data than the naïve approach described in [2]. Specifically, we relax the assumption about lighting and propose a method to enhance robustness to global illumination effects.

Under orthographic projection and directional lighting that moves on a view-centered circle, the plane spanned by the viewing direction and the surface normal direction of an isotropic [1] surface point can be recovered precisely according to the symmetry of the observed pixel intensity profile. In the camera local coordinate system, where the z-axis is aligned with the viewing direction, this plane gives the azimuth angle of the surface normal, which is the angle between the x-axis and the normal's projection in the xy-plane. We briefly explain this idea in Figure 1 (a). For more details, please refer to [2]. Figure 1 (a) shows the observed pixel intensities under 36 different lighting directions on a view-centered circle. The vertical axis of the chart indicates pixel intensities, while the horizontal axis is the range of azimuth angles. The red symmetry axis of these observations provides a good estimation of the azimuth angle. Once azimuth angles are computed, at each pixel, we can recover an iso-depth contour by tracing along the directions perpendicular to the xy-plane projection of the surface normal there. For easier reference, we refer this direction of a projected surface normal as the azimuth direction in this paper.

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-centered circle. So we compute spatially variant lighting directions at each pixel, and interpolate the desired observations from recorded pixel intensities.

We take the average depth of an object (computed from the reconstructed sparse 3D points in Section 4.2) 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-centered 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 rep-

---

[1]Note that the original algorithm is based on the bilateral symmetry. Here, we follow [2] to refer it as isotropy because bilateral symmetry is often observed for isotropic surfaces.
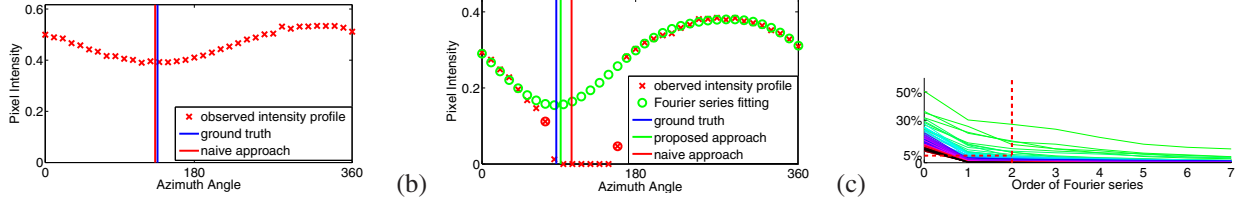
Figure 1. (a) The symmetry axis of intensity profiles tells the azimuth angle of a pixel's normal direction; (b) cast shadows can break this symmetry; (c) the intensity profile of most of isotropic BRDFs in [21] can be well represented by a 2-order Fourier series.
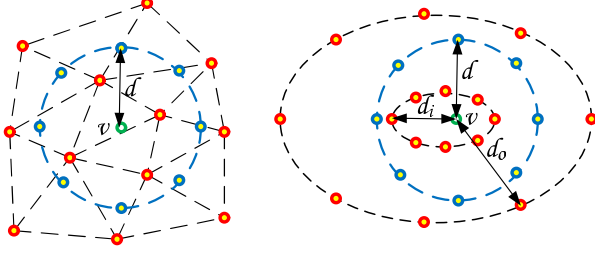

Figure 2. We compute a Delaunay triangulation of the original lighting directions (red dots) in the projective plane. The desired observations (blue dots) on a view-centered circle are generated by linear interpolation within these triangles. Left: the circle radius d is the mean distance between the 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.

resented by a 2D point $(x/z, y/z)$. As shown in the left of Figure 2, 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-centered 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 $\mathbf{v}$.

Global Illumination Effects To improve accuracy, we need to identify cast shadows, which break the symmetry of pixel intensities. Figure 1 (b) shows an example pixel with cast shadow. (This pixel is marked in red in the input image of the 'Buddha' example in Figure 7.) The original intensity profile marked by red '×' 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 '⊗' in Figure 1 (b). As shown in Figure 1 (b), the azimuth angle estimated by the naïve method in [2] is far from the ground truth at this point. Points in the penumbra also cause problems in the reflectance estimation in Section 5. 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 [21]. 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 1 (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 1 (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' [24] 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.

4.2. Multi-view depth propagation

A standard structure-from-motion algorithm such as [19, 30] 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
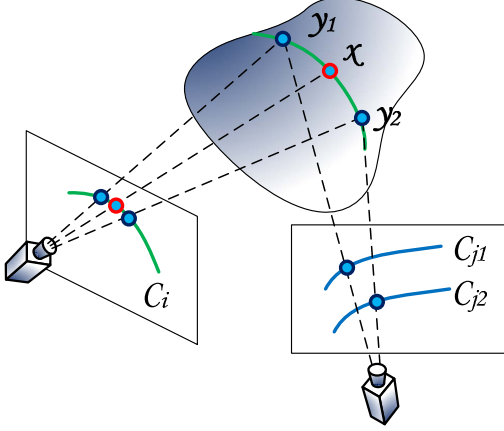
Figure 3. We propagate the depth of $\mathbf{x}$ to the iso-depth contour segment C_i that passes through its projection in the i-th view. This propagation generates new 3D points, e.g. $\mathbf{y}_1, \mathbf{y}_2$, whose depths in other images can also be propagated along their corresponding iso-depth contours C_{j1}, C_{j2}.

could be 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 3, 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 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
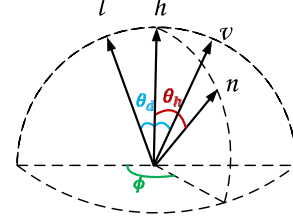

Figure 4. Definition of θ_h, θ_d and ϕ.

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 [2]. If different combinations of these L views all lead to consistent 3D 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 [15] to these points to obtain a triangulated surface. This surface is further optimized according to [23] by fusing the 3D point positions and their normal directions.

5. 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 [16] to estimate the basis BRDFs and their mixing weights at each point on the surface. We consider the general tri-

---

[2] 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.

variant isotropic BRDF, which is a function of θ_h, θ_d, ϕ as shown in Figure 4. 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 [26] 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 [16] 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 $\mathbf{W}$ at all surface points.

6. 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 algorithm 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 [4].

6.1. A Handheld 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 lighting directions. The light source positions and intensities were recorded with calibration spheres. An example is provided in Figure 5 (see the supplementary files for more results). Figure 5 (a) shows a sample input image. This example was captured from 10 viewpoints, which allow us to reconstruct part of its surface. To better visualize the recovered shape, we render it with uniform diffuse shading in (b). Most of the geometry details are successfully captured. (c) is a rendering according to the captured reflectance from the same viewpoint and lighting condition as the input image in (a). To provide a quantitative evaluation on shape capture, we visualize the shape reconstruction error (measured in millimeters) in (d). 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. Here, the object diameter is 250 millimeters.

6.2. A Ring-Light System

Setup To facilitate data capture, we built a simple device shown in Figure 6. 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 [3]. 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 LED light as shown in Figure 6. 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 2. When computing azimuth angles, we performed a Delaunay triangulation based interpolation as introduced in Section 4.1. 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

---

[3]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.
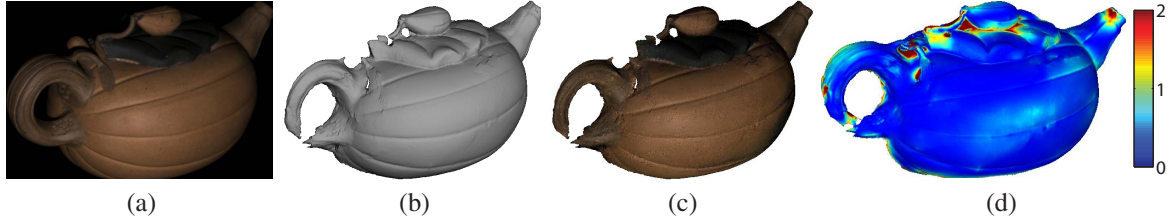
Figure 5. Results from the handheld system. (a) one of the input images, (b) the recovered shape rendered with uniform diffuse shading. (c) a rendering with the recovered reflectance model from the same viewpoint and lighting condition as the image in (a). (d) the color-coded shape error (in millimeters) compared to laser-scanned 'ground truth'.
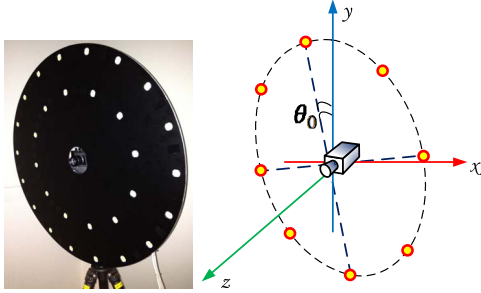

Figure 6. The hardware setup. Left: our device consists of a video camera and two circles of LED lights. Right: we need to calibrate one parameter θ_0 to determine lighting positions.
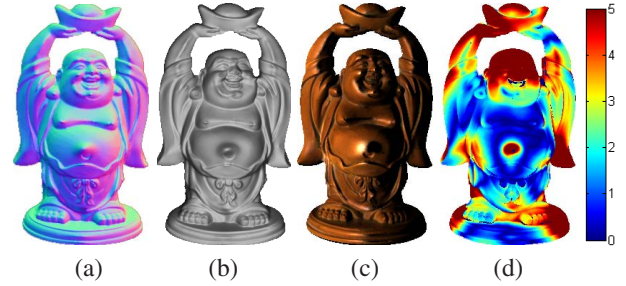

Figure 8. Results according to [3]. (a) the color coded normal map estimated. (b) the shape computed from the estimated normal according to [34]. (c) a rendering under novel illumination. (d) the color coded shape error (in millimeters).
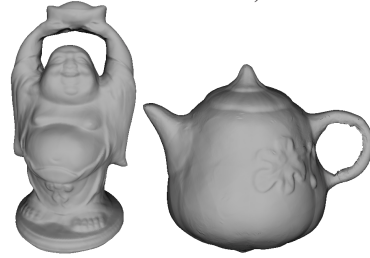

Figure 9. Results according to [12].

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 7. This example has focused and strong highlight. The object diameter is 120 millimeters. 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 (b) and (c). The median (or mean) shape error was 0.36 (or 0.57) millimeters in this example. Most of the large shape errors appeared at concave carvings with strong inter-reflection.

Another two examples, 'Cup' and 'Frog', are included in Figure 7 (more can be found in the supplementary files). Their diameters are 120 and 90 millimeters respectively. Our examples cover a wide range of different material. The rusted metal 'Cup' has quickly change reflectance over its surface. The painted 'Frog' also has significant spatial BRDF changes. We captured 30 and 34 viewpoints for the 'Cup' and 'Frog' examples respectively. Our method consistently performed well on all of them. Their median (or mean) shape reconstruction error was 0.29 and 0.25 (or 0.5 and 0.47) millimeters respectively. The 'Teapot2' example had relatively larger error at one side, mainly due to the imprecise structure-from-motion reconstruction caused by erroneous feature matching.

6.3. Comparison with Existing Methods

We compared our results with those obtained from [3] and [12]. We used the same code as the authors. The iterative shape and reflectance optimization in [3] is compli-

cated and slow. It took over 40 hours to compute the results of *one* viewpoint with 72 input images at resolution of 200×350. Figure 8 shows the results from [3]. (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 [34]. (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 9 shows the results from [12] which is designed for Lambertian surfaces, where most of the shape details are smoothed out.

6.4. 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

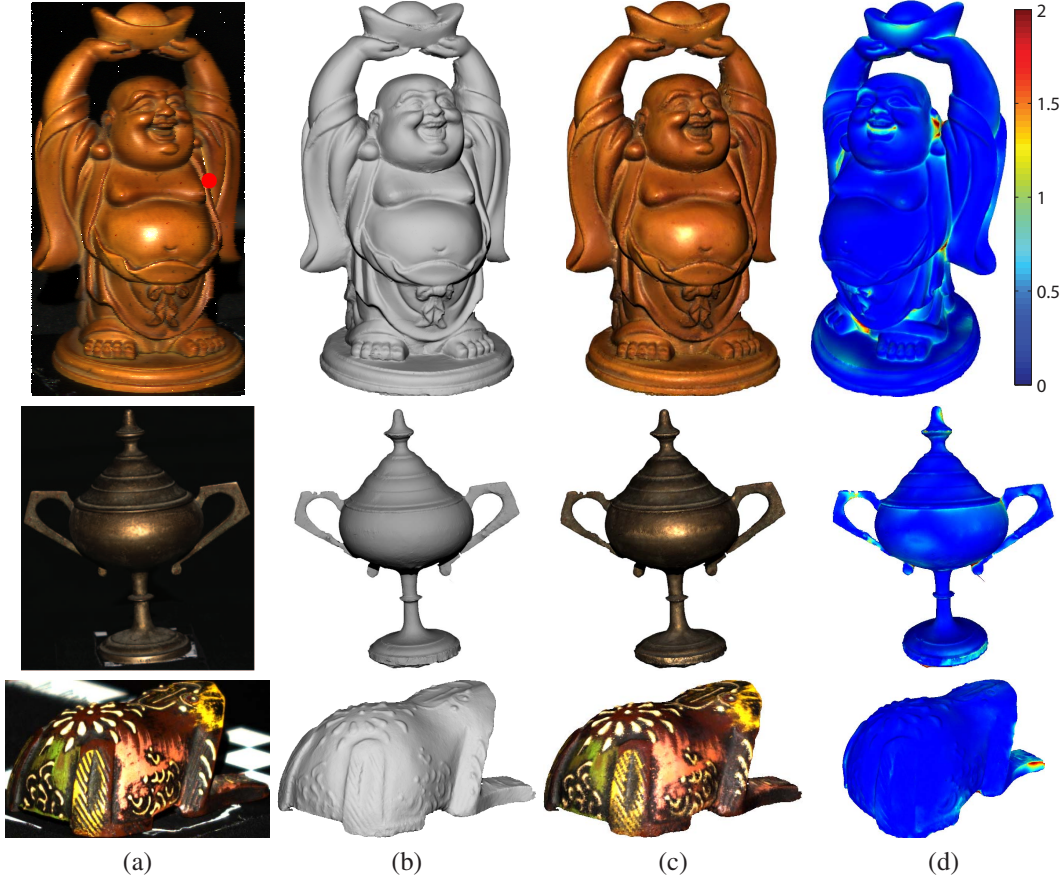(a)   (b)   (c)   (d)

Figure 7. Results from the ring-light system. From left to right, these figures are arranged in the same way as Figure 5.

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 optimization 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.

7. Discussion

We propose a method to capture both shape and reflectance of real objects with spatially variant isotropic reflectance. Our method requires a simple hardware setup and is able to capture 3D shapes accurate to 0.3 millimeters and reflectance with 9% relative RMSE error.

Our method 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 [22] to separate inter-reflection. However, it would significantly complicate the hardware setup. Alternatively, we might iteratively estimate the shape and inter-reflection. 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.) 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 [29] to solve this problem.

8. Acknowledgement

References

[1] D. Aliaga and Y. Xu. Photogeometric structured light: A self-calibrating and multi-viewpoint framework for accurate 3d modeling. In *Proc. of CVPR*, 2008. 1

[2] N. Alldrin and D. Kriegman. Toward reconstructing surfaceswith arbitrary isotropic reflectance : A stratified photometric stereo approach. In *Proc. of ICCV*, 2007. 1, 2, 3

[3] N. Alldrin, T. Zickler, and D. Kriegman. Photometric stereowith non-parametric and spatially-varying reflectance. In *Proc. of CVPR*, 2008. 2, 6

[4] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14:239–256, 1992. 5

[5] M. Chandraker, J. Bai, and R. Ramamoorthi. A theory of differential photometric stereo for unknown brdfs. In *Proc. of CVPR*, 2011. 2

[6] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(2), 2005. 1

[7] Y. Dong, J. Wang, X. Tong, J. Snyder, Y. Lan, M. Ben-Ezra, and B. Guo. Manifold bootstrapping for SVBRDF capture. *ACM Trans. Graph.*, 29(4), 2010. 2

[8] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32:1362–1376, 2010. 1

[9] A. Ghosh, T. Chen, P. Peers, C. A. Wilson, and P. Debevec. Estimating specular roughness and anisotropy from second order spherical gradient illumination. *Computer Graphics Forum*, 28, 2009. 1, 2

[10] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz. Shape and spatially-varying brdfs from photometric stereo. In *Proc. of ICCV*, pages 341–348, 2005. 1, 2

[11] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003. 1

[12] C. Hernandez, G. Vogiatzis, and R. Cipolla. Multiview photometric stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:548–554, 2008. 1, 2, 6

[13] M. Holroyd, J. Lawrence, G. Humphreys, and T. Zickler. A photometric approach for estimating normals and tangents. *ACM Trans. Graph.*, 27, 2008. 2

[14] M. Holroyd, J. Lawrence, and T. Zickler. A coaxial optical scanner for synchronous acquisition of 3d geometry and surface reflectance. *ACM Trans. Graph.*, 2010. 1, 2

[15] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proc. of Eurographics Symposium on Geometry processing*, pages 61–70, 2006. 4

[16] J. Lawrence, A. Ben-Artzi, C. DeCoro, W. Matusik, H. Pfister, R. Ramamoorthi, and S. Rusinkiewicz. Inverse shade trees for non-parametric material representation and editing. *ACM Trans. Graph.*, 25:735–745, July 2006. 1, 4, 5

[17] H. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H.-P. Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Trans. Graph.*, 22:234–257, 2003. 1

[18] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The digital michelangelo project: 3d scanning of large statues. In *Proc. of SIGGRAPH*, pages 131–144, 2000. 1

[19] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27:418–433, 2005. 1, 3

[20] W.-C. Ma, T. Hawkins, P. Peers, C.-F. Chabert, M. Weiss, and P. Debevec. Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Proc. of EGSR*, 2007. 2

[21] W. Matusik, H. Pfister, M. Brand, and L. McMillan. A data-driven reflectance model. *ACM Trans. Graph.*, 22:759–769, 2003. 3

[22] S. K. Nayar, G. Krishnan, M. D. Grossberg, and R. Raskar. Fast separation of direct and global components of a scene using high frequency illumination. *ACM Trans. Graph.*, 25:935–944, 2006. 7

[23] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi. Efficiently combining positions and normals for precise 3d geometry. *ACM Trans. Graph.*, 24:536–543, 2005. 1, 4

[24] R. Raskar, K.-H. Tan, R. Feris, J. Yu, and M. Turk. Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. *ACM Trans. Graph.*, 23:679–688, August 2004. 3

[25] P. Ren, J. Wang, J. Snyder, X. Tong, and B. Guo. Pocket reflectometry. *ACM Trans. Graph*, 30(4), 2011. 2

[26] F. Romeiro and T. Zickler. Inferring reflectance under real-world illumination. *Technical Report TR-10-10, Harvard School of Engineering and Applied Sciences*, 2010. 5

[27] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy. Real-time 3d model acquisition. *ACM Trans. Graph.*, 21, 2002. 1

[28] Y. Sato, M. D. Wheeler, and K. Ikeuchi. Object shape and reflectance modeling from observation. In *Proc. of SIGGRAPH*, pages 379–387, 1997. 1

[29] C. Schlick. An inexpensive BRDF model for physically-based rendering. *Computer Graphics Forum*, 13(3):233–246, 1994. 7

[30] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph.*, 25(3):835–846, 2006. 3

[31] P. Tan, S. P. Mallick, L. Quan, D. Kriegman, and T. Zickler. Isotropy, reciprocity and the generalized bas-relief ambiguity. In *Proc. of CVPR*, 2007. 2

[32] P. Tan, L. Quan, and T. Zickler. The geometry of reflectance symmetries. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33:2506–2520, 2011. 2

[33] G. J. Ward. Measuring and modeling anisotropic reflection. In *Proc. of SIGGRAPH*, pages 265–272, 1992. 2

[34] T.-P. Wu and C.-K. Tang. Visible surface reconstruction from normals with discontinuity consideration. In *Proc. of CVPR*, 2006. 6

[35] L. Zhang, B. Curless, A. Hertzmann, and S. M. Seitz. Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multi-view stereo. In *Proc. of ICCV*, 2003. 2

[36] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM Trans. Graph.*, 23:548–558, 2004. 1