Testing Spatial Patterns for Acquiring Shape and Subsurface Scattering Properties

Yitzchak David Lockerman, Samuel Brenner, Joseph Lanzone, Alexander Doronin, Holly Rushmeier; Yale University; New Haven, CT, USA

Abstract

Models of both the shape and material properties of physical objects are needed in many computer graphics applications. In many design applications, even if shape is not needed, it is desirable to start with the material properties of existing non-planar objects. We consider the design of a system to capture both shape and appearance of objects. We focus particularly on objects that exhibit significant subsurface scattering and inter-reflection effects. We present preliminary results from a system that uses coded light from a set of small, inexpensive projectors coupled with commodity digital cameras.

Introduction

We consider the problem of capturing digital models of the shape and material properties of physical objects. These models are needed in a variety of applications. A digital model with both shape and material is needed to visualize an object in a virtual scene in computer graphics. Material properties are needed in design tasks in which a new object is to be created with a material "look" similar to existing objects. Even if the shape is not of interest, if the object is not flat the shape of the object must be estimated to extract the material properties. Many methods have been developed to scan both shape and appearance. However these methods produce errors in both the shape and material properties when there is subsurface scattering in the material. In this project we build on recent work in separating light paths using spatially coded patterns of light to estimate both shape and material. Our goal is to have an inexpensive capture system that produces models that can be rendered in a computer graphics system with visually acceptable results.

Background and Previous Work

An image of an object is a function of the object itself, the incident lighting and the properties of the imaging system, as shown in Figure 1. Systems for capturing shape and material properties fundamentally depend on following light paths from a light source, to the object, and then to an camera sensor. The paths depend on the object shape and material scattering properties. Most systems assume simple paths, such as the one shown in Figure 1. Knowing the light source and camera positions, and the direction from each to a point on the object allows the calculation of the point position using triangulation. Knowing the magnitude of the reflected light can be used to estimate the surface reflectance property. The accuracy of the 3D point position depends on accurately locating the 2D point, shown in bright red in Figure 1, in the camera image. Locating the point in the image however can be difficult when the object diffuses the light into a region around the point of incidence as a result of subsurface scattering. Rather than a precise point on the image there is a blurred spot, as represented in pink in the figure. The appearance of the blurred spot also indicates that the material property can not be represented by a simple surface reflectance.

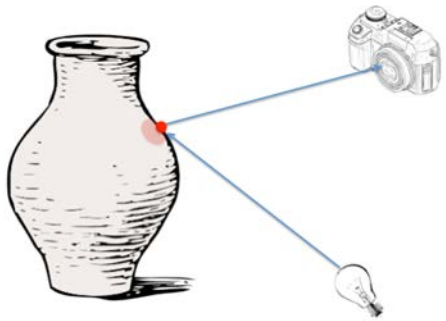

Figure 1. *Shape and material capture systems depend on estimating/controlling the parameters of object illumination and the imaging device.*

With the availability of inexpensive digital cameras (in particular the high resolution cameras on smart phones) and scanners (such as the NextEngine^{TM} and Kinect^{TM} there has been a great deal of interest in assembling systems for capturing digital content for 3D graphics applications, rather than modeling content from scratch. Early work in shape and material properties used laser scanners, cameras and lights [1]. Recent work includes limited acquisition (large, flat samples with statistically stationary textures) using just a smart phone camera [2] , simultaneous shape and surface reflectance capture with an RGB-D sensor [3], and sophisticated separation of light paths using an interferometer [4]. With the exception of [4] these systems do not account for the errors in shape introduced by subsurface scattering, and do not attempt to estimate subsurface scattering properties. In our project we seek a simple, inexpensive system that accounts for subsurface scattering.

In previous work [5] we considered the use of spatial patterns for estimating material properties for near-flat surfaces. We made use of the seminal work presented by Nayar et al. [6] for separating direct and indirection illumination effects in images by using projected spatial patterns. In addition to verifying the effects originally demonstrated in [6] we showed that indirect effects can be further separated by using direct/indirect separations from multiple illumination directions. Further we showed that while subsurface scattering effects are never spatially uniform, direct/indirect separations can be used to produce spatially maps on the material surface for where subsurface scattering is significant. Finally we proposed that subsurface scattering parameters could be estimated

by iteratively comparing images formed with light patterns and synthetic images of the material generated with varying scattering parameters. We seek to extend this work to arbitrarily shaped objects. Since we use projected patterns for examining the material properties, we wish to use projected patterns to estimate the shape as well.

The use of light patterns for shape acquisition has been studied extensively [7], and techniques such as temporal sequences of binary patterns have been used for decades [8]. In work following [6], Gu et al. [9] demonstrated that direct/indirect separation using spatial patterns could improve a wide range of shape measurement techniques developed in the field of computer vision. Arguing that separation techniques require large numbers of images and are vulnerable to noise, some subsequent research on shape acquisition has focused on creating robust patterns for shape that do not depend on explicit path separation [10, 11]. These projects however do not consider the simultaneous recovery of material properties, in particular subsurface scatting properties. We return to the idea of direct/indirect separation for sequences of binary patterns, with the goal of acquire both shape and subsurface scattering properties.

System Overview

The system we use for our experiments is an upgraded version of the setup we described in [5]. We have upgraded components of the system used in that system based on our experience with the captured results, and extend our processing pipeline to the estimation of arbitrary shapes.

Hardware

Figure 2. *Our hardware capture setup consists of three projector/camera pairs.*

Our hardware setup is shown in Figure 2. We built our system out of a number of consumer components. The heart of the system is three nodes each consisting of a projector/camera pair controlled by a Raspberry Pi single board computer. As in our previous system, we use AAXA KP-100-02 P2 Jr Pico projectors with 1920x1080 pixels at a price of about $200 each. We use Canon EOS Rebel T5 cameras, with 5184x3456 at a price of $300 each. While these are substantially more expensive than the Raspberry Pi cameras used in [5], the increased cost is justified by the increased resolution, ability to adjust field of view and reduced noise level. The camera model was selected from among other camera models based on its price and resolution, and the ability to operate the cameras using the gPhoto library framework [12]. The cameras still have the disadvantage that they cannot take high dynamic range (HDR) image directly, so for each image we must take multiple exposures to form an HDR image in postprocess. The Raspberry Pi's used were upgraded to the current Raspberry Pi 2-Model B-ARMv7 with 1Gb RAM, at $35 each.

The projector/camera nodes are connected to a control computer (a Dell Vostro 420 series desktop computer running Windows 7) using a gigabit network switch. We use a custom software stack to project the desired patterns and take photos of them. The system is designed with a fallback mode for when a node fails to take a picture, allowing an operator can recover the node and continue the scan.

Binary Patterns

The basic principle for separating luminance into direct light L_d and indirect light L_g is to use illumination patterns of high spatial frequency. When the fraction of a surface illuminated with the "on" portion of the pattern is c, the radiance L received by the camera from a directly lit spot is $L_d + cL_g$, and from other spots just cL_g. Reversing the "on" patterns of illumination to get two images, each location has a maximum value $L_d + cL_g$ and a minimum value cL_g. The values of L_d and L_g can then be calculated on a per pixel basis. Due to practical issues such as projectors not emitting perfectly sharp patterns, and the lack of one-to-one correspondence between projector and camera pixels, several shifted pairs of inverted patterns are needed to get an estimate of direct and indirect illumination with minimal visual artifacts.

Binary off/on patterns are also used in shape acquisition. A classic set of binary patterns is shown in Figure 3. In this case, three images would be captured, one for each of the patterns. A camera observing an object lit by these codes could identify which of the eight vertical strips in the projector image a point lies in. For example, a pattern of unlit-lit-unlit indicates that a point was in the third vertical strip from the right in the projector image. A companion set of projected horizontal patterns allows the localization of a point to the extent that projected features can be distinguished.

Normal binary codes such as those shown in Figure 3 are ill-suited to handle objects with strong subsurface scattering. In particular, binary codes with details smaller than the scattering distance will be blurred by the scattering causing a loss of information. In order to get a suitable three dimensional model we need to isolate the light that is *directly* reflected from the light source to the viewer. Our insight for designing a pattern to overcome this problem is that we can combine any pattern with a pattern used for direct/indirect separation to obtain the direct scattering of that first pattern. For clarity, we will call the first pattern, the one for which we seek the direct scattering, the primary pattern set, and the combined patterns the secondary pattern set.

Our combination works as follows. We view each pattern in the primary pattern set as a fixed illumination of the sample. We then subdivide the "on" areas of that pattern into a number of different regions. Two patterns are then projected. In the first, alternate regions are illuminated. In the second, the inverse set of regions are illuminated. Ideally, the sum of these two second order patterns should be the original primary pattern. However, due to the non-ideal nature of projectors, there will be artifacts in the

border between different regions. As such, we repeat the process above, with new divisions of regions until the entire original pattern is illuminated by at least one area that does not contain a border. For our use case, where the primary pattern is binary codes, we use the fact that rows and columns are processed separately to simplify our creation of second order patterns. In particular we, create regions by evenly spacing them along the direction orthogonal to the binary pattern. To create overlap, we move those regions along the direction of the binary pattern as if the stripes of the bindery pattern were a conveyor belt carrying the regions. An example of this is shown in Figure 4.

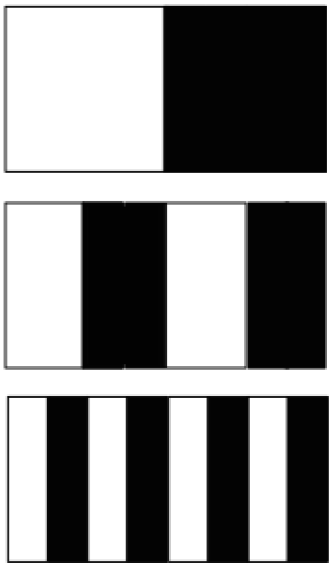

Figure 3. *A set of binary patterns classically used in shape acquisition. The vertical strip that a pixel is in can be identified by the pattern of lit and unlit for that pixel in a series of 3 images.*

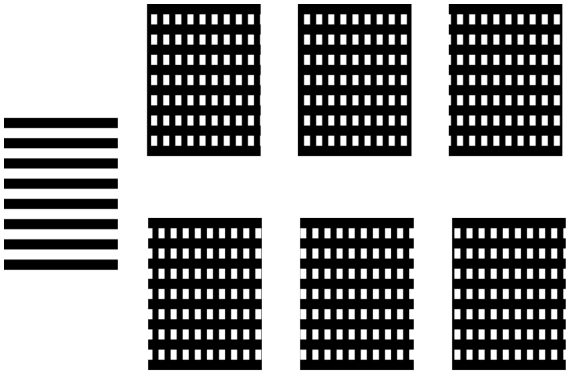

Figure 4. *A set of patterns is projected onto an object to identify unique points for matching to images in the camera views that is suited for cases with subsurface scattering. The primary pattern is shown at the left. The top row of small images show the secondary patterns dividing up the lit regions in the primary pattern. In each of the images in the top row the pattern is shifted slightly horizontally. The lower row of images show the patterns with the secondary patterns inverted. Note that the primary pattern is the union of the secondary patterns and their inverses.*

Processing

The images we acquire in the system must be processed to separate direct and indirect illumination effects and to compute the 3D locations of points on the object surface.

Separation of Direct and Indirect

The separation of direct and indirect components of primary patterns is performed, as discussed above, by taking the minimum and maximum of images lit with secondary patterns. For materials with high levels of subsurface scattering, the indirectly illuminated regions are never uniformly illuminated. As a result in the separated results there are always small but visible artifacts of slightly higher luminance on the boundaries of the patterns used. For creating the map of regions of subsurface scattering, we suggest that these artifacts might be mitigated by applying a blurring function that uses the ratio of the indirect to fully illuminated value.

The separation does not split apart direct effects (i.e. it doesn't separate specular and diffuse reflection) or indirect effects (it doesn't separate subsurface scattering and surface inter-reflections).To separate these effects, as shown in our previous work we can use three projector directions for each camera view, and use the results from the three camera views for the full object. For a single camera view, the use of different projector directions allows the identification of specular highlights in the direct illumination, which move on the surface. It also allows for the identification of surface inter-reflections in the indirect image, which depend on the orientation of the surface relative to the light. Specular reflections of the object onto itself that appear in the images can be eliminated using different camera views. While believe that a more advanced analysis is possible, our current approach simply averages colors from the different camera and projector directions to estimate a diffuse color.

Computing 3D Locations

We compute 3D points on the object surface using the primary/secondary codes just described. Each pixel in a projector image, which corresponds to a direction from the projector center to the scene, is characterized by a feature vector of on/off values that we have specified in our designed patterns. The camera pixels that correspond to projector pixels are found by examining their on/off values for direct illumination estimated from the sets of secondary patterns. This identification process proceeds by looping through all of the images of the row and column binary codes. After the identification process, the correspondences are reduced so that there is no more than one projector pixel matched with a camera pixel, and vice versa.

Given pixel matches between the projector and camera pixels, matches between different cameras and different projectors can be found by using the transitive property. This, in turn, allows for points to be uniquely identified across all cameras and projectors.

Using this full transitive matching, the relative camera and projector parameters and 3D point and camera locations can be computed using bundle adjustment in Bundler SFM [14] by passing the projector information from those software packages as "cameras". Alternatively we could use an SDK such as OpenCV [15] to perform this calculation. So long as the graph of cameras and projector overlap form a single connected component, all of

the camera and projector parameters can be found in the same coordinate system after matching each camera to each projector pairwise.

Results and Discussion

In this section we present results for fitting scattering parameters to data for a flat shape, and then results for 3D shapes.

Estimating Scattering Parameters

For our experiment fitting parameters, we use a piece of faux marble ($\approx 30 \times 30 \times 10$ *cm*) onto which a set of patterns were projected and recorded with our original system (i.e. using the Raspberry Pi cameras). Figure 5 shows gray scale images of the fully illuminated sample (left), and the separations into indirect (center) and direct diffusely reflected illumination (right).

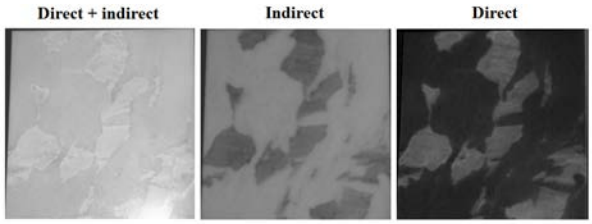

Figure 5. *A test sample of faux marble fully lit (left), and separated into indirect and direct components*

A complete scene description with the actual geometry, CCD camera settings, etc. has been set up and rendered with Mitsuba [16] utilizing optical properties outlined in [17]. Figure 6 shows the experimentally captured patterns compared with the results of the rendering.

In Figure 6, the top row left is the image of a pattern projected onto the sample, with the directly reflected light removed. The center and right images in the top row show Mitsuba renderings of the indirect illumination with the same camera and lighting parameters resulting when scattering parameters for skin and skim milk are used. In the second row, Mitsuba renderings are shown for the parameters for real marble on the left, and then with adjusted parameters to fit our captured results for the areas labeled "1" and "2". A comparison of the captured data and results for data for skin, skim milk and real marble show that real marble is by far the most similar. Starting with the data for real marble then, the value of the absorption coefficient is held constant, and the value of the scattering coefficient is adjusted to fit the two different regions on the sample. Plots of the results for a single row of pixels in each region are show in the bottom row of the figure. The final parameters estimated for the RGB image are show in Table 1.

Table 1: Scattering parameter μ_s in mm^{-1}

Material	R	G	B
Marble	2.19	2.62	3.0
Rendering 1	2.01	2.2	2.8
Rendering 2	3.29	4.82	5.3

These results show that scattering parameters can be estimated and refined using the images separated into direct and indirect illumination. However, while the approach takes into account spatial variations along the surface it assumes that the material is homogeneous in the third dimension. We can only estimate the scattering parameters that produce the same surface effects, not the actual scattering coefficients of layered materials.

Computing 3D Shape

Figure 7 shows a number of examples of the various stages of our 3D geometry pipeline to extract both properties and shape. We take 3 exposures of each pattern, which are used to create HDR images. Each LDR pattern takes between 20 and 30 MB of data. The reconstructed HDR image takes between 180 and 200 MB of data. In total we used 3090 patterns for a total of 9270 captures. A full capture (accounting for failures) can take 1 to 3 days. We perform our analysis on a server with two Intel Xeon E5-2680v3. It has a total of 24 cores and 128GB of RAM. The calculation from raw images to points takes several hours, with a major portion of the time taken by forming the HDR images.

In Figure 7 we show the objects under full illumination, a example pattern along with its direct/indirect separation and finally the resulting point-clouds. In all cases the points found still evidence a good deal of noise – more than would be expected from a laser scans. We show raw results only – standard techniques

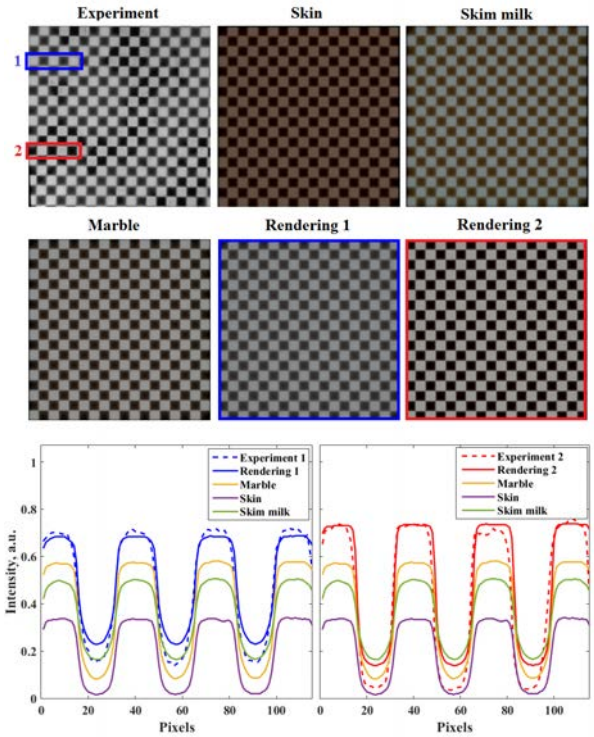

Figure 6. *A set of patterns projected onto a piece of faux marble. The top row left shows a captured image, with two regions highlighted with different scattering parameters. The other images are Mitsuba simulations using the same camera and lighting setup, using published parameters for skin, skim milk, real marble, and parameters tuned to match the highlighted regions. The plots at the bottom show the variation of intensity in the images for a single row of pixels.*

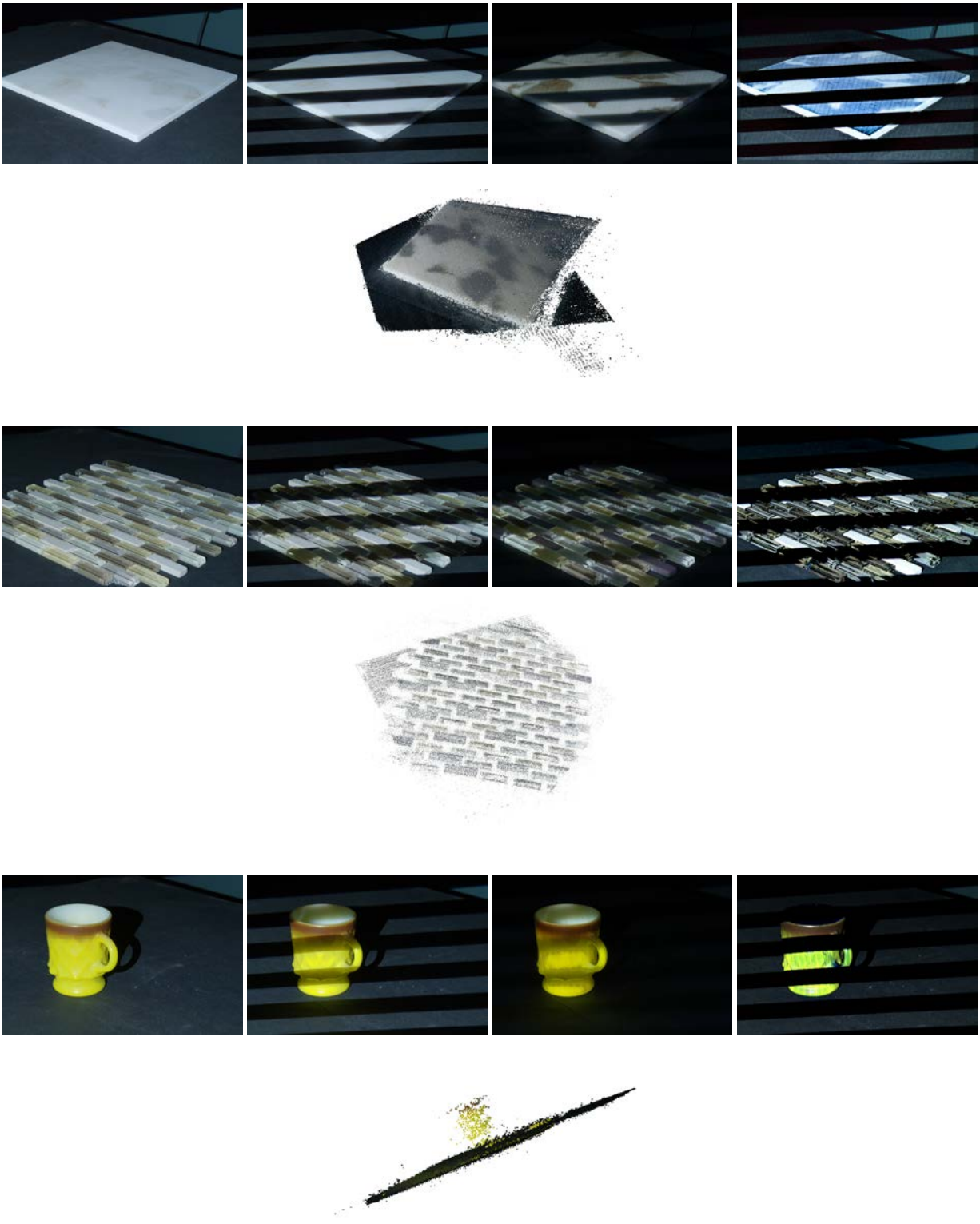

Figure 7. *A few examples of 3D geometry obtained using our system. In each example we show (from left to right) A single view of the object fully illuminated by a single projector, the object illuminated by a single pattern, the indirect illumination of that pattern, the direct illumination of that pattern, and (below the images) the point cloud we generate. The examples we show are (from top to bottom): A faux marble slab, an assortment of wall tiles, and a mug. Note that all the images in this example have been tone mapped from the original HDR version. Colors for the point clouds have been taken to match the image shown. Only points visible in that image are shown.*

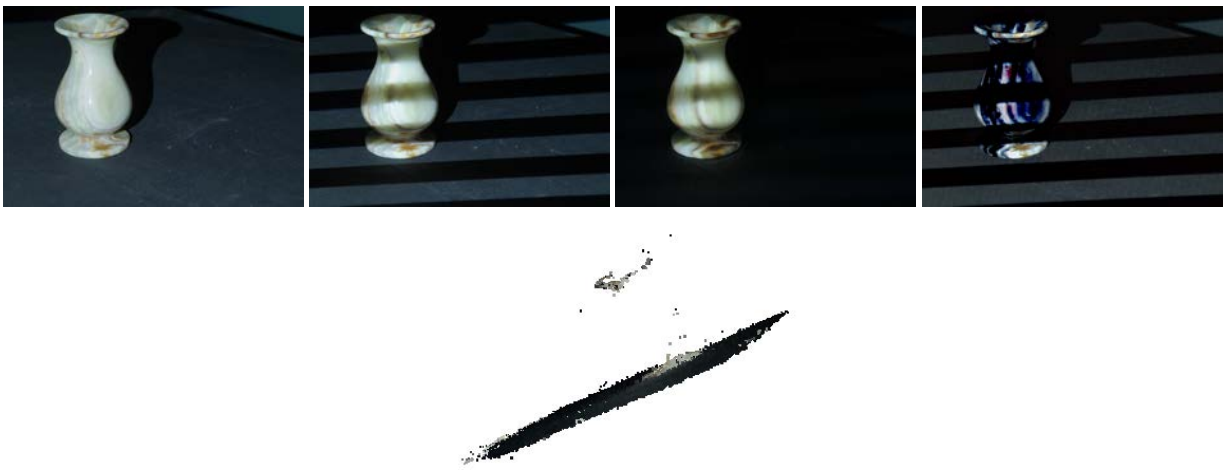

Figure 8. *Images from a vase which demonstrates a limitations of our method. In this case our method failed to capture points on the surface where there was a very small directly illuminated diffuse component. We are unable to locate patterns in those areas. This leads to missing points in the point cloud.*

for removing outliers and denoising the point cloud data would improve the results. The points could be used as a starting point for a shape from photometric stereo process that uses the directly illuminated images. They might also be useful to align the camera positions and separated direct/indirect images with a high resolution laser scan of the object to map material properties onto the laser scan.

We show a failure case, an onyx vase, in Figure 8. While our approach is robust with respect to the presences of subsurface scattering, it depends on having a detectable directly reflected diffuse component. On the vase, this assumption fails in several regions, which appear dark blue in the image of directly reflected light. The small regions that are captured however would still be useful for aligning the images used for scattering parameter estimation to a model of the vase obtained by other means – such as shape from silhouette.

Limitations

The use of consumer grade hardware does lead to some practical limitations. These products are more likely to malfunction in mid scan then more expensive equipment. For example, the cameras we use tend to get stuck in an inoperable state. When a component fails, most of the time our operator is able to diagnose the problem and complete the scan. To handle the issue with the cameras we installed a remotely operated power source.

Controlling the system over the network means that we don't have a synchronized clock between different components. Thus, a delay is needed to apply changes to the entire system. This, coupled with the inherent capture time of our camera stack, means that a full capture takes a number of hours.

Our system is set up to use a very large number of HDR patterns, requiring long acquisition times and large amounts of memory. In future work we need to greatly reduce the number of images required, perhaps using a small number of preliminary images to reduce the number of exposures and/or patterns.

Our method requires at least a minimal amount of direct illumination in order to distinguish between different primary patterns. Thus, if a section of an object is dominated by subsurface scattering, it will be ignored by our system. Future work is needed to test if in some cases the size of the secondary patterns might be adjusted to minimize this issue.

Conclusions

Using projected spatial patterns to detect different light paths is a promising area for further development of inexpensive acquisition systems. The continued advance in camera and projector quality translates directly into improved method for capturing material properties from existing objects.

Acknowledgments

This work was funded by NSF grants IIS-1064412 and IIS-1218515. The computational resources of the Yale Computer Science Cloud were used in this work.

References

[1] Lensch, H., Kautz, J., Goesele, M., Heidrich, W., and Seidel, H.-P., "Image-based reconstruction of spatial appearance and geometric detail," *ACM Transactions on Graphics (TOG)* **22**(2), 234–257 (2003).

[2] Aittala, M., Weyrich, T., and Lehtinen, J., "Two-shot svbrdf capture for stationary materials," *ACM Trans. Graph.* **34**, 110:1–110:13 (July 2015).

[3] Wu, H. and Zhou, K., "Appfusion: Interactive appearance acquisition using a kinect sensor," in [*Computer Graphics Forum*], Wiley Online Library (2015).

[4] Gkioulekas, I., Levin, A., Durand, F., and Zickler, T., "Micron-scale light transport decomposition using interferometry," *ACM Trans. Graph.* **34**, 37:1–37:14 (July 2015).

[5] Rushmeier, H., Lockerman, Y., Cartwright, L., and Pitera, D., "Experiments with a low-cost system for computer graphics material model acquisition," in [*IS&T/SPIE Electronic Imaging*], 939806–939806, International Society for Optics and Photonics (2015).

[6] Nayar, S. K., Krishnan, G., Grossberg, M. D., and Raskar, R., "Fast separation of direct and global components of a scene using high fre-

quency illumination," in [*ACM Transactions on Graphics (TOG)*], **25**(3), 935–944, ACM (2006).

[7] Salvi, J., Fernandez, S., Pribanic, T., and Llado, X., "A state of the art in structured light patterns for surface profilometry," *Pattern recognition* **43**(8), 2666–2680 (2010).

[8] Potsdamer, J. and Altschuler, M., "Surface measurement by space-encoded projected beam system," *Computer Graphics Image Processing* **18**, 1–17 (1982).

[9] Gu, J., Kobayashi, T., Gupta, M., and Nayar, S. K., "Multiplexed illumination for scene recovery in the presence of global illumination," in [*Computer Vision (ICCV), 2011 IEEE International Conference on*], 691–698, IEEE (2011).

[10] Couture, V., Martin, N., and Roy, S., "Unstructured light scanning robust to indirect illumination and depth discontinuities," *International Journal of Computer Vision* **108**(3), 204–221 (2014).

[11] Gupta, M., Agrawal, A., Veeraraghavan, A., and Narasimhan, S. G., "A practical approach to 3d scanning in the presence of interreflections, subsurface scattering and defocus," *International journal of computer vision* **102**(1-3), 33–55 (2013).

[12] gPhoto, "gPhoto," (2015). http://http://www.gphoto.org/.

[13] Wu, C., "Towards linear-time incremental structure from motion," in [*3D Vision - 3DV 2013, 2013 International Conference on*], 127–134 (June 2013).

[14] Snavely, N., Seitz, S. M., and Szeliski, R., "Photo tourism: exploring photo collections in 3d," in [*ACM transactions on graphics (TOG)*], **25**(3), 835–846, ACM (2006).

[15] OpenCV, "OpenCV," (2015). http://http://www.openv.org/.

[16] Jakob, W., "Mitsuba renderer," (2010). http://www.mitsuba-renderer.org.

[17] Jensen, H. W., Marschner, S. R., Levoy, M., and Hanrahan, P., "A practical model for subsurface light transport," in [*Proceedings of the 28th annual conference on Computer graphics and interactive techniques*], 511–518, ACM (2001).

Author Biography

Yitzchak David Lockerman is a doctoral candidate in the Computer Science Department at Yale University. His research focuses on capturing and reproducing spatially varying patterns for computer graphics applications.

Samuel Brenner is an undergraduate majoring in Computer Science at Yale University.

Joseph Lanzone is an undergraduate majoring in Computer Science at Yale University.

Alexander Doronin is a postdoctoral associate in Computer Science at Yale University. His research focuses on physically based rendering of translucent materials such as human skin.

Holly Rushmeier is a professor of Computer Science at Yale University. She received the BS, MS and PhD degrees in Mechanical Engineering at Cornell University. Prior to Yale she worked at Georgia Tech, the National Institute of Standards and Technology and IBM Watson Research Center. Her current research focuses on material appearance models for computer graphics, applications of perception in realistic image synthesis and applications of computer graphics in cultural heritage. She is a fellow of the Eurographics Association and the recipient of the ACM SIGGRAPH Outstanding Achievement Award.