Edge-Preserving Photometric Stereo via Depth Fusion

Qing Zhang[1] Mao Ye[1] Ruigang Yang[1] Yasuyuki Matsushita[2] Bennett Wilburn[2]
Huimin Yu[3]
University of Kentucky[1] Microsoft Research Asia[2] Zhejiang University[3]

Abstract

We present a sensor fusion scheme that combines active stereo with photometric stereo. Aiming at capturing full-frame depth for dynamic scenes at a minimum of three lighting conditions, we formulate an iterative optimization scheme that (1) adaptively adjusts the contribution from photometric stereo so that discontinuity can be preserved; (2) detects shadow areas by checking the visibility of the estimated point with respect to the light source, instead of using image-based heuristics; and (3) behaves well for ill-conditioned pixels that are under shadow, which are inevitable in almost any scene. Furthermore, we decompose our non-linear cost function into subproblems that can be optimized efficiently using linear techniques. Experiments show significantly improved results over the previous state-of-the-art in sensor fusion.

1. Introduction

The recent availability of consumer low-cost depth cameras brings exciting opportunities for computer vision. By providing the additional depth information, which is unavailable in traditional imaging techniques, they hold the potential to make a number of hard problems in computer vision more tractable in practice. It has already changed the way we interact with computers (in games) [25]. However, the depth quality from the current generation of depth cameras still has plenty of room for improvement. For example, in the most popular Kinect sensor [21], the quantization effect is visible and surface details are lost, as shown in Figure 1(b).

We are motivated by both the opportunities and limitations in commodity depth sensors. Our goal is to significantly improve the depth sensor quality so that it can be used for applications such as 3D modeling and view synthesis. In this paper we present a novel sensor fusion scheme that combines active stereo with photometric stereo. The complimentary nature of stereovision and photometric stereo (PS) has long been recognized. Stereovision is simple to set up and can generate metric measurement, but its accuracy is inversely proportional to the object distance. On the other hand, photometric stereo is known for capturing surface details, but the normal map produced by PS does not provide metric depth measurement. In addition, the resulting 3D surface may suffer from global non-uniform distortions. Therefore several methods have been developed to fuse depth maps with normal maps (*e.g.*, [9, 23, 2]). Our solution is different from all these previous methods in that we *explicitly model the discontinuity of the captured scene*. In other words, we envision that our method can be used as the foundation for the next generation depth sensors to

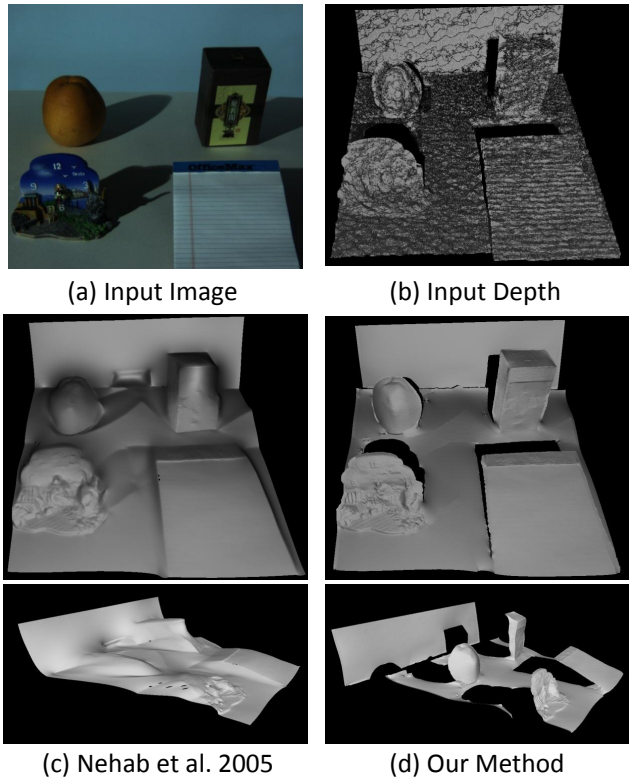


Figure 1. A reconstructed scene after fusion. (a) one of the three input images; (b) captured from a depth sensor; (c) and (d) the reconstructed mesh rendered from two view points, without and with discontinuity and shadow handling.

capture full-frame depth maps at high-quality, rather than a method for an object scanner.

Another constraint from our design consideration is the number of lighting variations. With dynamic scene capture in mind, we limit the number of lighting conditions to three – the minimum to resolve the surface normal. In almost any real world scene, shadows due to occlusions will unavoidably appear and photometric stereo will fail in there areas. We therefore develop a framework to first identify areas in shadow and then allow insufficient number of lighting conditions.

In summary, this paper makes the following technical contributions. First, we develop an automatic fusion algorithm to use the intensity variations under different illumination conditions to refine the depth map produced by a depth sensor. Unlike previous fusion algorithms, an *adaptive weighting* scheme is devised to *preserve surface discontinuities*. Secondly, instead of requiring a given normal map from a complete photometric stereo process, we extend the fusion scheme to use *insufficient number of light sources*. We cast the full optimization problem into a set of linear problems that can be solved efficiently. Our method allows a principled way to deal with pixels in shadow. As we will see later, we demonstrate quality results using only two lights. Finally, our fusion scheme also enables us to identify shadow regions in a way similar to *shadow mapping* in computer graphics.

2. Related Work

Our proposed work is related to both photometric stereo (PS) and sensor fusion. The foundation for PS is described in the pioneering work of Woodham in the eighties [27], in which the surface unit normal of a Lambertian object can be estimated given three distant light sources and corresponding images. The focus of later PS research has two major threads. One is on extending PS to non-Lambertian surfaces (*e.g.*, [11, 24, 15]). While in this paper we make the basic Lambertian surface assumptions, some of these techniques can be incorporated into our framework.

The other thread of PS research is to integrate the normal field from PS to a surface. Horn and Brooks [16] propose the basic method to integrate from measured gradient field using calculus of variations. Frankot and Chellappa [12] use the Fourier basis function to project the possibly non-integrable gradient field onto the nearest integrable slopes. Agrawal *et al*. [1] summarize a collection of integration methods from the traditional approach to the affine transform approach. Harker and O'Leary [13] discretize the traditional cost function to determine a unique least square solution up to an integration constant. Basri *et al*. [5] use the product of normal and surface gradient to avoid the rim error of the gradients and the explicit normal integration.

Shadows in Photometric Stereo PS requires at least three distinctive lighting conditions to resolve the surface normal. Common approaches to address shadows are to use more lighting conditions and check the consistency with the result where the minimum requirement can be met [10, 4, 7]. In the minimum case of three lights, [14] modifies the cost in [7] to detect and deal with pixel shadows. The fundamental assumption made there is that the entire captured surface is continuous and smooth. That is exactly the type of assumption that we in this paper *do not* choose to assume.

Depth Sensor Fusion Given the tremendous interest in commodity depth cameras, there have been several papers on improving depth sensor quality by using sensor fusion. For example, stereo and time-of-flight sensors are fused together for improved quality(*e.g.*, [28, 18]). There are also numerous papers on the general problem of combining positions with normals (*e.g.*, [9, 22, 8, 23, 26]). To deal with dynamic scenes, [17, 3, 2] utilize color information for non-rigid objects. Color lights will interfere the image quality and therefore we *do not* choose. Moreover, none of these methods explicitly models the discontinuities in the scene.

3. Adaptive Depth-Normal Fusing

We first present notations and preliminaries and then describe our adaptive weighting method to combine the depth and normal maps while preserving depth discontinuities. For now, we assume that the corresponding normal map has been calculated and given.

3.1. Fusion of depths and normals

Suppose the 3D surface $S(u,v) = [x,y,z]^T$ is parameterized in a 2D field $\Omega = [u,v]$ and the initial measured surface is S^0. We use the superscript ^{0} to denote the initial measure in this paper. Assume the field $[u,v]$ coincides with the image grid $[i,j]$, and the normal and depth are denoted as $\mathbf{n}(i,j) = N_{ij}$ and $z(i,j) = Z_{ij}$, respectively. Considering the perspective projection of 3D positions and pixels, the surface can be represented in terms of the depth map Z_{ij}:

$$S(i,j) = \left[\frac{i-p_x}{f_x}, \frac{j-p_y}{f_y}, 1\right]^T Z_{ij}, \quad (1)$$

$$\mu_{ij} := \left[\frac{i-p_x}{f_x}, \frac{j-p_y}{f_y}, 1\right]^T, \quad (2)$$

where $[f_x, f_y]$ and $[p_x, p_y]$ are the camera focal length and principal point, respectively. Then the distance of the surface to the measurement is represented using depths:

$$E_p = \sum_{ij} \|\mu_{ij}\|^2 \left(Z_{ij} - Z_{ij}^0\right)^2. \quad (3)$$

To use the normal information, we define the following cost function ("$\cdot$" refers to the dot product of 3D vectors in this paper):

$$E_n = \sum_{ij} \left(N^0_{ij} \cdot \frac{\partial S}{\partial u} \right)^2 + \left(N^0_{ij} \cdot \frac{\partial S}{\partial v} \right)^2, \quad (4)$$

where the derivatives of the surface are:

$$\frac{\partial S}{\partial u}(i,j) = \mu_{ij} \frac{\partial Z}{\partial u}(i,j) + \left[\frac{Z_{ij}}{f_x}, 0, 0 \right]^T, \quad (5)$$

$$\frac{\partial S}{\partial v}(i,j) = \mu_{ij} \frac{\partial Z}{\partial v}(i,j) + \left[0, \frac{Z_{ij}}{f_y}, 0 \right]^T, \quad (6)$$

and $\frac{\partial Z}{\partial u}$ and $\frac{\partial Z}{\partial v}$ are discrete derivatives. Due to the truncated error and the Gibbs phenomenon arising near the gradient discontinuities, we choose a 3-point derivative formula similar to [13].

Combining all terms above, the desired surface is obtained by minimizing the following function in terms of depth values:

$$\begin{aligned} E &= \lambda_p \sum_{i,j} \|\mu_{ij}\|^2 \left(Z_{ij} - Z^0_{ij} \right)^2 \\ &+ \lambda_n \sum_{i,j} \left((N^0_{ij} \cdot \mu_{ij}) \, \frac{\partial Z}{\partial u}\Big|_{ij} + \frac{N^0_{xij}}{f_x} Z_{ij} \right)^2 \\ &+ \lambda_n \sum_{i,j} \left((N^0_{ij} \cdot \mu_{ij}) \, \frac{\partial Z}{\partial v}\Big|_{ij} + \frac{N^0_{yij}}{f_y} Z_{ij} \right)^2, \end{aligned} \quad (7)$$

where $\lambda_p + \lambda_n = 1$ and $\lambda_p, \lambda_n \geq 0$ are blending weights for each penalty. Since the differential operator over the depth map will finally represent a matrix multiplication, the total energy is a quadratic form of depth values and therefore can be solved by an over-constrained linear least square system:

$$\begin{bmatrix} \lambda_p I \mu \\ \lambda_n \left[(N^0 \cdot \mu) \frac{\partial}{\partial u} + \frac{N^0_x}{f_x} \right] \\ \lambda_n \left[(N^0 \cdot \mu) \frac{\partial}{\partial v} + \frac{N^0_y}{f_x} \right] \\ \lambda_s \nabla^2 \end{bmatrix} [Z] = \begin{bmatrix} \lambda_p Z^0 \mu \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad (8)$$

where $[Z]$ means stacking all the depth variables into a column vector, and the multiplications taken in the left hand side are arranged in the order of each corresponding pixels. The smoothness term is to suppress the quantization error and Gibbs phenomenon, where ∇^2 denotes a Laplacian operator on the 4-neighbor image grid and the weight λ_s is chosen small. λ_p and λ_n adjust how the depth and normal affect the final reconstructed surface. In practice, we choose a large λ_n and small λ_p to ensure high quality details as long as the depth bias of the result is comparable to the original one.

3.2. Adaptive weighting algorithm

Traditionally, the fusion weights λ_p and λ_n are prespecified constants, which are only suitable for a single continuous object. Adjusting them in a pixelwise manner turns out ineffective to deal with discontinuities and may result in unwanted artifacts. In order to handle generic scenes, we develop a new automatic weighting algorithm without requiring any manual hard segmentation. This is possible because the discontinuity at different objects' boundaries can be reliably detected from the measured depth map.

According to the first order discrete form of eq. 4, we can consider each pixel in the reconstructed surface as a quadrilateral (shown in Figure 2) that: each quadrilateral is a tiny plane perpendicular to the measured normal and connected to its four direct neighbors. The derivative is then approximated by the center difference $(Z_{i+1} - Z_{i-1})/2$, which can be further decomposed to an average of the forward and backward difference $(Z_{i+1} - Z_i)/2 + (Z_i - Z_{i-1})/2$ as the dotted line. The idea of our adaptive weighting algorithm is

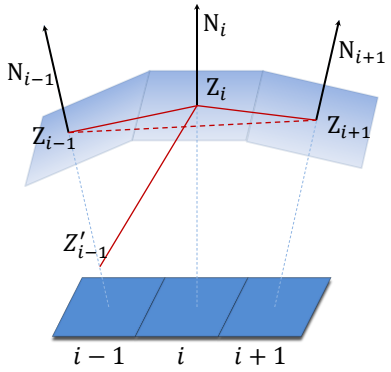


Figure 2. The approximation of surface gradient in one dimension.

to design the differential operator as a bilateral filter: considering both neighbors and the depth range. By choosing a proper range filter, we can approximate the derivative across the discontinuity by only using one side difference. If the range difference is large (*e.g.*, Z'_{i-1} in Figure 2), the range filter will assign a small weight to the "steep" side. If the range difference is small, the operator goes back to the center difference. In other words, in the continuous part all neighbors are weighted involved, and meanwhile on boundaries false neighbors are occluded adaptively.

We choose a shift-invariant Gaussian filter for the depth range at the location $\mathbf{p}_i$ given a general pixel $\mathbf{p}$:

$$s(\mathbf{p}, \mathbf{p}_i) = \exp\left(-\frac{|Z_{\mathbf{p}} - Z_{\mathbf{p}_i}|^2}{2\sigma^2} \right). \quad (9)$$

The weighted differential operators are:

$$\frac{\partial Z_{ij}}{\partial u} = \frac{s_{i-1,j}(Z_{ij} - Z_{i-1,j}) + s_{i+1,j}(Z_{i+1,j} - Z_{ij})}{s_{i-1,j} + s_{i+1,j}}, \quad (10)$$

$$\frac{\partial Z_{ij}}{\partial v} = \frac{s_{i,j-1}(Z_{ij} - Z_{i,j-1}) + s_{i,j+1}(Z_{i,j+1} - Z_{ij})}{s_{i,j-1} + s_{i,j+1}}, \quad (11)$$

where $s_{i-1,j}$ is short for $s((i-1,j),(i,j))$ and so on. The standard deviation σ should be small to make a good distinction, and we choose $\sigma = 0.1$ in our experiment. Besides, in the singular case where both sides are very steep, we do not normalize the operator when the divisor $s_{i-1,j} + s_{i+1,j}$ or $s_{i,j-1} + s_{i,j+1}$ is close to zero. In addition, the Laplacian operator in eq. 8 is also modified in a similar form containing four neighbors.

4. Dealing with Insufficient Lighting

In a general scene, shadows will be inevitable for photometric stereo. For dynamic scene capture, the smaller number of lights is preferred. Under our minimum 3-light setup, shadow pixels in any of the three images will not have sufficient information to fully resolve the normal. In this section we extend our fusion algorithm to deal with an arbitrary number of lights, including these ill-conditioned ones when a pixel is lit by less than three lights. We first formulate the fusion as a nonlinear optimization problem and later present our efficient solution method. We then analyze the case of optimizing the depth with $n \leq 2$-light conditions. Note that in this section, we do not assume a normal map is given. Instead, we directly work with image intensities.

4.1. Problem Statement

Suppose we have images I^k illuminated by distant light sources L_k from different directions $k = 1, \ldots, n$. Instead of computing the normal map a priori from the images, we directly put the image appearance discrepancy to the objective function eq. 7. The goal is to optimize depths in the following form:

$$\begin{aligned} Z &= \arg\min_Z \lambda_p \sum_{i,j} \|\mu_{ij}\|^2 \left(Z_{ij} - Z^0_{ij}\right)^2 \\ &+ \lambda_n \sum_{k=1}^{n} \sum_{i,j} \left(L_k \cdot N_{ij} - I^k_{ij}\right)^2 \\ &+ \lambda_n \sum_{i,j} \left(N_{ij} \cdot \mu_{ij} \left.\frac{\partial Z}{\partial u}\right|_{ij} + \frac{N_{xij}}{f_x} Z_{ij}\right)^2 \\ &+ \lambda_n \sum_{i,j} \left(N_{ij} \cdot \mu_{ij} \left.\frac{\partial Z}{\partial v}\right|_{ij} + \frac{N_{yij}}{f_y} Z_{ij}\right)^2. \end{aligned} \quad (12)$$

Differently from eq. 7, this optimization is nonlinear in the depth map Z_{ij} due to unknown normals N_{ij}. This optimization is experimentally found difficult to converge [26]. Our approach solves this problem by alternately optimizing depths and normals. It not only allows convergence even in the ill-conditioned region, but also avoids the use of nonlinear optimization.

We decouple eq. 12 by iteratively optimizing two steps: firstly, we compute an intermediate normal map $\tilde{N}$ from the following subproblem:

$$\begin{aligned} \min_{\tilde{N}^t_{ij}} \quad & d(\tilde{N}^t_{ij}, N^{t-1}_{ij}) \\ \text{subject to} \quad & \tilde{N}^t_{ij} \cdot L_k = I^k_{ij}, \quad k = 1, 2, \ldots, n, \end{aligned} \quad (13)$$

where $d(\cdot,\cdot)$ represents the geodesic distance between two vectors, the superscript t indicates the iteration number and N^t_{ij} is computed from the mesh after t times iterations. The intermediate normals $\tilde{N}^t_{ij}$ are then combined into the fusion step by solving eq. 8 as our second step. If the number of lights is $n \geq 3$, the solution is unique and hence the whole optimization will converge after one iteration. When the number of lights is insufficient, experiments show that our optimization usually converges in at most 10 iterations.

4.2. $n \leq 2$ light sources

We illustrate the case of $n = 2$ in Figure 3. The constraints restrict the solution space to a 3D line l intersected by two planes $\Pi_1 : \tilde{N} \cdot L_1 = I^1$ and $\Pi_2 : \tilde{N} \cdot L_2 = I^2$. The discussion in this section is for a single pixel and thus we omit the subscript for conciseness.

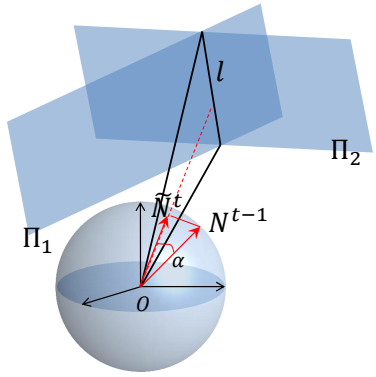


Figure 3. The illustration of optimizing the normal when only two light conditions are given.

By projecting the line onto the unit sphere, we obtain a semi-circle representing the 1-DOF solution space of the normal. The goal of eq. 13 is to find the closest point on the arc to the given normal N^{t-1}. We observe that the line l cannot pass the origin since intensities cannot be zeros. Also the line cannot be parallel to the direction of the given normal N^{t-1}, otherwise N^{t-1} will be parallel to both planes and hence be perpendicular to both plane normals L_1 and L_2, *i.e.*, $L_1 \cdot N^{t-1} = 0$ and $L_2 \cdot N^{t-1} = 0$, which is impossible in our case.

Therefore, we can find the closest vector by projecting the normal N^{t-1} onto the plane that contains both the line

l and origin O:

$$\tilde{N}^t = N^{t-1} - (I_2 L_1 - I_1 L_2) \frac{N^{t-1} \cdot (I_2 L_1 - I_1 L_2)}{\|I_2 L_1 - I_1 L_2\|^2}, \quad (14)$$

and normalizing $\tilde{N}^t$. We note that since the solution space is a semi-circle, there exists a failure case: $\tilde{N}^t$ locates in the other half of the sphere and does not intersect with the arc. This may happen when given severely incorrect initial surface normal. In such a case, additional information or assumptions are required. In our implementation, we simply skip the optimization if 1) the projected normal indicates a negative albedo according to lighting constraints; 2) the angle between $\tilde{N}^t$ and N^{t-1} exceeds the threshold (60^o in our experiment). We do not require any regularization like the curl-free integrability constraints [14] because we did not find any improvement by using them.

It is worth noting that our approach can be extended to deal with the scene under ambient light. Given intensities from three light sources, the ambient light A is constantly added to the constraints, $\tilde{N} \cdot L_k = I^k + A$, $k = 1, 2, 3$. By subtracting from the one with the maximum intensity, we obtain two virtual lights and their Lambertain constraints. Consequently, the ambient light case becomes equivalent to the two-light case.

For the $n = 1$ light source case, we have a hemisphere as the solution space and are unable to search for a reasonable solution. In $n \leq 1$ region, we do not optimize the normals but preserve the Laplacian coordinates [20]. Our experiments show that although we do not perform any optimization directly on normals, the recovered mesh is robust to the noise in $n \leq 1$ region and maintains the local neighborhood well without large deformations.

4.3. Fusion Framework Summary

The above analysis of the well-defined and the degenerated cases provides us with all the elements needed to formulate the normal optimization using arbitrary number of light sources over the image domain. Although the total optimization is nonlinear, we include the pseudo-code of our fusion scheme in Algorithm 1 to emphasize both OPTIMIZE_NORMAL and OPTIMIZE_DEPTH steps are linear in both space and time. The PREPROCESS smooths the captured depth map using bilateral filtering. The COMPUTE_NORMAL triangles the depth map to a 3D mesh by camera intrinsic matrix and returns the facet normal for each pixel. The VISIBILITY detects pixels in shadow.

Shadow Pixel Detection Differently from the previous shadow segmentation by checking the consistency with four images [10, 4] or the graph-cut based scheme [7, 14], we detect shadow map using the reconstructed 3D mesh because: 1) it provides a principled way for shadow detection; 2) the shadow map is robust to image noise and insensitive to the dark colored or specular material, where the image-based approach will likely categorize such regions as the invisible part from all light sources. In our implementation, we render the reconstructed scene in each iteration by Z-buffer and determine that each pixel is visible to which light source or totally invisible.

Algorithm 1 Pseudo-code for the fusion framework.

OPTIMIZE(Z^0, L, I)
$Z^0 \leftarrow$ PREPROCESS(Z^0)
$Z \leftarrow Z^0$
while Z is not converged **do**
  $vis \leftarrow$ VISIBILITY(Z, L, I)
  $N \leftarrow$ COMPUTE_NORMAL(Z)
  $\tilde{N} \leftarrow$ OPTIMIZE_NORMAL(Z, N, vis, L, I)
  $Z \leftarrow$ OPTIMIZE_DEPTH($Z^0, \tilde{N}$)
end while

5. Evaluation and Results

Our approach is evaluated using both synthetic data and real scenes. Using synthetic data, we assess the quantitative accuracy with respect to the ground truth. Using real data, we show the ability of our algorithm to automatically preserve surface discontinuities and also demonstrate the capability to deal with shadows, two light sources or three lights plus ambient light. Error analysis is conducted in a simple case with the ground truth. We shall notice that in our results we do not perform any hard segmentation on the mesh but just not render the triangles that have long edges (greater than 15mm in our result). We set weights $\lambda_d = 0.01$, $\lambda_n = 0.99$, $\lambda_s = 0.1$ described in eq. 8 for all of our experiment. For the sake of comparison, we also implement the method in [23] by plugging in their gradient calculation method in eq. 8. The global weights are set to be the same as ours.

Our Matlab code run on an Intel i5 3.2GHz and 16GB memory PC in less than 50 seconds per iteration at the resolution of 1024×768, in which the sparse linear equation solver takes up about 30 seconds. We run 3 iterations for a 3-light scene and 10 iterations for a 2-light scene.

5.1. Synthetic Data

We render a plane and sphere scene using three directional lights as shown in Figure 4. In the convex case, the hemisphere "pops up" from the plane and cast shadows can be observed on the plane. In the concave case, the hemisphere "digs" into the plane and the shadow region contains both one-light and two-light case, in particular the center part is totally invisible. The sphere radius is 400mm and the center is 1200mm away from the camera center. Initially the ground truth depth map is artificially corrupted with white

noise of maximum 100mm as input. Results from both our method and [23] are compared with the ground truth and shown in Table 1. Notice the significant improvement in accuracy with our method.

	Convex		Concave	
	Mean	Max	Mean	Max
Our method	0.883	75.1	3.2	18.4
Method from [23]	28.8	153.8	19.7	120.8

Table 1. Quantitative evaluation (depth error in mm) on synthetic data.

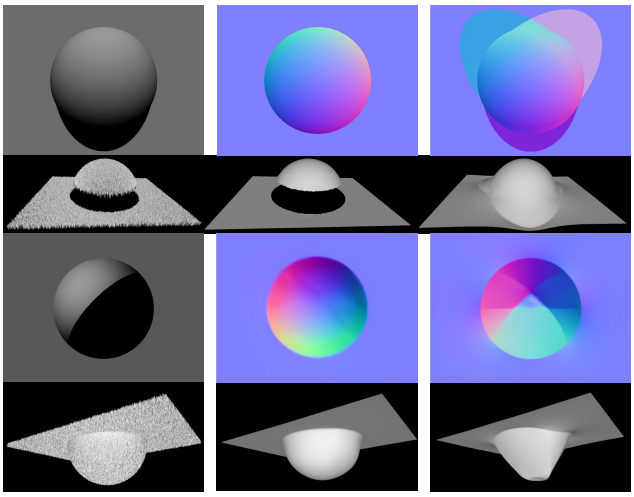


Figure 4. The synthetic data under three different illuminations.

5.2. Real Capture System

We build up the real scene capture system using a depth camera - Kinect [21], and replace its low resolution and nonlinear response color camera with the Point Grey Research Flea2, which has a linear response and 1024×768 resolution at 30Hz. Figure 5 shows the actual hardware of our system. We use three white Luxeon K2 LEDs that are inexpensive, bright and switch instantly. We design a simple microcontroller circuit to trigger the camera at 30Hz and turn on each LED in a sequential manner. Only one of lights is turned on throughout an exposure time. Kinect captures independently and reports the latest buffered depth map. The Flea2 and Kinect cameras are calibrated as well as light directions using the color checker [17]. The depth maps are finally registered and converted to the camera coordinate of the Flea2, and upsampled to dense using the joint bilateral upsampling [19].

Figure 6 shows the result of two painted balls where the bigger ball is occluded by the smaller one. We compute the error map using the ground truth that is obtained by measuring the diameter of each ball in both 2D and 3D, and back

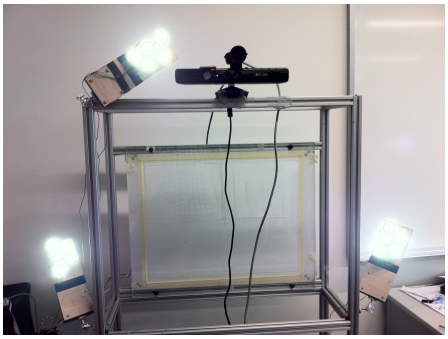


Figure 5. The dynamic scene capture setup.

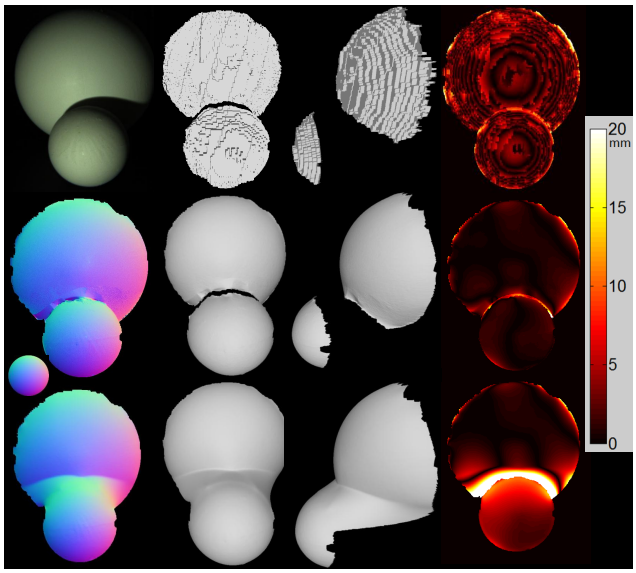
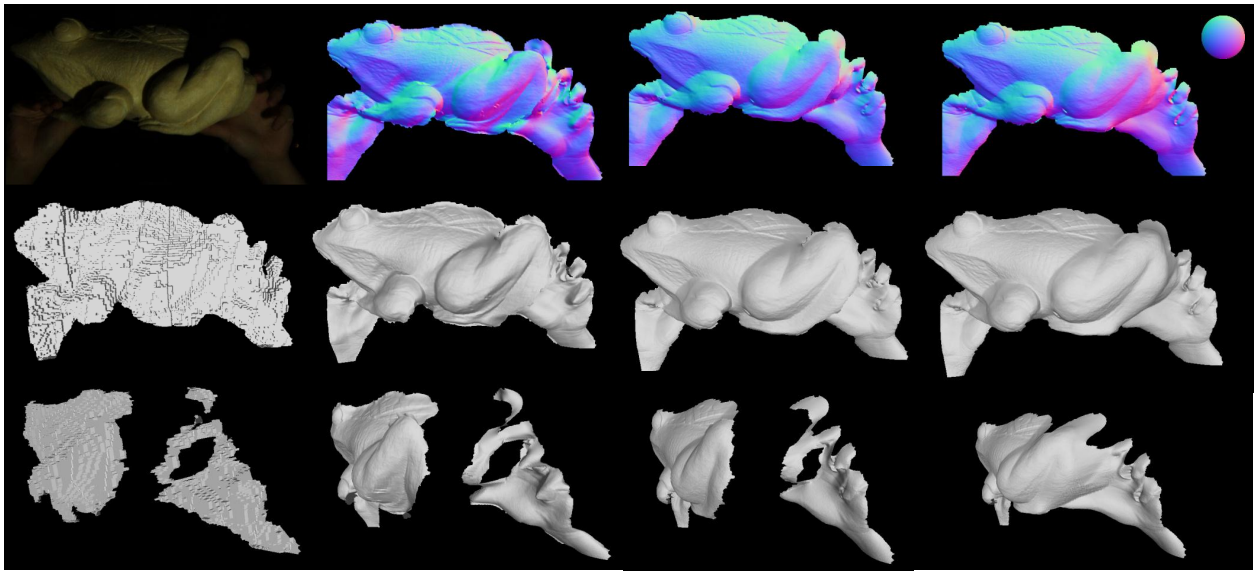


Figure 6. Reconstruction of two balls. The first row contains one input image and the depth map from Kinect. The second row is our result, and the third row is the result of [23]. In the fourth column, the error map of the reconstruction result is visualized.

perspective-projecting the ball center to a 3D location. Using the existing fusion method, the shape recovered in the shadow region is totally smoothed out and stretches balls closer to each other. Our method correctly optimizes the normal map and the depth in the two-light region. Note that artifacts arising around the lower part of the bigger ball are in the one-light region, we do not perform any optimization but smoothing.

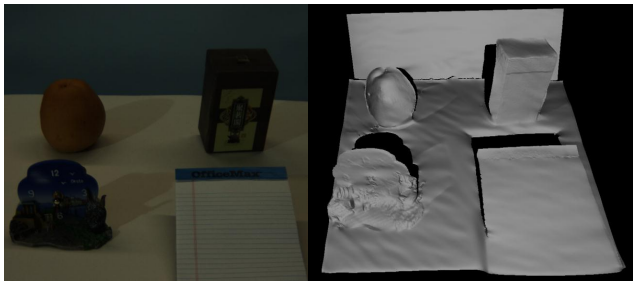
In Figure 7, we show an extreme case by setting one of visibility maps to false, the whole scene is therefore lit by only two lights. Compared with the result of three lights, our result misses some details but still contains much more details than the input. Meanwhile, compared with the existing approach, our final result using the adaptive differential operator correctly maintains discontinuities between its two legs.

Figure 8 shows the same static scene as Figure 1 with



(a) Input Image and Depth (b) Our Method - 2 Lights (c) Our Method - 3 Lights (d) Method from [22] - 3 Lights

Figure 7. The frog model taken under two and three illuminations. The held leg is behind and mostly occluded by the frontal leg in the capture view.



3 lights + ambient light Our Method

Figure 8. The three-light sources plus ambient light result.

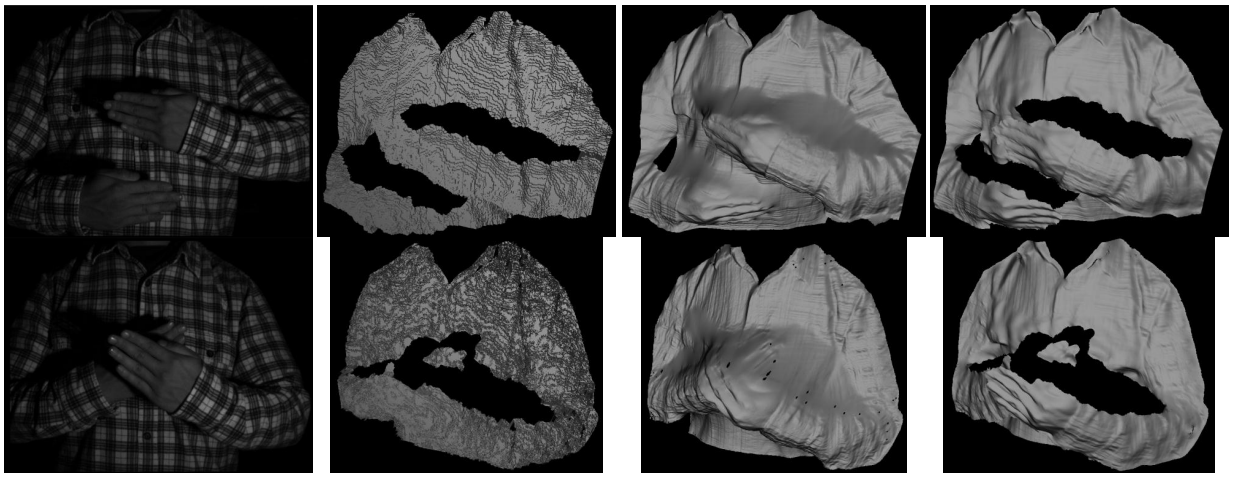
ambient light on. Compared with the ambient-light-free result, artifacts arise in shadows where the constraints reduce to 1-light case.

Figure 9 shows a dynamic sequence of crossing hands in front of the actor's body. Because our camera is low speed (at 30Hz), we have to align pixels across frames. We compute both forward and backward optic flows using the method of Black and Anandan [6] every other three frames. For example, at frame 3, we warp frame 4 using one third of the backward flow between frame 1 and frame 4, also, warp frame 2 using one third of the forward flow between frame 2 and frame 5. Shadows can appear on both hands and the body.

6. Summary and Future Work

In this paper, we have presented an efficient approach for combining captured depth map and images under various illuminating conditions to significantly improve the scene reconstruction quality. The quality improvement is achieved by explicitly modeling discontinuities and shadow areas. Our fusion framework is designed to work with the minimum number of lighting variations and even with ambient light. With its modest computational overhead, we believe our approach can extend the application of current depth capture techniques by simply setting several LED lights around and automatically generating much more detailed shapes.

There are certainly rooms for improvement in our current approach. It will be stuck at local minima when the input depth resolution is too low or provides a severely wrong normal in the ill-conditioned region (*e.g.*, shadows). This can only be remedied by performing strong smoothing. In addition, our method cannot improve the results in areas that receive only one or even no illumination. Fortunately areas like these do not take up too much space in the scene. Other limitations of our current method are like traditional photometric stereos: error occurs in the presence of non-Lambertian reflectance, interreflection, *etc*. The use of the depth map, which is generated by active or passive stereo, avoids catastrophic failure in such cases. A practical concern in our setup is the directional light assumption. We have observed that within the effective distance of the depth sensor, lights can become near and area, introducing soft shadows. We hope to formulate the light source as a light field in next step to get more precise results.



(a) Input Image (b) Input Depth (c) Method from [22] (d) Our Method

Figure 9. Two reconstructed frames from a dynamic sequence.

Acknowledgements

This work is supported in part by University of Kentucky Research Foundation, US National Science Foundation award IIS-0448185, CPA-0811647, MRI-0923131, National Science Foundation of China grant No. 60872069, and Zhejiang Provincial Natural Science Foundation of China grant 2011C11053.

References

[1] A. K. Agrawal, R. Raskar, and R. Chellappa. What is the range of surface reconstructions from a gradient field? In *ECCV*, pages 578–591, 2006. 2

[2] R. Anderson, B. Stenger, and R. Cipolla. Color photometric stereo for multicolored surfaces. In *ICCV*, 2011. 1, 2

[3] R. Anderson, B. Stenger, and R. Cipolla. Augmenting depth camera output using photometric stereo. In *Machine Vision Applications*, June 2011. 2

[4] S. Barsky and M. Petrou. The 4-source photometric stereo technique for three-dimensional surfaces in the presence of highlights and shadows. *PAMI*, 25(10):1239–1252, 2003. 2, 5

[5] R. Basri, D. W. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *IJCV*, 72(3):239–257, 2007. 2

[6] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *ICCV*, pages 231–236, 1993. 7

[7] M. K. Chandraker, S. Agarwal, and D. J. Kriegman. Shadowcuts: Photometric stereo with shadows. In *CVPR*, pages 1–8, 2007. 2, 5

[8] C.-Y. Chen, R. Klette, and C.-F. Chen. Shape from photometric stereo and contours. In *CAIP*, pages 377–384, 2003. 2

[9] J. E. Cryer, P.-S. Tsai, and M. Shah. Integration of shape from shading and stereo. *Pattern Recognition Society*, 28(7):1033–1043, 1995. 1, 2

[10] M. S. Drew. Reduction of rank–reduced orientation–from–color problem with many unknown lights to two-image known-illuminant photometric stereo. In *In: IEEE International Symposium on Computer Vision*, pages 419–424, 1995. 2, 5

[11] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 2

[12] R. T. Frankot, R. Chellappa, and S. Member. A method for enforcing integrability in shape from shading algorithms. *PAMI*, 10:439–451, 1988. 2

[13] M. Harker and P. O'Leary. Least squares surface reconstruction from measured gradient fields. In *CVPR*, 2008. 2, 3

[14] C. Hernández, G. Vogiatzis, and R. Cipolla. Shadows in three-source photometric stereo. In *ECCV*, pages 290–303, 2008. 2, 5

[15] T. Higo, Y. Matsushita, and K. Ikeuchi. Consensus photometric stereo. In *CVPR*, pages 1157–1164, 2010. 2

[16] B. K. P. Horn and M. J. Brooks. The variational approach to shape from shading. *Computer Vision, Graphics, and Image Processing*, 33(2):174–208, 1986. 2

[17] H. Kim, B. Wilburn, and M. Ben-Ezra. Photometric stereo for dynamic surface orientations. In *ECCV*, pages 59–72, 2010. 2, 6

[18] Y. M. Kim, C. Theobalt, J. Diebel, J. Kosecka, B. Micusik, and S. Thrun. Multi-view image and tof sensor fusion for dense 3d reconstruction. In *Proc. of 3DIM 2009*, 2009. 2

[19] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. *SIGGRAPH*, 26(3):96, 2007. 6

[20] M. Liao, Q. Zhang, H. Wang, R. Yang, and M. Gong. Modeling deformable objects from a single depth camera. In *ICCV*, pages 167–174, 2009. 5

[21] Microsoft. Kinect camera. http://www.xbox.com/en-US/kinect/default.htm, 2010. 1, 6

[22] M. G.-H. Mostafa, S. M. Yamany, and A. A. Farag. Integrating shape from shading and range data using neural networks. In *CVPR*, pages 2015–2020, 1999. 2

[23] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi. Efficiently combining positions and normals for precise 3D geometry. *SIGGRAPH*, 24(3), aug 2005. 1, 2, 5, 6

[24] M. Oren and S. K. Nayar. Generalization of the lambertian model and implications for machine vision. *IJCV*, 14(3):227–251, 1995. 2

[25] J. Shotton, A. FItzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, June 2011. 1

[26] G. Vogiatzis, C. Hernández, and R. Cipolla. Reconstruction in the round using photometric normals and silhouettes. In *CVPR*, pages 1847–1854, 2006. 2, 4

[27] R. J. Woodham. Photometric method for determining surface orientation from multiple images. In *Optical Engineering*, volume 19(1), pages 139–144, 1980. 2

[28] J. Zhu, L. Wang, R. Yang, and J. Davis. Fusion of time-of-flight depth and stereo for high accuracy depth maps. In *CVPR*, 2008. 2