

# Robust Multiview Photometric Stereo using Planar Mesh Parameterization

Jaesik Park, *Member, IEEE*, Sudipta N. Sinha, *Member, IEEE*,  
 Yasuyuki Matsushita, *Senior Member, IEEE*, Yu-Wing Tai, *Senior Member, IEEE*,  
 and In So Kweon, *Member, IEEE*.

**Abstract**—We propose a robust uncalibrated multiview photometric stereo method for high quality 3D shape reconstruction. In our method, a coarse initial 3D mesh obtained using a multiview stereo method is projected onto a 2D planar domain using a planar mesh parameterization technique. We describe methods for surface normal estimation that work in the parameterized 2D space that jointly incorporates all geometric and photometric cues from multiple viewpoints. Using an estimated surface normal map, a refined 3D mesh is then recovered by computing an optimal displacement map in the same 2D planar domain. Our method avoids the need of merging view-dependent surface normal maps that is often required in conventional methods. We conduct evaluation on various real-world objects containing surfaces with specular reflections, multiple albedos, and complex topologies in both controlled and uncontrolled settings and demonstrate that accurate 3D meshes with fine geometric details can be recovered by our method.

**Index Terms**—Multiview Photometric Stereo, Planar Mesh Parametrization.

## 1 INTRODUCTION

ACCURATE 3D shape reconstruction from images is of broad interest in computer vision. Recent advances in geometric techniques including structure from motion (SfM) [19] and multiview stereo (MVS) [28] have enabled accurate 3D reconstruction of challenging scenes. These methods generally rely on recovering dense pixel correspondences in images recorded from multiple viewpoints and estimate depth maps, from which a 3D shape of the scene is determined. On the other hand, photometric methods, such as shape-from-shading [14] and photometric stereo [36], use shading cues to estimate per-pixel surface normals. Such methods avoid the difficulty of inferring dense pixel correspondence but only recover the shape information in the form of surface normals. These two types of approaches have complementary strengths and have been combined in prior work on 3D reconstruction [8], [11], [21], [24], [37], [38], [40], [41].

In this paper, we present an efficient and robust multiview photometric stereo method that naturally combines all available geometric and photometric cues for accurate 3D scene reconstruction. The input to our method is a set of images captured from multiple viewpoints under varying light sources for each viewpoint. Our method first recovers a coarse 3D mesh of a scene using existing SfM and MVS methods, which serves as an initialization for the proposed

method. The key idea of our method is to transform the initial 3D mesh into a parameterized 2D space using a distortion minimizing mesh parameterization technique [29], and estimate a surface normal of each point using all the observations of the scene obtained from varying viewpoints under different lightings. Finally, the initial coarse 3D shape is refined by estimating an optimal displacement map represented in the same 2D domain. Camera poses and light source directions are both automatically calibrated in our method.

For surface normal estimation, we present two methods. The first method operates on dense blocks of an observation matrix formed by stacking pixel intensities from the input images captured from multiple viewpoints under different lightings. This method can efficiently deal with missing elements in the observation matrix, which arise due to the lack of visibility of local surface patches from certain viewpoints, given that the number of missing elements is not significant. For handling more challenging cases, where the number of missing elements is significant or the number of observations is small, we develop the second method that is based on a low-rank matrix factorization approach. The second method shows greater robustness against a higher amount of missing data at the cost of more computation.

Once we obtain the surface normal estimates, subsequently we refine the initial coarse 3D mesh based on the surface normal. The problem of mesh refinement is formulated as recovering the optimal displacement map in the same parameterized 2D space. The estimated 2D displacement map encodes fine geometric details of a final 3D shape. Unlike prior method that merges view-dependent normal maps [24], our method performs both surface normal estimation and shape refinement in a unified manner in the parameterized 2D space.

The proposed method has a few advantages over existing techniques. First, with our method, multiview phot-

- Jaesik Park is with Intel Labs, CA, USA,  
*E-mail: jaesik.park@intel.com*
- Sudipta N. Sinha is with Microsoft Research, WA, USA,  
*E-mail: Sudipta.Sinha@microsoft.com*
- Yasuyuki Matsushita is with Osaka University, Japan,  
*E-mail: yasumat@ist.osaka-u.ac.jp*
- Yu-Wing Tai is with SenseTime Group Limited, Hong Kong,  
*E-mail: yuwing@gmail.com*
- In So Kweon (corresponding author) is with KAIST, Republic of Korea.  
*E-mail: iskweon77@kaist.ac.kr*

Manuscript received XXXX XX, XXXX; revised XXXX XX, XXXX.

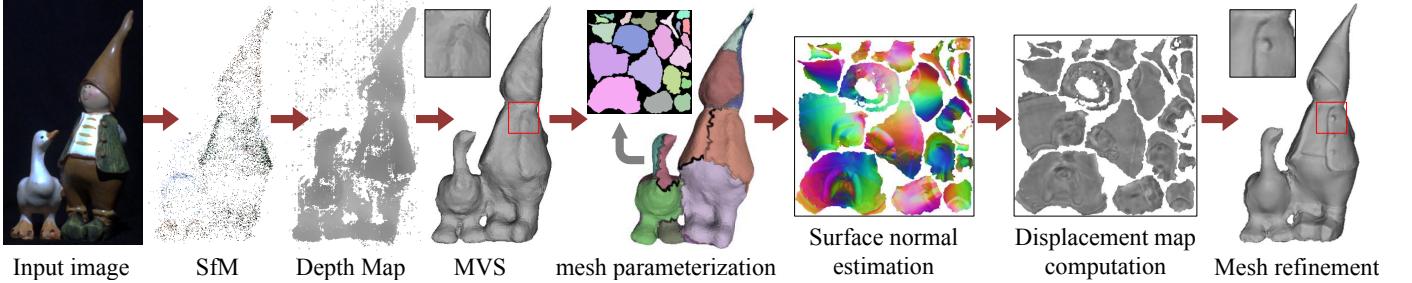


Fig. 1. Overview of our method. Structure-from-motion is used to calibrate cameras and multiview stereo is used to recover a coarse mesh. After parameterizing the coarse mesh, multiview photometric stereo and mesh refinement are performed in a parameterized 2D space.

TABLE 1

Comparison of our method with prior techniques [11], [24]. We use a different 3D shape representation, which is key to the high accuracy and efficiency of the normal map estimation and mesh refinement steps in our method.

	Base mesh	Calibration	3D shape representation	Optimize by	Normals estimated in
Nehab <i>et al.</i> [24]	3D scanner	Manual	Regular 3D mesh	Mesh refinement	Individual image
Hernandez <i>et al.</i> [11]	Visual hull	Auto	Regular 3D mesh	Mesh refinement	Face normals of mesh
Our method	SfM+MVS	Auto	Base mesh+displacement map	Displacement map	Parameterized 2D space

tometric stereo can be naturally formulated in the 2D domain since all the input images captured from multiple viewpoints under different illumination can be warped into the same 2D space. In contrast to prior methods, where surface normals are first estimated independently in each view and then subsequently merged, our method treats all the images from multiple viewpoints simultaneously and avoids the need for post-merging. Second, our method can easily increase the resolution of the normal and displacement maps in the 2D domain and efficiently recovers a detailed 3D mesh by leveraging the full resolution of the input images. Third, since our method performs mesh refinement in the 2D space, it requires fewer parameters than directly working in the original 3D mesh representation. These advantages make the multiview photometric stereo problem more tractable.

This paper extends the preliminary version of our work [27] in a few respects. First, we introduce a robust matrix factorization-based technique that can handle more challenging datasets by exploiting the low-rank structure of the observation matrix. The new method can handle more general visibility patterns and outliers at the cost of more computation. We evaluate the original dense block-based and matrix factorization-based methods under various camera-light configurations and objects with different reflectances, and their individual merits are discussed. We also show a 3D reconstruction result obtained from a fully uncontrolled setting, where the target object is manually rotated and illuminated by a hand-held light source.

## 2 RELATED WORK

The idea of fusing geometric and photometric cues for high-quality 3D scanning is gaining attention due to their complementary strengths; multiview geometric approaches are getting mature and reliable enough for obtaining coarse shape information, but the recovered shape suffer from textureless regions of the scene. On the other hand, photometric approaches can generally recover fine details by estimating per-pixel surface normals, while it does not directly provide

a metric depth information. Recent successful 3D reconstruction methods effectively combine these two types of approaches.

Nehab *et al.* [24] propose an efficient method for 2D depth map refinement using a surface normal map by adjusting depth values based on orthogonality between depth gradients and surface orientations. Zhang *et al.* [42] extend their method to better preserve depth discontinuities. Okatani and Deguchi [26] propose a probabilistic framework for shape refinement using the first-order derivative of surface normals. Lu *et al.* [21] use a giga-pixel camera to estimate ultra high-resolution surface normals via photometric stereo to refine a low-resolution depth map captured by a structured light. While these methods are effective and can be used for recovering a full 3D mesh, they require additional processing for registering and merging multiple view-dependent depth maps. Instead of refining geometry, Zhou *et al.* [44] propagate sparse points that are recovered by SfM according to iso-depth contour segments derived from intensity profiles. To acquire the reliable intensity profiles, they capture dozens of images at each viewpoint. Their method does not require an optimization step for fusing photometric/geometric information and works well for non-Lambertian objects.

For 3D mesh refinement, Nehab *et al.* [24] state that their 2D depth refinement method can be extended to handle a 3D mesh while it is not explicitly shown in the paper. Lensch *et al.* [17] introduce a generalized method for modeling non-Lambertian surfaces using wavelet-based BRDFs and use it for mesh refinement. Hernandez *et al.* [11] propose an effective approach that jointly estimates geometry and per-triangle normal in an unified manner. It can substantially deform the initial geometry by minimizing a quadratic energy function. In addition, a robust approach for self-calibration of light directions is also discussed. Wu *et al.* [37] use the spherical harmonics representation to estimate global illumination and refine a preliminary mesh using photometric stereo by minimizing  $\ell_1$  penalties. In their extended approach [38], geometric details are added using shape-from-shading under natural lightings. Vlasic *et*

al. [34] integrate view-dependent normal maps into partial meshes, then deform them with thin-plate offsets to improve the alignment with preserving geometric details. Choe *et al.* [3] use near infra-red images of Kinect to refine the geometry reconstructed via KinectFusion [15].

Table 1 summarizes the similarities and differences among our approach and the two closely related methods proposed by Nehab *et al.* [24] and Hernandez *et al.* [11]. These 3D mesh refinement methods generally use a high-resolution mesh in order to attain high frequency details recovered by photometric methods. Determining the proper mesh resolution a priori is not straightforward in these techniques because it depends on the viewpoint and the image resolution. To work around, these techniques use an over-sampled mesh resolution that is high enough not to lose the geometric details. In contrast, our method allows the mesh resolution to be derived directly from the normal map resolution and avoids the problem of under-sampling or over-sampling mesh vertices. In addition, our 2D parameterization approach performs mesh refinement efficiently, where only 1D vertex displacements are optimized rather than directly updating the 3D vertex coordinates of the mesh. The parameterized 2D space jointly handles the input images from multiple viewpoints, and it allows us to apply recent low-rank factorization techniques [2], [25], [39] because of the increased number of aligned observations.

### 3 PROPOSED METHOD

This section describes the key elements of the proposed method. For now, let us assume that all the cameras are calibrated and the initial base mesh is available. The methods for auto-calibration and obtaining the initial base mesh are explained later. After describing the mesh parameterization scheme, we explain how surface normal estimation and mesh refinement is performed in the parameterized 2D space. Figure 1 shows an overview of our approach.

#### 3.1 Mesh Parameterization

In our method, the base triangle mesh, denoted by  $\mathcal{M}$ , is mapped to a parameterized 2D space using a piecewise continuous function  $f_{\mathcal{M}} : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ . The mapping is referred to as mesh parameterization [29]. While this process is not restricted to a particular mesh parameterization method, in this paper, we use the iso-charts method proposed by Zhou *et al.* [43] that minimizes non-uniform distortions of the original mesh by finding optimal cuts that partition the mesh into segments. Each connected segment is mapped by its own mapping function to a single *chart* in the parameterized space. We denote the parameterized 2D space as  $\mathcal{U}$ , which contains an arbitrary arrangement of the charts. Using iso-charts, we also obtain a one-to-one inverse mapping  $f_{\mathcal{M}}^{-1}$  from a 2D point  $\mathbf{u} = [u, v]^T$  in  $\mathcal{U}$  to a 3D point  $\mathbf{x}$  on the mesh  $\mathcal{M}$ . For maximally benefiting from the photometric stereo estimates, the resolution of  $\mathcal{U}$ , or referred as *texture map resolution* is set proportional (80%) to the input image resolution.

#### 3.2 Image Warping

Using the camera calibration and the inverse mapping  $f_{\mathcal{M}}^{-1}$ , we warp input images  $\mathcal{I}$  to images  $\mathcal{I}'$  in the  $\mathcal{U}$  coordinates.

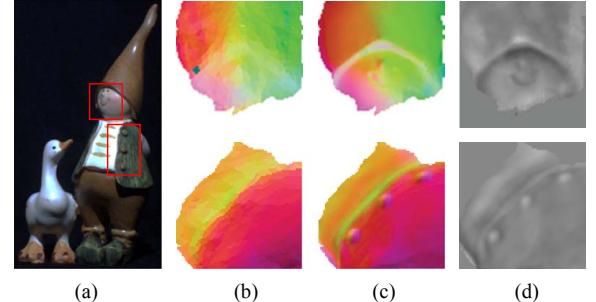


Fig. 2. Example of surface normal map and displacement map estimation. (a) Input image. (b) Initial normal map obtained from a base mesh, in  $\mathcal{U}$ . (c) Disambiguated normals from photometric stereo in  $\mathcal{U}$ . Here, unit 3D vectors have been linearly mapped to RGB. (d) Estimated displacement map.

---

#### Algorithm 1: Image Warping

---

**Input:** Image  $\mathcal{I}$ , camera projection matrix  $\mathcal{P}$ , mesh  $\mathcal{M}$  and its face visibility

**Output:** Warped image  $\mathcal{I}'$

**for** each pixel  $\mathbf{u} \in \mathcal{U}$  **do**

```

    Find triangle  $t \in \mathcal{U}$  that contains  $\mathbf{u}$ 
    Find barycentric coefficients,  $\mathbf{w}_t$  for  $\mathbf{u}$  in  $t$ 
    Find face  $f \in \mathcal{M}$  that maps to  $t$  and its vertices  $\{\mathbf{x}_t\}$ 
    if  $f$  is visible then
         $\mathbf{x}' \leftarrow$  Barycentric-interpolation( $\{\mathbf{x}_t\}$ ,  $\mathbf{w}_t$ )
         $\mathcal{I}'(\mathbf{u}) \leftarrow \mathcal{I}(\mathcal{P}\mathbf{x}')$ 
    
```

---

The images are warped using the standard inverse mapping technique, *i.e.*, we begin with a pixel  $\mathbf{u}$  in the  $\mathcal{U}$  coordinates and determine its corresponding pixel location in the input image  $\mathcal{I}$  via the inverse mapping function  $f_{\mathcal{M}}^{-1}$ . Since  $f_{\mathcal{M}}^{-1}$  is discrete, we use a piece-wise linear interpolation to approximate continuous  $f_{\mathcal{M}}^{-1}$ . Specifically, our method finds the projected mesh face that encloses pixel  $\mathbf{u}$  of the  $\mathcal{U}$  coordinates and determines 3D position  $\mathbf{x}'$  that corresponds to pixel  $\mathbf{u}$  using barycentric interpolation [33].

Finally, the intensity of pixel  $\mathbf{u}$  in  $\mathcal{I}'$  is determined by mapping the pixel in image  $\mathcal{I}$  via the 3D scene point  $\mathbf{x}'$ . This procedure is summarized in Algorithm 1. We use kd-trees [23] to accelerate the search for the 2D triangle. The warping function is computed only once, and it is applied to multiple images recorded from the viewpoint but illuminated under varying lighting conditions. During the image warping, only visible mesh faces are considered, and  $z$ -buffering is used for checking the visibility of faces from the camera.

#### 3.3 Surface Normal Estimation

One of the key benefits of the distortion minimizing mesh parameterization scheme [43] is that pixels of the multiview images are well aligned in the parameterized 2D space without significant error caused by viewpoint variations. Unlike single-view photometric stereo, in our case, we have more observations from nearby viewpoints that are reasonably well aligned using the base mesh geometry. Therefore, the parameterization allows the images from multiple viewpoints under varying lightings to be effectively used for multiview photometric stereo.

We now describe two methods for estimating surface normals given warped images  $\mathcal{I}'$ ; one is an efficient block-

based method, and the other is a matrix factorization-based method that is robust against outliers. To simplify discussions, we begin with assuming a Lambertian image formation model. We will see later that this assumption can be relaxed to handle more diverse reflectances. By denoting the image intensities in the form of an observation matrix  $\mathbf{O} \in \mathbb{R}^{p \times q}$ , where  $p$  is the number of valid pixels in  $\mathcal{U}$ , and  $q$  is the number of images (multiple views  $\times$  multiple light directions), the Lambertian image formation model can be expressed as  $\mathbf{O} = \mathbf{NL}^T$ . Here,  $\mathbf{N} \in \mathbb{R}^{p \times 3}$  is an albedo-scaled surface normal matrix, and  $\mathbf{L} \in \mathbb{R}^{q \times 3}$  represents a lighting matrix.

In our case, the observation matrix  $\mathbf{O}$  has many missing elements because of the visibility of a scene point from various viewpoints. To handle this problem, the first approach finds dense sub-matrices in  $\mathbf{O}$  and aggregates individual solutions to obtain a full surface normal map. It is computationally efficient but susceptible to difficult visibility patterns and non-Lambertian pixels that break the dense block structures. The second approach is built upon robust matrix factorization that can better handle these challenging cases at the cost of more computation. In the following, we describe these two techniques; dense block-based and matrix factorization-based methods, denoted as *Block* method and *Factorization* method, respectively.

### 3.3.1 Dense Block-based Method

The *Block* method computes the surface normal  $\mathbf{N}$  using subsets of the observations, which form dense block sub-matrices in  $\mathbf{O}$ . In general, finding dense sub-matrices  $\mathbf{O}_S$  from the matrix  $\mathbf{O}$  is a NP-hard problem. However, if the columns of  $\mathbf{O}$  are arranged in the order of image recording, which is easy to do in our setting, valid intensity observations usually appear in the consecutive columns of  $\mathbf{O}$ . The problem of finding the sub-matrices is, therefore, reduced to finding maximum cliques in an interval graph. We use the method proposed by Gupta *et al.* [6] for finding multiple and overlapping dense sub-matrices in the original observation matrix  $\mathbf{O}$ .

Given a set of sub-matrices  $\{\mathbf{O}_S\}$ , we apply uncalibrated photometric stereo [9] to each  $\mathbf{O}_S$ . Each sub-matrix  $\mathbf{O}_S$  can be factored and approximated by a product of two rank-3 matrices via singular value decomposition (SVD) as

$$\mathbf{O}_S = \mathbf{U}\Sigma\mathbf{V}^T \approx \mathbf{U}_3\Sigma_3\mathbf{V}_3^T, \quad (1)$$

where  $\Sigma$  is a diagonal singular value matrix, and  $\mathbf{U}$  and  $\mathbf{V}$  are orthonormal matrices containing singular vectors.  $\Sigma_3$ ,  $\mathbf{U}_3$ , and  $\mathbf{V}_3$  are rank-3 approximations of  $\Sigma$ ,  $\mathbf{U}$ , and  $\mathbf{V}$ , respectively. Corresponding surface normal and light matrices  $\mathbf{N}_S$  and  $\mathbf{L}_S$  can be recovered up to a linear ambiguity as

$$\mathbf{O}_S \approx \mathbf{U}_3\Sigma_3\mathbf{V}_3^T = (\mathbf{N}_S\mathbf{A}^{-1})(\mathbf{A}\mathbf{L}_S^T), \quad (2)$$

where  $\mathbf{N}_S = \mathbf{U}_3\Sigma_3^{\frac{1}{2}}$ ,  $\mathbf{L}_S^T = \Sigma_3^{\frac{1}{2}}\mathbf{V}_3^T$ , and  $\mathbf{A}$  is a non-singular  $3 \times 3$  matrix that represents a general linear shape-light ambiguity that exists in uncalibrated photometric stereo.

To automatically resolve the linear ambiguity  $\mathbf{A}$ , we use a mesh normal matrix  $\mathbf{N}_f \in \mathbb{R}^{p \times 3}$  obtained from the base mesh, which is coarse yet contains sufficient low-frequency information of the surface normal. Specifically, we regard

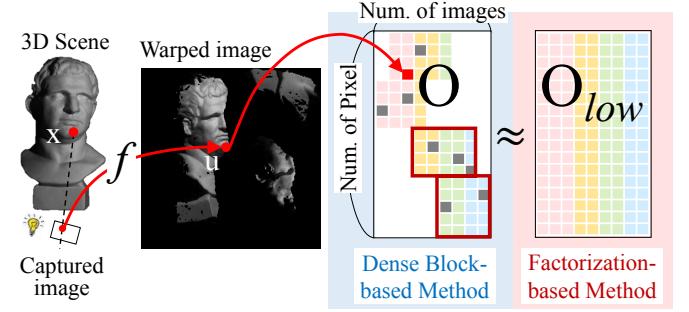


Fig. 3. Illustration of the mapping of pixel intensities to the observation matrix  $\mathbf{O}$ . It shows surface normal can be estimated by solving dense block matrices (Sec. 3.3.1) or by applying factorization-based method (Sec. 3.3.2).

$\mathbf{N}_f \approx \mathbf{N}_S\mathbf{A}^{-1} = \mathbf{U}_3\Sigma_3^{\frac{1}{2}}\mathbf{A}^{-1}$ . Using the pseudo-inverse of  $\mathbf{N}_f$ , we solve for  $\mathbf{A}$  and obtain the surface normal estimate  $\hat{\mathbf{N}}_S$  as

$$\begin{cases} \mathbf{A} & \leftarrow (\mathbf{N}_f^T\mathbf{N}_f)^{-1}\mathbf{N}_f^T\mathbf{U}_3\Sigma_3^{\frac{1}{2}}, \\ \hat{\mathbf{N}}_S & = \mathbf{U}_3\Sigma_3^{\frac{1}{2}}\mathbf{A}^{-1}, \end{cases} \quad (3)$$

where  $\hat{\mathbf{N}}_S$  is a disambiguated surface normal matrix that corresponds to the subset  $S$ . To combine multiple solutions for different overlapping sub-matrices  $\mathbf{O}_S$ , we compute a weighted sum to consolidate them as

$$\mathbf{n}_p(\mathbf{u}) = \frac{1}{M} \sum_S (\mathbf{n}_f(\mathbf{u})^T \mathbf{n}_S(\mathbf{u})) \mathbf{n}_S(\mathbf{u}), \quad (4)$$

where  $\mathbf{n}_f(\mathbf{u}), \mathbf{n}_S(\mathbf{u}) \in \mathbb{R}^3$  are unit vectors taken from a row of  $\mathbf{N}_f$  and  $\mathbf{N}_S$ , respectively, for pixel  $\mathbf{u}$ .  $\mathbf{n}_f^T \mathbf{n}_S (= w_S)$  is a weighting factor for reducing the effect of outliers, which is the cosine of the angle between the face normal and estimated normal vectors, and the term  $M = \Sigma w_S$  normalizes the weighted sum. Figure 2 (c) shows an example of the computed surface normal maps in the described manner.

The *Block* method computes the surface normals efficiently by finding dense block structures in the observation matrix. This method, however, has the following limitations:

- The observation matrix  $\mathbf{O}$  tends to have an uneven distribution of non-zero elements, which often makes it difficult to find dense blocks with an appropriate size.
- When  $\mathbf{O}$  contains non-Lambertian observations (e.g., specular and shadow), the solution of Eq. (3) based on the least squares solution is affected by outliers.

We address these issues by proposing a new method described in the next section.

### 3.3.2 Factorization-based Method

We now describe a robust approach to surface normal estimation via a low-rank matrix factorization method [2]; a *Factorization* method. To estimate surface normals  $\mathbf{N}$  and lightings  $\mathbf{L}$  from the observation  $\mathbf{O}$ , we solve the following optimization problem.

$$\hat{\mathbf{N}}, \hat{\mathbf{L}} = \underset{\mathbf{N}, \mathbf{L}}{\operatorname{argmin}} \|\mathbf{W} \odot (\mathbf{O} - \mathbf{NL}^T)\|_1 + \frac{\lambda}{2} (\|\mathbf{N}\|_F^2 + \|\mathbf{L}\|_F^2), \quad (5)$$

where  $\odot$  denotes a Hadamard product operator,  $\mathbf{W} \in \{0, 1\}$  is an indicator matrix to represent observed (1) or unobserved elements (0). Equation (5) explicitly enforces the rank of  $\mathbf{O}$  to be 3 since it is expressed by the product of the two rank-3 matrices  $\mathbf{N}$  and  $\mathbf{L}$ . Because there are many possible factorization, we use the second term for making the decomposition unique, *i.e.*, selecting one where the matrices  $\mathbf{N}$  and  $\mathbf{L}$  have Frobenius norm as small as possible. They also make the objective to be less sensitive to the issue of stacking in local minima [2].

The estimates  $\tilde{\mathbf{N}}$  and  $\tilde{\mathbf{L}}$  in Eq. (5) also have a shape-light ambiguity  $\mathbf{A}$  as described in the previous section. We regard  $\mathbf{N}_f \mathbf{A} \approx \tilde{\mathbf{N}}$  and compute the pseudo-inverse of  $\mathbf{N}_f$  to obtain the disambiguated surface normals  $\hat{\mathbf{N}}$  as

$$\begin{cases} \mathbf{A} & \leftarrow (\mathbf{N}_f^T \mathbf{N}_f)^{-1} \mathbf{N}_f^T \tilde{\mathbf{N}}, \\ \hat{\mathbf{N}} & = \tilde{\mathbf{N}} \mathbf{A}^{-1}. \end{cases} \quad (6)$$

By solving Eqs. (5) and (6), we obtain a low-rank approximation of  $\mathbf{O}$  denoted as  $\mathbf{O}_{low} = \tilde{\mathbf{N}} \tilde{\mathbf{L}}^T$ . When a sufficient number of observations are given, the method effectively recovers the low-rank approximation of  $\mathbf{O}$  without being affected by outliers. However, recording a large number of images in practice is often inconvenient unless the imaging setup is fully automatic.

To make the method available for fewer images, we additionally enforce  $\mathbf{N}$  to be similar to  $\mathbf{N}_f$  by adding a  $\ell_2$  regularizer of  $\mathbf{N}$  to Eq. (5) as

$$\begin{aligned} \hat{\mathbf{N}}, \hat{\mathbf{L}} &= \underset{\mathbf{N}, \mathbf{L}}{\operatorname{argmin}} \|\mathbf{W} \odot (\mathbf{O}' - \mathbf{Z})\|_1 \\ &+ \frac{\lambda_1}{2} (\|\mathbf{N}\|_F^2 + \|\mathbf{L}\|_F^2) + \frac{\lambda_2}{2} (\|\mathbf{N} - \mathbf{N}_f\|_F^2), \\ &\text{subject to } \mathbf{Z} = \mathbf{N} \mathbf{L}^T. \end{aligned} \quad (7)$$

Compared to Eq. (5), now there is an additional term  $\|\mathbf{N} - \mathbf{N}_f\|_F^2$  and an auxiliary variable  $\mathbf{Z}$  that is used for efficient optimization. We refer the method with the regularizer (Eq. (7)) as the *Factorization with regularization* method. Equation (7) uses a new matrix  $\mathbf{O}'$  instead of  $\mathbf{O}$ . Its purpose is to remove the effect of surface albedo in  $\mathbf{O}$ , since  $\mathbf{N}$  is compared with normalized surface normals  $\mathbf{N}_f$  in Eq. (7).  $\mathbf{O}'$  is computed by dividing elements in each row of  $\mathbf{O}$  with the maximum value in each row. This approximation is based on the assumption that for at least one observation, the corresponding surface normal and light direction vectors are parallel. Under this approximation, the maximum observed intensity is equal to the surface albedo since the dot product between the surface normal and light direction vectors becomes one. We found that this heuristic works reasonably well on the various acquisition configurations and surface BRDFs as we will see in the experiment section.

We optimize Eq. (7) by an Augmented Lagrange Multiplier (ALM) method [18]. Let us rewrite Eq. (7) as

$$\begin{aligned} \underset{\mathbf{Z}, \mathbf{N}, \mathbf{L}, \mathbf{Y}, \alpha}{\operatorname{argmin}} & \|\mathbf{W} \odot (\mathbf{O}' - \mathbf{Z})\|_1 + \frac{\lambda_1}{2} (\|\mathbf{N}\|_F^2 + \|\mathbf{L}\|_F^2) \\ &+ \frac{\lambda_2}{2} (\|\mathbf{N} - \mathbf{N}_f\|_F^2) + \langle \mathbf{Y}, \mathbf{Z} - \mathbf{N} \mathbf{L}^T \rangle + \frac{\alpha}{2} \|\mathbf{Z} - \mathbf{N} \mathbf{L}^T\|_F^2, \end{aligned} \quad (8)$$

where we introduce a Lagrange multiplier  $\mathbf{Y}$  and a scalar  $\alpha$ . The optimization procedure of the ALM method [18] is

as follows. First, it solves subproblems of Eq. (8) for  $\mathbf{N}$ ,  $\mathbf{L}$ , and  $\mathbf{Z}$  iteratively (called inner-loop). Second, using the previous estimates of  $\mathbf{N}$ ,  $\mathbf{L}$ , and  $\mathbf{Z}$ , the values of  $\mathbf{Y}$  and  $\alpha$  are updated. Third, the inner-loop repeats with updated  $\mathbf{Y}$  and  $\alpha$  until convergence (called outer-loop). Next, let us derive the equations for the iterative optimization technique.

The sub-problems of Eq. (8) for  $\mathbf{N}$  and  $\mathbf{L}$  can be solved in closed form by taking the first order derivatives of Eq. (8) and setting them to 0. We get the solution as

$$\begin{aligned} \mathbf{N} &= ((\alpha \mathbf{Z} + \mathbf{Y}) \mathbf{L} + \lambda_2 \mathbf{N}_f) (\alpha \mathbf{L}^T \mathbf{L} + \lambda_1 \mathbf{I}_3 + \lambda_2 \mathbf{I}_3)^{-1}, \\ \mathbf{L} &= (\alpha \mathbf{Z} + \mathbf{Y})^T \mathbf{N} (\alpha \mathbf{N}^T \mathbf{N} + \lambda_1 \mathbf{I}_3)^{-1}, \end{aligned} \quad (9)$$

where  $\mathbf{I}_3$  is a  $3 \times 3$  identity matrix. This derivation is possible because the expression in Eq. (8) is quadratic in  $\mathbf{N}$  and  $\mathbf{L}$ . In our implementation, we empirically set  $\lambda_1$  to  $\frac{1}{\sqrt{p}}$  and set  $\lambda_2$  to  $10\lambda_1$ . After substituting  $\mathbf{N}$  and  $\mathbf{L}$ , we can rewrite Eq. (8) as a subproblem of  $\mathbf{Z}$  written as

$$\underset{\mathbf{Z}}{\operatorname{argmin}} \|\mathbf{W} \odot (\mathbf{O}' - \mathbf{Z})\|_1 + \frac{\alpha}{2} \|\mathbf{Z} - (\mathbf{N} \mathbf{L}^T - \frac{\mathbf{Y}}{\alpha})\|_F^2. \quad (10)$$

The solution to Eq. (10) can also be obtained in closed form and is of the following form [2]:

$$\begin{aligned} \mathbf{Z} &= \mathbf{W} \odot \left( \mathbf{O}' - \mathcal{S}_{\frac{1}{\alpha}} (\mathbf{O}' - \mathbf{N} \mathbf{L}^T + \frac{\mathbf{Y}}{\alpha}) \right) \\ &+ \bar{\mathbf{W}} \odot (\mathbf{N} \mathbf{L}^T - \frac{\mathbf{Y}}{\alpha}), \end{aligned} \quad (11)$$

where we use  $\mathcal{S}_a(b) = \max(0, b - a)$  as an element-wise shrinkage operator, and  $\bar{\mathbf{W}}$  denotes the complement of  $\mathbf{W}$ . We repeat the inner-loop, which solves Eq. (9) and Eq. (11) sequentially, until the decrease of residual error  $e$  of Eq. (8) becomes very small. We stop the iterations when  $|e_t - e_{t-1}| < 10^{-12} \times e_{t-1}$ , where  $e_t$  and  $e_{t-1}$  denote the residual errors at the  $t$ -th and  $(t-1)$ -th iterations, respectively.

After  $\mathbf{N}$ ,  $\mathbf{L}$ , and  $\mathbf{Z}$  are optimized,  $\mathbf{Y}$  is updated as

$$\mathbf{Y} = \mathbf{Y} + \alpha (\mathbf{Z} - \mathbf{N} \mathbf{L}^T), \quad (12)$$

where  $\alpha$  is reset to  $\min(1.5\alpha, 10^{20})$ . Using the value of updated  $\mathbf{Y}$  and  $\alpha$ , we repeat the inner-loop if  $\|\mathbf{O}' - \mathbf{N} \mathbf{L}^T\|_F^2 > 10^{-9} \times \|\mathbf{O}'\|_F$ . For initialization, we set  $\mathbf{N}$ ,  $\mathbf{L}$  and  $\mathbf{Z}$  to random values sampled from a zero-mean normal distribution.  $\alpha$  and  $\mathbf{Y}$  are initialized to  $10^{-3}$  and 0, respectively.

The illustration shown in Fig. 3 compares the two surface normal estimation methods discussed in this section. The first method is computationally more efficient than the second method since the dimensions of each sub-matrix is much smaller than the full dimensions of the observation matrix. In terms of accuracy, both the methods can recover accurate surface normals when the input images recorded from densely sampled viewpoints are available. On the other hand, the second method is more robust to outliers and can produce more reliable results with the fewer input images as it uses the full information at a time by exploiting the low-rank structure. An evaluation of these two methods is reported in the experiment section.

### 3.4 Shape Refinement

Another advantage of working in the parameterized 2D space is that mesh refinement can be performed by simply estimating a displacement map in the 2D domain. The shape refinement problem can be formulated as finding the optimal displacement  $d \in \mathbb{R}$  per pixel  $\mathbf{u}$  as

$$\mathbf{x}^*(\mathbf{u}) = \mathbf{x}(\mathbf{u}) + d(\mathbf{u})\mathbf{n}_f(\mathbf{u}), \quad (13)$$

where  $\mathbf{n}_f$  is a unit face normal of the triangle in  $\mathcal{M}$  to which  $\mathbf{x} \in \mathbb{R}^3$  is mapped, and  $\mathbf{x}^*$  is the refined 3D position. Notice that the geometry refinement is defined in the parameterized 2D space using  $\mathbf{u}$  as indices. Now, given photometric normals  $\mathbf{n}_p \in \mathbb{R}^3$  obtained via photometric stereo and the initial position  $\mathbf{x}$ , we estimate the displacement  $\hat{d}$  by minimizing the following energy function:

$$\hat{d} = \underset{d}{\operatorname{argmin}} \sum_{\mathbf{u} \in \mathcal{U}} \left( \mathbf{n}_p(\mathbf{u})^T \frac{\partial \mathbf{x}^*(\mathbf{u})}{\partial \mathbf{u}} \right)^2 + \lambda \sum_{\mathbf{u} \in \mathcal{U}} d^2(\mathbf{u}). \quad (14)$$

The first term of Eq. (14) is a data term that encourages the surface gradient at  $\mathbf{x}^*$  to be orthogonal to the orientation of photometric normal  $\mathbf{n}_p$ . This term is related to the one proposed by Nehab *et al.* [24]. However, we estimate only a single displacement for each 3D point, optimizing a single scalar instead of three coordinates thereby reducing mesh refinement to the optimal displacement map estimation. We use a cross-shape operator for computing partial derivatives, *i.e.*,  $[-1, 0, 1]$  for  $\frac{\partial}{\partial u}$ , and  $[1, 0, -1]^T$  for  $\frac{\partial}{\partial v}$ . To define partial derivatives at pixels on the boundary of two charts, we use their respective inverse mappings to look up neighboring 3D points on the mesh. In this manner, every boundary pixel is connected just like the interior pixels within a chart. This operation is important as it prevents seams from occurring on the chart boundaries by encouraging points across seams to have similar displacement values. The second term of Eq. (14) is a regularization term that discourages large displacements.

We optimize Eq. (14) by using an off-the-shelf sparse linear solver. Specifically, we substitute  $\mathbf{x}^*$  in Eq. (14) with Eq. (13), and take the first order derivative on the equation. By setting it to 0, we have

$$\begin{aligned} \mathbf{0} = \sum_{\mathbf{u} \in \mathcal{U}} & \left( \mathbf{n}_p(\mathbf{u})^T \frac{\partial \mathbf{x}(\mathbf{u})}{\partial \mathbf{u}} + \mathbf{n}_p(\mathbf{u})^T \frac{\partial(d(\mathbf{u})\mathbf{n}_f(\mathbf{u}))}{\partial \mathbf{u}} \right) \\ & + \lambda' \sum_{\mathbf{u} \in \mathcal{U}} d(\mathbf{u}). \end{aligned} \quad (15)$$

Here, we treat  $\mathbf{n}_p^T(\mathbf{u}) \frac{\partial \mathbf{n}_f(\mathbf{u})}{\partial \mathbf{u}}$  as a small constant  $c$  and  $\lambda'$  is defined as  $\frac{\lambda}{c}$ . Equation (15) constitutes sparse linear system of variable  $d(\mathbf{u})$ , *e.g.*,  $\mathbf{M}\mathbf{d} = \mathbf{b}$ . We use a QR decomposition-based sparse linear solver [4] for the problem.  $\lambda$  is empirically set as  $\lambda = 0.3$ .

An example of an estimated displacement map is shown in Fig. 2 (d). In our method, the level of geometric details is controlled by the resolution of  $\mathcal{U}$  regardless of the resolution of the base mesh. For example, a base mesh with as few as  $2K$  vertices with a  $512 \times 512$  displacement map can generate  $262K$  effective vertices. Since our approach directly estimates a displacement map on a coarse mesh,

our 3D models can be efficiently stored and rendered using modern graphics hardware that supports displacement mapping [31].

## 4 RECONSTRUCTION PIPELINE

This section describes our reconstruction pipeline, in particular the SfM [30] and MVS pre-processing steps.

**Stereo matching.** Using the visibility of the SfM point cloud, we estimate a depth range for each camera viewpoint and then perform plane-sweep stereo matching for each viewpoint using two other images captured from adjacent viewpoints under identical lighting. Using normalized cross correlation as the matching cost and semi-global matching based cost aggregation [12], we first estimate a dense depth map with discrete depth estimates. Sub-pixel refinement is then performed on these depth maps using a standard local parabolic refinement of the aggregated matching costs [12]. We compute per-pixel confidence associated with the depth map using the ratio of the minimum and the second smallest costs to measure distinctiveness and prune depth estimates at pixels with very low confidence. See Fig. 1 for an example of such a depth map.

**Mesh extraction.** The filtered depth maps are fused using an energy minimization framework based on volumetric graph-cuts [35]. The step computes an implicit 3D shape of a closed object by labeling voxels on a uniform 3D grid with binary labels – *occupied*, or *empty*. This optimization is formulated using a discrete binary Markov Random Field using unary and pairwise terms on a 6-connected voxel grid with a typical resolution of  $100^3$ .

The unary potentials are computed using free space occupancy of the 3D points in the depth map [10], where the contributions from depth maps are weighted by their confidences. The pairwise potentials are derived from the sub-voxel positions of these 3D points. As our acquisition setup allows simple foreground silhouette extraction, we also include a silhouette-based unary term in the energy – voxels that are projected outside the silhouette are given a high penalty for taking the label *occupied*. The optimal binary labeling can be exactly computed in an efficient manner using graph cuts [1]. Finally, from the labeled grid, we recover a triangulated mesh  $\mathcal{M}$  using marching cubes [20]. We prefer MVS in computing our base mesh over a visual-hull based approach [11], since MVS usually yields more accurate mesh in our experience, especially for objects with large concavities or complex topologies.

## 5 EXPERIMENT RESULTS

We evaluate our method using both synthetic and real-world datasets. In the synthetic data experiment, we quantitatively evaluate the two key steps of our method; surface normal estimation and shape refinement. In the experiments using real data, we show extremely detailed 3D reconstructions of various real-world objects.

### 5.1 Experiments on Synthetic Data

In this experiment, the input images are synthesized by rendering the ground truth model from varying viewpoints

under different lightings. To simulate real-world capturing, we use a perspective camera model with  $25^\circ$  field of view and the pixel intensities are normalized to fit in the range of 0 to 255.

We use publicly available BUNNY, GARGOYLE, HAPPY-BUDDHA, and SITTING-BUDDHA models that show different topologies and geometric details. Given the ground truth mesh  $\mathcal{G}$ , we measure the accuracy of the refined mesh  $\mathcal{R}$  by computing either of the *Accuracy* or the *Completeness* metrics that are used in the Middlebury multiview stereo benchmark [28]. These are based on asymmetric distances  $\text{dist}_{\mathcal{R} \rightarrow \mathcal{G}}$  and  $\text{dist}_{\mathcal{G} \rightarrow \mathcal{R}}$ , where  $\text{dist}_{A \rightarrow B}$  represents the minimum distance from vertices of  $A$  to vertices of  $B$ . *Accuracy* refers to the distance  $d \in \text{dist}_{\mathcal{R} \rightarrow \mathcal{G}}$  such that  $x\%$  of the points are within distance  $d$  to  $\mathcal{G}$ . *Completeness* refers to the proportion of vertices, where  $\text{dist}_{\mathcal{G} \rightarrow \mathcal{R}}$  is less than threshold  $\text{dist}_{th}$ . In the experiments, we set  $x = 90$  and  $\text{dist}_{th} = 0.01$ . For consistency, the 3D models are scaled by setting the radius of their tightest bounding spheres to unity. We use the screened Poisson surface reconstruction [16] for producing the final mesh that is used for quantitative evaluation.

For the purpose of assessing the accuracy of surface normal estimation and mesh refinement, we prepare the base mesh synthetically instead of computing it via SfM. The base meshes are generated by the following procedures applied to the ground truth mesh.

**Mesh perturbation.** We use the Taubin operator [32] for mesh smoothing and then add random vertex displacements as noise. The degree of noise is defined in three levels; the level 1 adds 0.05% of uniform noise, the level 2 adds 0.1% of uniform noise, and the level 3 adds 0.15% of uniform noise.

**Mesh resolution.** The numbers of triangles for the original meshes vary from  $70K$  to  $1M$ . In this procedure, we generate low-resolution meshes with approximately  $25K$ ,  $50K$ , and  $70K$  faces by a mesh simplification technique [13].

In the following, we describe the experimental results of surface normal estimation and mesh refinement.

### 5.1.1 Evaluation of Surface Normal Estimation

We first evaluate the accuracy of three surface normal estimation methods described in Sec. 3.3, *i.e.* the *Block*, *Factorization*, and *Factorization with regularization* methods. For this experiment, we use the BUNNY model with the perturbation level 3 as the degraded base mesh. The rendered images have a resolution of  $500 \times 500$  pixels.

We designed five geometric configurations of cameras and light sources to simulate different patterns of non-zero elements in the observation matrix  $\mathbf{O}$ . The five capture configurations consist of *Hemisphere*, *Ring*, and *Ring Light* with *sparse* and *dense* viewpoints as illustrated in Fig. 4(a). In this experiment, the light directions are treated unknown; thus, it is a multiview uncalibrated photometric stereo setting. For rendering, besides the Lambertian reflectance, we have also included non-Lambertian BRDFs, such as Blinn-Phong and Alum-bronze BRDFs provided by Matusik *et al.* [22] to evaluate the robustness of our algorithms in the presence of such non-Lambertian materials. The BUNNY model rendered with the different reflectances are shown in Fig. 4(b).

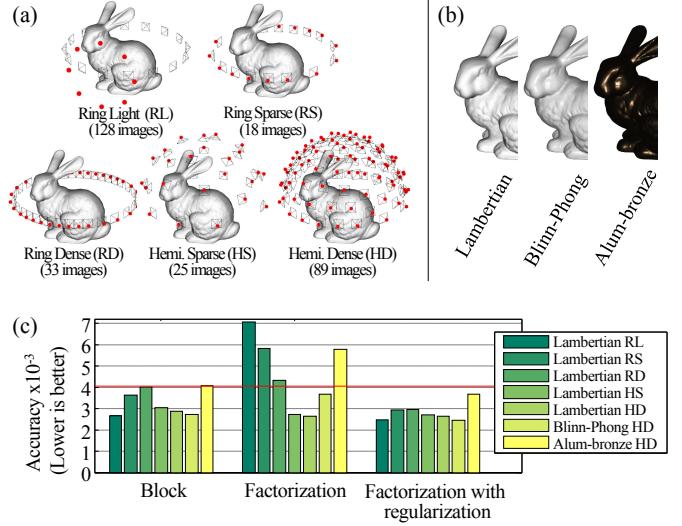


Fig. 4. Evaluation of the normal estimation algorithms on the five synthetic capture setups (shown in (a)) and the three different BRDFs (shown in (b)). A red dot in (a) indicates a point light source. In (c), each group of bar plots shows the *Accuracy* metric ( $\times 10^{-3}$ ) for the three surface normal estimation methods. The *Factorization with regularization* method is most stable and accurate in various capture configurations and BRDFs (see Sec. 5.1.1 for more details). These results were obtained with the BUNNY model with perturbation level 3 as the base mesh. The *Accuracy* metric of the base mesh is shown by the red line.

The experiment results are shown in Fig. 4(c). In the case of *Hemi. Dense* (HD) camera setup with the Lambertian reflectance, all the three normal estimation methods have similar accuracy. In the *Ring Light* (RL), *Ring Sparse* (RS), and *Ring Dense* (RD) setups, where rank-deficient matrices can occur due to co-planar light source positions, the *Factorization* method failed to refine the base mesh, whereas the *Factorization with regularization* method consistently achieved the best results. This is due to the addition of the regularization term  $\|\mathbf{N} - \mathbf{N}_f\|_F^2$ , which effectively avoids degenerate solutions. The *Block* method also achieved stable results with rank-deficient input matrices. We observe that the *Block* method can also avoid co-planarity because it utilizes the approximate surface normals  $\mathbf{N}_f$  to obtain a good local solution. Due to the avoidance of the co-planar light source positions, the results of the *Hemi. Sparse* (HS) setup outperforms the results of RD setup even though the HS setup has fewer input images.

Regarding the accuracy variation with non-Lambertian BRDFs, we have observed that the *Factorization* method is less accurate in the presence of non-Lambertian reflectances due to the large amount of outliers in the observation matrix. In contrast, the *Block* and *Factorization with regularization* methods yield more accurate results because of the guidance of approximate surface normals. The results of *Factorization with regularization* are superior to the results of *Block* since the *Factorization with regularization* method can robustly handle outliers with a  $\ell_1$  penalty function defined in Eq. (7). Close-up views of the refined BUNNY models from this experiment are shown in Fig. 5.

### 5.1.2 Evaluation of Light Direction Estimation

As a by-product of the surface normal estimation, our approach also estimates the light directions via Eq. (8).

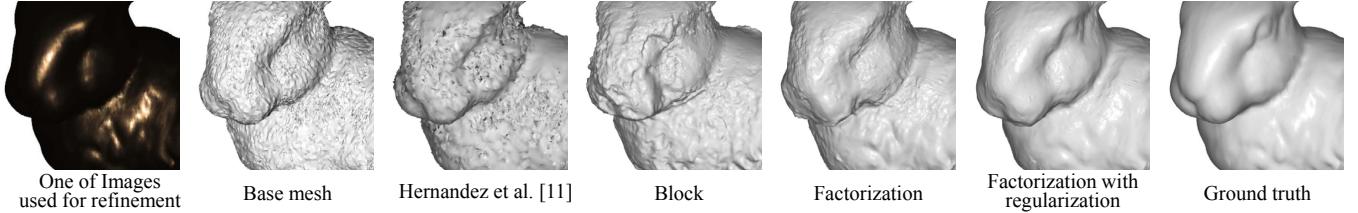


Fig. 5. Close-ups of the BUNNY model refined by the three surface normal estimation methods and Hernandez *et al.* [11]. The model is rendered with the Alum-bronze BRDF and the input images were captured in the Hemi-Dense (HD) light and camera configuration shown in Fig. 4.

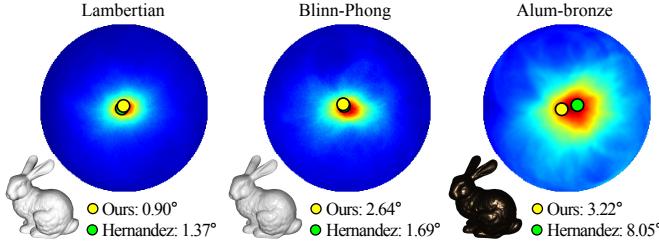


Fig. 6. Comparisons of the light directions estimated by our approach and that of Hernandez *et al.* [11]. The directional errors for the three synthetic datasets are displayed. The spherical heatmaps, which is only used by Hernandez *et al.* [11], indicate the number of inlier points as a function of light direction. The center of the circle corresponds to the ground truth light direction. The weak consensus of the heat map (particularly in Alum-bronze) indicates weak confidence on the estimated light direction because there are many probable light directions that have many inlier points. Our estimated light directions (yellow dots) and results of Hernandez *et al.* [11] (green dots) are overlaid on the heat maps for comparisons.

The method for estimating light directions proposed by Hernandez *et al.* [11] is closely related to our approach because it is also robust to outliers and utilizes Lambertian shading model.

The method described in [11] picks any three points on an object and solves for the light direction using Lambertian shading model. The synthesized image intensities using this light direction are compared with the observed image intensities, and number of inlier pixels are counted. This step is iteratively performed, and the light direction with the maximum number of inliers is selected as the final estimate.

Figure 6 compares the light directions estimated using our approach and that of Hernandez *et al.* [11]. For this experiment, we use the synthetic dataset introduced in Sec. 5.1.1. Although the surface normal of the base mesh is very noisy (as shown in Fig. 5) and some of the dataset exhibit non-Lambertian shading, both methods estimate reasonably accurate light directions with small angular error. This is not surprising because both approaches are built upon the theory of robust statistics. In case of Alum-bronze, the inlier heat map obtained in [11] shows a weak consensus which results in an erroneous light direction estimate. This also happens when surface albedo is non-uniform as discussed in [11]. Their method directly refines the geometry using a least squares method that involves estimating the light directions and requires an assumption that the surface has Lambertian reflectance properties. As shown in Fig. 5, this approach is not resilient to strong outliers and fails to refine the geometry when the Alum-bronze BRDF is used.

### 5.1.3 Evaluation of Mesh Refinement

This test focuses on comparing the quality of our refined mesh to those obtained using the state-of-the-art ap-

proaches [11], [24]. In order to alleviate other factors, which may affect the quality of the refined mesh, we allow all the methods to use the known directions of distant light sources and camera poses. For this experiment, we use the BUNNY, GARGOYLE and HAPPY-BUDDHA models. They are rendered at  $712 \times 712$  pixel resolution with 8 different light directions and 16 distinct viewpoints. Therefore, 128 images are used in each case. We use the low-resolution meshes as the base meshes.

In this experiment, we report the quality of the refined geometry by applying the *Block* method for surface normal estimation, since we found that the *Factorization with regularization* method has the same refinement quality. To simulate the real-world setting, we designed two more tests referred to as the *Mesh perturbation test* and the *Mesh resolution test* both of which utilizes the degraded base meshes generated using the *Mesh perturbation* and *Mesh resolution* steps described earlier.

Table 2 shows the quantitative result of these tests. For the *Mesh perturbation test*, our method consistently outperforms Nehab *et al.* [24]. This is because our method naturally avoids mesh flipping and overlapping triangles. In this test, the *Accuracy* and *Completeness* metrics of our results are comparable to those of Hernandez *et al.* [11] as the effective resolution of the base mesh is enough to express geometric details.

As our approach estimates a displacement map whose resolution is derived from the original image resolution, our method recovers fine geometric details regardless of the resolution of the base mesh. According to the evaluation metrics of the *Mesh resolution test* shown in Table 2, the quality of our refined mesh is independent of the resolution of the base mesh.

Figure 7 shows the cumulative error distributions of the *Accuracy* metric. The percentage of vertices within an accuracy threshold is plotted for different values of the threshold. The plot shows that our method is consistently the most accurate, except for the HAPPY-BUDDHA model where our method is comparable to [11]. The refined meshes for HAPPY-BUDDHA are shown in Fig. 8. Our method faithfully reconstructs fine details such as the necklace and the flower in the model.

### 5.1.4 Computation Time

To observe the trade-off between the mesh resolution, reconstruction accuracy, and computation time, we additionally compare our approach with Nehab *et al.* [24]. Because of the different pre-processing stages of the two methods prior to the mesh refinement, we report timings for only the mesh refinement stage and the *Accuracy* metric of the output in Fig. 9. We use the SITTING-BUDDHA model in

TABLE 2

Comparison on synthetic dataset. In this experiment, each method refines degraded meshes, and the results are evaluated using ground truth. Each cell of the table shows the *Accuracy* ( $\times 10^{-3}$ ) and *Completeness* (%) metrics for two experiments, *mesh perturbation test* and *mesh resolution test* (see Sec. 5.1.3 for more details).

Mesh Perturbation	BUNNY			GARGOYLE			HAPPY-BUDDHA		
	Level 1	Level 2	Level 3	Level 1	Level 2	Level 3	Level 1	Level 2	Level 3
Nehab <i>et al.</i> [24]	2.87, 99.7	2.93, 99.7	6.31, 93.6	4.72, 99.9	4.56, 99.9	4.58, 99.9	4.01, <b>99.8</b>	3.95, 99.8	4.06, 99.8
Hernandez <i>et al.</i> [11]	1.66, 99.7	2.30, 99.7	5.51, 92.4	<b>3.03</b> , 100.0	<b>3.40</b> , <b>100.0</b>	4.15, 100.0	<b>2.92</b> , 99.8	<b>3.42</b> , <b>99.9</b>	<b>3.04</b> , <b>99.9</b>
Ours	<b>1.50</b> , <b>100.0</b>	<b>1.94</b> , <b>100.0</b>	<b>2.67</b> , <b>100.0</b>	3.43, <b>100.0</b>	3.47, 100.0	<b>3.70</b> , <b>100.0</b>	3.50, 99.5	3.65, 99.5	3.50, 99.5
Mesh Resolution	70K	50K	25K	70K	50K	25K	70K	50K	25K
Nehab <i>et al.</i> [24]	3.56, 99.9	5.44, 94.4	7.54, 67.8	6.02, 98.5	8.38, 83.6	11.56, 49.3	4.82, 99.2	6.51, 91.2	8.37, 64.8
Hernandez <i>et al.</i> [11]	<b>1.11</b> , 99.8	1.43, 96.7	1.67, 76.0	3.64, 96.7	4.14, 89.8	4.88, 64.5	<b>2.76</b> , 98.3	<b>3.43</b> , 93.4	4.29, 72.1
Ours	1.39, <b>100.0</b>	<b>1.40</b> , <b>100.0</b>	<b>1.41</b> , <b>100.0</b>	<b>3.33</b> , <b>100.0</b>	<b>3.37</b> , <b>100.0</b>	<b>3.45</b> , <b>99.9</b>	3.45, <b>99.6</b>	3.48, <b>99.5</b>	<b>3.49</b> , <b>99.5</b>

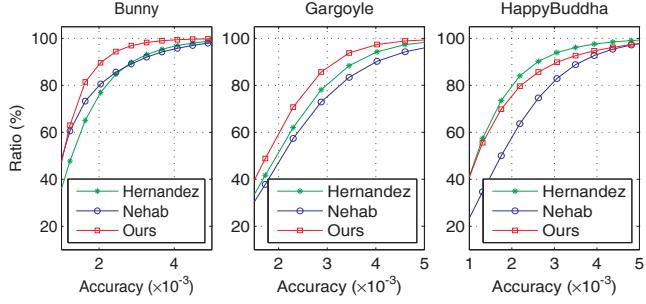


Fig. 7. Cumulative error distributions of the *Accuracy* metric for three synthetic dataset; BUNNY, GARGOYLE, and HAPPY-BUDDHA. The graph corresponds to the *mesh perturbation test* in Table 2 when perturbation level is 3. On BUNNY and GARGOYLE, our method is consistently the most accurate. On HAPPY-BUDDHA, the accuracy is very similar to Hernandez *et al.* [11].

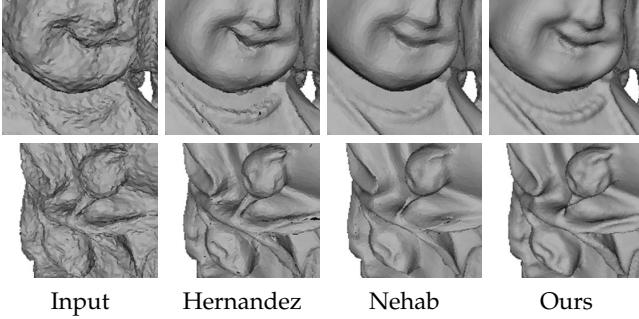


Fig. 8. Perturbed HAPPY-BUDDHA model (denoted as input) and refinement results by three methods; Hernandez [11], Nehab [24], and ours. The result corresponds to *Mesh perturbation test* in Table 2 where perturbation level is 3.

this experiment to observe the refinement of fine detailed regions. The base mesh has only 50K faces, which is not enough to convey the full geometric details of the original model. To recover the fine details, Nehab *et al.* [24] requires very high mesh resolution (1.8M faces) whereas our method uses a much coarser base mesh. As a result, our mesh refinement step is computationally more efficient than that of Nehab *et al.* [24] when both the methods are configured to produce the results with the comparable accuracy. We report the timings of Nehab *et al.* [24] method by using the implementation provided by the authors.

On average, the overall computational time of our method (excluding data acquisition and pre-processing)

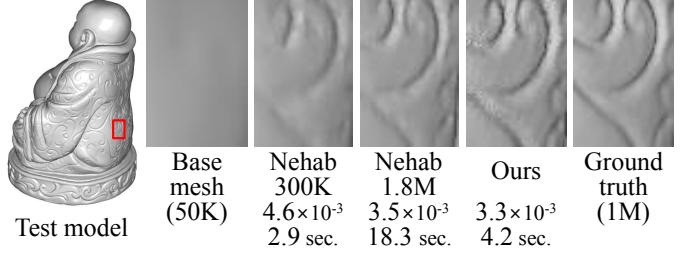


Fig. 9. Comparison with Nehab *et al.* [24] on the computational cost and accuracy of the refined mesh. Mesh resolutions, accuracies, and computation times are shown under the sub-figures. Our method does not require tuning the mesh resolution because it is automatically determined by the input image resolution.

takes less than a minute to run if we utilize the *Block* method for the surface normal estimation. If we use the *Factorization with regularization* method, the running time become 30 minutes on average. Our methods are implemented in C++ without optimization. All the reported timings are measured on a system equipped with an Intel i7 quad-core 3.06GHz CPU and 8GB memory.

## 5.2 Experiments on Real-world Data

In this section, we show the real-world experiments. We first describe two imaging setups for real-world 3D reconstruction, and then discuss the quality of the results. We calibrate the camera intrinsics and the camera response function a priori and assume that they remain unchanged during the data acquisition. The extrinsic parameters are estimated using a generic SfM pipeline [30], and light directions are also estimated via our method.

### 5.2.1 Two Flexible Acquisition Setups

Since the proposed method does not require geometric calibration of the light source and the camera, we can have a flexible imaging setup with various camera and light configurations. In this work, we demonstrate two practical setups, one automated and one manual.

**Automated imaging setup.** Our automated acquisition setup sequentially captures images using programmed devices. The system consists of a rotation stage, a LED array, and two cameras as illustrated in Fig. 10 (a). All images of the target object are captured automatically using a remotely controlled rotation stage with synchronized cameras

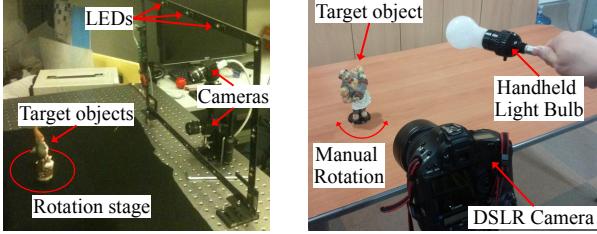


Fig. 10. Two imaging setups for real world object capture. (left) Automated setup. It consists of a rotation stage, a light array and two cameras. (right) Manual setup. It only requires a camera and a light bulb. In both setups, several images are captured under varying lighting for a particular camera orientation.

and LEDs. A typical acquisition captures 312 images (24 viewpoints, 15 degrees apart illuminated by 13 different LEDs) and takes about three minutes. The image resolution is  $1024 \times 768$  pixels. In this setup, instead of the generic SfM approach [30], camera pose estimation methods tailored to turn-tables [5] can also be used.

**Manual imaging setup.** In this setup, the target object, the camera and the light source are moved manually. We use an incandescent light bulb and Canon Mark-1D DSLR camera, and the target scene is recorded in a dim environment with low ambient light. Specifically, we record the target object from a fixed viewpoint multiple times under varying lightings, and rotate the object manually and repeat the same for acquiring the input data. The image resolution is  $3908 \times 2600$ , and in this test, we recorded images of seven object poses under six light source directions.

Capturing multiple images by varying the light position for each viewpoint is preferable way to capture the dataset if there are at least a few viewpoint variations covering the target object. It makes surface normal estimation more accurate because any observable 3D points on the surface have intensity profiles with perfect pixel alignment for certain viewpoint. As an example, 3 viewpoints and 8 light variations for each viewpoint ( $3V \times 8L$ ) shown in Fig. 11 indicates lower intensity reconstruction error and better refined geometry compared to  $24V \times 1L$  or  $8V \times 3L$ .

### 5.2.2 Reconstruction Results

Figure 17 shows the result of five real scenes whose images are captured using the automated imaging setup. Since the two normal estimation methods (the *Block* and *Factorization with regularization* methods) show very similar results, especially for objects with Lambertian reflectance, and the second method consistently outperforms the first method in terms of accuracy and robustness, we present the results from the *Block* method as the lower bound accuracy of our reconstruction method. Also, from our observation, the *Block* method consistently works well when the viewpoint sampling rate is high enough, which is the case in our automated setup. The first two objects shown in Fig. 17, BUDDHA-STATUE and AGRIPPA, have mostly uniform albedos. However, BUDDHA-STATUE is made of copper and has many specular reflections. Even though our *Block* method assumes Lambertian reflectance, the normal aggregation method described in Eq. (4) effectively handles outliers arising from non-Lambertian reflectances.

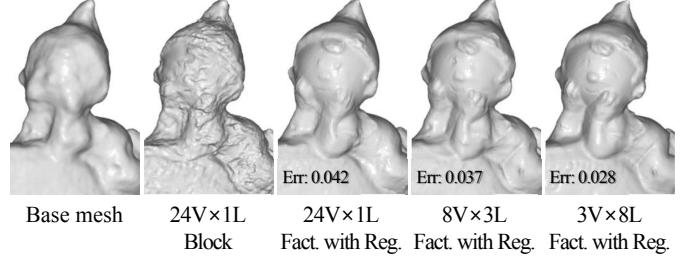


Fig. 11. Reconstruction of the DOLL-2 dataset with different surface normal estimation methods and imaging scenarios. We only use 24 images out of the 312 images in this experiment.  $V$  indicates the number of viewpoints and  $L$  is the number of light variations for each viewpoint. Every 24 image set has viewpoint variations covering the target object. The average intensity reconstruction error  $\|O - O_{low}\|_1$  is shown on the sub-figures.



Fig. 12. Two color images from the ACCORDION MAN dataset.

The other three objects in Fig. 17, DOLL-1, DOLL-2, and TEAPOT, have more interesting topologies and multiple albedos. In DOLL-1, we can observe the detailed shape of buttons on the jacket of the right doll as well as facial expression of the dolls, which cannot be seen in the original base mesh. The English characters in the middle region of DOLL-2 are clearly visible in the final mesh. The geometric details on the TEAPOT model are faithfully reconstructed. Note that these embossed patterns are only a few millimeters deep. On the other hand, an artifact can be seen below the left doll's skirt in DOLL-1 as indicated by red rectangles in Fig. 17. Since no valid normal could be estimated from any of the viewpoints, our method is unable to refine the coarse mesh in this region.

Figure 13 shows the reconstruction result of ACCORDION MAN which is captured using the manual setup. For reference, an example RGB image of the ACCORDION MAN sequence is shown in Fig. 13. The height of the object is approximately 22cm. As shown in Fig. 12, the ACCORDION MAN model has rich surface albedo and a complex 3D shape. Due to the high-resolution of the input images, the geometry of the base mesh is quite reliable. From this base mesh, the *Block* method is applied and fine geometric details are recovered in the refined mesh. The geometric features on the face and the hair of the model can be clearly seen in the refined 3D model. It is worth noting that photometric cues are able to add sub-millimeter level of geometric details which is challenging for multiview stereo algorithms.

To study the effect of the observation matrix structure on the two surface normal estimation methods, we experimented with the DOLL-2 sequence and uniformly reduced the number of input images to simulate the case where only one light source is used for each viewpoint. The results are shown in Fig. 11. Compared to the *Block* method, the

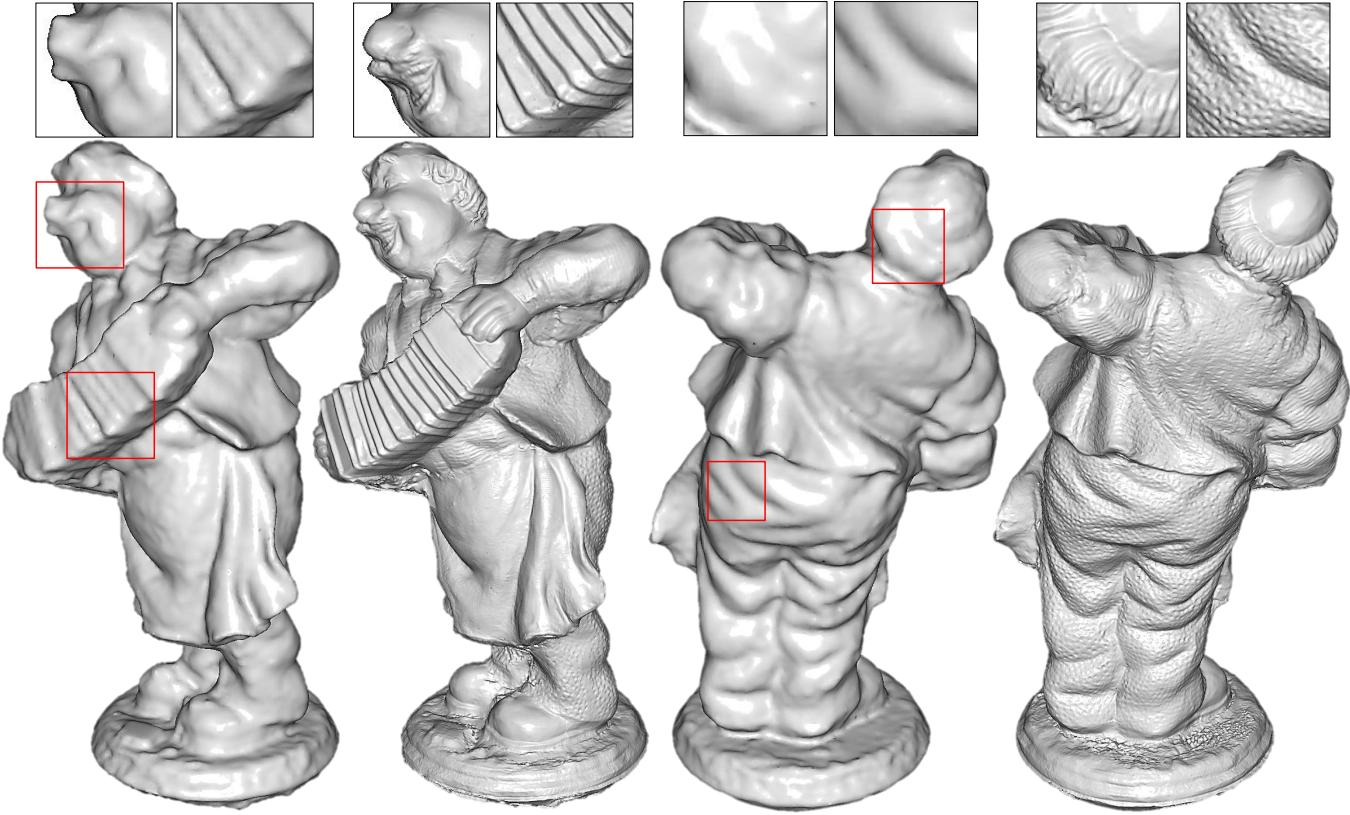


Fig. 13. Reconstruction of ACCORDION MAN. Comparison between the base mesh (left) and refined mesh (right) rendered from two different viewpoints. The geometric details on the face, hair and bumpy texture on the pants as shown in the insets on the top. Note that facial expression, hair geometry and bumpy texture on the pants appears on the refined geometry which has sub-millimeter level accuracy. The input images for this result were captured with the manual imaging setup shown in Fig. 10(b).

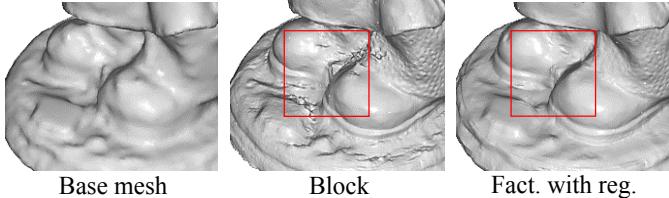


Fig. 14. Comparison of the reconstruction quality using the *Block* and *Factorization with regularization* methods. The *Block* method is more susceptible to outliers but reconstructed details are slightly sharper.

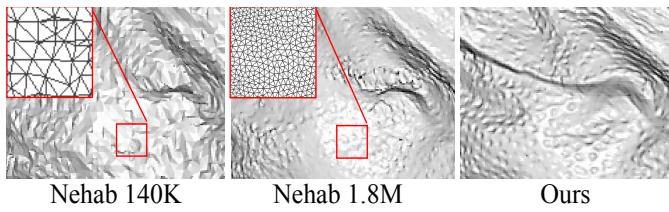


Fig. 15. Zoom-in views of the ACCORDION MAN models obtained using the method of [24] and our method. Our models encode the geometric details using displacement maps ( $1000^2$  pixels in this case without requiring high resolution mesh.).

performance of the *Factorization with regularization* method degrades gracefully when the pattern of the observation matrix is highly unstructured.

In Fig. 14, we show the qualitative comparisons of two ACCORDION MAN models reconstructed using the estimated normals from the *Block* and *Factorization with regularization* methods, respectively. The two approaches generate similar results although there are subtle differences. The

result from the *Block* method shows sharper details than the result from the *Factorization with regularization* method. However, the *Block* method results in a few artifacts in the refined geometry whereas the result from the *Factorization with regularization* method is more stable and robust to outliers. Looking at the result of the *Block* method shown in Fig. 14, where the observed pixels are quite dark, we can see uneven geometry with the *Block* method, while the *Factorization with regularization* method consistently avoids these problems.

Figure 15 shows the 3D meshes obtained using our approach and those generated by the method of Nehab *et al.* [24] at two different resolutions. The micro-geometry on the shoes is only captured in our result due to the effectiveness of the high resolution displacement maps. Here, we used the same light directions for both methods.

### 5.2.3 Metric Error Analysis For Different Configurations

Figure 16 shows 3D reconstructions of the CICERO dataset, where the images were captured using the manual imaging setup. We used a structured light based 3D object scanner [7] to acquire the ground truth geometry in metric units. Since the 3D model reconstructed by our method has a scale ambiguity, we rescale the reconstructed model using Geomagic studio<sup>TM</sup> software to align it with the ground truth.

Various configurations of our approach are evaluated in this experiment as shown in Fig. 16. When the MVS geometry is utilized as the initial geometry, *Factorization with regularization* methods outperform the *Block* methods although the latter group produces models with sharper

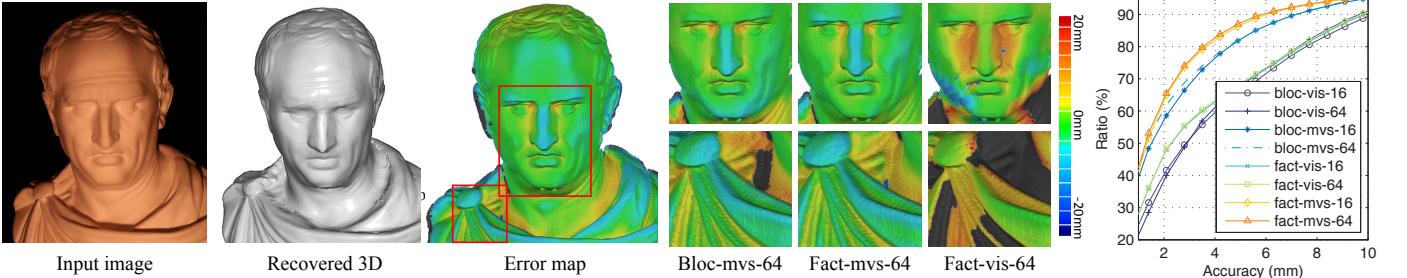


Fig. 16. Reconstruction results obtained using our method for the CICERO dataset. The metric errors of the reconstructed 3D models are shown. In this experiment, we measure accuracy by running our method with different configurations. On the right side, the error plot displays these configurations. For example, bloc-vis-16 indicates the block based method with visual hull as the initial geometry and 16 input images. The black color in the error map indicates that the errors are out of the color map range.

geometric details. When comparing the different number of images (16 or 64 images), we observe that using more images generally helps to achieve better accuracy. In another experiment we used a visual hull reconstruction for the initial geometry instead of the model obtained using MVS. In this case, highly concave regions such as nose or shoulder of the CICERO model shows significant errors because our approach cannot handle large errors in the initial geometry. On average, the metric error is 3.0017 mm in case of fact-mvs-64. The overall error maps from this experiment can be found in the supplementary material.

## 6 DISCUSSIONS

Our 3D reconstruction method enables acquisition of high-fidelity 3D models by fusing photometric and geometric cues. The key benefit of planar mesh parameterization is the maximum usage of photometric cues by avoiding both oversampling and undersampling problems.

We have proposed two approaches for reliable surface normal estimation. The first approach, *Block* method, has a merit in its computational efficiency, and it is applicable when the observation matrix is well-structured. The other approach, *Factorization with regularization* method, outputs reliable results regardless of the observation matrix structures at a cost of more computation. As a general guideline, when the number of observations is large and the target scene has a near Lambertian reflectance, we can rely on the *Block* method. Otherwise, it is recommended to use *Factorization with regularization* method for obtaining accurate results.

Our algorithm assumes that the initial geometry is reasonably accurate and the misalignment between the overlapping images is not severe on the smooth surfaces. In the future, we plan to explore a joint optimization approach that simultaneously estimates surface normals and scene depth for greater accuracy and robustness. Recovering surface reflectance in addition to accurate 3D shape is another interesting direction for future work.

## ACKNOWLEDGMENTS

This work was partly supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2010-0028680) and JSPS KAKENHI Grant Number JP16H01732.

## REFERENCES

- [1] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in computer vision. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, 26(9):1124–1137, 2004.
- [2] R. Cabral, F. D. la Torre, J. P. Costeira, and A. Bernardino. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In *Proc. of Int'l Conf. on Computer Vision (ICCV)*, 2013.
- [3] G. Choe, J. Park, Y.-W. Tai, and I. S. Kweon. Exploiting shading cues in kinect IR images for geometry refinement. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [4] T. A. Davis. Algorithm 915, suitesparseQR: Multifrontal multi-threaded rank-revealing sparse QR factorization. *ACM Transactions on Mathematical Software*, 38(1):8:1–8:22, 2011.
- [5] A. W. Fitzgibbon, G. Cross, and A. Zisserman. Automatic 3D model construction for turn-table sequences. In *3D Structure from Multiple Images of Large-Scale Environments*, Lecture Notes in Computer Science, pages 155–170. Springer-Verlag, 1998.
- [6] U. I. Gupta, D. T. Lee, and J. Y.-T. Leung. Efficient algorithms for interval graphs and circular-arc graphs. *Networks*, 12(4):459–467, 1982.
- [7] H. Ha, J. Park, and I. S. Kweon. Dense depth and albedo from a single-shot structured light. In *International Conference on 3D Vision (3DV)*. IEEE, 2015.
- [8] Y. Han, J.-Y. Lee, and I. S. Kweon. High quality shape from a single rgbd image under uncalibrated natural illumination. In *Proc. of Int'l Conf. on Computer Vision (ICCV)*, 2013.
- [9] H. Hayakawa. Photometric stereo under a light source with arbitrary motion. *Journal of the Optical Society of America (JOSA)*, 11(11):3079–3089, 1994.
- [10] C. Hernández, G. Vogiatzis, and R. Cipolla. Probabilistic visibility for multi-view stereo. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.
- [11] C. Hernández, G. Vogiatzis, and R. Cipolla. Multi-view photometric stereo. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, 30(3):548–554, 2008.
- [12] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, 30(2):328–341, 2008.
- [13] H. Hoppe. New quadric metric for simplifying meshes with appearance attributes. In *Proc. of IEEE Visualization Conference*, pages 59–66, 1999.
- [14] B. K. P. Horn. Shape from shading: A method for obtaining the shape of a smooth opaque object from one view. *PhD thesis, MIT*, 1970.
- [15] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *ACM Symposium on User Interface Software and Technology*, 2011.
- [16] M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *ACM Trans. on Graph. (ToG)*, 32(3):29:1–29:13, 2013.
- [17] H. P. A. Lensch, J. Kautz, M. Goesele, W. Heidrich, and H. Peter Seidel. Image-based reconstruction of spatial appearance and geometric detail. *ACM Trans. on Graph. (ToG)*, 22:234–257, 2003.
- [18] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. In *Mathematical Programming*, 2010.

- [19] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 193:133–135, 1981.
- [20] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proc. of SIGGRAPH*, pages 163–169, 1987.
- [21] Z. Lu, Y.-W. Tai, M. Ben-Ezra, and M. S. Brown. A framework for ultra high resolution 3d imaging. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [22] W. Matusik, H. Pfister, M. Brand, and L. McMillan. A data-driven reflectance model. *ACM Trans. on Graph. (ToG)*, 22(3):759–769, 2003.
- [23] D. M. Mount and S. Arya. ANN: A library for approximate nearest neighbor searching. <http://www.cs.umd.edu/~mount/ANN/>, 2010.
- [24] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi. Efficiently combining positions and normals for precise 3D geometry. *ACM Trans. on Graph. (ToG)*, 24(3):536–543, 2005.
- [25] T.-H. Oh, H. Kim, Y.-W. Tai, J.-C. Bazin, and I. S. Kweon. Partial sum minimization of singular values in rpca for low-level vision. In *Proc. of Int'l Conf. on Computer Vision (ICCV)*, 2013.
- [26] T. Okatani and K. Deguchi. Optimal integration of photometric and geometric surface measurements using inaccurate reflectance/illumination knowledge. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 254–261, 2012.
- [27] J. Park, S. N. Sinha, Y. Matsushita, Y.-W. Tai, and I. S. Kweon. Multiview photometric stereo using planar mesh parameterization. In *Proc. of Int'l Conf. on Computer Vision (ICCV)*, 2013.
- [28] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 519–528, 2006.
- [29] A. Sheffer, E. Praun, and K. Rose. Mesh parameterization methods and their applications. *Foundations and Trends in Computer Graphics and Vision*, 2(2):105–171, 2006.
- [30] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. In *Proc. of SIGGRAPH*, pages 835–846. ACM Press, 2006.
- [31] L. Szirmay-Kalos and T. Umenhoffer. Displacement mapping on the GPU - State of the Art. *Computer Graphics Forum*, 27(1), 2008.
- [32] G. Taubin. A signal processing approach to fair surface design. In *Proc. of SIGGRAPH*, pages 351–358, 1995.
- [33] A. Ungar. Barycentric calculus in euclidean and hyperbolic geometry: A comparative introduction. World Scientific, 2010.
- [34] D. Vlasic, P. Peers, I. Baran, P. Debevec, J. Popović, S. Rusinkiewicz, and W. Matusik. Dynamic shape capture using multi-view photometric stereo. *ACM Trans. on Graph. (ToG)*, 28(5), 2009.
- [35] G. Vogiatzis, C. Hernández Esteban, P. H. S. Torr, and R. Cipolla. Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, 29(12):2241–2246, Dec. 2007.
- [36] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980.
- [37] C. Wu, Y. Liu, Q. Dai, and B. Wilburn. Fusing multiview and photometric stereo for 3d reconstruction under uncalibrated illumination. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 17(8):1082–1095, 2011.
- [38] C. Wu, B. Wilburn, Y. Matsushita, and C. Theobalt. High-quality shape from multi-view stereo and shading under general illumination. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 969–976, 2011.
- [39] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Robust photometric stereo via low-rank matrix completion and recovery. In *Proc. of Asian Conf. on Computer Vision (ACCV)*, pages 703–717, 2010.
- [40] L.-F. Yu, S.-K. Yeung, Y.-W. Tai, and S. Lin. Shading-based shape refinement of rgbd images. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [41] 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 Int'l Conf. on Computer Vision (ICCV)*, pages 618–625, 2003.
- [42] Q. Zhang, M. Ye, R. Yang, Y. Matsushita, B. Wilburn, and H. Yu. Edge-preserving photometric stereo via depth fusion. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 2472–2479, 2012.
- [43] K. Zhou, J. Snyder, B. Guo, and H.-Y. Shum. Iso-charts: Stretch-driven mesh parameterization using spectral analysis. In *Eurographics Symposium on Geometry Processing*, pages 45–54, 2004.
- [44] Z. Zhou, Z. Wu, and P. Tan. Multi-view photometric stereo with spatially varying isotropic materials. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 1482–1489, 2013.



**Jaesik Park** received his Bachelor degree (Summa cum laude) in media communication engineering from Hanyang University in 2009. He received his Master degree and Ph.D. degree in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST) in 2011 and 2015, respectively. He joined Intel Labs as a research scientist in 2015. His research interests include depth map refinement, image-based 3D reconstruction. He is a member of the IEEE.



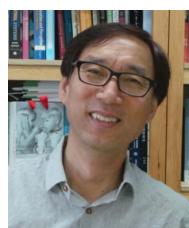
**Sudipta N. Sinha** received his B.Tech. degree in Computer Science and Engineering from IIT Kanpur in 2000 and his M.S. and Ph.D. degrees in Computer Science from the University of North Carolina at Chapel Hill in 2005 and 2009 respectively. He is a researcher in the Adaptive Systems and Interaction group in Microsoft Research. His research interests lie in image-based modeling and rendering, structure from motion, stereo matching and real-time vision-based localization. He is a member of the IEEE.



**Yasuyuki Matsushita** received his B.S., M.S. and Ph.D. degrees in EECS from the University of Tokyo in 1998, 2000, and 2003, respectively. From April 2003 to March 2015, he was with Visual Computing group at Microsoft Research Asia. In April 2015, he joined Osaka University as a professor. His research area includes computer vision, machine learning and optimization. He is on the editorial board of IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), International Journal of Computer Vision (IJCV), IPSJ Journal of Computer Vision and Applications (CVA), The Visual Computer Journal, and Encyclopedia of Computer Vision. He served/is serving as a Program Co-Chair of PSIVT 2010, 3DIMPVT 2011, ACCV 2012, ICCV 2017, and a General Co-Chair for ACCV 2014. He is a senior member of IEEE.



**Yu-Wing Tai** is a principle research scientist of SenseTime Group Limited. He was an associate professor working at the Korea Advanced Institute of Science and Technology (KAIST) from July 2009 to August 2015. He received his Ph.D. degree in Computer Science from the National University of Singapore (NUS) in 2009. From Sept 2007 to June 2008, he worked as a full-time student internship in the Microsoft Research Asia (MSRA). He was awarded the Microsoft Research Asia Fellowship in 2007, and the KAIST 40th Anniversary Academic Award for Excellent Professor in 2011 respectively. He received a M.Phil and B.Eng (First Class Honors) degree in Computer Science from the Hong Kong University of Science and Technology (HKUST) in 2005 and 2003 respectively. He is a senior member of the IEEE.



**In So Kweon** received the BS and MS degrees in mechanical design and production engineering from Seoul National University in 1981 and 1983, respectively, and the Ph.D. degree in robotics from the Robotics Institute, Carnegie Mellon University, in 1990. He joined the Department of Automation and Design Engineering, KAIST in 1992, where he is now a professor with the Department of Electrical Engineering. He is a recipient of the best student paper runner-up award at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09). His research interests are in camera and 3D sensor fusion, color modeling and analysis, visual tracking, and visual SLAM. He was the program co-chair for the Asian Conference on Computer Vision (ACCV '07) and was the general chair for the ACCV '12. He is also on the editorial board of the International Journal of Computer Vision. He is a member of the IEEE and the KROS.

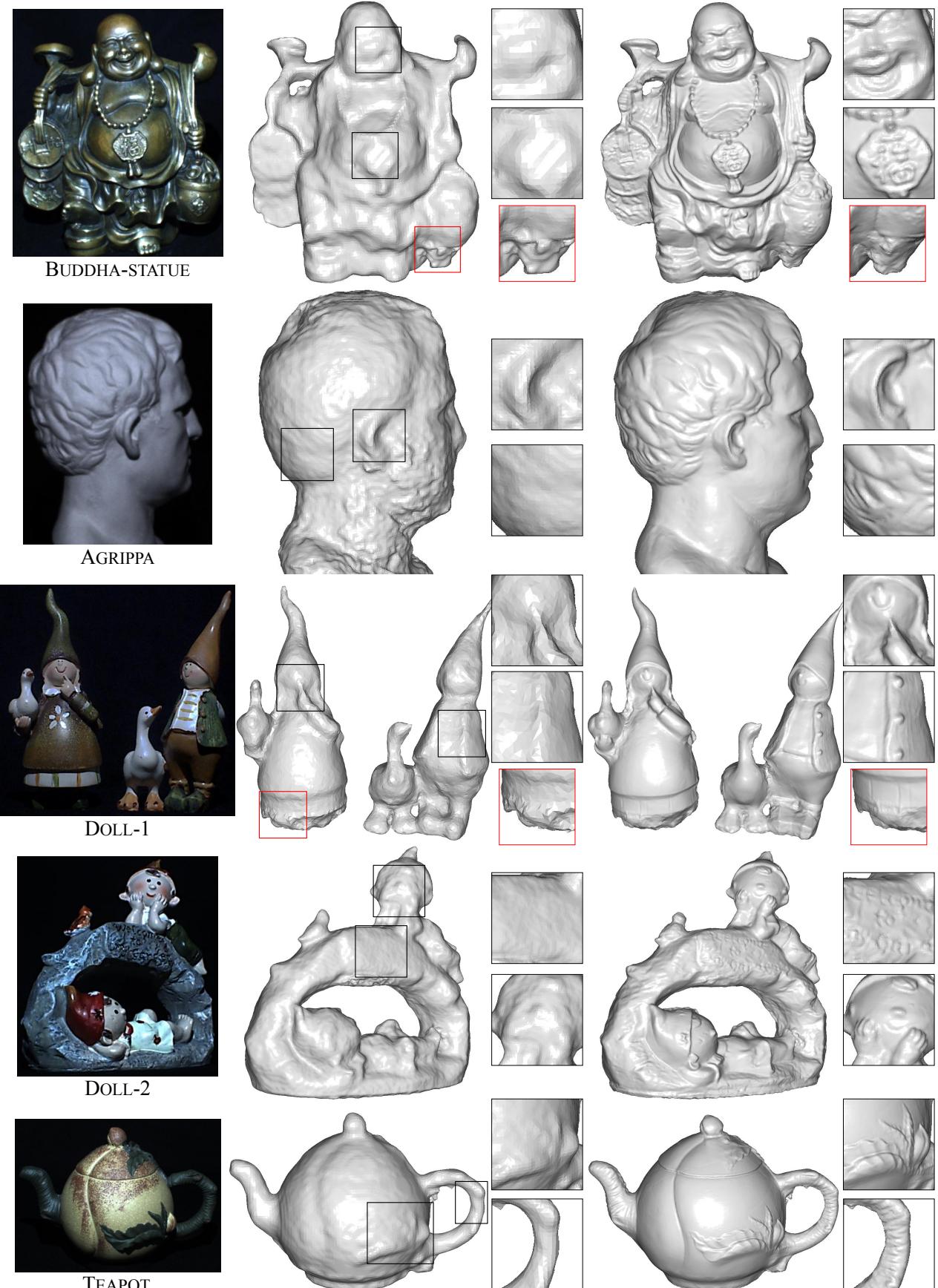


Fig. 17. Reconstructions obtained using our method for BUDDHA-STATUE, AGRIPPA, DOLL-1, DOLL-2, and TEAPOT scenes. Each row shows from left to right one of the input images, the base mesh from MVS, and the refined mesh. The corresponding surface normal and displacement maps are shown in the supplementary material. Areas where our method performed poorly are highlighted by red rectangles; these occur at texture-less dark regions. The input images for these results are captured with the automated imaging setup as shown in Fig. 10(a).