Topology-adaptive Multi-view Photometric Stereo

Yusuke Yoshiyasu
Graduate School of Science and Technology
Keio University, Japan
yusuke_2_ax_es@z6.keio.jp

Nobutoshi Yamazaki
Department of Mechanical Engineering
Keio University, Japan
yamazaki@mech.keio.ac.jp

Abstract

In this paper, we present a novel technique that enables capturing of detailed 3D models from flash photographs integrating shading and silhouette cues. Our main contribution is an optimization framework which not only captures subtle surface details but also handles changes in topology. To incorporate normals estimated from shading, we employ a mesh-based deformable model using deformation gradient. This method is capable of manipulating precise geometry and, in fact, it outperforms previous methods in terms of both accuracy and efficiency. To adapt the topology of the mesh, we convert the mesh into an implicit surface representation and then back to a mesh representation. This simple procedure removes self-intersecting regions of the mesh and solves the topology problem effectively. In addition to the algorithm, we introduce a hand-held setup to achieve multi-view photometric stereo. The key idea is to acquire flash photographs from a wide range of positions in order to obtain a sufficient lighting variation even with a standard flash unit attached to the camera. Experimental results showed that our method can capture detailed shapes of various objects and cope with topology changes well.

1. Introduction

Image-based modeling methods have received attention because of their high-quality results without the need to use specialized hardware. Multi-view stereo is a well-studied technique and its state-of-the-art methods [11, 12] require only texture cues for reconstruction. Multi-view photometric stereo [3, 14], on the other hand, can recover highly-detailed models of texture-less objects exploiting shading and silhouette cues. One of the difficulties with using this method is that it requires a good initial approximation to the object surface in a geometrical and topological sense. In fact, doing so is not always trivial as it demands a visual hull computed from high-quality silhouettes, which can easily be violated by self shadows, etc. Also, the lighting setup is somewhat cumbersome, i.e., the lighting position must be adjusted in every capture and the experimental room must be dark.

In this study, we attempt to make the multi-view photometric stereo technique more practical. Our method accepts a rough initial shape whose geometry and topology is far from that of the final shape, and deforms this to match with shading and silhouettes in the images. Our optimization framework is a hybrid deformation approach which combines the explicit and implicit surface. Our key insight is to use the explicit surface (mesh deformation) to capture geometry from images and let the implicit method concentrate on topology adaptation. This combination allows us to efficiently capture surface details while also handling topological changes. In addition, we introduce a simple setup to achieve multi-view photometric stereo which uses only a digital camera and a flash unit (no control of lighting positions and ambient lighting).

The remainder of this paper is organized as follows: In Section 2, we briefly review related work. We overview our method in Section 3. Our acquisition setup is described in Section 4. In Sections 5 and 6, we present our optimization framework. We then show experimental results in Section 7. Finally, we conclude our work in Section 8.

2. Related Work

Multi-view stereo Multi-view stereo reconstructs 3D models using texture cues. While the problem we solve differs slightly from it, it is worth referring to the classification of multi-view stereo here as a guide for choosing our shape representation. Following Seitz et al. [24], multi-view stereo algorithms can be classified into four categories: voxel-based, mesh-based, depth map based and patch-based approaches. Voxel-based approaches [9, 27] typically start from a bounding box and accommodate geometry and topology changes. However, they are computationally expensive and the result is not exact because of spatial discretization. In contrast, mesh-based approaches [13], in general, must have a good initial approximation to the object surface (same topology and similar geometry) but, when it is given, they are rather fast and produce high-

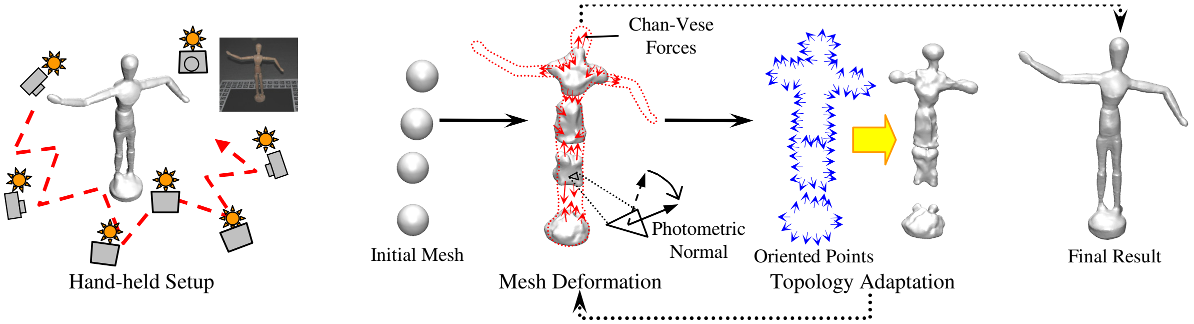

Figure 1. Overview of our topology-adaptive multi-view photometric stereo.

quality results. Depth map based [12] and patch-based [11] approaches are flexible in that they obtain points using a classical binocular stereo technique and do not necessarily require silhouettes. The results for all the viewpoints are then merged into a full model with the surface reconstruction software [19]. Although recent techniques achieve high-quality results, they have difficulty dealing with texture-less objects.

Binocular stereo and photometric stereo fusion Stereo and photometric stereo fusion approaches produce good results for both textured and texture-less objects [15, 20]. This class of methods sheds light on the object and captures video footage. From the video streams, these methods sequentially or simultaneously optimize surface normals and depths incorporating shading and texture cues. The technique of Higo et al. [15] is closely related to our method in terms of the problem solved and the setup used. However, it is difficult to recover a full 3D model using their method because of their relatively time-consuming optimization procedure which requires a large set of input images. Our method can achieve a full model reconstruction.

Multi-view photometric stereo Multi-view photometric stereo reconstructs detailed full 3D models integrating shape-from-silhouettes and photometric stereo [3, 14]. Hernandez et al. [14] acquired detailed full 3D models of figurines by refining a visual hull to agree with photometric normals using a mesh deformation method. They captured multiple images with a turn-table setup using a fixed single camera and changed lighting positions manually for every capture. Ahmed et al. [1] and Vlasic et al. [26] developed systems for capturing the time-varying geometry of moving objects using video cameras and calibrated lights. Although the result of multi-view photometric stereo is impressive, current methods demand high-quality silhouettes and precise control of lighting.

2.1. Contributions

We go a step further to maximize the potential of multi-view photometric stereo [14] by improving the optimization technique and setup used. The proposed optimization technique, which is our main contribution, is closely related to the ones developed in [6, 14, 17, 21]. However, our method can not only handle topological changes but also control surface details precisely and efficiently. In addition, we propose a simple setup for *multi-view* photometric stereo introducing an effective way of obtaining lighting variations. Our setup is similar to those proposed in Higo et al. [15] and Paterson et al. [22] in the sense that we use a hand-held camera. However, we use a standard flash-equipped camera and capture photographs from a wide range of position in order to obtain sufficient lighting variations. Our method therefore requires a smaller set of input images than [15] allowing *multi-view* reconstruction within a reasonable time.

3. Method Overview

Overview of our method is depicted in Fig.1. Our method performs multi-view photometric stereo with a hand-held setup (Section 4). The input of our algorithm is flash photographs captured from a wide range of viewpoints. Starting from a rough initial shape which may be very different from the final one, we recover detailed geometry by integrating silhouettes and shading. We fuse these cues by employing mesh deformation based on the deformation gradient (Section 5). To solve the topology problem, we propose a simple but effective idea (Section 6). We first compute oriented points for all the triangles in the mesh. We then convert these oriented points into an implicit surface. Finally, we convert this back to a mesh representation. This way, we can solve the topology problem easily. We alternate mesh deformation and topology adaptation and the final result is a detailed full 3D model.

4. Setup and Acquisition

We use a standard digital camera with a flash unit. To calibrate the camera, we use a calibration board with self-identifying markers [2] attached on it.

Our method moves the camera widely around the object acquiring L photographs using a flash. Although the distance between the camera and the flash is short, this provides sufficient variations of illumination to estimate accurate normals from shading.

The pose of a camera in l-th viewpoint is represented as a 3×3 rotation matrix $\mathbf{r}_l$ and a translation $\mathbf{o}_l \in \mathbb{R}^3$. These extrinsic camera parameters as well as the intrinsic camera parameters are computed from the camera calibration. Because the flash is fixed to the camera, the light position can also be obtained with the camera calibration. Let $\mathbf{b}$ be the position of the flash in the camera coordinate, then the light position of l-th viewpoint in the global coordinate $\mathbf{x}_l$ is obtained by:

$$\mathbf{x}_l = \mathbf{r}_l^{-1}(\mathbf{b} - \mathbf{o}_l) \tag{1}$$

The position of the flash in the camera coordinate $\mathbf{b}$ is obtained prior to acquisition.

We select the manual exposure mode and keep the aperture, shutter speed and radiance of the flash unchanged during the capture. We set the aperture small and the shutter speed fast to eliminate the effect of ambient light. These functions are available from many high-end compact cameras and DSLRs.

5. Mesh Deformation

From the flash photographs, we reconstruct detailed 3D models of a texture-less object integrating shading and silhouette cues using the mesh deformation method. We assume that the camera pose ($\mathbf{o}_l$ and $\mathbf{r}_l$) and the light position $\mathbf{x}_l$ are known. Also, we assume the radiance of the flash does not change between the images. There is a possibility that we cannot control the radiance of the flash perfectly, resulting in low-frequency biases of a normal field obtained from photometric stereo. However, we can address this by extracting high-frequency components and incorporating only those into the optimization. Input images have to have dark backgrounds and light-colored foregrounds, but our method does not require high-quality silhouettes (i.e., self shadows, background objects, etc. exist in the images).

The mesh consists of n vertices and m triangles. The vertex position is defined by $\mathbf{v}_p \in \mathbb{R}^3$, $p \in 1...n$, and we concatenate all the vertices into a $n \times 3$ matrix, $\mathbf{v} = [\mathbf{v}_1 \ldots \mathbf{v}_n]^T$. The index of a triangle is represented by $i \in 1...m$. We denote the centroid of triangle i by $\mathbf{c}_i$. We also denote that the normal of triangle i (the mesh normal) by $\mathbf{N}_i$ and the vertex normal of vertex p by $\mathbf{n}_p$. In addition, we use $\tilde{\mathbf{v}}$ to define the deformed vertices.

Energy terms Our optimization framework comprises three energy terms: the photometric normal term $E^{\mathrm{ps}}(\mathbf{v})$, the silhouette position term $E^{\mathrm{sil}}(\mathbf{v})$ and the regularization term $E^{\mathrm{reg}}(\mathbf{v})$. Overall, we minimize the following energy:

$$E(\mathbf{v}) = w^{\mathrm{ps}}E^{\mathrm{ps}}(\mathbf{v}) + w^{\mathrm{sil}}E^{\mathrm{sil}}(\mathbf{v}) + w^{\mathrm{reg}}E^{\mathrm{reg}}(\mathbf{v}) \tag{2}$$

where w^{ps}, w^{sil} and w^{reg} are the respective weights.

The main challenge is the derivation of the photometric normal term. In the case of the level-set method, normals are simply normalized gradients of the implicit surface. As for the mesh deformation method, however, we cannot directly relate normals to vertex positions. Therefore, we use deformation gradients to associate surface normals with mesh vertices.

Deformation gradient Consider triangle i with its three vertices before and after deformation $[\mathbf{v}_{i1}, \mathbf{v}_{i2}, \mathbf{v}_{i3}]$ and $[\tilde{\mathbf{v}}_{i1}, \tilde{\mathbf{v}}_{i2}, \tilde{\mathbf{v}}_{i3}]$. Given the tangential vectors before and after deformation $\mathbf{V}_i = [\mathbf{v}_{i2} - \mathbf{v}_{i1}, \mathbf{v}_{i3} - \mathbf{v}_{i1}]$ and $\tilde{\mathbf{V}}_i = [\tilde{\mathbf{v}}_{i2} - \tilde{\mathbf{v}}_{i1}, \tilde{\mathbf{v}}_{i3} - \tilde{\mathbf{v}}_{i1}]$, we can approximate the deformation gradient $\mathbf{T}_i \in \mathbb{R}^{3\times3}$ by:

$$\mathbf{T}_i = \tilde{\mathbf{V}}_i \mathbf{V}_i^+ \tag{3}$$

where $\mathbf{V}_i^+$ is the pseudo-inverses of the tangential vectors before deformation. Thus, the computation of the $3m \times 3$ deformation gradients $\mathbf{T} = [\mathbf{T}_1 \ldots \mathbf{T}_m]^T$ can achieved linearly from the $n \times 3$ deformed vertex positions $\tilde{\mathbf{v}} = [\tilde{\mathbf{v}}_1 \ldots \tilde{\mathbf{v}}_n]^T$. Using a $3m \times n$ deformation gradient operator $\mathbf{G}$ which contains the pseudo-inverses of the tangential vectors before deformation, we can compute $\mathbf{T}$, such that:

$$\mathbf{T} = \mathbf{G}\tilde{\mathbf{v}} \tag{4}$$

5.1. Photometric normal term

In this section, we derive the photometric normal term. We first compute a photometric normal $\mathbf{N}_i^{\mathrm{ps}}$ by solving a nearby point-source photometric stereo problem [16] for each triangle. We model the flash as a point light source with its radiance unchanged between images. Assuming a Lambertian surface, the problem is posed as:

$$\underset{\mathbf{N}_i^{\mathrm{ps}}, \rho_i}{\arg\min} \sum_{l \in \mathcal{V}_i} \| I_{i,l} - \rho_i \mathbf{N}_i^{\mathrm{ps}} \cdot \mathbf{l}_{i,l} / {r_{i,l}}^2 \|^2, \mathrm{s.t} \| \mathbf{N}_i^{\mathrm{ps}} \| = 1 \tag{5}$$

where $I_{i,l}$, ρ_i, $\mathbf{l}_{i,l}$ and $r_{i,l}$ respectively signify, the intensity of triangle i on the l-th image, the albedo, the unit light vector at the centroid of triangle i, and the light object distance. Note that we excluded the radiance of the light source from the equation following the assumption we made. $\mathcal{V}_i$ is the visibility map containing information in which images triangle i is apparent. We can estimate $\mathcal{V}_i$ from the current mesh. $\mathbf{l}_{i,l}$ and $r_{i,l}$ are obtained from the triangle's centroid $\mathbf{c}_i$ and the light position $\mathbf{x}_l$ as follows:

$$r_{i,l} = \| \mathbf{c}_i - \mathbf{x}_l \|, \ \mathbf{l}_{i,l} = (\mathbf{c}_i - \mathbf{x}_l)/r_{i,l} \tag{6}$$

Thus, the solution of Eq.(5) is obtained via a linear least-squares minimization. If the normalized residual of Eq.(5) exceeds the threshold, then we replaced $\mathbf{N}_i^{\mathrm{ps}}$ with $\mathbf{N}_i$. We set the threshold value as 0.8.

It is known that a normal field obtained using photometric stereo are prone to low-frequency biases. This is particularly true if the radiance of the flash changes between the images and the assumption is not met. Although we checked empirically that our assumption is satisfied during the capture, we avoid possible low frequency-biases by extracting high-frequency components using the method similar to those proposed in [21].

First, we smoothed the mesh normal by:

$$\bar{\mathbf{N}}_i = \sum_{j \in \{i\} \cup \mathrm{adj}(i)} \mathbf{N}_j / N_i \tag{7}$$

where $\mathrm{adj}(i)$ is the set of triangles adjacent to triangle i, and N_i is the number of triangles in $\{i\} \cup \mathrm{adj}(i)$. We iterate this operation until we can remove high-frequency noise in the mesh normals. Similarly, we smoothed the photometric normal $\mathbf{N}_i^{\mathrm{ps}}$ with the same amount to obtain $\bar{\mathbf{N}}_i^{\mathrm{ps}}$. In our case, we found that 40 iterations are sufficient. After smoothing of the normals, we can get the following relations.

$$\bar{\mathbf{N}}_i = \mathbf{R}_i^{\mathrm{smooth}} \mathbf{N}_i \tag{8}$$

$$\mathbf{N}_i^{\mathrm{ps}} = \mathbf{R}_i^{\mathrm{detail}} \bar{\mathbf{N}}_i^{\mathrm{ps}} \tag{9}$$

where $\mathbf{R}_i^{\mathrm{smooth}}$ and $\mathbf{R}_i^{\mathrm{detail}}$ are 3×3 rotation matrices. The first one brings the original mesh normal to the smoothed normal and the second one brings the smoothed photometric normal back to the original photometric normal. Using these relationships, we can then obtain the photometric normals free from low-frequency biases as:

$$\hat{\mathbf{N}}_i = \mathbf{R}_i^{\mathrm{detail}} \mathbf{R}_i^{\mathrm{smooth}} \mathbf{N}_i \tag{10}$$

To associate the new normal $\hat{\mathbf{N}}_i$ with the new vertices $\tilde{\mathbf{v}}$, we use deformation gradients. Given $\mathbf{R}_i = \mathbf{R}_i^{\mathrm{detail}} \mathbf{R}_i^{\mathrm{smooth}}$, the photometric normal term is defined by minimizing the difference of the actual deformation gradient $\mathbf{T}_i$ and the rotation $\mathbf{R}_i$, such that:

$$\begin{aligned} E^{\mathrm{ps}}(\mathbf{v}) &= \sum_{i=1}^{m} \|\mathbf{T}_i - \mathbf{R}_i\|_F^2 \\ &= \|\mathbf{G}\tilde{\mathbf{v}} - \mathbf{R}\|_F^2 \end{aligned} \tag{11}$$

where $\mathbf{R}$ is a $3m \times 3$ matrix, $\mathbf{R} = [\mathbf{R}_1 \ldots \mathbf{R}_m]^T$.

5.2. Silhouette position term

We construct the silhouette position term using Chan and Vese's force [5]. If the silhouettes are clean, a signed distance based force [13], for instance, could be used. Unfortunately, we cannot extract clean silhouettes because an intensity thresholding fails due to shading and the tunnel-effect of the flash. We also tried edge-based forces [29], but the mesh was attracted slightly inside the silhouettes because the intensity of the foreground changes gradually toward silhouettes due to shading. Therefore we chose to use Chan and Vese's region-based force.

We first project the vertices onto the input images using a perspective projection and find a set of vertices that corresponds to the silhouettes. In l-th image, we can find $K(l)$ vertices defined by $\mathbf{v}_{\mathrm{idx}(k(l))}$, $k(l) \in 1 \ldots K(l)$ where $\mathrm{idx}(k(l))$ is the index of the vertex corresponding to the silhouette.

Now Chan-Vese's force for vertex $\mathbf{v}_{\mathrm{idx}(k(l))}$ can be computed by:

$$\mathbf{f}_{k(l)} = \|i_{k(l)} - c_{1,l}\|^2 \mathbf{n}_{\mathrm{idx}(k(l))} - \|i_{k(l)} - c_{2,l}\|^2 \mathbf{n}_{\mathrm{idx}(k(l))} \tag{12}$$

where $c_{1,l}$ and $c_{2,l}$ are the average intensities of the foreground and the background of l-th image, $i_{k(l)}$ is the intensity at the 2D coordinate where $\mathbf{v}_{\mathrm{idx}(k(l))}$ is projected, and $\mathbf{n}_{\mathrm{idx}(k(l))}$ is the vertex normal of $\mathbf{v}_{\mathrm{idx}(k(l))}$. We obtain $c_{1,l}$ and $c_{2,l}$ by averaging intensities of the interior region and exterior region of the current mesh's silhouette. The target silhouette position $\mathbf{q}_{k(l)}$ is computed as

$$\mathbf{q}_{k(l)} = \mathbf{v}_{\mathrm{idx}(k(l))} + \Delta t \; \mathbf{f}_{k(l)} \tag{13}$$

where Δt is the time-step.

Once this procedure is done for all images, we concatenate the target silhouette points of all the images into $\mathbf{q}^{\mathrm{sil}} \in \mathbb{R}^{K \times 3}$ where K is the number of all the target silhouette points. Likewise, we obtain $\mathbf{F}^{\mathrm{sil}} \in \mathbb{R}^{K \times 3}$ containing the silhouette forces of all the images. Now we can rewrite Eq.(13) with the following matrix form:

$$\mathbf{q}^{\mathrm{sil}} = \mathbf{C}^{\mathrm{sil}} \mathbf{v} + \Delta t \; \mathbf{F}^{\mathrm{sil}} \tag{14}$$

where $\mathbf{C}^{\mathrm{sil}}$ is a $K \times n$ matrix having 1 at $\mathrm{idx}(k(l))$ column and otherwise 0. With $\mathbf{C}^{\mathrm{sil}}$, we can choose the vertices that correspond to the silhouettes using a matrix multiplication as follows:

$$\mathbf{C}^{\mathrm{sil}} \mathbf{v} \;=\; \begin{bmatrix} \\ \cdots \mathbf{1} \cdots \\ \\ \end{bmatrix} \begin{bmatrix} \vdots \\ \mathbf{v}_{\mathrm{idx}(k(l))} \\ \vdots \end{bmatrix} \tag{15}$$

Finally, the silhouette position term can be defined as

$$\begin{aligned} E^{\mathrm{sil}}(\mathbf{v}) &= \sum_{l=1}^{L} \sum_{k(l)=1}^{K(l)} \|\tilde{\mathbf{v}}_{\mathrm{idx}(k(l))} - \mathbf{q}_{k(l)}\|^2 \\ &= \|\mathbf{C}^{\mathrm{sil}} \tilde{\mathbf{v}} - \mathbf{q}^{\mathrm{sil}}\|_F^2 \end{aligned} \tag{16}$$

To set Δt an appropriate size, we specify desired accuracy A^d. Δt is computed from A^d as

$$\Delta t = A^d / \|\bar{\mathbf{F}}^{\mathrm{sil}}\| \tag{17}$$

where $\|\bar{\mathbf{F}}^{\mathrm{sil}}\|$ is the average force strength. We used the range of $A^d = [1 \; 3]$ in this paper.

5.3. Vertex optimization

Regularization To avoid large deformation, we incorporate the regularization term keeping deformations of adjacent triangles as similar as possible. This can be achieved by minimizing the differences of deformation gradients of adjacent pairs.

$$\begin{aligned} E^{\mathrm{reg}}(\mathbf{v}) &= \sum_{i=1}^{m} \sum_{j \in \mathrm{adj}(i)} \|\mathbf{T}_i - \mathbf{T}_j\|_F^2 \\ &= \left\| \begin{bmatrix} \cdots \mathbf{I} \cdots - \mathbf{I} \cdots \end{bmatrix} \begin{bmatrix} \vdots \\ \mathbf{T}_i^T \\ \vdots \\ \mathbf{T}_j^T \\ \vdots \end{bmatrix} \right\|_F^2 \\ &= \|\mathbf{M}\mathbf{G}\tilde{\mathbf{v}}\|_F^2 \end{aligned} \qquad (18)$$

$\mathbf{M}$ is a $3P \times 3m$ matrix having a 3 × 3 identity matrix $\mathbf{I}$ at $3(i-1)+1$ to $3i$ columns and $-\mathbf{I}$ at $3(j-1)+1$ to $3j$ columns, where P is the number of triangle pairs.

Optimization Finally, the overall energy is defined from Eq.(11), (16) and (18). Thus, we can obtain the new vertices $\tilde{\mathbf{v}}$ by minimizing:

$$\begin{aligned} E(\mathbf{v}) &= \left\| \begin{bmatrix} w^{\mathrm{ps}}\mathbf{G} \\ w^{\mathrm{sil}}\mathbf{C}^{\mathrm{sil}} \\ w^{\mathrm{reg}}\mathbf{M}\mathbf{G} \end{bmatrix} \tilde{\mathbf{v}} - \begin{bmatrix} w^{\mathrm{ps}}\mathbf{R} \\ w^{\mathrm{sil}}\mathbf{q}^{\mathrm{sil}} \\ \mathbf{0} \end{bmatrix} \right\|_F^2 \\ &= \|\mathbf{A}\tilde{\mathbf{v}} - \mathbf{B}\|_F^2 \end{aligned} \qquad (19)$$

where $\mathbf{A}$ is a large sparse matrix whose size is $(3m + K + 3P) \times n$ and $\mathbf{B}$ is a $(3m + K + 3P) \times 3$ matrix. Equation (19) is solved via a normal equation:

$$\tilde{\mathbf{v}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{B} \qquad (20)$$

We use CHOLMOD [7] for constructing $\mathbf{A}$, factoring $\mathbf{A}^T\mathbf{A}$, and back substitutions. We have empirically determined the ranges of $w^{\mathrm{ps}} = 1$, $w^{\mathrm{sil}} = [0.1\ \ 0.25]$ and $w^{\mathrm{reg}} = [0.8\ 1]$ to work well. w^{ps} is set to 1 from the beginning to prevent large deformation. The erroneous normals are removed through the thresholding of Eq.(5).

6. Topology Adaptation

Mesh-deformation methods, in general, are not capable of handling topology changes. On the other hand, methods using an implicit surface such as level-sets can cope with topology changes well. Here, we developed a simple method to solve this problem by alternating the surface between the explicit and implicit representations. Firstly, from the resulting mesh after deformation, we compute the centroid $\tilde{\mathbf{c}}_i$ and the mesh normal $\tilde{\mathbf{N}}_i$ for each triangle. We call the point $\tilde{\mathbf{c}}_i$ coupled with the normal $\tilde{\mathbf{N}}_i$ the oriented point. We then convert these oriented points into an implicit surface. Because the oriented points can be thought of as gradients of an implicit surface, we can obtain the implicit surface from the oriented points by integration. The resulting function is a signed distance field or an indicator function (inside the surface is 1 and outside is 0 [19]). This can then be converted into a mesh representation using an iso-surface extraction such as marching cubes. With this explicit-implicit conversion, the self-intersecting regions can be properly filtered and topology changes are handled effectively. To convert the oriented points into an implicit function and then to a mesh, we use Poisson surface reconstruction software because of its ease of use and efficiency [19]. A similar approach is presented in [28] in the context of the elastic simulation. Because their aim is to preserve original surface details, topology adaptation is applied only to the region where self-intersection occurs at the time it happens. However, since our method adds surface details, we chose to refine all the regions of the mesh for every iteration. The advantages of our method are that it can produce high-quality triangles and that it is quite easy to implement. The downside of our method is that it is relatively time consuming. When executing Poisson surface reconstruction, we use the depth level 7–9 depending on the surface to recover.

7. Experiments

In this section, we first show our results using real images. We then show experimental results using synthetic models. We evaluate the influence of each energy term to the final result and compared our algorithm with previous methods [14, 21].

7.1. Real images

We tested our method on 4 examples (Figs.2, 3, 4, and 5). All the photographs are 1200×1600 pixels in size. We alternate mesh deformation and topological adaptation 50–100 times. It took approximately 20–40 min to reconstruct one example using Matlab on a recent machine.

Initial mesh Since our method can handle extreme geometrical and topological changes, the initial approximation does not necessarily have to be a good one. In this study, we use a visual hull (containing errors and holes) or primitive meshes (spheres or ellipsoids) as an initial shape. The silhouettes for the visual hull is obtained using the interactive foreground extraction software, GrabCut [23]. The origins and radii of the spheres or ellipsoids are set interactively. The former provides a better initial approximation, whereas the latter requires less user-effort.

Result Our method captures detailed shapes of various objects. Our method can recover thin objects such as a wooden puppet (Fig.2). Also, concave regions such as around the shoulder blades and chest are recovered well (Fig.3). We also compared the results with and without the photometric normal term (Fig.3 (b)). This illustrates how the photometric term contributes to capturing surface details.

The surface evolution process (Fig.2 (c)) shows that our method is capable of extreme topological changes (For the surface evolution processes of other examples, see our supplemental material). In fact, our method can capture a teapot handle starting from a sphere as shown in Fig.4. Our method can also start from a visual hull. For the visual hull reconstruction, we selected 14 silhouettes (Fig.2 (b)) with small extraction errors from 30 views, but the result contains cracks and holes (Fig.2 (d)). Our method with topology adaptation can produce a detailed 3D model (Fig.2 (e)) even starting from it. The result of a naive mesh deformation exhibits reconstruction errors (Fig.2 (f)).

We also evaluated the robustness our method for background objects in the images, and textures and materials of the object surface. Our method works well even though some objects exist in the background (Figs.2 and 5). Also, our method can capture a slightly textured object (Fig. 4). Finally, our method is applicable to a surface which is composed of different materials (Fig.5).

7.2. Synthetic evaluation

Silhouettes vs Shading Here we discuss how each energy term influences the result. To do so, we reconstructed a model using 12 rendered images of a known shape (a muscular body with a height of 1000 mm) starting from an ellipsoid (Init). We have performed experiments incorporating only the photometric normal term (E^{ps}), only the silhouette position term (E^{sil}), and both ($E^{\mathrm{sil+ps}}$). In Fig.6 and Table 1, we showed the output models and reconstruction errors which are measured with distances between the ground truth and the outputs. The photometric normal term on its own can not capture the overall shape. On the other hand, using only the silhouette position term loses surface details. The combination of the photometric term and the silhouette term produces a visually pleasing and accurate result.

Comparison with previous methods Next, we compare our algorithm with previous methods proposed in [14] and [21]. Here, we only compare the results of mesh deformation i.e., topology adaptation is not included in the comparison. Our method fits the mesh normals to the photometric normals using deformation gradients and solve a linear system with a direct method. The method of Hernandez et al. [14] computes displacement vectors that modify orientations of triangles to match with photometric normals. These vectors are averaged at each vertex and then used for mesh deformation using a gradient descent. Nehab et al. [21] minimizes the dot products of mesh's tangent vectors and photometric normals solving a linear system with a direct method. To evaluate accuracy and computational time of the methods, we deform the smoothed mesh by fitting its normals to the normals of the original model (Fig.7). We evaluate accuracy with the average value of angles between the original normals and the deformed mesh's normals.

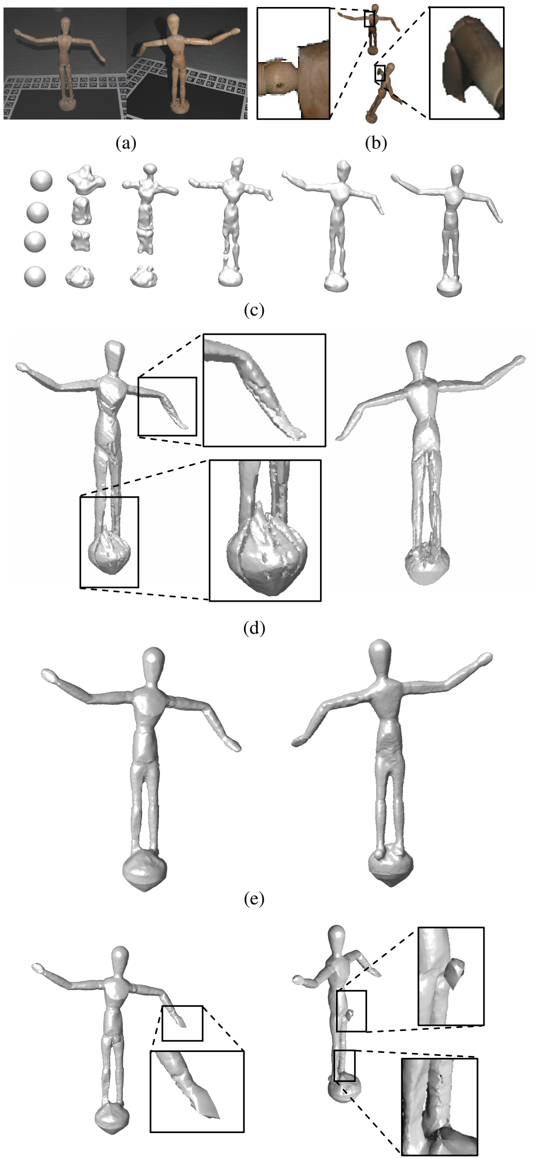

Figure 2. The result of a wooden puppet. We used 30 input images (a). Extracted silhouettes are not clean (b). Starting from 4 spheres, the model is reconstructed after extreme topological and geometrical changes (c). The reconstructed visual hull has holes and cracks due to self-shadows (d). Our method reconstruct a detailed full 3D model starting from the imperfect visual hull (e). The result of a naive mesh deformation exhibits errors (f).

At first glance, because our method uses a direct method to solve a large system, it seems that an iterative method using a gradient descent is more efficient. However, the iterative method is fast only if few iterations are required. For

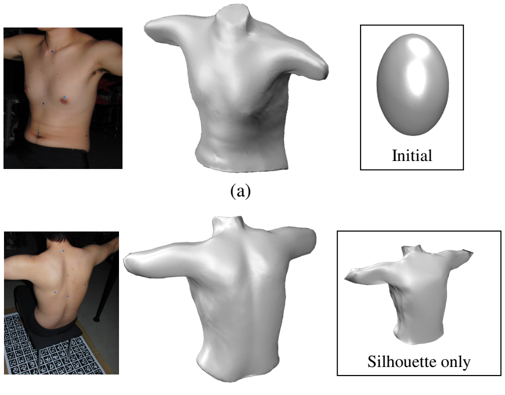

Figure 3. The reconstructed result of an upper body (input: 14 images, output: 23126 triangles). Our method captures concave regions such as the area around chest and shoulder blades. The photometric term contributes to recover surface details (b).

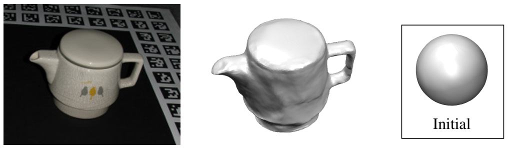

Figure 4. A teapot is reconstructed from a sphere (input: 24 images, output: 95079 triangles). Notice that our method captures a teapot handle.

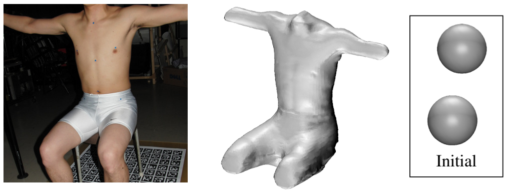

Figure 5. Our method captures a surface covered with different materials (input: 14 images, output: 49334 triangles).

normal fitting, it requires over 100 iterations to converge. So in this case, the direct method such as ours is more efficient than the iterative method like [14]. Related discussions can be found in [8] in the context of mesh fairing.

Also, experimental results show that our method is the most accurate of the three methods. The result of [21] contains noise and is not smooth because their method only minimizes dot products of tangents and photometric normals. On the other hand, our method minimizes Eq.(11) that amounts to solve the following Poisson system [4]:

$$\Delta_s(\tilde{\mathbf{v}}) = \mathrm{div}(\mathbf{R}) \tag{21}$$

Here, Δ_s is the cotangent Laplace operator and div is the divergence operator. This method is equivalent to the mesh gradient fitting [30] whose result is a detailed but smooth mesh. The method of Hernandez et al. [14] can also be represented by a Poisson system. However, from the definition of their force, the uniform weight Laplacian is used as in [13] instead of the cotangent one. As a result, because the cotangent weight is more suitable for irregular triangles with arbitrary shapes, our method is more accurate than [14].

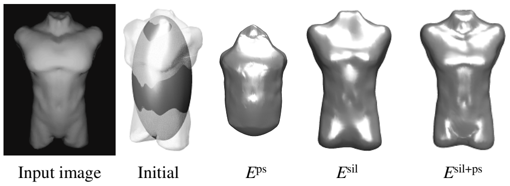

Figure 6. Synthetic evaluation (12 input images). We used an ellipsoid for the initial. The photometric normal term on its own can not capture the overall shape. On the other hand using only the silhouette position term loses details. The combination of the photometric term and the silhouette term produces a visually pleasing result. The result of the quantitative evaluation shown in Table 1 also supports this fact.

Table 1. Quantitative evaluation (mean errors [mm]) of Fig.6.

Init	E^{ps}	E^{sil}	$E^{\mathrm{sil+ps}}$
25.9	19.9	4.3	2.83

8. Conclusion

We have presented a novel method for capturing detailed 3D shapes using a flash-equipped camera. To recover a highly detailed model, we showed an effective way of integrating silhouettes and shading using mesh deformation. Our topology adaptation method is simple and easy to implement and thus it might also be useful for multi-view stereo reconstruction and image segmentation.

Our method also has limitations that need to be overcome. Our method cannot capture when broad areas are covered with dark color. The silhouette force we use fails to recover the region where the intensity is different from that of other areas, such as legs in Fig.5, due to shading and fall-off of the flash. Therefore we would like to replace our silhouette force with the one that is more local. Our method does not model specular reflections explicitly, which needs to be addressed in future work.

Because we use flash photographs as an input, our method can be used in a wide variety of environment. It would therefore be interesting to use our method to capture large objects located outdoors. In this case, the method of calibration is the key to the problem. Structure-from-motion using silhouettes [10] or feature points [12] might be good candidates. Another possible direction is to use our method to enhance the multi-view stereo technique. This might improve high-frequency details of the result. We believe that these extensions would contribute to the development of a 3D scanning technology that can capture objects with com-

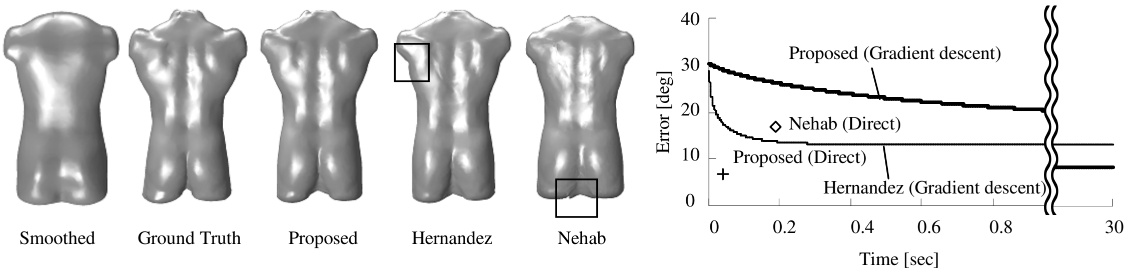

Figure 7. Comparison of algorithms. The proposed method is the most efficient and accurate of the three methods (Proposed, Hernandez [14], Nehab [21]). Given the smoothed mesh (15000 triangles) and the normals of the ground truth model, our method recovers the original detail well. The result using Hernandez et al. [14] loses some detail. The result of Nehab et al. [21] contains noise. One iteration of a gradient descent took approximately 0.001 sec. It took 200 iterations for [14] to converge. Our method took approx. 0.04 sec to solve a linear system. Therefore our method is about 5 times faster than [14]. Although the result of the gradient descent version of our method is as accurate as that of the direct version, it converges slowly (approx. 30 sec).

plex shapes, textures and materials using only a consumer-grade digital camera.

Acknowledgments

We would like to thank Carlos Hernandez for providing us detailed information about their multi-view photometric stereo algorithm.

References

[1] Ahmed, N., Theobalt, C., Dobrev, P., Seidel, H.P., Thrun, S.: Robust fusion of dynamic shape and normal capture for high-quality reconstruction of time-varying geometry. In Proc. CVPR (2008)

[2] ARToolkitPlus. `http://studierstube.icg.tu-graz.ac.at/handheld_ar/artoolkitplus.php`.

[3] Birkbeck, N., Cobzas, D., Sturm, P., Jagersand, M.: Variational shape and reflectance estimation under changing light and viewpoints. In Proc. ECCV (2006)

[4] Botsch, M., Sumner, R., Pauly, M., Gross, M.: Deformation transfer for detail-preserving surface editing. In Proc. Vision, Modeling and Visualization (2006) 357–364

[5] Chan, T., Vese, L.: An active contour model without edges. In Scale-Space Theories in Computer Vision (1999) 141–151

[6] Chang, J., Lee, K., Lee, S.: Multi-view normal field integration using level set methods, In Proc. CVPR (2007)

[7] Chen, Y., Davis, T., Harger, W., Rajamanickam, S.: Algorithm 8xx: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate. Technical report TR-2006-005.

[8] Desbrun, M., Meyer, M., Schroder, P., Barr, A.H.: Implicit fairing of arbitrary meshes using diffusion and curvature flow. In Proc. SIGGRAPH (1999)

[9] Faugeras, O., Keriven. R.: Variational principles, surface evolution, pdes, level set methods, and the stereo problem. IEEE Trans. on Image Processing 7 3 (1998) 336–344

[10] Furukawa, Y., Sethi, A., Ponce, J., Kriegman, D.: Structure and motion from images of smooth textureless objects. In Proc. ECCV (2004).

[11] Furukawa, Y., Ponce, J.: Accurate,dense, and robust multiview stereopsis. In Proc. CVPR (2007)

[12] Goesele, M., Snavely, N., Curless, B., Hoppe, H., Seitz, S.M.: Multi-view stereo for community photo collections. In Proc. ICCV (2007)

[13] Hernandez, C., Schmitt, F.: Silhouette and stereo fusion for 3D object modeling. CVIU 96 3 (2004) 367–392

[14] Hernandez, C., Vogiatzis, G., Cipolla, R.: Multi-view photometric stereo. IEEE Trans. PAMI 30 3 (2008) 548–554

[15] Higo, T., Matsushita, Y., Joshi, N., Ikeuchi. K.: A hand-held photometric stereo camera for 3-D modeling. In Proc. ICCV (2009)

[16] Iwahori, Y., Sugie, H., Ishii, N.: Reconstructing shape from shading images under point light source illumination, In Proc. ICPR, (1990) 83–87

[17] Jin, H., Cremers, D., Yezzi, A.J., Soatto, S.: Shedding light on stereoscopic segmentation. In Proc. CVPR (2004)

[18] Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. IJCV 1 4 (1987) 321–331

[19] Kazhdan. M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In Proc. SGP (2006) 26–28

[20] Lim, J., Ho, J., Yang, M.H., Kriegman, D.: Passive photometric stereo from motion. In Proc. ICCV (2005) 1635–1642

[21] Nehab, D., Rusinkiewicz, S., Davis, J., Ramamoorthi, R.: Efficiently combining positions and normals for precise 3d geometry. In Proc. SIGGRAPH (2005) 536–543

[22] Paterson, J., Claus, D., Fitzgibbon. A.: Brdf and geometry capture from extended inhomogeneous samples using flash photography. In Proc. Eurographics 24 3 (2005) 383–391

[23] Rother, C., Kolmogorov, V., Blake, A.: GrabCut: Interactive foreground extraction using iterated graph cuts. ACM TOG (2004)

[24] Seitz, S., Curless, B., Diebel, J., Scharstein, D., Szeliski. R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In Proc. CVPR (2006) 519–528

[25] Sumner, R.W., Popovic, J.: Deformation transfer for triangle meshes. In Proc. SIGGRAPH (2004) 399–405

[26] Vlasic, D., Peers, P., Baran, I., Debevec, P., Popovic, J., Rusinkiewicz, S., Matusik, W.: Dynamic shape capture using multi-view photometric stereo. In Proc. SIGGRAPH Asia (2009)

[27] Vogiatzis, G., Torr, P.H.S., Cipolla, R.: Multi-view stereo via volumetric graph-cuts. In Proc. CVPR (2005) 391–398

[28] Wojtan, C., Thurey, N., Gross, M., Turk, G.: Deforming meshes that split and merge. In Proc. SIGGRAPH (2009)

[29] Xu, C., Prince, J.L.: Snakes, shapes, and gradient vector flow. IEEE Trans. Image Processing 7 3 (1998) 359–369

[30] Yu, Y., Zhou, K., Xu, D., Shi, X., Bao, H., Guo. B., Shum. H.: Mesh editing with Poisson-based gradient field manipulation. In Proc. SIGGRAPH (2004)