Recovering Surface Details under General Unknown Illumination Using Shading and Coarse Multi-view Stereo

Di Xu[1], Qi Duan[1], Jianming Zheng[1], Juyong Zhang[2*], Jianfei Cai[1] and Tat-Jen Cham[1]
[1] Nanyang Technological University, 50 Nanyang Ave, Singapore 639798
[2] University of Science and Technology of China, Hefei, Anhui, China 230026
[1]{xudi,duan0013,ASJMZheng,ASJFCai,ASTJCham}@ntu.edu.sg, [2]juyong@ustc.edu.cn

***Abstract*—Reconstructing the shape of a 3D object from multi-view images under unknown, general illumination is a fundamental problem in computer vision and high quality reconstruction is usually challenging especially when high detail is needed. This paper presents a total variation (TV) based approach for recovering surface details using shading and multi-view stereo (MVS). Behind the approach are our two important observations: (1) the illumination over the surface of an object tends to be piecewise smooth and (2) the recovery of surface orientation is not sufficient for reconstructing geometry, which were previously overlooked. Thus we introduce TV to regularize the lighting and use visual hull to constrain partial vertices. The reconstruction is formulated as a constrained TV-minimization problem that treats the shape and lighting as unknowns simultaneously. An augmented Lagrangian method is proposed to quickly solve the TV-minimization problem. As a result, our approach is robust, stable and is able to efficiently recover high quality of surface details even starting with a coarse MVS. These advantages are demonstrated by the experiments with synthetic and real world examples.**

I. Introduction

This paper considers the problem of recovering the surface details of 3D shape of an object from multi-view images. With advance and wide availability of various acquisition devices, 3D shape recovery under general unknown illumination conditions is of significant interest in practice. Extensive research has been done in this area and many techniques have been developed. In particular, multi-view stereo (MVS) methods [16] compute depth from corresponding views of the same point in multiple images and reconstruct the overall shape of an object well. Shape-from-shading (SfS) [30] and photometric stereo (PS) methods [5], [22] use shading information to estimate shapes and are able to recover high-frequency surface details. Recently much work aims to combine different techniques to improve the performance of reconstruction [2], [3], [5], [8]–[10], [14], [20], [24], [25], [28].

Though the state-of-the-art has achieved great success, many methods still have various underlying requirements that limit their application scope in practice. For example, SfS and PS methods require the surface reflectance properties to be known in advance or assume varying controlled lighting conditions (in the studio lighting environment) in order to compute the normals of surfaces accurately [27], [28]. Wu *et al.* present an excellent method that combines multi-view stereo and shading-based optimization for high quality reconstruction [25], but their method assumes a good initial MVS mesh. Usually such a mesh is also very dense.

This paper restricts the reconstruction to a static Lambertian object from multi-view images captured under general, unknown illumination conditions. This modest assumption makes the applicable range wide. Similar to Wu *et al.* [25], we are interested in integrating the strengths of MVS and SfS. Different from [25], however, we aim at a shading-based geometry refinement that is able to produce a high resolution model with fine surface details from a coarse initial MVS mesh. This will make our work insensitive to the initial MVS result and easy to be used with the state-of-the-art. To this end, we have made some observations regarding lighting and geometry, proposed a total variation (TV) based formulation for integrating multi-view stereo and shading cues, and used visual hull to constrain partial vertices. An incident illumination vector is introduced to each vertex to simulate the overall lighting effect including self-shadowing and occlusion. A TV term is imposed to the illumination vector for regularization and also for preserving the piecewise smooth property that the illumination vector tends to have. Our algorithm starts with a coarse initial MVS mesh, refines it using classic subdivision scheme, and uses the proposed TV-based minimization with visual hull constraint to optimize the mesh and estimate the illumination as well. The major contributions of the paper are as follows:

- We have observed that the recovery of surface orientation is not sufficient for reconstructing geometry and thus suggest using the visual hull to constrain partial vertices of the initial MVS mesh, which helps enhance the robustness of our reconstruction method.
- We have proposed a brand new formulation of shape optimization and lighting estimation based on TV-minimization, which enables us to generate high fidelity 3D models from images captured under general lighting

*Corresponding author: Juyong Zhang.

conditions. Different from [25] which alternates lighting estimation and geometry optimization, our method estimates both the geometry and illumination together. This helps speed up the computation and reduce the chance of converging to a wrong solution or diverging.
- We have developed an augmented Lagrangian solver to effectively solve the TV-minimization problem. The solver may be adapted for other TV-based problems.

II. Related Work

An excellent survey on MVS is given in [16] where the state-of-the-art MVS algorithms are compared and evaluated. These algorithms are classified into four categories: volumetric methods [17], [18], [21], surface evolution methods [19], [29], depth map fusion methods [4], [12], [13] and surface region growing methods [6], [7]. One essential part of MVS methods is to find point matches from multi-view images and then to calculate depth through triangulation. If good correspondence cannot be found in smoothly shading regions, high-frequency shape detail may not be recovered well. The MVS evaluation benchmark [16] shows that even the-state-of-the-art MVS algorithms may miss high-frequency surface details while they can recover rough shapes well.

In contrast to MVS, SfS estimates surface normal from shading cues [30]. The shape is then reconstructed based on the normal field. In this way, the high-frequency shape detail can be well generated. However, the SfS usually imposes some assumptions on the illumination. For example, the illumination comes from a single direction or is known as a prior [30].

Due to the nature of MVS and SfS, it is interesting to combine them to maximize the strengths of both techniques [5], [8]–[10], [24], [25], [28]. For example, Jin *et al.* [8], [9] introduce variational frameworks that combine various MVS and shading constraints to estimate the shape of Lambertian objects, surface albedo and lighting condition through surface evolution and variational minimization methods. Joshi *et al.* [10] proposed to merge the depth map and surface normal field to reconstruct objects model while Hernández *et al.* [5] first recover the surface normal and then refine the MVS model. Although these two methods can produce good 3D reconstruction, they require a very large number of sample images in different viewpoints and vary light conditions in each viewpoint. Moreover, they both assume a single point light source in a dark room and cannot handle general situations. Yoshiyasu *et al.* [28] introduce a topology adaptive mesh evolution method by evolving the object model to match MVS boundary and shading constraints. Compared with [5], [10], this method reduces the number of required sample images but still assumes a single light source with known position and directions in a dark environment. The work most related to ours is Wu *et al.*'s method [25] that uses sphere harmonic functions to model general illumination conditions and refines the geometric model through shading. The method can efficiently recover surface details. While our method estimates the geometry and lighting simultaneously and just requires a coarse initial MVS mesh, their method performs the lighting estimation and geometry refinement separately and assumes a good initial guess of the geometry to ensure convergence of the iterations.

III. Surface detail recovery using shading and coarse MVS

Given multiple images taken from different viewpoints under general, unknown, but fixed and distant illumination, we first use existing MVS methods to generate an initial MVS mesh. The camera parameters are also recovered if they are unknown. The mapping between the mesh and the multi-view images is then established so that for each mesh vertex, its corresponding intensity value captured in each of multi-view images can be found. It is worth noting that while a dense MVS mesh with a good guess of the geometry is usually needed for previous work such as [25], a coarse MVS mesh is sufficient for our method. This makes our method simple and fast in building the initial MVS mesh. Once we obtain a coarse initial mesh, we use classic mesh subdivision schemes such as the Butterfly subdivision to refine the mesh, generating a dense mesh.

Next we use the shading cues to optimize the position of the mesh vertices in order to deliver a high-resolution triangular mesh that recovers high-frequency surface details in addition to the overall geometry shape. This is done by iteratively minimizing a TV-based objective function in terms of geometry and lighting (see Section III-B). The lighting is formulated by an illumination vector introduced for each vertex (see Section III-A) and the total variation is applied to constrain the lighting. The optimal geometry is the one whose surface details best reflect the shading variations in the multi-view images. An augmented Lagrangian method is proposed to effectively solve the minimization problem (see Section IV).

A. Vertex overall illumination vector

The image formation can be approximately described by the Lambertian reflectance model [11]:

$$I_o(v) = \int_{\Omega(v)} \rho(v) I_i(v,\omega) \max(\omega \cdot \mathbf{n}(v), 0) V(v,\omega) d\omega \quad (1)$$

where $I_o(v)$ is the reflected radiance of the object at vertex v, $\rho(v)$ is the bidirectional reflectance distribution function (BRDF), ω is the incident direction, $I_i(v,\omega)$ is the incident radiance along ω, $\mathbf{n}(v)$ is the unit surface normal at v, $\Omega(v)$ represents a hemisphere of incident directions at v, and $V(v,\omega)$ stands for a binary visibility function of vertex v to direction ω.

Denote by $\Omega'(v)$ the subset of Ω for which $\omega \cdot \mathbf{n}(v) > 0$ and $V(v,\omega) = 1$. Then the model (1) can be rewritten as

$$I_o(v) = \left(\int_{\Omega'(v)} \rho(v) I_i(v,\omega) \omega d\omega \right) \cdot \mathbf{n}(v). \quad (2)$$

Let

$$L(v) = \int_{\Omega'(v)} \rho(v) I_i(v,\omega) \omega d\omega. \quad (3)$$

We call $L(v)$ the *vertex overall illumination vector* at v. It can be used to represent the overall effect of all incident lights (see Figure 1 for an illustration). From $L(v)$, the reflected radiance can be easily computed: $I_o(v) = L(v) \cdot \mathbf{n}(v)$. Thus instead of estimating individual incident lights and computing the visibility function of each vertex to each possible incident light direction, we propose to estimate the overall illumination vector for each vertex. This approach can handle concave surfaces and self-occlusion automatically, and simplify the reconstruction process.

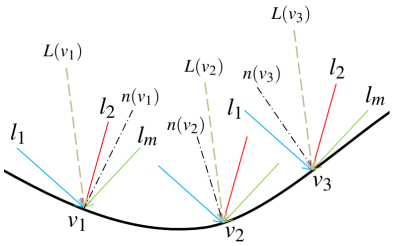


Figure 1. At each point v_i on the surface, the vertex overall illumination vector $L(v_i)$ represents the overall effect of all incident lights such as $l_1, l_2, \cdots, l_m$ from different directions.

It can be seen from (3) that the vertex overall illumination vector $L(v)$ depends on surface normal $\mathbf{n}(v)$, visibility function $V(v,\omega)$, BRDF $\rho(v)$ and the lighting condition. If $\rho(v)$ is constant or varies smoothly and the lighting condition can be approximated by a few distant light sources, then $L(v)$ will exhibit to be piecewise constant or piecewise smooth over the surface of the object. Fig. 2 visualizes $L(v)$ of a teapot model lit by several directional light sources. The rendering indicates that the overall illumination vector is piece-wise constant over the teapot surface.

Figure 2. Visualization of $L(v)$, which is mapped to RGB, over a teapot model lit by several directional light sources.

B. Iterative TV-minimization

Assume the obtained initial MVS mesh is composed of N vertices $\{v_i^{in}\}$ and a set of triangles. We denote the normal of the mesh at v_i^{in} by $\mathbf{n}_i^{in}$. In the next step, we fix the connectivity of the mesh, but move vertices v_i^{in} to new positions v_i such that the new mesh better matches the intensity captured in the multi-view images. Our basic approach is to treat vertices v_i and their respective overall illumination vectors $L(v_i)$ as variables and find them as a solution to the minimization problem

$$\min \left\{ \frac{\alpha}{2} E_f + \frac{\beta}{2} E_{sh} + \frac{\eta}{2} E_{lap} + E_{tv} \right\} \quad (4)$$

where α, β and η are weights. Unless the difference between new positions v_i and v_i^{in} is within a prescribed threshold, we replace v_i^{in} by v_i and repeat solving the minimization problem. The objective function of (4) consists of four terms accounting for position, shading, smoothness and lighting constraints, which are explained below.

(1) **Fidelity term**:

$$E_f = \sum_{i=1}^{N} (\|v_i - v_i^{in}\|^2 + \|\mathbf{n}(v_i) - \mathbf{n}_i^{in}\|^2).$$

This term is introduced to prevent the refined vertices v_i and their normals $\mathbf{n}(v_i)$ from deviating from their counterparts of the initial MVS mesh too much. The normal $\mathbf{n}(v_i)$ is calculated by the sum of normals of the neighboring triangle faces surrounding v_i. If the vertices on the 1-ring neighbors of v_i are $v_{i,1}, v_{i,2}, \cdots, v_{i,a}$, the normal can be computed by

$$\mathbf{N}(v_i) = v_{i,1} \times v_{i,2} + v_{i,2} \times v_{i,3} + \cdots + v_{i,a} \times v_{i,1}$$

and $\mathbf{n}(v_i) = \frac{\mathbf{N}(v_i)}{\|\mathbf{N}(v_i)\|}$. Thus E_f is a function of v_i.

(2) **Shading term**:

$$E_{sh} = \sum_{i=1}^{N} \|L(v_i) \cdot \mathbf{n}(v_i) - c_i\|^2 + \sum_{(v_i,v_j) \in E} \|(L(v_i) \cdot \mathbf{n}(v_i) - L(v_j) \cdot \mathbf{n}(v_j)) - (c_i - c_j)\|^2$$

where E is the set of all edges of the mesh and c_i is the average of the intensity values corresponding to vertex v_i in all the multi-view images. The first term of E_{sh} is the intensity error measuring the difference between the computed reflected radiance and the average of the captured intensities. The second term of E_{sh} is the gradient error measuring the difference between the gradients of the computed reflected radiance and the average of the captured intensities. As pointed out in [25], the gradient error is more stable than the intensity error. Moreover, the gradient error term imposes another constraint on the normal changes that reflect the high frequency surface details [25], which the conventional MVS methods have difficulty to recover. E_{sh} is a function of both vertices v_i and vertex overall illumination vectors $L(v_i)$.

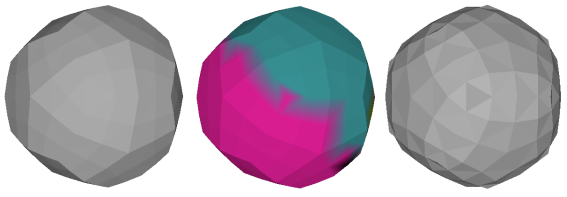
(3) **Laplacian term**: $E_{lap} = \sum_{i=1}^{N} \|v_i - \overline{v}_i\|^2$ where $\overline{v}_i$ is the average of all the 1-ring neighboring vertices of v_i. This term is computed as the squared sum

of the Laplacian of all vertices, which helps avoid generating singular or invalid triangles in the mesh updating process and make the updated mesh smooth.

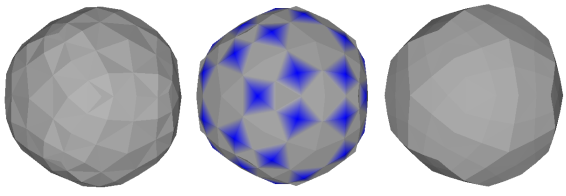
(4) **TV term**: $E_{tv} = \sum_{i=1}^{N} \|\nabla L(v_i)\|$ where ∇ is the discrete operator of the intrinsic gradient on the triangular mesh. The computation of ∇ can refer to [23]. E_{tv} is a total variation regulation term that enables good edge preservation while removing noise [15]. It has been observed that the overall illumination vectors tend to be piece-wise constant or piece-wise smooth. The TV term E_{tv} is introduced to preserve this property in the optimization process.

C. Visual hull constraint

In [25] it is pointed out that generating 3D geometry from recovered normal fields for general surfaces is non-trivial. In fact, the situation is even worse than that. Sometimes it is impossible to uniquely determine the geometry from the normal fields. For example, for two Lambertian surface meshes lit by directional light sources, if one is a scale of the other, they could have the same normal fields and same intensities. This observation implies that the formulation only relying on shading cues in SfS methods is likely under-constrained for 3D geometry. Thus it may produce artifacts of prick-shapes in reconstructed geometry or it may not deliver accurate vertex positions even if the recovered normal field is correct. Fig. 3 shows such an example, where (a) is a ground truth, (b) is the lighting condition, (c) is a mesh obtained by perturbing the ground truth and it is used as the initial mesh for reconstruction and (d) is the reconstruction result using the TV-minimization. Note that in this case the objective function has actually reached its minimum value 0 which suggests that Fig. 3(d) is an optimal solution, but obviously the shape in Fig. 3(d) is different from the ground truth.



(a) Ground truth (b) Lighting condition (c) Initial mesh



(d) Output mesh without constraints (e) Constraint candidates (in blue) (f) Output mesh with constraints

Figure 3. Reconstruction with and without constraints

The above example implies that we need to impose further constraints in the minimization process in order to precisely recover the ground truth. Here we propose a heuristic approach using the visual hull which is created by the intersection of silhouette cones. The visual hull contains the reconstructed object. We compare the initial MVS mesh with the visual hull and identify those vertices on the MVS mesh, which are close to the visual hull and at which the MVS mesh is approximately tangential to the visual hull. The blue region in Fig. 3(e) show such vertices. We believe that these vertices are likely on the ground truth and thus fix them in the reconstruction. In this way, the TV-minimization problem becomes a constrained one and the variables are a subset of all vertices, which usually makes the minimization problem have a unique solution. Fig. 3(f) is the reconstruction result with those vertices in the blue region being fixed, which reproduces the ground truth.

IV. Augmented Lagrangian-based Solver

Solving the minimization problem (4) is difficult due to the non-differentiability of the total variation term and the non-linearity of the unit normal vectors. In this section we propose an augmented Lagrangian (ALM)-based solver to solve (4). Augmented Lagrangian methods are known as a good alternative to penalty methods for solving constrained optimization problems in that they replace a constrained optimization problem by a series of unconstrained problems.

Let $\mathcal{V} = [v_1, v_2, \cdots, v_N]$, $\mathcal{L} = [L(v_1), \cdots, L(v_N)]$ and denote $[\nabla L(v_1), \cdots, \nabla L(v_N)]$ by $\nabla\mathcal{L}$. We introduce new variable $\mathcal{P} = [P_1, P_2, \cdots, P_N]$ and reformulate the TV-based minimization problem (4) to the following constrained problem:

$$\begin{aligned} &\min \left\{ \tfrac{\alpha}{2} E_f + \tfrac{\beta}{2} E_{sh} + \tfrac{\eta}{2} E_{lap} + R(\mathcal{P}) \right\} \\ &s.t. \quad \mathcal{P} = \nabla L(v) \end{aligned}$$

where $R(\mathcal{P}) = \sum_{i=1}^{N} \|P_i\|$. To solve this problem, we define the augmented Lagrangian functional

$$\begin{aligned} \mathcal{G}(\mathcal{V}, \mathcal{L}, \mathcal{P}; \lambda) &= \frac{\alpha}{2} E_f + \frac{\beta}{2} E_{sh} + \frac{\eta}{2} E_{lap} + R(\mathcal{P}) \\ &+ \lambda \cdot (\mathcal{P} - \nabla\mathcal{L}) + \frac{r}{2} \|\mathcal{P} - \nabla\mathcal{L}\|^2 \end{aligned} \quad (5)$$

where $\lambda = [\lambda_1, \cdots, \lambda_N]$ is the Lagrange multiplier, r is a positive constant, and $\frac{r}{2}\|\mathcal{P} - \nabla\mathcal{L}\|^2$ is the augmented term. Consider the following saddle-point problem: find $(\mathcal{V}^*, \mathcal{L}^*, \mathcal{P}^*; \lambda^*)$ such that

$$\mathcal{G}(\mathcal{V}^*, \mathcal{L}^*, \mathcal{P}^*; \lambda) \le \mathcal{G}(\mathcal{V}^*, \mathcal{L}^*, \mathcal{P}^*; \lambda^*) \le \mathcal{G}(\mathcal{V}, \mathcal{L}, \mathcal{P}; \lambda^*)$$

for all $(\mathcal{V}, \mathcal{L}, \mathcal{P}; \lambda)$.

According to [26], the saddle-point problem has at least one solution and all the saddle-points $(\mathcal{V}^*, \mathcal{L}^*, \mathcal{P}^*; \lambda^*)$ have the same $\mathcal{V}^*$ and $\mathcal{L}^*$ which are the solution to the original problem (4). Thus we solve (5) by iteratively solving two subproblems: the $\mathcal{VL}$-subproblem and the $\mathcal{P}$-subproblem.

- $\mathcal{VL}$-subproblem: Given $\mathcal{P}$, solve

$$\min_{\mathcal{V},\mathcal{L}}\{\frac{\alpha}{2}E_f+\frac{\beta}{2}E_{sh}+\frac{\eta}{2}E_{lap}+\frac{r}{2}\|(\mathcal{P}^{(k)}+\frac{\lambda^{(k)}}{r})-\nabla\mathcal{L}\|^2\}$$

for $\mathcal{V}$ and $\mathcal{L}$ where k is the previous iteration number. This problem refines both the vertices and the vertex overall illumination vectors. It is a non-linear least squares optimization problem that can be solved through Levenberg-Marquardt algorithm.

- $\mathcal{P}$-subproblem: Given $\mathcal{V}$ and $\mathcal{L}$, solve

$$\min_{\mathcal{P}}\{R(\mathcal{P})+\lambda^{(k)}\cdot\mathcal{P}+\frac{r}{2}\|\mathcal{P}-\nabla\mathcal{L}\|^2\}$$

for $\mathcal{P}$. This problem is decomposable and thus can be solved for each P_i independently. That is, for each i, we solve

$$\min_{P_i}\{\|P_i\|+\lambda_i^{(k)}\cdot P_i+\frac{r}{2}\|P_i-\nabla L(v_i)\|^2\}.$$

By a simple geometric analysis, it can be found that the above problem has a closed form solution

$$P_i=\max(0,1-\frac{1}{r\|w_r\|})w_r$$

where $w_r=\nabla L(v_i)-\lambda_i^{(k)}/r$.

The whole augmented Lagrangian solver is given in Algorithm 1.

Algorithm 1 Augmented Lagrangian Solver

Input: Initial MVS mesh $\{v_i^{in}\in\mathbb{R}^3|i=1,\cdots,N\}$ and the average intensity $\{c_i\in\mathbb{R}^1|i=1,\cdots,N\}$; r,ϵ;

Output: Optimal $\{v_i\in\mathbb{R}^3|i=1,\cdots,N\}$ and overall illumination vector $\{L(v_i)\in\mathbb{R}^3|i=1,\cdots,N\}$;

1: initialization: $v_i^{(0)}=v_i^{in}$, $\mathcal{L}^{(0)}=0$, $\mathcal{P}^{(0)}=0$, $\lambda^{(0)}=0$;
2: **repeat**
3: Solve $\mathcal{VL}$-subproblem:

$$\min_{\mathcal{V},\mathcal{L}}\{\frac{\alpha}{2}E_f+\frac{\beta}{2}E_{sh}+\frac{\eta}{2}E_{lap}+\frac{r}{2}\|(\mathcal{P}^{(k)}+\frac{\lambda^{(k)}}{r})-\nabla\mathcal{L}\|^2\}$$

4: Solve $\mathcal{P}$-subproblem:

$$\min_{\mathcal{P}}\{R(\mathcal{P})+\lambda^{(k)}\cdot\mathcal{P}+\frac{r}{2}\|\mathcal{P}-\nabla\mathcal{L}^{(k+1)}\|^2\}$$

5: Update Lagrange multiplier λ:

$$\lambda^{k+1}=\lambda^k+r(\mathcal{P}^{(k+1)}-\nabla\mathcal{L}^{(k+1)})$$

6: **until** $\sum_{i=1}^{N}\|L(v_i)^{(k+1)}-L(v_i)^{(k)}\|^2<\epsilon$

V. Experiments

We validate our algorithm using two synthetic datasets: buddha and bunny models (Fig. 4), and four real world datasets: the dinoRing and templeRing datasets (Fig. 7) from Middlebury [1], as well as the fish and angel datasets (Fig. 6 and Fig. 5(d)) from [25]. For all the cases, we assume we do not have any prior knowledge about the lighting conditions. To show that our algorithm only needs a coarse initial mesh, unless specified, we use Meshlab to heavily simplify the existing models (either groundtruth or some MVS results) followed by subdivision to generate the initial mesh models as the input to our optimization framework. Table I lists out the different model size information. Experimentally, the parameters r and ϵ are set to 0.5 and $1.0e-10$ for all the cases, and parameters α, β and η are set to $1.0e{+}6$, $1.0e{+}5$, 100 for datasets from Middlebury, $1.0e+5$, $1.0e+5$, 100 for bunny and buddha, and $1.0e+6$, $5.0e+5$, 100 for fish and angel.

Table I
Model sizes, percentage of the visual hull constrained vertices, and the corresponding optimization runtime

# of vertices	bunny	buddha	angel	dinoRing	templeRing	fish∗
original (k)	20	25	42	100	100	200
simplified (k)	8	14	10	24	20	NA
subdivided (k)	27	46	40	140	140	NA
VH constrained	7%	8%	6%	5%	5%	5%
runtime	33 m	45 m	1 h	4 h	4 h	6 h

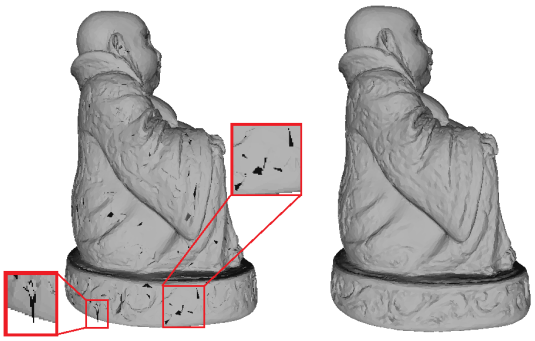
Synthetic data: Fig. 4 shows the results of the synthetic datasets. Here, since we have the groundtruth models, we directly use them for simplification to generate our inputs. Compared with the geometry ground truth in Fig. 4(e), our results in Fig. 4(b) successfully recover many high-frequency shape details, which do not exist in the simplified meshes in Fig. 4(a). To further demonstrate the robustness of our method, we generate another input, which is a random perturbed version of the ground truth, i.e. random vertex displacement up to 0.5% of bounding box, as shown in Fig. 4(c). Even with such highly distorted inputs, our algorithm can still recover the surface details successfully, as shown in Fig. 4(d).

Table II gives the quantitative evaluation results of the reconstruction errors (using input2 in Fig. 4) w.r.t. the groundtruth models. Note that we did not give a numerical evaluation on the case of using input1 since the output1 in Fig. 4) has different number of vertices from that of the groundtruth. From Table II, it can be seen that our method significantly reduces the reconstruction errors by 20% and 71% for budda and 67% and 83% for bunny for the mean position error and the mean normal error respectively. We also list down the reconstruction errors using our method but without the visual hull constraint in Table II, which achieves results worse than the inputs. This further verifies that without the visual hull constraint our algorithm becomes under-constrained and leads to unwanted results. Fig. 5(a) and (b) gives a visual comparison of the buddha example w/o and with the visual hull constraint, where the result w/o the visual hull constraint contain the artifacts of prick-shape

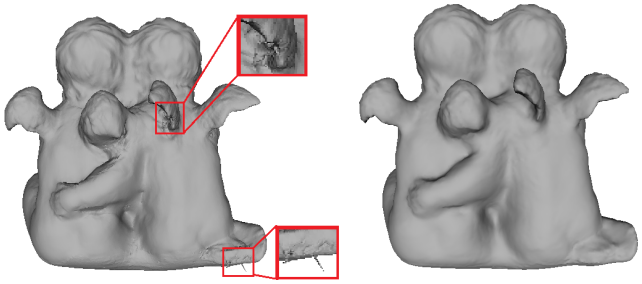
∗The MVS fish model is provided by the author of [25]. It has not been simplified or subdivided for a direct comparison

(a) Input1(simplified) (b) Output1 (c) Input2(distorted) (d) Output2 (e) Ground Truth

Figure 4. Results of the synthetic data (buddha and bunny) using our proposed method.



(a) Buddha result w/o VH constraint (b) Buddha result with VH constraint



(c) Angel result from [25] w/o VH constraint (d) Our angel result with VH constraint

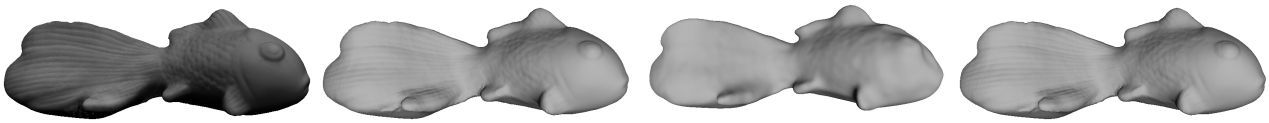
Figure 5. A comparison of the reconstruction results w/o (left) and with (right) the visual hull constraint.

Table II
QUANTITATIVE EVALUATION ON THE SYNTHETIC DATA. FIRST COLUMN: POSITION ERROR (IN ‰OF BOUNDING BOX DIMENSION). SECOND COLUMN: ERROR IN SURFACE NORMAL DIRECTION IN DEGREE.

	Position(‰)		Normal(deg.)	
	mean	std	mean	std
Buddha model				
Input (distorted)	3.57	1.37	23.17	20.88
Output	**2.83**	**1.31**	**6.68**	**6.23**
Output w/o VH	4.21	2.73	27.38	29.42
Bunny model				
Input (distorted)	4.35	1.66	21.83	19.11
Output	**1.45**	**0.88**	**3.77**	**4.09**
Output w/o VH	3.72	3.75	18.68	20.52

vertices.

Real-world datasets: For the real-world datasets, the original dinoRing and templeRing models are generated by the MVS method [18], and then they are simplified and subdivided to generate the inputs to our algorithm. The MVS fish model is provided by the author of [25], which is directly used as our input. The angel model, however, is simplified and subdivided to generate the input(see Table I). Fig. 6 shows the comparison between our result and that of Wu's method [25], which represents the state-of-the-art approach on utilizing lighting and shading information for 3D reconstruction under general unknown illumination. It can be seen that our result of the fish dataset is comparable to that of [25]. Fig. 5(c) and (d) show the comparison of Wu's angle result [25] and ours. Note that Wu's result is directly obtained from the authors of [25]. Without the visual hull constraint, their method suffers from the artifacts of prick-



(a) Captured image (b) Result from [25] (c) Wu's (and also our) input (d) Our result

Figure 6. Comparison with Wu's result for the fish model [25].

shape vertices.

Fig. 7 shows the results for the Middlebury templeRing dataset. Since there is no result available for Wu's method for Middlebury dataset, we compare our result with the original model obtained from the volumetric MVS algorithm [18], which is one of the high-performance MVS methods. It can be seen that our method recovers more surface details such as the pillars of the temple. We also seek for the standard Middlebury MVS evaluation and obtain the accuracy scores of 0.62mm and 0.44mm and the completeness scores of 98.1% and 99.3% for templeRing and dinoRing, respectively, which are reasonable but not at the top. We argue that the standard Middlebury MVS evaluation criteria are not in favor of recovering surface details.

Computational cost: As described in Algorithm 1, in our framework, the vertices and their overall illumination vectors are optimized iteratively until convergence. In each iteration, the computational cost is mainly on the $\mathcal{VL}$-sub problem while the $\mathcal{P}$-sub problem and the λ update can be solved in a very short period of time. The Levenberg-Marquardt algorithm is used to solve the $\mathcal{VL}$-sub problem, which takes around $5 \sim 6$ minutes for the bunny dataset. The iteration is usually repeated about $2 \sim 3$ times until the stopping criteria are satisfied. Thus, the entire energy minimization problem is solved around 30 minutes for the bunny dataset on a standard PC with unoptimized codes. In contrast, Wu's method takes less than half an hour to estimate the lighting and the visibility function and another $1 \sim 4$ hours to refine the mesh [25]. A detailed runtime for all example are shown in Table I.

Limitations: Since our method optimizes all the vertices and overall illumination vectors at the same time, the memory cost of our framework is very high, which limits the number of vertices we can process. In our current implementation, our unoptimized codes cannot handle meshes with more than 300 thousand vertices, which is also one of the reasons that we cannot get higher completeness score from Middlebury benchmark.

VI. Conclusion

We have described a new algorithm for recovering surface details of an object from multi-view images captured under general unknown illuminations. The algorithm is based on a TV-minimization formulation which integrates MVS, shading and visual hull cues. The shape refinement and lighting estimation are obtained simultaneously by solving the minimization problem using the proposed augmented Lagrangian method. It is shown that our method can efficiently reconstruct geometric models with high-frequency surface details. Compared to the existing method, our algorithm has less requirement on the initial MVS mesh.

Acknowledgement: This research, which is carried out at BeingThere Centre, is supported by Singapore MoE AcRF Tier-1 Grant RG30/11 and the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

References

[1] Middlebury multi-view stereo evaluation. http://vision.middlebury.edu/mview/eval/.

[2] T. Beeler, B. Bickel, P. Beardsley, B. Sumner, and M. Gross. High-quality single-shot capture of facial geometry. *ACM Transactions on Graphics (TOG)*, 29(4):40, 2010.

[3] T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Beardsley, C. Gotsman, R. W. Sumner, and M. Gross. High-quality passive facial performance capture using anchor frames. In *ACM Transactions on Graphics (TOG)*, volume 30, page 75. ACM, 2011.

[4] N. D. F. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *Computer Vision - ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I*, pages 766–779, 2008.

[5] C. H. Esteban, G. Vogiatzis, and R. Cipolla. Multiview photometric stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(3):548–554, 2008.

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

[7] M. Habbecke and L. Kobbelt. A surface-growing approach to multi-view stereo reconstruction. In *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA*, 2007.

[8] H. Jin, D. Cremers, D. Wang, E. Prados, A. J. Yezzi, and S. Soatto. 3-d reconstruction of shaded objects from multiple images under unknown illumination. *International Journal of Computer Vision*, 76(3):245–256, 2008.

[9] H. Jin, S. Soatto, and A. J. Yezzi. Multi-view stereo reconstruction of dense shape and complex appearance. *International Journal of Computer Vision*, 63(3):175–189, 2005.

[10] N. Joshi and D. J. Kriegman. Shape from varying illumination and viewpoint. In *IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007*, pages 1–7, 2007.

(a) Captured image (b) MVS [18] (c) Our input (d) Our result

Figure 7. Recovery results of the Middlebury templeRing dataset.

[11] J. T. Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '86, pages 143–150, New York, NY, USA, 1986. ACM.

[12] J. Li, E. Li, Y. Chen, L. Xu, and Y. Zhang. Bundled depth-map merging for multi-view stereo. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, pages 2769–2776, 2010.

[13] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nistér, and M. Pollefeys. Real-time visibility-based fusion of depth maps. In *IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007*, pages 1–8, 2007.

[14] D. Nehab, S. Rusinkiewicz, J. Davis, and R. Ramamoorthi. Efficiently combining positions and normals for precise 3d geometry. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 536–543. ACM, 2005.

[15] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[16] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA*, pages 519–528, 2006.

[17] S. N. Sinha, P. Mordohai, and M. Pollefeys. Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In *IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007*, pages 1–8, 2007.

[18] P. Song, X. Wu, and M. Y. Wang. Volumetric stereo and silhouette fusion for image-based modeling. *The Visual Computer*, 26(12):1435–1450, 2010.

[19] R. Tylecek and R. Sara. Refinement of surface mesh for accurate multi-view reconstruction. *International Journal of Virtual Reality*, 9(1):45–54, 2010.

[20] L. Valgaerts, C. Wu, A. Bruhn, H.-P. Seidel, and C. Theobalt. Lightweight binocular facial performance capture under uncontrolled lighting. *ACM Trans. Graph.*, 31(6):187, 2012.

[21] G. Vogiatzis, C. H. 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.*, 29(12):2241–2246, 2007.

[22] R. J. Woodham. Photometric method for determining surface orientation from multiple images, shape from shading, 1989.

[23] C. Wu, J. Deng, F. Chen, and X.-C. Tai. Scale-space analysis of discrete filtering over arbitrary triangulated surfaces. *SIAM J. Imaging Sciences*, 2(2):670–709, 2009.

[24] C. Wu, Y. Liu, Q. Dai, and B. Wilburn. Fusing multiview and photometric stereo for 3d reconstruction under uncalibrated illumination. *IEEE Trans. Vis. Comput. Graph.*, 17(8):1082–1095, 2011.

[25] C. Wu, B. Wilburn, Y. Matsushita, and C. Theobalt. High-quality shape from multi-view stereo and shading under general illumination. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 969–976, 2011.

[26] C. Wu, J. Zhang, Y. Duan, and X.-C. Tai. Augmented lagrangian method for total variation based image restoration and segmentation over triangulated surfaces. *J. Sci. Comput.*, 50(1):145–166, 2012.

[27] K.-J. Yoon, E. Prados, and P. Sturm. Joint estimation of shape and reflectance using multiple images with known illumination conditions. *International Journal of Computer Vision*, 86:192–210, 2010. 10.1007/s11263-009-0222-4.

[28] Y. Yoshiyasu and N. Yamazaki. Topology-adaptive multi-view photometric stereo. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 1001–1008, 2011.

[29] A. Zaharescu, E. Boyer, and R. Horaud. Transformesh : A topology-adaptive mesh-based approach to surface evolution. In *Computer Vision - ACCV 2007, 8th Asian Conference on Computer Vision, Tokyo, Japan, November 18-22, 2007, Proceedings, Part II*, pages 166–175, 2007.

[30] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah. Shape-from-shading: a survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(8):690–706, 1999.