

# Robust Photometric Stereo via Low-Rank Matrix Completion and Recovery <sup>\*</sup>

Lun Wu<sup>\*</sup>, Arvind Ganesh<sup>†</sup>, Boxin Shi<sup>‡</sup>, Yasuyuki Matsushita<sup>§</sup>, Yongtian Wang<sup>\*</sup>  
and Yi Ma<sup>†,§</sup>

<sup>\*</sup>School of Optics and Electronics, Beijing Institute of Technology, Beijing

<sup>†</sup>Coordinated Science Lab, University of Illinois at Urbana-Champaign

<sup>§</sup>Visual Computing Group, Microsoft Research Asia, Beijing

<sup>‡</sup>Key Laboratory of Machine Perception, Peking University, Beijing

lun.wu@hotmail.com, abalasu2@illinois.edu, shiboxin@cis.pku.edu.cn,  
yasumat@microsoft.com, wyt@bit.edu.cn, mayi@microsoft.com

**Abstract.** We present a new approach to robustly solve photometric stereo problems. We cast the problem of recovering surface normals from multiple lighting conditions as a problem of recovering a low-rank matrix with both missing entries and corrupted entries, which model all types of non-Lambertian effects such as shadows and specularities. Unlike previous approaches that use Least-Squares or heuristic robust techniques, our method uses advanced convex optimization techniques that are guaranteed to find the correct low-rank matrix by simultaneously fixing its missing and erroneous entries. Extensive experimental results demonstrate that our method achieves unprecedentedly accurate estimates of surface normals in the presence of significant amount of shadows and specularities. The new technique can be used to improve virtually any photometric stereo method including uncalibrated photometric stereo.

## 1 Introduction

Photometric stereo [1, 2] estimates surface orientations from photographs taken from a fixed viewpoint under different lighting conditions. Since photometric stereo can produce a dense normal field at the level of detail that cannot be achieved by any other triangulation-based approaches, it has generated a lot of interest for accurate shape reconstruction.

It is well understood that when a Lambertian surface is illuminated by at least three known lighting directions, the surface orientation at each visible point can be uniquely determined from its intensities. From different perspectives, it has long been shown that if there are no shadows, the appearance of a convex Lambertian scene illuminated from different lighting directions span a three-dimensional subspace [3] or an illumination cone [4]. Basri and Jacobs [5] and Georgiades *et al.* [6] have further shown that the images of a convex-shaped object with cast shadows can also be well-approximated by a low-dimensional linear subspace. The aforementioned works indicate that there exists a degenerate structure in the appearance of Lambertian surfaces under variation in illumination. This is the key

---

<sup>\*</sup> This work was supported by grants ONR N00014-09-1-0230, NSF CCF 09-64215, and NSF ECCS 07-01676. Boxin Shi will be moving to the University of Tokyo in Fall 2010.

property that all photometric stereo methods harness to determine the surface normals.

Previously, photometric stereo algorithms for Lambertian surfaces generally find surface normals as the *Least Squares* solution to a set of linear equations that relate the observations and known lighting directions, or equivalently, try to identify the low-dimensional subspace using conventional Principal Component Analysis (PCA) [7]. Such a solution is known to be optimal if the measurements are corrupted by only *i.i.d.* Gaussian noise of small magnitude. Unfortunately, in reality, photometric measurements rarely obey such a simplistic noisy linear model: the intensity values at some pixels can be severely affected by specular reflections (deviation from the basic Lambertian assumption), sensor saturations, or shadowing effects. As a result, the Least Squares solution normally ends up with incorrect estimates of surface orientations in practice.

To overcome this problem, researchers have explored various approaches to eliminate such deviations by treating the corrupted measurements as outliers, *e.g.*, using a RANSAC scheme [8, 9], or a median-based approach [10]. To identify the different types of corruptions in images more carefully, Mukaiegawa *et al.* [11] have proposed a method for classifying diffuse, specular, attached, and cast shadow pixels based on RANSAC and outlier elimination.

*Contributions:* In this paper, we propose a simple but principled solution to photometric stereo that can deal with any kind of deviation from the basic Lambertian assumption in a unified framework. We cast the photometric stereo problem as a problem of recovering and completing a low-rank matrix subject to sparse, gross errors like corrupted and missing pixels. Unlike previous heuristic methods, under fairly broad conditions, the new method is guaranteed to correctly recover the low-rank Lambertian diffuse component from the highly corrupted and incomplete observations. Based on advanced convex optimization tools for nuclear norm and  $\ell_1$ -norm minimization, the new method can efficiently obtain highly accurate estimates of surface orientations. Our method can be used to improve virtually any existing photometric stereo method, including uncalibrated photometric stereo [12], where traditionally, corruption in the data (*e.g.*, by specularity) is either neglected or ineffectively dealt with conventional heuristic robust estimation methods.

In contrast to previous robust approaches, our method is computationally more efficient and provides theoretical guarantees for robustness to large errors. More importantly, our method is able to use all the available information simultaneously for obtaining the optimal result, instead of discarding informative measurements, *e.g.*, by either selecting the best set of illumination directions [9] or using the median estimator [10].

## 2 Photometric Stereo as Low-Rank Matrix Recovery with Sparse Errors

In this section, we formulate the problem of estimating the normal map as a rank minimization problem. We first review the basic Lambertian image forma-

tion model, and then discuss how to model large deviations like shadows and specularities. In the following discussion, we make a few assumptions:

- The relative position of the camera and object is fixed across all images.
- The object is illuminated by a point light source at infinity.
- The sensor response is linear.

*Lambertian Image Formation Model.* The appearance  $I$  of a Lambertian scene observed under a lighting direction  $\mathbf{l} \in \mathbb{R}^3$  is described as the inner product:

$$I = \rho \mathbf{n} \cdot \mathbf{l} \quad (1)$$

where  $\rho$  is the diffuse albedo, and  $\mathbf{n} \in \mathbb{R}^3$  is the surface normal. Suppose that we are given  $n$  images  $I_1, \dots, I_n$  of a scene under different lighting conditions. Let the region of interest be composed of  $m$  pixels in each image.<sup>1</sup> We order the pixel locations with a single index  $k$ , and let  $I_j(k)$  denote the observed intensity at pixel location  $k$  in image  $I_j$ . With this notation, we have the following relation about the observation  $I_j(k)$ :

$$I_j(k) = \rho_k \mathbf{n}_k \cdot \mathbf{l}_j \quad (2)$$

where  $\rho_k$  is the albedo of the scene at pixel location  $k$ ,  $\mathbf{n}_k \in \mathbb{R}^3$  is the (unit) surface normal of the scene at pixel location  $k$ , and  $\mathbf{l}_j \in \mathbb{R}^3$  represents the normalized lighting direction vector corresponding to image  $I_j$ .<sup>2</sup> We assume that the light intensity is constant across images to simplify the discussion, although the proposed method is not limited to such a condition.

*Low-rank Matrix Structure.* Consider the matrix  $D \in \mathbb{R}^{m \times n}$  constructed by stacking all the vectorized images  $\text{vec}(I)$  as

$$D = [\text{vec}(I_1) \mid \dots \mid \text{vec}(I_n)] \quad (3)$$

where  $\text{vec}(I_j) = [I_j(1), \dots, I_j(m)]^T$  for  $j = 1, \dots, n$ . It follows from Eq. (2) that  $D$  can be factorized as follows:

$$D = NL \quad (4)$$

where  $N \doteq [\rho_1 \mathbf{n}_1 \mid \dots \mid \rho_m \mathbf{n}_m]^T \in \mathbb{R}^{m \times 3}$ , and  $L \doteq [\mathbf{l}_1 \mid \dots \mid \mathbf{l}_n] \in \mathbb{R}^{3 \times n}$ . Suppose that the number of images  $n \geq 3$ . Clearly, irrespective of the number of pixels  $m$  and the number of images  $n$ , the rank of the matrix  $D$  is at most 3.

*Modeling Corruptions as Sparse Errors.* The low-rank structure of the observation matrix  $D$  described above is seldom observed with real images. This is due to the presence of shadows and specularities in real images.

- **Shadows** arise in real images in two possible ways. Some pixels are not visible in the image because they face away from the light source. Such dark pixels are referred to as *attached shadows* [13]. In deriving Eq. (4) from Eq. (2), we have

<sup>1</sup> Typically,  $m$  is much larger than the number of images  $n$ .

<sup>2</sup> The convention here is that the lighting direction vectors point from the surface of the object to the light source.

implicitly assumed that all pixels of the object are illuminated by the light source in each image. However, if the pixel faces away from the light source, then the relation no longer holds. Mathematically, this implies that Eq. (2) must be rewritten as follows:

$$I_j(k) = \max \{ \rho_k \mathbf{n}_k \cdot \mathbf{l}_j, 0 \} \quad (5)$$

Shadows can also occur in images when the shape of the object’s surface is not convex: parts of the surface can be occluded from the light source by other parts. Even though the normal vectors at such occluded pixels may form a sharp angle with the lighting direction, these pixels appear entirely dark. We refer to such dark pixels as *cast shadows*. Irrespective of the type, all shadows occur in images as dark pixels with very small, if not zero, intensity values.

- **Specularities.** Specular reflection arises when the object of interest is not perfectly diffusive, *i.e.*, when the surface luminance is not purely isotropic. Thus, the intensity of reflected light depends on the viewing angle, and light is reflected in a mirror-like fashion accompanied by a specular lobe when viewed from certain angles. This gives rise to some bright spots or shiny patches on the surface of the object that significantly deviate from the Lambertian assumption.

Suppose we represent all these deviations from the ideal low-rank diffusive model Eq. (4) by an error matrix  $E \in \mathbb{R}^{m \times n}$ . Thus, instead of Eq. (4), the image measurements should be modeled as

$$D = NL + E \quad (6)$$

where the matrix  $E$  accounts for corruption by shadows or specularities. Now suppose that only a small fraction of the pixels in each image exhibit strong specular reflectance and that a large majority of the pixels are illuminated by the light source. Then, most pixels in the input images obey the low-rank diffusive model given by Eq. (4), and hence, most entries in the error matrix  $E$  will be zero, *i.e.*,  $E$  is a sparse matrix. If the matrix  $L$  of lighting directions is known, then we can compute the surface normals, provided that we can decompose  $D$  as the sum of a low-rank matrix and a sparse error matrix. Thus, the problem can be stated more formally as follows:

*Let  $I_1, \dots, I_n$  be  $n$  images of an object under different illumination conditions. If  $D \in \mathbb{R}^{m \times n}$  is defined as given in Eq. (3), then find a sparse matrix  $E$  such that the matrix  $A \doteq D - E$  has the lowest possible rank.*

Using a Lagrangian formulation, we can write the above problem as the following optimization problem:

$$\min_{A, E} \text{rank}(A) + \gamma \|E\|_0 \quad \text{s.t.} \quad D = A + E \quad (7)$$

where  $\|\cdot\|_0$  denotes the  $\ell_0$ -norm (number of non-zero entries in the matrix), and  $\gamma > 0$  is a parameter that trades off the rank of the solution  $A$  versus the sparsity of

the error  $E$ . Let  $(\hat{A}, \hat{E})$  be the optimal solution to Eq. (7). If the lighting directions  $L$  are given, we can easily recover the matrix  $N$  of surface normals from  $\hat{A}$  as:

$$N = \hat{A}L^\dagger \quad (8)$$

where  $L^\dagger$  denotes the Moore-Penrose pseudo-inverse of  $L$ . The surface normals  $\mathbf{n}_1, \dots, \mathbf{n}_m$  can be estimated by normalizing each row of  $N$  to have unit norm.

While Eq. (7) follows from our formulation, it is not tractable since both rank and  $\ell_0$ -norm are non-convex and discontinuous functions. Solving this optimization problem efficiently will be the topic of discussion in the next section.

### 3 Efficient Solution via Convex Programming

As discussed above, the optimization problem given in Eq. (7) is extremely difficult (NP-hard in general) to solve. In this section, we propose to solve it efficiently based on recent advances in algorithms for matrix rank minimization [14–16].

#### 3.1 Convex relaxation and modification

Recently, Wright *et al.* [14] and Chandrasekaran *et al.* [15] have proposed that the problem in Eq. (7) can be solved by replacing the cost function with its convex surrogate, provided that the rank of the matrix  $A$  is not too high and the number of non-zero entries in the matrix  $E$  is not too large. This convex relaxation, dubbed *Principal Component Pursuit* (PCP) in [14], replaces  $\text{rank}(\cdot)$  with the *nuclear norm* (sum of the singular values of the matrix) and the  $\ell_0$ -norm with the matrix  $\ell_1$ -norm (sum of the absolute values of all entries of the matrix). Under quite general conditions, it has been proved in [14, 15] that the following optimization problem has the same optimal solution as Eq. (7):

$$\min_{A,E} \|A\|_* + \lambda \|E\|_1 \quad \text{s.t.} \quad D = A + E \quad (9)$$

where  $\|\cdot\|_*$  and  $\|\cdot\|_1$  represent the nuclear norm and  $\ell_1$ -norm, respectively, and  $\lambda > 0$  is a weighting parameter. Theoretical considerations in [14] suggest that  $\lambda$  must be of the form  $C/\sqrt{\max\{m, n\}}$ , where  $C$  is a constant, typically set to unity. It is interesting to note that the equivalence between Eq. (7) and Eq. (9) is not affected by the magnitude of the singular values of the solution  $A$  or by the magnitude of the non-zero entries of the error matrix  $E$ .

In the framework of PCP, the locations of the non-zero entries in the sparse matrix  $E$  are assumed to be unknown a priori. But if the locations of some of the corrupted entries are known, then we can incorporate that information into the recovery procedure and hence, make the problem somewhat easier to solve. This is similar in spirit to the matrix completion problem [17–19]. Notice that although both shadows and specularities corrupt the low-rank matrix, they have different characteristics. While the locations of the specular pixels are hard to detect, especially that of pixels in specular lobes, it is relatively easy to detect the location of shadows in an image (*e.g.*, by a simple thresholding of the pixel values). Thus, we have more information about the shadows than specularities, and such

information can greatly help finding the correct solution. So mathematically, we have a problem of recovering a low-rank matrix with both missing entries (the shadows) and unknown corrupted entries (the specularities).

We denote by  $\Omega$  the locations of missing entries in the observed matrix  $D$ , defined in Eq. (3), that correspond to shadows in the input images. By a slight abuse of notation, we also denote by  $\Omega$  the linear subspace of  $m \times n$  matrices with support in  $\Omega$ . Let  $\pi_\Omega$  represent the orthogonal projection operator corresponding to the subspace  $\Omega$ . Thus, we modify the PCP problem in Eq. (9) to the following one which does both matrix completion and error correction:

$$\min_{A,E} \|A\|_* + \lambda \|E\|_1 \quad \text{s.t.} \quad \pi_{\Omega^c}(D) = \pi_{\Omega^c}(A + E) \quad (10)$$

where  $\Omega^c$  denotes the linear subspace complementary to  $\Omega$ , and  $\pi_{\Omega^c}$  is the associated projection operator. The above problem is almost identical to the PCP problem (Eq. (9)), except that the linear equality constraint is now applied only on the set  $\Omega^c$  of pixels that are not affected by the detected shadows.

### 3.2 Fast Algorithm using Augmented Lagrange Multiplier

The optimization problem in Eq. (10) can be re-cast as a semidefinite program and solved using interior-point methods. Although interior-point methods have excellent convergence properties, they are not very scalable for large problems. Fortunately, there has been a flurry of work recently on developing scalable algorithms for high-dimensional nuclear-norm minimization [16, 20, 21]. In this section, we show how one such algorithm, the Augmented Lagrange Multiplier (ALM) method [16, 22], can be adapted to efficiently solve Eq. (10).

The basic idea of the ALM method is to minimize the augmented Lagrangian function instead of directly solving the original constrained optimization problem. For our problem Eq. (10), the augmented Lagrangian is given by

$$\mathcal{L}_\mu(A, E, Y) = \|A\|_* + \lambda \|E\|_1 + \langle Y, \pi_{\Omega^c}(D - A - E) \rangle + \frac{\mu}{2} \|\pi_{\Omega^c}(D - A - E)\|_F^2 \quad (11)$$

where  $Y \in \mathbb{R}^{m \times n}$  is a Lagrange multiplier matrix,  $\mu$  is a positive constant,  $\langle \cdot, \cdot \rangle$  denotes the matrix inner product,<sup>3</sup> and  $\|\cdot\|_F$  denotes the Frobenius norm. For appropriate choice of the Lagrange multiplier matrix  $Y$  and sufficiently large constant  $\mu$ , it can be shown that the augmented Lagrangian function has the same minimizer as the original constrained optimization problem [22]. The ALM algorithm iteratively estimates both the Lagrange multiplier and the optimal solution. The basic ALM iteration is given by

$$\begin{cases} (A_{k+1}, E_{k+1}) = \operatorname{argmin}_{A,E} \mathcal{L}_{\mu_k}(A, E, Y_k) \\ Y_{k+1} = Y_k + \mu_k \pi_{\Omega^c}(D - A_{k+1} - E_{k+1}) \\ \mu_{k+1} = \rho \cdot \mu_k \end{cases} \quad (12)$$

where  $\{\mu_k\}$  is a monotonically increasing positive sequence ( $\rho > 1$ ).

<sup>3</sup>  $\langle X, Y \rangle \doteq \operatorname{trace}(X^T Y)$ .

We now focus our attention on solving the non-trivial first step of the above iteration. Since it is difficult to minimize  $\mathcal{L}_{\mu_k}(\cdot)$  with respect to both  $A$  and  $E$  simultaneously, we adopt an alternating minimization strategy as follows:

$$\begin{cases} E_{j+1} = \operatorname{argmin}_E \lambda \|E\|_1 - \langle Y_k, \pi_{\Omega^c}(E) \rangle + \frac{\mu_k}{2} \|\pi_{\Omega^c}(D - A_j - E)\|_F^2 \\ A_{j+1} = \operatorname{argmin}_A \|A\|_* - \langle Y_k, \pi_{\Omega^c}(A) \rangle + \frac{\mu_k}{2} \|\pi_{\Omega^c}(D - A - E_{j+1})\|_F^2 \end{cases} \quad (13)$$

Without loss of generality, we assume that the  $Y_k$ 's and the  $E_k$ 's (and hence,  $Y$  and  $E$ , respectively) have their support in  $\Omega^c$ . Then, the above minimization problems in Eq. (13) can be solved as described below.

We first define the *shrinkage* (or soft-thresholding) operator for scalars as follows:

$$\operatorname{shrink}(x, \alpha) = \operatorname{sign}(x) \cdot \max\{|x| - \alpha, 0\} \quad (14)$$

where  $\alpha \geq 0$ . When applied to vectors or matrices, the shrinkage operator acts element-wise. Then, the first step in Eq. (13) has a closed-form solution given by

$$E_{j+1} = \operatorname{shrink}\left(\pi_{\Omega^c}(D) + \frac{1}{\mu_k} Y_k - \pi_{\Omega^c}(A_j), \frac{\lambda}{\mu_k}\right) \quad (15)$$

Since it is not possible to express the solution to the second step in Eq. (13) in closed-form, we adopt an iterative strategy based on the Accelerated Proximal Gradient (APG) algorithm [23, 21, 20] to solve it. The iterative procedure is given as:

$$\begin{cases} (U_i, \Sigma_i, V_i) = \operatorname{svd}\left(\frac{1}{\mu_k} Y_k + \pi_{\Omega^c}(D) - E_{j+1} + \pi_{\Omega}(Z_i)\right) \\ A_{i+1} = U_i \operatorname{shrink}\left(\Sigma_i, \frac{1}{\mu_k}\right) V_i^T \\ Z_{i+1} = A_{i+1} + \frac{t_i - 1}{t_{i+1}}(A_{i+1} - A_i) \end{cases} \quad (16)$$

where  $\operatorname{svd}(\cdot)$  denotes the singular value decomposition operator, and  $\{t_i\}$  is a positive sequence satisfying  $t_1 = 1$  and  $t_{i+1} = 0.5 \left(1 + \sqrt{1 + 4t_i^2}\right)$ . The entire algorithm to solve Eq. (10) has been summarized as Algorithm 1.

## 4 Experiments

In this section, we verify the effectiveness of the proposed method using both synthetic and real-world images. We compare our results with a simple Least Squares (LS) approach, which assumes the ideal diffusive model given by Eq. (4). However, we do not use those pixels that were classified as shadows (the set  $\Omega$ ). Thus, the LS method can be summarized by the following optimization problem:

$$\min_N \|\pi_{\Omega^c}(D - N L)\|_F \quad (17)$$

We first test our algorithm using synthetic images whose ground-truth normal maps are known [24]. In these experiments, we quantitatively verify the correctness of our algorithm by computing the angular errors between the estimated normal map and the ground-truth. We then test our algorithm on more challenging real images. Throughout this section, we denote by  $m$  the number of pixels in the region of interest in each image, and by  $n$  the number of input images (typically,  $m \gg n$ ).

**Algorithm 1 (Matrix Completion and Recovery via ALM).**

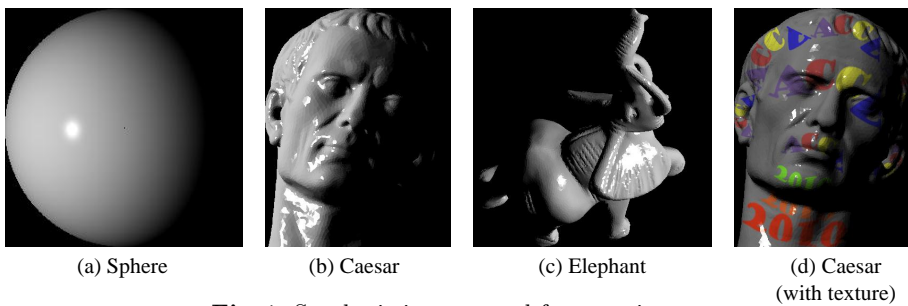

---

**INPUT:**  $D \in \mathbb{R}^{m \times n}$ ,  $\Omega \subset \{1, \dots, m\} \times \{1, \dots, n\}$ ,  $\lambda > 0$ .  
Initialize  $A_1 \leftarrow 0$ ,  $E_1 \leftarrow 0$ ,  $Y_1 \leftarrow 0$ .  
**while** not converged ( $k = 1, 2, \dots$ ) **do**  
 $A_{k,1} = A_k$ ,  $E_{k,1} = E_k$ ;  
**while** not converged ( $j = 1, 2, \dots$ ) **do**  
 $E_{k,j+1} = \text{shrink} \left( \pi_{\Omega^c}(D) + \frac{1}{\mu_k} Y_k - \pi_{\Omega^c}(A_{k,j}), \frac{\lambda}{\mu_k} \right)$ ;  
 $t_1 = 1$ ;  $Z_1 = A_{k,j}$ ;  $A_{k,j,1} = A_{k,j}$ ;  
**while** not converged ( $i = 1, 2, \dots$ ) **do**  
 $(U_i, \Sigma_i, V_i) = \text{svd} \left( \frac{1}{\mu_k} Y_k + \pi_{\Omega^c}(D) - E_{k,j+1} + \pi_{\Omega}(Z_i) \right)$ ;  
 $A_{k,j,i+1} = U_i \text{shrink} \left( \Sigma_i, \frac{1}{\mu_k} \right) V_i^T$ ,  $t_{i+1} = 0.5 \left( 1 + \sqrt{1 + 4t_i^2} \right)$ ;  
 $Z_{i+1} = A_{k,j,i+1} + \frac{t_i - 1}{t_{i+1}} (A_{k,j,i+1} - A_{k,j,i})$ ,  $A_{k,j+1} = A_{k,j,i+1}$ ;  
**end while**  
 $A_{k+1} = A_{k,j+1}$ ;  $E_{k+1} = E_{k,j+1}$ ;  
**end while**  
 $Y_{k+1} = Y_k + \mu_k \pi_{\Omega^c}(D - A_{k+1} - E_{k+1})$ ,  $\mu_{k+1} = \rho \cdot \mu_k$ ;  
**end while**  
**OUTPUT:**  $(\hat{A}, \hat{E}) = (A_k, E_k)$ .

---

**4.1 Quantitative evaluation with synthetic images**

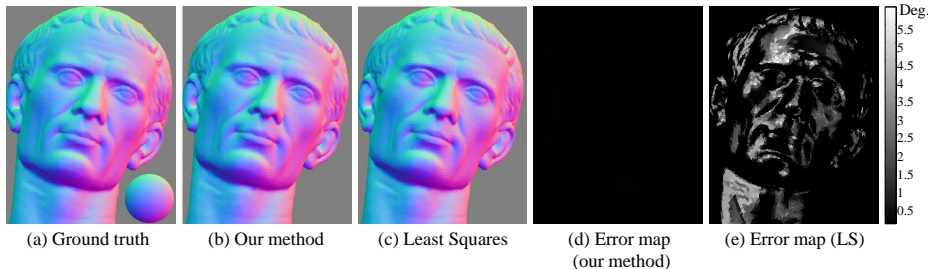
In this section, we use synthetic images of three different objects (see Fig. 1(a)-(c)) under different scenarios to evaluate the performance of our algorithm. Since these images are free of any noise, we use a pixel threshold value of zero to detect shadows in the images. Unless otherwise stated, we set  $\lambda = 1/\sqrt{m}$  in Eq. (10).

**Fig. 1.** Synthetic images used for experiments.

**a. Specular scene.** In this experiment, we generate images of an object under 40 different lighting conditions, where the lighting directions are chosen at random from a hemisphere with the object placed at the center. The images are generated with some specular reflection. For all our experiments, we use the Cook-Torrance reflectance model [25] to generate images with specularities. Thus, there are two sources of corruption in the images – attached shadows and specularities.

A quantitative evaluation of our method and the Least Squares approach is presented in Table 1. The estimated normal maps are shown in Fig. 2(b),(c). We use the RGB channel to encode the 3 spatial components (XYZ) of the normal





**Fig. 2. Specular scene.** 40 different images of Caesar were generated using the Cook-Torrance model for specularities. (a) Ground truth normal map with reference sphere. (b) and (c) show the surface normals recovered by our method and LS, respectively. (d) and (e) show the pixel-wise angular error w.r.t. the ground truth.

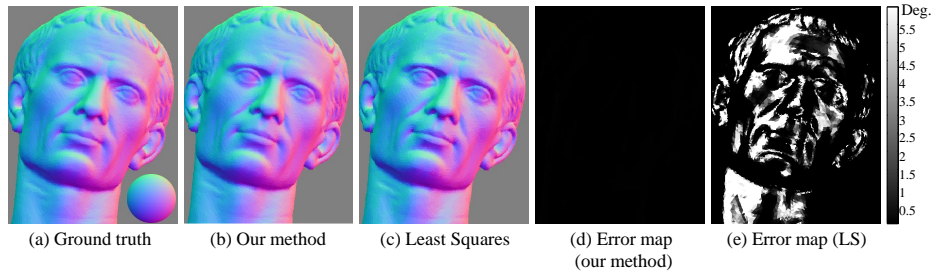
Object	Mean error (in degrees)		Max. error (in degrees)		Avg. % of corrupted pixels	
	LS	Our method	LS	Our method	Shadow	Specularity
Sphere	0.99	$5.1 \times 10^{-3}$	8.1	<b>0.20</b>	18.4	16.1
Caesar	0.96	$1.4 \times 10^{-2}$	8.0	<b>0.22</b>	20.7	13.6
Elephant	0.96	$8.7 \times 10^{-3}$	8.0	<b>0.29</b>	18.1	16.5

**Table 1. Specular scene.** Statistics of angle error in the normals for different objects. In each case, 40 images were used. In the rightmost column, we indicate the average percentage of pixels corrupted by attached shadows and specularities in each image.

map for display purposes. The error is measured in terms of the angular difference between the ground truth normal and the estimated normal at each pixel location. The pixel-wise error maps are shown in Fig. 2(d),(e). From the mean and the maximum angular error (in degrees) in Table 1, we see that our method is much more accurate than the LS approach. This is because specularities introduce large magnitude errors to a small fraction of pixels in each image whose locations are unknown. The LS algorithm is not robust to such corruptions while our method can correct these errors and recover the underlying rank-3 structure of the matrix. The column on the extreme right of Table 1 indicates the average percentage of pixels in each image (averaged over all images) that were corrupted by shadows and specularities, respectively. We note that even when more than 30% of the pixels are corrupted by shadows and specularities, our method can efficiently retrieve the surface normals.

**b. Textured scene.** We also test our method using a textured scene. Like the traditional photometric stereo approach, our method does not have a dependency on the albedo distribution and works well on such scenes.

We use 40 images of Caesar for this experiment with each image generated under a different lighting condition (see Fig. 1(d) for example input image). The estimated normal maps as well as the pixel-wise error maps are shown in Fig. 3. We provide a quantitative comparison in Table 2 with respect to the ground-truth normal map. From the mean and maximum angular errors, it is evident that our method performs much better than the LS approach in this scenario.



**Fig. 3. Textured scene with specularities.** 40 different images of Caesar were generated with texture, using the Cook-Torrance model for specularities. (a) Ground truth normal map with reference sphere. (b) and (c) show the surface normals recovered by our method and LS, respectively. (d) and (e) show the pixel-wise angular error w.r.t. the ground truth.

Object	Mean error (in degrees)		Max error (in degrees)	
	LS	Our method	LS	Our method
Caesar	2.4	<b>0.016</b>	32.2	<b>0.24</b>

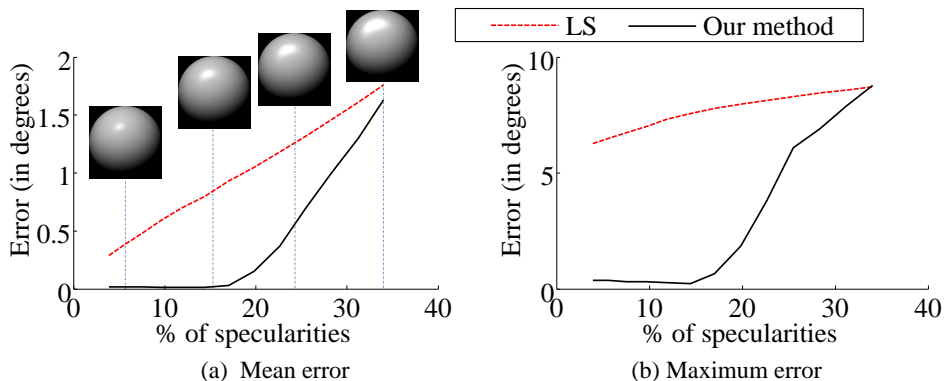
**Table 2. Textured scene with specularities:** Statistics of angle errors. We use 40 images under different illuminations.

**c. Effect of the number of input images.** In the above experiments, we have used images of the object under 40 different illuminations. In this experiment, we study the effect of the number of illuminations used. In particular, we would like to find out empirically the minimum number of images required for our method to be effective. For this experiment, we generate images of Caesar using the Cook-Torrance reflectance model, where the lighting directions are generated at random. The mean percentage of specular pixels in the input images is maintained approximately constant at 10%. The angular difference between the estimated normal map and the ground truth is used as a measure of accuracy of the estimate.

Num of images		5	10	15	20	25	30	35	40
Mean error (in degrees)	LS	<b>4.5</b>	0.52	0.51	0.53	0.62	0.59	0.59	0.57
	Our method	15.1	<b>0.23</b>	<b>0.036</b>	<b>0.026</b>	<b>0.015</b>	<b>0.019</b>	<b>0.017</b>	<b>0.013</b>
Max. error (in degrees)	LS	<b>88.2</b>	<b>34.5</b>	<b>13.7</b>	9.0	8.4	7.6	7.6	7.0
	Our method	127.9	56.6	25.6	<b>5.8</b>	<b>0.42</b>	<b>0.48</b>	<b>0.37</b>	<b>0.37</b>

**Table 3. Effect of number of input images.** We use synthetic images of Caesar under different lighting conditions. The number of illuminations is varied from 5 to 40. The angle error is measured with respect to the ground truth normal map. The illuminations are chosen at random, and the error has been averaged over 20 different sets of illumination.

We present the experimental results in Table 3. We observe that with 5 input illuminations, estimates of both algorithms are very inaccurate but our method is worse than LS. However, when the number of illuminations is larger than 10, we observe that the mean error in the LS estimate becomes higher than that our method. Upon increasing the number of images further, the proposed method consistently outperforms the LS approach. If the number of input images is less



**Fig. 4. Effect of increasing size of specular lobes.** We use synthetic images of Caesar under 40 randomly chosen lighting conditions. (a) Mean angular error, (b) Maximum angular error w.r.t. the ground truth. The illuminations are chosen at random, and the error has been averaged over 10 different sets of illumination. (a) contains illustrations of increasing size of specular lobe.

than 20, then the maximum error in the LS estimate is smaller than that of our method. However, our method performs much better when at least 25 different illuminations are available. Thus, the proposed technique performs significantly better as the number of input images increases.

**d. Varying amount of specularity.** From the above experiments, it is clear that the proposed technique is quite robust to specularities in the input images when compared to the LS method. In this experiment, we empirically determine the maximum amount of specularity that can be handled by our method. We use the Caesar scene under 40 randomly chosen illumination conditions for this experiment. On an average, about 20% of the pixels in each image is corrupted by attached shadows. We vary the size of the specular lobe in the input images (as illustrated in Fig. 4(a)), thereby varying the number of corrupted pixels. We compare the accuracy of our method against the LS technique using the angular error of the estimates with respect to the ground-truth.

The experimental results are illustrated in Fig. 4. We observe that our method is very robust when up to 16% of all pixels in the input images are corrupted by specularities. The LS method, on the other hand, is extremely sensitive to even small amounts of specularities in the input images. The angular error in the estimates of both methods rises as the size of the specular lobe increases.

**e. Enhancing performance by better choice of  $\lambda$ .** We recall that  $\lambda$  is a weighting parameter in our formulation given by Eq. (10). In all the above experiments, we have fixed the value of the parameter  $\lambda = 1/\sqrt{m}$ , as suggested by [14]. While this choice promises a certain degree of error correction, it may be possible to correct larger amounts of corruption by choosing  $\lambda$  appropriately, as demonstrated in [26] for instance. Unfortunately, the best choice of  $\lambda$  depends on the input images, and cannot be determined analytically.

We demonstrate the effect of the weighting parameter  $\lambda$  on a set of 40 images of Caesar used in the previous experiments. In this set of images, approximately 20%

of the pixels are corrupted by attached shadows and about 28% by specularities. We choose  $\lambda = C/\sqrt{m}$ , and vary the value of  $C$ . We evaluate the results using angular error with respect to the ground-truth normal map. We observe from Table 4 that the choice of  $C$  influences the accuracy of the estimated normal map. For real-world applications, where the data is typically noisy, the choice of  $\lambda$  could play an important role in the efficacy of our method.

$C$	1.0	0.8	0.6	0.4
Mean error (in degrees)	1.42	0.78	0.19	0.029
Max. error (in degrees)	8.78	8.15	1.86	0.91

**Table 4. Handling more specularities by appropriately choosing  $\lambda$ .** We use 40 images of Caesar under different lighting conditions with about 28% specularities and 20% shadows, and set  $\lambda = C/\sqrt{m}$ .

**f. Computation.** The core computation of our method is solving a convex program Eq. (10). For the specular Caesar data (Fig. 1(b)) with 40 images of  $450 \times 350$  resolution, a single-core MATLAB implementation of our method takes about 7 minutes on a Macbook Pro with a 2.8 GHz Core 2 Duo processor and 4 GB memory, as against 42 seconds taken by the LS approach. While our method is slower than the LS approach, it is much more accurate in a wide variety of scenarios and is more efficient than other existing methods (*e.g.* [10]).

#### 4.2 Qualitative evaluation with real images

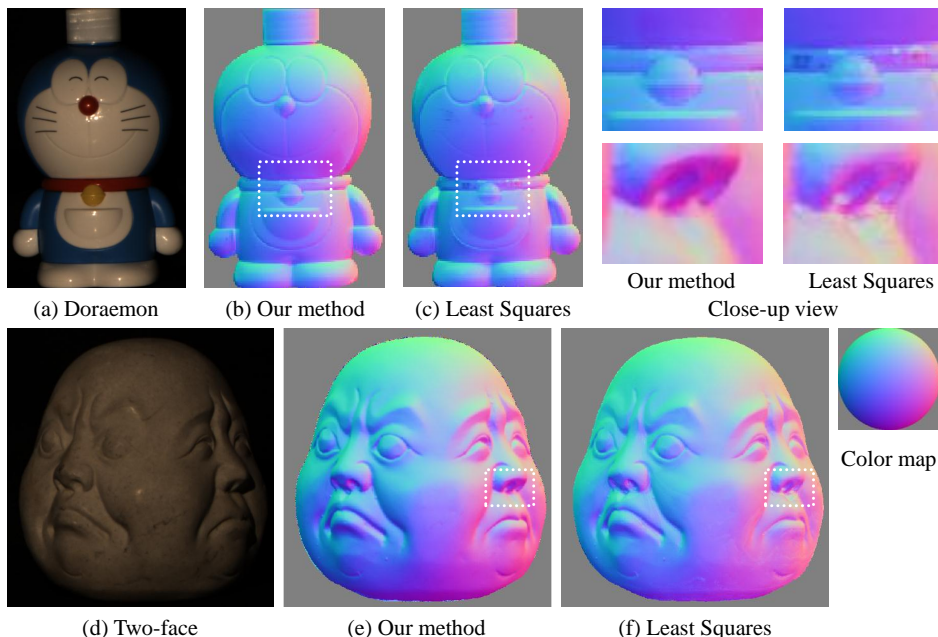
We now test our algorithm on real images. We use a set of 40 images of a toy Doraemon and Two-face taken under different lighting conditions (see Fig. 5(a), (d)). A glossy sphere was placed in the scene for light source calibration when capturing the data. We used a Canon 5D camera with the RAW image mode without Gamma correction. These images present new challenges to our algorithm. In addition to shadows and specularities, there is potentially additional noise inherent to the acquisition process as well as possible deviations from the idealistic Lambertian model illuminated by distant lights. In this experiment, we use a threshold of 0.01 to detect shadows in images.<sup>4</sup> We also found experimentally that setting  $\lambda = 0.3/\sqrt{m}$  works well for these datasets.

Since the ground truth normal map is not available for these scenes, we compare our method and the LS approach by visual inspection of the output normal maps shown in Fig. 5(b),(c),(e),(f). We observe that the normal map estimated by our method appears smoother and hence, more realistic. This can be observed particularly around the necklace area in Doraemon and nose area in Two-face (see Fig. 5) where the LS estimate exhibits some discontinuity in the normal map.

## 5 Discussion and Future Work

In this paper, we have presented a new computational framework to aid in photometric stereo. We have formulated the basic photometric stereo problem as a convex optimization problem that can be solved efficiently. The efficacy of our method is demonstrated using synthetic and real images. The biggest advantage

<sup>4</sup> All pixels are normalized to have intensity between 0 and 1.



**Fig. 5. Qualitative comparison on real data.** We use images of Doraemon and Two-face taken under 40 different lighting conditions to qualitatively evaluate the performance of our algorithm against the LS approach. (a),(d) Sample input images. (b),(e) Normal map estimated by our method. (c),(f) Normal map estimated by Least Squares. Close-up views of the dotted rectangular areas (top-right) where the normal map estimate of our method is much more smoother and realistic than that of Least Squares.

of the proposed technique is its ability to handle shadows, specularities, and other kinds of large-magnitude, non-Gaussian errors in the data.

The new framework also opens up several avenues for future research. Currently, we assume that all the images are noise-free and perfectly aligned with each other at the pixel level. However, in real world scenarios, small noise and misalignment are commonplace in any data acquisition process. By exploring the low-rank structure described in this work, we believe that the proposed technique can be extended to simultaneously handle small noise and misalignment in the input images.

## References

1. Woodham, R.: Photometric method for determining surface orientation from multiple images. *Optical Engineering* **19** (1980) 139–144
2. Silver, W.: Determining shape and reflectance using multiple images. Master’s thesis, MIT (1980)
3. Shashua, A.: Geometry and photometry in 3d visual recognition. Ph.D dissertation, Department of Brain and Cognitive Science, MIT (1992)
4. Belhumeur, P., Kriegman, D.: What is the set of images of an object under all possible lighting conditions? In: *Proc. of CVPR*. (1996) 270–277

5. Basri, R., Jacobs, D.: Lambertian reflectance and linear subspaces. *PAMI* **25** (2003) 218–233
6. Georghiades, A.S., Kriegman, D.J., Belhumeur, P.N.: From few to many: illumination cone models for face recognition under variable lighting and pose. *PAMI* **23** (2001) 643–660
7. Jolliffe, I.: *Principal Component Analysis*. Springer-Verlag (1986)
8. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model-fitting with applications to image analysis and automated cartography. *Communications of the ACM* **24** (1981) 381–395
9. C. Hernández, G.V., Cipolla, R.: Multi-view photometric stereo. *PAMI* **30** (2008) 548–554
10. Miyazaki, D., Hara, K., Ikeuchi, K.: Median photometric stereo as applied to the segonko tumulus and museum objects. *IJCV* **86** (2010) 229–242
11. Mukaigawa, Y., Ishii, Y., Shakunaga, T.: Analysis of photometric factors based on photometric linearization. *JOSA* **24** (2007) 3326–3334
12. Hayakawa, H.: Photometric stereo under a light source with arbitrary motion. *JOSA* **11** (1994) 3079–3089
13. Knill, D.C., Mamassian, P., Kersten, D.: The geometry of shadows. *JOSA* **14** (1997) 3216–3232
14. Wright, J., Ganesh, A., Rao, S., Peng, Y., Ma, Y.: Robust principal component analysis: Exact recovery of corrupted low-rank matrices by convex optimization. In: *Proc. of Neural Information Processing Systems*. (2009)
15. Chandrasekaran, V., Sanghavi, S., Parrilo, P.A., Willsky, A.S.: Sparse and low-rank matrix decompositions. In: *Proc. of IFAC Symp. on System Identification*. (2009)
16. Lin, Z., Chen, M., Wu, L., Ma, Y.: The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Preprint (2009)
17. Recht, B., Fazel, M., Parillo, P.: Guaranteed minimum rank solution of matrix equations via nuclear norm minimization. to appear in *SIAM Review* (2008)
18. Candès, E., Recht, B.: Exact matrix completion via convex optimization. *Found. of Comput. Math.* (2008)
19. Candès, E., Tao, T.: The power of convex relaxation: Near-optimal matrix completion. to appear in *IEEE Transactions on Information Theory* (2009)
20. Ganesh, A., Lin, Z., Wright, J., Wu, L., Chen, M., Ma, Y.: Fast algorithms for recovering a corrupted low-rank matrix. In: *Proc. of CAMSAP*. (2009)
21. Toh, K., Yun, S.: An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. *Pacific Journal of Optimization* (accepted) (2009)
22. Bertsekas, D.P.: *Nonlinear Programming*. Athena Scientific (2004)
23. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problem. *SIAM Journal on Imaging Sciences* (2008) 183–202
24. <http://www-roc.inria.fr/gamma/gamma/download/download.php>: 3D meshes research database by INRIA gamma group (2008)
25. Cook, R.L., Torrance, K.E.: A reflectance model for computer graphics. *SIGGRAPH Comput. Graph.* **15** (1981) 307–316
26. Ganesh, A., Wright, J., Li, X., Candès, E., Ma, Y.: Dense error correction for low-rank matrices via principal component pursuit. In: *Proc. of ISIT*. (2010)