Noname manuscript No.
(will be inserted by the editor)

Solving Uncalibrated Photometric Stereo Using Fewer Images by Jointly Optimizing Low-rank Matrix Completion and Integrability

S. Sengupta[1], H. Zhao[1], W. Forkel[2], R. Basri[3], T. Goldstein[4], D. Jacobs[1]

Received: date / Accepted: date

Abstract We introduce a new, integrated approach to uncalibrated photometric stereo. We perform 3D reconstruction of Lambertian objects using multiple images produced by unknown, directional light sources. We show how to formulate a single optimization that includes rank and integrability constraints, allowing also for missing data. We then solve this optimization using the Alternate Direction Method of Multipliers (ADMM). We conduct extensive experimental evaluation on real and synthetic data sets. Our integrated approach is particularly valuable when performing photometric stereo using as few as 4-6 images, since the integrability constraint is capable of improving estimation of the linear subspace of possible solutions. We show good improvements over prior work in these cases.

Keywords Photometric Stereo · 3D reconstruction · Low rank optimization

1 Introduction

Uncalibrated photometric stereo (UPS) is the problem of recovering the 3D shape of an object and associated lighting conditions, given images taken with varying, unknown illumination. In this work we replace the existing pipeline for solving UPS with an integrated approach. This paper, like much prior work [14, 4, 33, 31, 2, 11, 22], focuses on Lambertian objects illuminated by a single distant point light source in each image. Existing methods, pioneered by [14], formulate UPS as the problem of finding a low-rank factorization of the measurements. Specifically, given m images each with p pixels, let M denote the $m \times p$ matrix containing the pixel intensities. These methods optimize

$$\min_{\hat{M}} \|\hat{M} - M\|_F^2 \quad \text{s.t.} \quad \text{rank}(\hat{M}) = 3. \tag{1}$$

This problem can be solved by SVD, from which we produce a family of solutions, each consisting of a set of light sources, albedos, and surface normals. These solutions are related by a 3×3 ambiguity matrix. The surface normals provided by SVD are in general inconsistent with the partial derivatives of the surface (i.e. they are not *integrable*). Consequently, existing methods apply an additional sequence of steps aimed at reducing the ambiguity and fitting a surface to the recovered normals.

In this paper we propose instead to optimize:

$$\min_{\hat{M}} \|\hat{M} - M\|_F^2 \tag{2}$$

s.t. $\hat{M}$ is rank 3 and produced by an integrable surface.

Eq. (1) optimizes over rank 3 matrices, which can represent sets of images produced by any set of surface normals. In contrast, in (2) we optimize over only those rank 3 matrices that correspond to integrable surfaces.

Intuitively, a single optimization over all constraints may produce a better global optimum than a sequence of optimizations in which constraints are used one at a time to increasingly narrow the solution (see illustration in Figure 1). Specifically in UPS the measurement matrix may contain many errors due to shadows and specular effects. Therefore, while in theory UPS can be solved with as few as three images, SVD can properly handle these modeling errors only when many images are supplied. Indeed, current methods [2, 11] typically use 10 or more images. With fewer images SVD results tend to provide noisy solutions. Our

S. Sengupta
[1]Center for Automation Research, University of Maryland, College Park, MD, USA.
E-mail: sengupta@umd.edu

[2]TU Dresden, Germany.
[3]Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel.
[4]Department of Computer Science, University of Maryland, College Park, MD, USA.

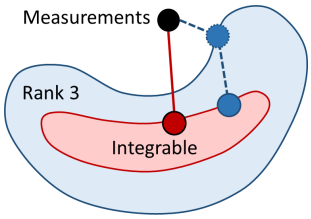


Fig. 1: An illustration of our approach. Blue represents the set of rank 3 matrices, while red represents the subset of those that correspond to integrable surfaces. Our optimization seeks to find the integrable matrix (red dot) that is closest to the measurements (black dot). If instead we first find the nearest rank 3 matrix and then select an integrable matrix (the blue dots) we may produce a suboptimal solution.

method incorporates integrability into this estimation, providing valuable additional constraints that help us to find a better subspace in the presence of fewer images with noisy estimates. In presence of large number of images SVD based methods can handle noises and outliers to solve an over-constrained problem. However for fewer images, a joint optimization based on rank constraint can obtain a better subspace. Our experiments indicate that our method can produce reasonable reconstructions with as few as 4 images and good reconstruction with 6 images, significantly improving over state-of-the-art methods with these few images.

For our approach we optimize a cost function based on (2) over the surface, lighting, normals, and (restored) error-free observations. The cost ensures that normals and lighting are consistent with the measurements, which must have low rank. We use constraints that ensure integrability. This is somewhat tricky because rank constraints apply to the measurements while integrability constraints apply to the normals. We show that by constructing a rank 3 matrix that contains normals, measurements, and lighting, we can impose the rank and integrability constraints together. Specifically, we use a truncated nuclear norm approach [15] to enforce the rank constraint, while integrability is represented by linear equalities. This leads to a single non-convex problem that we solve using a series of Alternating Direction Method of Multipliers (ADMM) operations [5, 13].

Our formulation allows us to easily account for missing data in the measurement matrix. This commonly occurs when pixels are dark due to shadows, or saturated due to specularities. In some of the prior approaches, this can be solved with a preprocessing step, which may lead to a pipeline with yet another optimization [31]. We handle missing data using matrix completion based on the rank constraint. We initialize our optimization using prior approaches, since non-convex optimization requires a good initialization.

2 Background and Previous Work

In this section we introduce in detail the problem of uncalibrated photometric stereo for Lambertian objects and review past work. We assume that we view an object in multiple images from a fixed viewpoint. In each image the object is illuminated by a single, distant point light source. We represent lighting in image i with $l_i \in R^3$, in which the direction of l_i represents the direction to the lighting, and $\|l_i\|$ represents its magnitude. We represent the object using a set of surface normals $\hat{n}_j \in R^3$, and albedos $\rho_j \in R$ for each pixel. We then obtain images with the equation:

$$M_{ij} = \max(0, \rho_j l_i^T \hat{n}_j) \tag{3}$$

where M_{ij} represents the j-th pixel of the i-th image. We define the surface normal $\hat{n}_j = \frac{n_j}{\|n_j\|}$, $n_j = (-z_x, -z_y, 1)^T$, where z_x and z_y represent partial derivatives of the surface $z(x, y)$ at pixel j. Negative values of $\rho_j l_i^T \hat{n}_j$ are set to 0; these appear as attached shadows.

We now describe the creation of all images using matrix operations. We define S to be a $3 \times p$ matrix in which column j contains $\rho_j \hat{n}_j$. Given m images, we can stack the light into the matrix L of dimension $m \times 3$, where each row denotes one light per image. We concatenate all the images to form an observation matrix M of dimension $m \times p$, where p is the number of pixels. Now, in the absence of shadows, we can write the equation of UPS as:

$$M = LS. \tag{4}$$

Classical work on photometric stereo (e.g. [30], see a recent review in [1]) has assumed that known lighting is obtained by careful calibration. With L known, (4) can be solved as a linear least squares problem. A more general and challenging case is unconstrained photometric stereo, in which the L is unknown. A common approach, which we use as a baseline algorithm, follows the steps in Algorithm 1.

Algorithm 1 Baseline

Input : M
Output : Z
Factorization : Perform SVD on M to obtain light and scaled surface normals $M = \tilde{L}\tilde{S}$ [14].
Integrability : Follow Yuille and Snow [33] to resolve ambiguity after the factorization using integrability. In $M = LS = \tilde{L}A^{-1}A\tilde{S}$, we solve for A, such that $S = A\tilde{S}$ approximately forms a set of integrable surface normals.
Depth Reconstruction : Obtain the depth map Z from the set of integrable surface normals S as, e.g, in [3].

Belheumer *et al.* [4] showed that in UPS the integrable set of surface normals can only be recovered up to a Generalized Bas-Relief transformation (GBR). A number of recent papers have concentrated on methods of solving the GBR ambiguity. Researchers have used priors on the albedo distribution [2], reflectance extrema [11], total variation norm

[24], grouping based on image appearance and color [25], inter-reflections [9], isotropy and symmetries [29], and specularity [10] as constraints while solving for the GBR. All of these methods have first used the above mentioned baseline described in Algorithm 1 to obtain a solution up to the GBR.

Recent works have explored a variety of other research directions in photometric stereo. Mecca *et al.* [17] proposed an integrated, PDE based approach to calibrated photometric stereo that uses a mere two images under perspective projection. It is not clear how to extend this to uncalibrated photometric stereo. Basri *et al.* [3] extended the baseline to handle multiple light sources in each image using a spherical harmonics formulation. Chandraker *et al.* [8] proposed a method to handle attached and cast shadows in the case of multiple light sources per image. In [27] the authors determine the visibility subspace for a set of images to remove the cast and attached shadows for performing UPS. Various works have addressed non-Lambertian materials (e.g., Georghiades *et al.* [12] and Okabe *et al.* [20]).

In the context of Lambertian UPS, Georghiades *et al.* [12] proposed to remove shadows and specularities and recover the missing pixel values using matrix completion algorithms, e.g., using the damped Wiberg [21] or Cabral's algorithm [6]. Wu *et al.* [31] proposed a Robust PCA formulation for calibrated Photometric Stereo. Their approach seeks a low-rank (not necessarily rank 3) approximation to M while removing outlier pixels (corresponding to shadows and specularities). Oh *et al.* [18, 19] applied Robust PCA in the context of calibrated photometric stereo, replacing the Nuclear Norm with a Truncated Nuclear Norm (TNN) regularizer [15]. In [11], Favaro *et al.* have used Robust PCA as preprocessing to the baseline algorithm for UPS.

3 Our Approach

In this section we introduce our integrated formulation that enforces integrability of surface normals in solving the uncalibrated photometric stereo problem. We recall from (4) that the measurement matrix M can be factored into $M = LS$. To access the derivatives of $z(x, y)$ we write S as a product

$$S = N\Lambda, \tag{5}$$

where N is a $3 \times p$ matrix whose j'th column is $n_j = (-z_x, -z_y, 1)^T$ and $\Lambda = \mathrm{diag}(\lambda_i, \lambda_2, ..., \lambda_p)$ with $\lambda_j = -\rho_j / \|n_j\|$. We next define the matrix:

$$X = \begin{bmatrix} X^I & X^N \\ X^L & X^M \end{bmatrix} = \begin{bmatrix} I & N \\ L & M\Lambda^{-1} \end{bmatrix}, \tag{6}$$

where X is $(3 + m) \times (3 + p)$. The matrices X, Λ, and the depth values ($z(x, y)$) form the unknowns in our optimization. Note that, because $LN = M\Lambda^{-1}$, the following holds for any 3×3 non-degenerate matrix A

$$X = \begin{bmatrix} A^{-1} \\ LA^{-1} \end{bmatrix} \begin{bmatrix} A & AN \end{bmatrix}. \tag{7}$$

This shows that X is rank 3. The matrix A represents a linear ambiguity. However, forcing the normals in N to be integrable will reduce this ambiguity to the GBR.

To force integrability we denote by $\mathbf{z} = (z_1, ..., z_p)^T$ the vector of unknown depth values and require

$$X^N = \begin{bmatrix} D_x\mathbf{z}, \, D_y\mathbf{z}, \, -\mathbf{1} \end{bmatrix}^T, \tag{8}$$

where D_x, D_y denote respectively the x- and y-derivative operators and $\mathbf{1}$ denotes the vector of all 1's.

Additional constraints are obtained by noticing that, because $0 \leq \rho_j \leq 1$ and $\|n_j\| \geq 1$,

$$-1 \leq \lambda_j \leq 0 \tag{9}$$

and

$$X^I = I_{3\times 3}. \tag{10}$$

We are now ready to define our optimization function. Let W be a binary, $m \times p$ matrix so that $W_{ij} = 0$ if M_{ij} is missing and $W_{ij} = 1$ otherwise, and let

$$f_{data}(X, \Lambda) = \frac{1}{2}||W \odot (M - X^M\Lambda)||_F^2, \tag{11}$$

where $\odot$ denotes element-wise multiplication. Then (2) can be written as

$$\begin{aligned} \min_{X,\Lambda,\mathbf{z}} \quad & f_{data}(X, \Lambda) \\ \text{s.t.} \quad & \mathrm{rank}(X) = 3, \ (8), \ (9), \text{and } (10). \end{aligned} \tag{12}$$

Handling the rank-3 constraint: Enforcing the non-convex constraint $\mathrm{rank}(X) = 3$ can be challenging. In the context of matrix completion a recent paper [15] proposed using the Truncated Nuclear Norm (TNN) regularization term:

$$f_{tnn}(X) = ||X||_* - \sum_{k=1}^{3} \sigma_k(X), \tag{13}$$

where $||X||_*$ denotes the nuclear norm of X and $\sigma_k(X)$ is the k-th largest singular value of X. Clearly, $f_{tnn}(X) = 0$ if and only if $\mathrm{rank}(X) \leq 3$. We use f_{tnn} as a regularizer and solve

$$\begin{aligned} \min_{X,\Lambda,\mathbf{z}} \quad & f_{data}(X, \Lambda) + c\, f_{tnn}(X) \\ \text{s.t.} \quad & (8), \ (9), \text{and } (10), \end{aligned} \tag{14}$$

where c is a preset scalar.

There are several different ways of handling the rank constraint. One such technique is to use explicit factorization of M into L, N and ρ with the integrability constraint

over N. This tri-linear decomposition can be solved with alternate steps. Alternate optimizations are sometimes slow to converge. Another technique widely used in matrix completion is Nuclear Norm (NN) relaxation : $\sum \sigma_k(X)$. Even though NN relaxation is convex, TNN [15] regularization is shown to outperform the former for matrix completion problems. An alternate approach is to directly impose $rank(X) = 3$ with ADMM. We have implemented both NN relaxation and $rank(X) = 3$ and empirically observed TNN to outperform both of them. Thus in this paper we have used TNN regularization to handle the rank constraint.

4 Optimization using ADMM

In this section we introduce a method for solving (14). This is a challenging problem because both f_{data} and f_{tnn} are non-convex. Specifically, f_{data} (11) is bilinear in X and Λ, while f_{tnn} (13) is a difference between two convex functions. Our solution is based on a nested iteration in which the outer loop uses majorization to decrease f_{tnn} whereas the inner loop uses the scaled ADMM with alternation to decrease f_{data}.

Outer loop: Following [15] at each iteration of the outer loop we replace $f_{tnn}(X)$ with a majorizer. Specifically, at iteration k let $X^{(k)} = U\Sigma V^T$ be the singular value decomposition of $X^{(k)}$, and let U_3 (and V_3) be the matrices containing the left (right) singular vectors corresponding to the three largest singular values of $X^{(k)}$. U_3 and V_3 are determined in the outer loop and are held constant throughout the inner loop. We then define

$$f^{(k)}_{maj}(X) = \|X\|_* - \text{trace}(U_3^T X V_3). \tag{15}$$

It was shown in [15] that $f^{(k)}_{maj}(X) \geq f_{tnn}(X)$ for all X and that $f^{(k)}_{maj}(X^{(k)}) = f_{tnn}(X^{(k)})$, and so decreasing f_{maj} leads to decreasing f_{tnn}.

Inner loop: In the inner loop we seek to minimize

$$\begin{aligned} \min_{X,\Lambda,\mathbf{z}} \quad & f_{data}(X,\Lambda) + c f^{(k)}_{maj}(X) \\ \text{s.t.} \quad & (8),\ (9), \text{and } (10), \end{aligned} \tag{16}$$

We use scaled ADMM, a variant of the augmented Lagrangian method that splits the objective function and aims to solve the different subproblems separately. We maintain a second copy of X, which we denote by Y and form the augmented Lagrangian of (16) as follows

$$\begin{aligned} \max_{\Gamma} \min_{X,\Lambda,\mathbf{z},Y} \quad & \frac{1}{2}||W \odot (M - X^M \Lambda)||_F^2 + \\ & c\left(||Y||_* - \text{trace}(U_3^T Y V_3)\right) + \frac{\tau}{2}||Y - X + \Gamma||_F^2 \\ s.t.\ X^I = I_{3\times 3},\ & -1 \leq \lambda_j \leq 0\ \forall j,\ X^N = \left[D_x\mathbf{z},\ D_y\mathbf{z},\ -\mathbf{1}\right]^T, \end{aligned} \tag{17}$$

where $||Y - X + \Gamma||_F^2$, denotes the Lagrangian penalty; τ is a constant, and Γ is a matrix of Lagrange multipliers the same size as X that is updated by the ADMM steps [5, 13]. We next describe the ADMM steps (applied iteratively).

Step 1: Solving for $(X, \Lambda, \mathbf{z})$.
In each iteration, k, we solve the following sub-problems:

1. Optimize w.r.t. X^I: $X^{I\,(k+1)} = I_{3\times 3}$.
2. Optimize w.r.t. X^L:
$$\begin{aligned} X^{L\,(k+1)} &= \underset{X^L}{\text{argmin}} ||Y^{L\,(k)} - X^L + \Gamma^{L\,(k)}||_F^2 \\ &= Y^{L\,(k)} + \Gamma^{L\,(k)}. \end{aligned} \tag{18}$$
3. Optimize w.r.t. X^N and $\mathbf{z}$:
$$\begin{aligned} (X^{N\,(k+1)}, \mathbf{z}^{(k+1)}) &= \underset{X^N,\mathbf{z}}{\text{argmin}} ||Y^{N\,(k)} - X^N + \Gamma^{N\,(k)}||_F^2 \\ &\text{s.t. } X^N = \left[D_x\mathbf{z},\ D_y\mathbf{z},\ -\mathbf{1}\right]^T. \end{aligned} \tag{19}$$
The problem is solved by setting the third row of $X^{N\,(k+1)}$ to $-\mathbf{1}$ and by substituting $D_x\mathbf{z}$ and $D_y\mathbf{z}$ for the first two rows of X^N in the objective, obtaining linear least squares equations in $\mathbf{z}$ that can be solved directly.
4. Optimize w.r.t. X^M and Λ:
$$\begin{aligned} (X^{M\,(k+1)}, \Lambda^{(k+1)}) =& \underset{X^M,\Lambda}{\text{argmin}} \frac{1}{2}\|W \odot (M - X^M\Lambda)\|_F^2 \\ &+ \frac{\tau}{2}\|Y^{M\,(k)} - X^M + \Gamma^{M\,(k)}\|_F^2 \\ \text{s.t.} \quad & -1 \leq \lambda_j \leq 0\ \forall j. \end{aligned}$$

We will separate this into the known and unknown pixels based on W. For an **unknown pixel** j in frame i ($W_{ij} = 0$) the first term vanishes and the minimization only determines the respective entry of X^M so that:

$$X^{M\,(k+1)}_{ij} = Y^{M\,(k)}_{ij} + \Gamma^{M\,(k)}_{ij}. \tag{20}$$

For the **known pixels**, since Λ is diagonal we can write these equations separately for each column j (corresponding to the j-th pixel):

$$\begin{aligned} (X^{M\,(k+1)}_j, \lambda^{(k+1)}_j) =& \underset{X^M_j,\lambda_j}{\text{argmin}} \frac{1}{2}\|(W_j \odot (M_j - \lambda_j X^M_j)\|_2^2 \\ &+ \frac{\tau}{2}\|Y^{M\,(k)}_j - X^M_j + \Gamma^{M\,(k)}_j\|_2^2 \\ \text{s.t.} \quad & -1 \leq \lambda_j \leq 0. \end{aligned} \tag{21}$$

The problem (21) is non-convex. We will solve it with *alternate optimization*. X^M and Λ are updated by the following steps until convergence.
$\mathbf{X^M}$: Let $\tilde{M}_j = W_j \odot M_j$, $\tilde{X}_j = W_j \odot X^M_j$ and $\tilde{A}^{M\,(k)}_j = W_j \odot (Y^{M\,(k)}_j + \Gamma^{M\,(k)}_j)$. Then,

$$\begin{aligned} \tilde{X}_j &= \underset{\tilde{X}_j}{\text{argmin}} \frac{1}{2}||\tilde{M}_j - \lambda_j\tilde{X}_j||_2^2 + \frac{\tau}{2}||\tilde{A}^{M\,(k)}_j - \tilde{X}_j||_2^2 \\ &= \frac{\lambda_j\tilde{M}_j + \tau\tilde{A}^{M\,(k)}_j}{\lambda_j^2 + \tau}. \end{aligned} \tag{22}$$

$\mathbf{\Lambda}$:

$$\lambda_j = \underset{\lambda_j}{\operatorname{argmin}} \frac{1}{2}\|\tilde{M}_j - \lambda_j \tilde{X}_j\|_2^2 \text{ s.t. } -1 \leq \lambda_j \leq 0,$$
$$= \min(0, \max(-1, \tilde{X}_j^T \tilde{M}_j / \|\tilde{X}_j\|_2^2)). \quad (23)$$

Step 2: Solving for Y. Solving for Y requires a solution to

$$Y^{(k+1)} = \underset{Y}{\operatorname{argmin}}\, c\left(\|Y\|_* - \operatorname{trace}(U_3^T Y V_3)\right) + \frac{\tau}{2}\|Y - X^{(k+1)} + \Gamma^{(k)}\|_F^2. \quad (24)$$

Below we show that this problem can be solved in closed form by applying the shrinkage operator, obtaining

$$Y^{(k+1)} = D_{c/\tau}(X^{(k+1)} - \Gamma^{(k)} + \frac{c}{\tau} U_3 V_3^T), \quad (25)$$

where the shrinkage operator $D_t(.)$ is defined as follows. For a scalar s we define $D_t(s) = \operatorname{sign}(s) \times \max(|s| - t, 0)$. For a diagonal matrix $S = \operatorname{diag}(s_1, s_2, ...)$ with non-negative entries we define $D_t(S) = \operatorname{diag}(D_t(s_1), D_t(s_2), ...)$. Finally, for a general matrix Υ, let $\Upsilon = \tilde{U} S \tilde{V}^T$ be its singular value decomposition, then $D_t(\Upsilon) = \tilde{U} D_t(S) \tilde{V}^T$.

To derive (25), we rewrite (24) as:

$$Y^{(k+1)} = \underset{Y}{\operatorname{argmin}} \|Y\|_* + \frac{\tau}{2c}\|Y - X^{(k+1)} + \Gamma^{(k)} - \frac{c}{\tau} U_3 V_3^T\|_F^2 - T, \quad (26)$$

where $T = \operatorname{trace}(V_3 U_3^T (X^{(k+1)} - \Gamma^{(k)})) + \frac{c}{2\tau}\|U_3 V_3^T\|_F^2$ is independent of Y. Equation (26) is of the general form $\min_Y \|Y\|_* + \frac{1}{2t}\|Y - C\|_F^2$, for which the solution is $D_t(C)$, as is shown in [7], implying (25).

Step 3: Update of Γ. The matrix Γ contains Lagrange multipliers that are used in the saddle-point formulation (17) to enforce the equality constraint $X = Y$. The following update is a gradient ascent step that acts to maximize the augmented Lagrangian (17) for Γ. For details, see [5, 13].

$$\Gamma^{(k+1)} = \Gamma^{(k)} + (Y^{(k+1)} - X^{(k+1)}). \quad (27)$$

The entire optimization process is listed in Algorithm 2. We will make the code available.

Algorithm 2 TNN formulation solved with ADMM

Input: M, W.
Output: X, $\mathbf{z}$.
Initialization: Initialize X^L and X^N by running Baseline algorithm (without resolving GBR). Initialize $X^M = -M$, $\Lambda = -I$, and $c = 1$. Set $X^{(0)} = X$, $Y = X$, $\Gamma = 0$, and $\tau = 1$.
$k = 0$.
while not converged **do**
  Perform SVD over $X^{(k)}$ to obtain U_3 and V_3.
  Run ADMM:
  while not converged **do**
    Update of X, $\mathbf{z}$ and Λ.
    Update $X^{I(k+1)} = I_{3\times 3}$.
    Update $X^{L(k+1)}$ using (18).
    Update $X^{N(k+1)}$ and $\mathbf{z}$ using (19).
    while not converged **do**
      for each pixel j **do**
        Update $X_j^{M(k+1)}$ using (22) and $\lambda_j^{(k+1)}$ using (23).
      end for
      for each pixel j in each image i **do**
        if W_{ij} = 0 i.e. pixel j is not known **then**
          Update $X_{ij}^{M(k+1)}$ using (20).
        end if
      end for
    end while
    Update Y using (25).
    Update of Γ using (27).
    $k = k + 1$.
  end while
end while

5 Experimental Results

In this section we evaluate and compare the performance of our algorithm with two versions of the baseline algorithm, in both real world and synthetic examples. We compare the following methods:

Baseline: Algorithm 1 described in Section 2. This method is used in [2, 11, 25, 9, 29, 10].

RPCA: Images are preprocessed using Robust PCA [31], parameters are chosen as suggested by [11]. Then we apply the baseline algorithm to the obtained matrix. This method is used in [11]. RPCA solves a sparse low rank optimization to detect shadows and other non-Lambertian effects. The method uses L_1 regularization to identify outlier pixels, even when they do not result in intensities near 0 or 1. The refined intensities obtained from RPCA may not be in the range of [0,1]. Obtuse angle between the surface normal and the light can cause negative intensity and specularity can cause intensity more than 1.

Our(NC): Our proposed formulation as described in Section 4 using $W = 1$, i.e., no completion. This allows comparison to Baseline, which also does not perform matrix completion.

Our(MC): Our proposed formulation as in Section 4 with $w_{ij} \in \{0, 1\}$, allowing for matrix completion. In both versions of our algorithm we use $c = 1$. We identify missing pixels as those with normalized intensity outside the range of $(0.02, 0.98)$. We use RPCA algorithm to perform UPS, and the obtained normals and lights are used to initialize our algorithm as highlighted in Algorithm 2.

All the tested methods solve for the surface only up to a GBR ambiguity. To compare the results with ground truth, we find the GBR that optimizes the fit to ground truth, and measure the residual error.

In the presence of a large number of images with noise and non-Lambertian effects, we expect the sequential pipeline of Baseline and RPCA, involving SVD, to produce accurate

solutions, because the problem solved by SVD is heavily overconstrained. In the presence of fewer images, our integrated method will be able to produce a more accurate decomposition by using both rank and integrability constraints to find the right linear subspace. Thus we expect our integrated approach to improve over the Baseline and RPCA as we reduce the number of images. In the following subsection we will show results with synthetic and real world data that supports our claim.

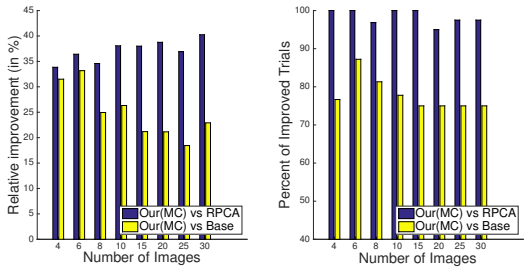
5.1 Experiments on Synthetic Data

We use five real objects ("cat", "owl", "rock", "horse", "buddha") to produce synthetic images, their shape is obtained by applying calibrated photometric stereo to a publicly available dataset [16]. We use the normals and albedos from these objects to generate images. Each image is generated by a randomly selected light source which lies at 30 degrees of the viewing direction on average. We clip the intensities outside the range [0,1] to create shadows and specularity. All images are of size 512 $\times$ 340 with objects occupying 29-72K pixels. A segmentation mask is also supplied. To show the variation of performance with the number of images N_I, we use sets of 4, 6, 8, 10, 15, 20, 25 and 30 images respectively. We add Gaussian noise with standard deviation ranging from 1% to 7% (in steps of 2%) of the maximum intensity. For each choice of noise, we run 5 different trials with random noise and lighting to generate the synthetic images. Thus we have 5 objects, 4 levels of noise and 5 random simulations, making a total of 100 experiments for each of the 8 different sets of 4, 6, 8, 10, 15, 20, 25 and 30 images. As a measure of performance, we calculate the error in the reconstructed depth map. Let the ground truth surface be Z_T and the reconstructed surface be Z_{rec}. We measure error in depth as $Z_{err} = 100 \times \frac{||Z_T - Z_{rec}||}{||Z_T||}$. To compare two algorithms (say, algorithm A vs. algorithm B), we define the following two terms :

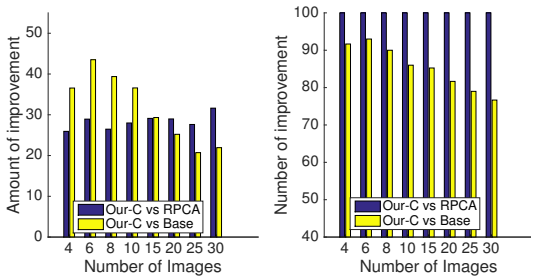
Relative Improvement (in %) : Denote e_k^a and e_k^b as the depth error for each trial k by using algorithm A and B respectively. The Relative Improvement of algorithm B over A is the average of $\frac{(e_k^a - e_k^b)}{e_k^a}$ over all trials K for each choice of N_I expressed in percentage.

Percent of Improved Trials : This denotes the number of trials in which algorithm B improves over A. In terms of notation introduced previously, this is $\frac{1}{K}\sum_{k=1}^{K} \mathbb{I}(e_k^{\mathrm{a}} < e_k^{\mathrm{b}})$, where $\mathbb{I}(.)$ is in indicator variable and K is the total number of trials for each choice of N_I. The measure is expressed in percentage.

In Figure 2 we compare performance of Our(MC) with Baseline and RPCA, on synthetic data in the presence of Gaussian noise. We initialize our methods with RPCA. We observe that as the number of images decreases, our method

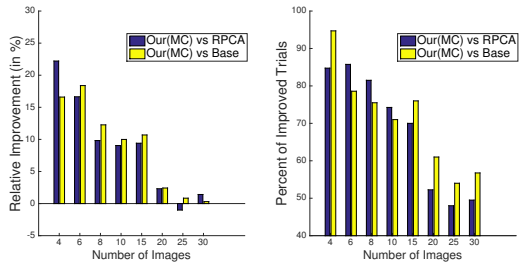


(a) Median error in Depth

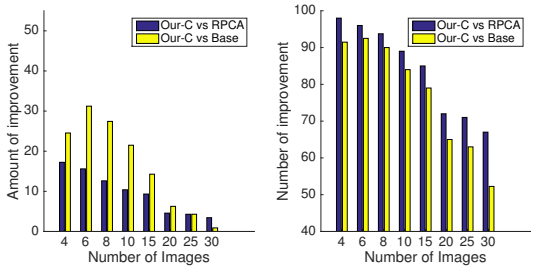


(b) Median error in Normal

Fig. 2: Performance comparison of Our(MC) algorithm to RPCA (in blue) and Baseline (yellow) for different numbers of input images with gaussian noise under a pure lambertian model. The left bar plot shows the amount of relative improvement achieved with our algorithm, and the right plot shows the percent of trials in which our algorithm out performed each one of the competing algorithms.



(a) Median error in Depth



(b) Median error in Normal

Fig. 3: Performance comparison of Our(MC) algorithm to RPCA (in blue) and Baseline (yellow) for different numbers of input images with gaussian noise under the Phong model. The left bar plot shows the amount of relative improvement achieved with our algorithm, and the right plot shows the percent of trials in which our algorithm out performed each one of the competing algorithms.

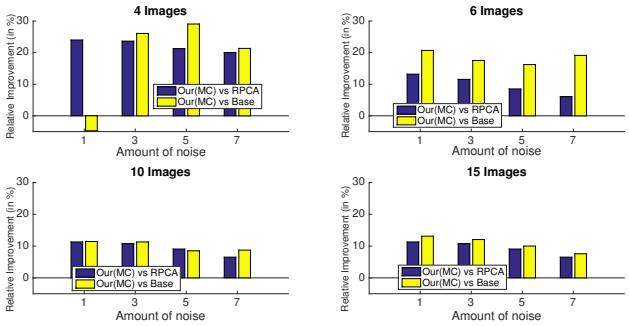


Fig. 4: Performance comparison of Our(MC) with RPCA and Baseline with varying noise created using the Phong model.

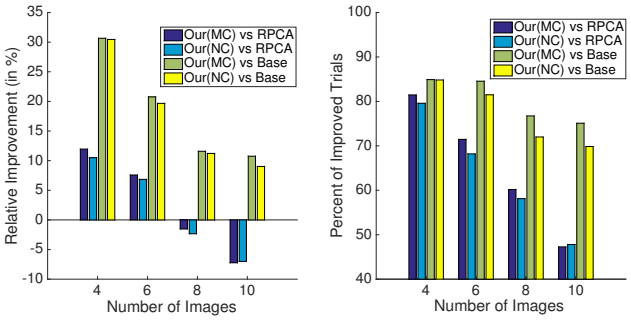


Fig. 5: Performance comparison of Our (MC) and Our (NC) algorithms to RPCA and Baseline with real images.

improves compared to Baseline and RPCA. With simple Gaussian noise RPCA doesn't produce additional advantages as there are no outliers.

In Figure 3 we compare the performance of our methods on synthetic data with Gaussian noise and with specularities generated by the Phong reflectance model [28, 23]. Mathematically each image M_i can be represented as :

$$M_i = L_i S + k_s (VR)^{\alpha}, \tag{28}$$

where V is the viewing direction and R denotes the directions of perfect reflection for incoming light L_i for each pixel j. Larger α produces sharper specularities, while larger k_s causes more light to be reflected as specularity. We use $k_s = 0.2$ and $\alpha = 10$. We observe that the advantage of Our(MC) degrades as the number of images increases, as expected. This experiment shows that even though our method is designed specifically for Lambertian objects it can tolerate a certain amount of model irregularities such as specularity. With 4 images our method beats RPCA in 85% of the all trials with a relative improvement of 22.12%.

In Figure 4 we compare Our(MC) with Baseline and RPCA with variation of noise for different subsets of images (4,6,10 and 15). We can conclude that our method is robust to noise and its advantages do not degrade with an increase in noise.

5.2 Experiments on Real World Data

5.2.1 Lambertian Objects

To test our approach on real data, we used the two publicly available data sets [16] and [32] consisting of 5 and 7 objects respectively. We perform Uncalibrated Photometric Stereo over a set of images and use the result of Calibrated Photometric Stereo as Ground Truth for comparison. The datasets provide calibrated lighting, which we use to perform calibrated photometric stereo. We use the code provided by [32, 16] along with the lighting information to obtain normals and depth map. The obtained depth map, albedo, and surface normals from calibrated PS are considered as ground-truth for photometric stereo with unknown lighting similar to [2]. To show the variation of performance with the number of images, we select subset of 4, 6, 8 and 10 images for each object. We perform 10 random selections of subset of images for each of the 12 objects. Thus we have 120 experiments for every subset of images.

In Figure 5 we compare the performance of our methods, Our(MC) and Our(NC), with Baseline and RPCA with variation in the number of images. We see that for fewer images our methods outperform Baseline and RPCA by a significant amount and are comparable to RPCA for more images. For 4 images Our(MC) outperforms Baseline in 84.9% cases with a relative improvement of 30.6% and outperforms RPCA in 81.4% cases with a relative improvement of 12%. However for 10 images we beat Baseline in 75% cases with a relative improvement of 10.7% and beat RPCA in only 47.3% cases with a relative improvement of -7.2%.

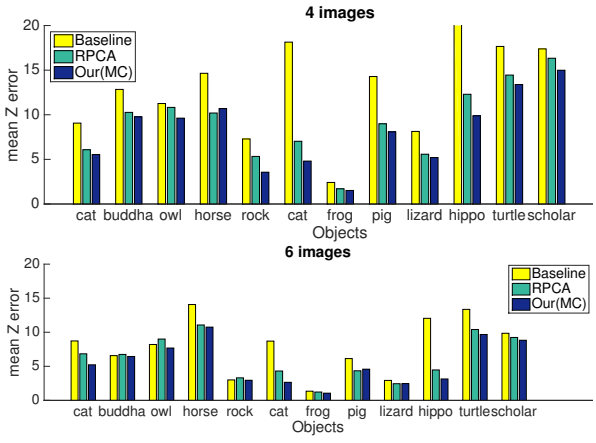


Fig. 6: Average surface reconstruction error with 4 (top) and 6 (bottom) real images of 12 objects over 10 random trials using Our(MC), RPCA and Baseline.

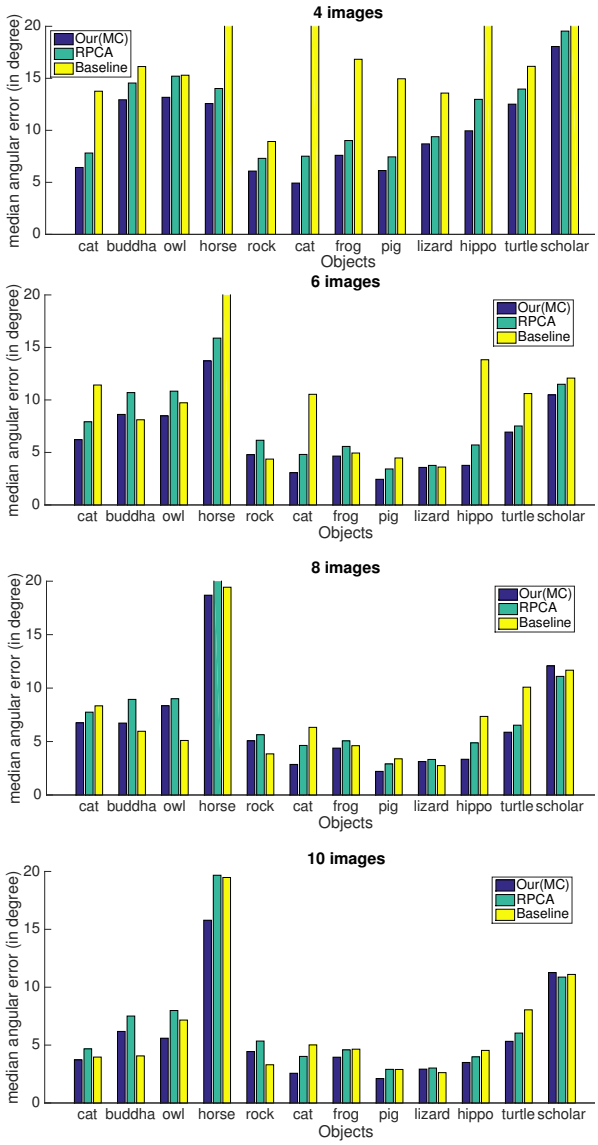


Fig. 7: Average median angular error in surface normal with 4, 6, 8 and 10 real images of 12 objects over 10 random trials using Our(MC), RPCA and Baseline.

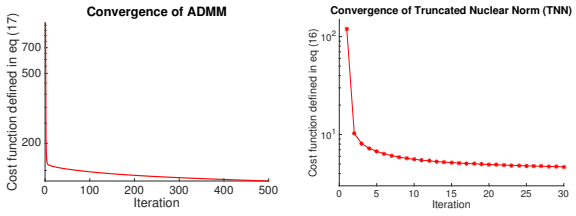


Fig. 8: Convergence of ADMM algorithm for each iteration of TNN as shown in equation 17 (left). Convergence of TNN regularized cost function as shown in equation 16 (right).

Figure 6 shows the average reconstruction error obtained by Our(MC), RPCA and Baseline on 12 real-world objects over 10 random simulations. We observe that Our(MC) outperforms RPCA on 11 out of 12 objects for 4 images and 10 out of 12 objects for 6 images (and is comparable in 1). With 10 images the average reconstruction error using Our(MC) over all objects and all trials is 4.6%. This increases to 8.1% with four images, and is only 5.4% with six images. This shows that we have reasonable reconstruction with 4 images and good reconstruction with as few as 6 images.

Figure 7 shows the average of median angular error in surface normal obtained by Our(MC), RPCA and Baseline on 12 real-world objects over 10 random simulations. Our(MC) outperforms both RPCA and baseline on all 12 objects for 4 images and 10 out of 12 objects for 6 images. This shows that also in terms of surface normal reconstruction error our algorithm outperforms RPCA and Baseline when fewer images are available.

In Figure 9 we compare the error in surface reconstruction between Baseline, RPCA, and Our(MC) on some of our real world examples. Figure 10 shows two views of surfaces reconstructed using Our(MC) algorithm using 4 images, showing reasonable surface reconstruction. These results suggest that our joint approach to enforcing rank and integrability constraints can significantly improve the performance of photometric stereo in the presence of a few images.

In general, we see that incorporating matrix completion into our formulation results in a slight improvement, with Our(MC) somewhat outperforming Our(NC). This indicates that the improvement of our method compared to RPCA or Baseline is mostly due to the joint optimization formulation and not due to matrix completion. We further note that RPCA seems to significantly improve over Baseline. RPCA is able to identify outliers and use that extra information for better recovery. This also suggests that the robust error function used by RPCA is important. However our integrated approach, which does not have a robust cost function like RPCA, still outperforms RPCA for 4 and 6 images and is almost equal for 8 or 10 images. This shows that an integrated approach is very useful for a small number of images and provides similar gain compared to RPCA for more images. It would be an interesting topic of future work to amend the cost function of Our(MC) to include RPCA's robust handling of error, to see if this further improves its performance.

5.2.2 Non Lambertian Objects

We also test our method on 8 objects of a Non Lambertian Objects Dataset [26]. We compare our method with RPCA and Baseline. For each object we choose 5 different random sub-samples to 4, 6, 8, 10, 15, 20, 30 and 40 images. In Table 1 - 9 we show the results of Our(MC), RPCA and Baseline for different number of images. The results show that our method is also robust in the presence of Non Lambertian objects.

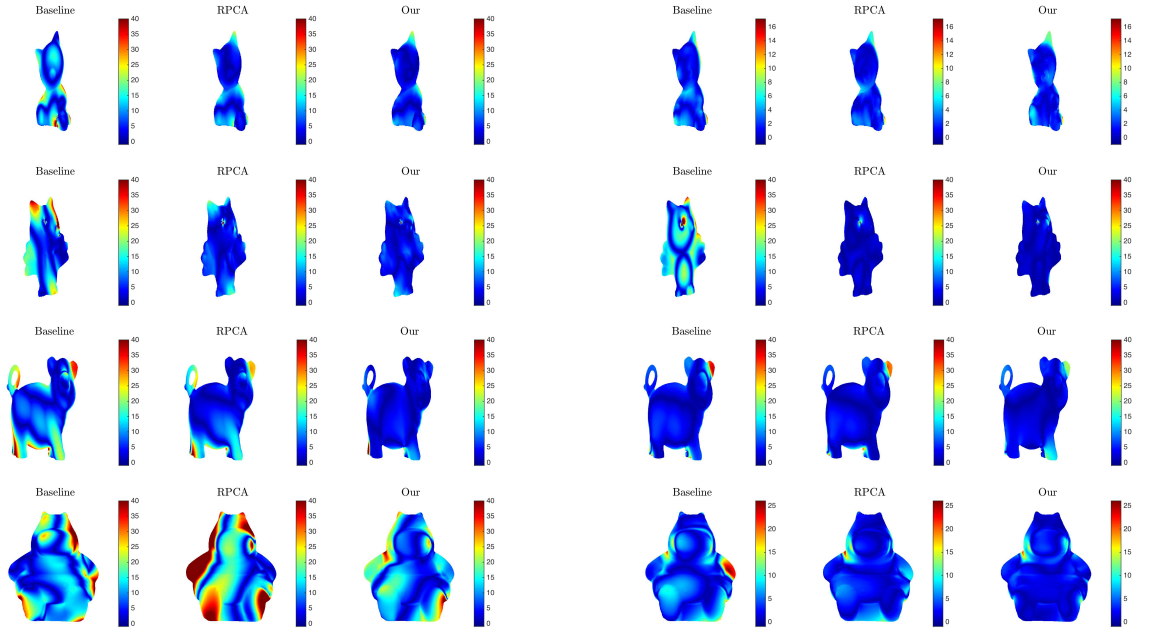


Fig. 9: Reconstruction error $|Z_T - Z_{rec}|$ for Baseline, RPCA and Our(MC) on "Cat", "Owl", "Pig" and "Hippo" shown in each row. The left column shows results for 4 images, the right shows results for 10.

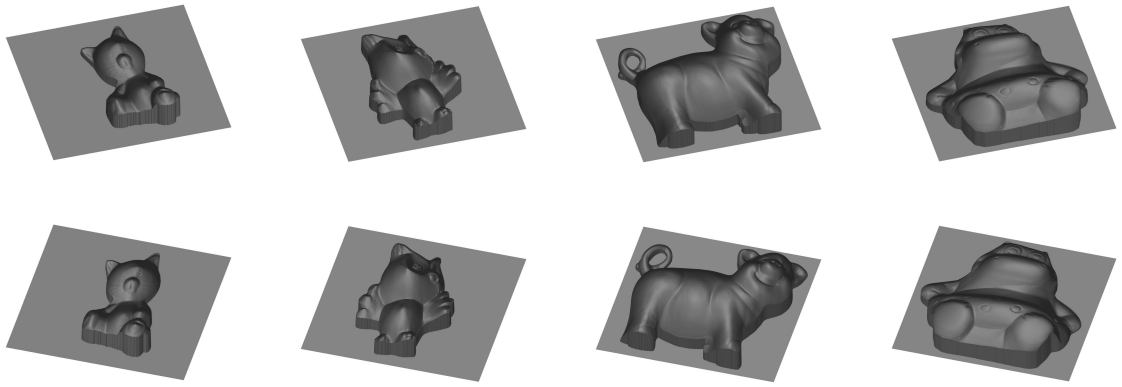


Fig. 10: Two views of surfaces reconstructed with Our(MC) algorithm for 4 images. Each column shows two images of surfaces reconstructed on "Cat", "Owl", "Pig" and "Hippo" respectively.

The result shows that Our(MC) consistently performs better than Baseline and RPCA. Even for large number of images, like 96, Our(MC) improves over RPCA by 6.19% on average over all objects in the dataset.

For an image of size 512×340 with an object occupying an area of 30K pixels, our algorithm takes 20 minutes on a 2.7 GHz Intel Core i5 machine. In figure 8 we show a typical sample convergence graph for our ADMM algorithm which solves the optimization problem (17) on the left and the convergence graph for TNN-ADMM algorithm which solves our original optimization (16) on the right. We empirically observe that ADMM converges to a local minimum. Since the problem is non-convex there is no guarantee of convergence to the global minimum.

6 Conclusion and Future Work

In this paper we have introduced a new low-rank constrained optimization method for solving uncalibrated photometric stereo using fewer images. The key to this approach is to combine rank and integrability constraints in a single optimization problem. This relies on a novel formulation that exposes both depth and surface normals to the optimization, linking them with an integrability constraint. We then

Table 1: Median Surface normal Reconstruction error for 4 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	**9.46**	15.02	11.20	52.88	10.30	43.24	18.38	49.39
RPCA	10.48	11.63	**8.37**	39.49	**8.32**	41.86	**12.62**	42.41
Our(MC)	9.74	**9.91**	8.50	**38.95**	8.39	**41.00**	13.95	**42.40**

Table 2: Median Surface normal Reconstruction error for 6 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	7.34	13.02	7.73	43.35	9.11	**36.83**	15.22	46.74
RPCA	7.02	10.68	**7.23**	39.98	7.77	34.78	12.18	39.68
Our(MC)	**6.75**	**9.03**	7.41	**39.61**	**7.34**	**33.96**	**11.91**	**38.96**

Table 3: Median Surface normal Reconstruction error for 8 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	7.01	12.73	7.86	38.39	8.83	36.20	14.76	45.61
RPCA	6.68	10.78	**7.49**	36.36	7.71	31.11	12.11	39.84
Our(MC)	**6.30**	**9.13**	7.63	**33.60**	**7.27**	**30.40**	**11.06**	**39.37**

Table 4: Median Surface normal Reconstruction error for 10 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	6.99	12.56	8.18	36.46	8.41	35.07	14.72	45.25
RPCA	6.64	10.88	**7.76**	35.19	**7.66**	32.55	12.69	39.75
Our(MC)	**6.18**	**9.19**	8.15	**31.48**	7.73	**32.02**	**11.38**	**39.45**

Table 5: Median Surface normal Reconstruction error for 15 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	6.99	11.95	7.85	45.11	7.86	31.29	15.27	43.52
RPCA	6.61	11.03	**7.80**	34.89	7.49	31.58	13.40	**40.50**
Our(MC)	**5.89**	**9.36**	8.26	**30.98**	**6.71**	**31.08**	**11.87**	40.55

Table 6: Median Surface normal Reconstruction error for 20 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	6.81	11.77	**7.68**	46.55	7.78	30.70	14.87	43.02
RPCA	6.56	11.08	7.76	33.30	7.43	30.17	13.54	**40.20**
Our(MC)	**5.78**	**9.52**	8.34	**31.40**	**6.91**	**29.49**	**11.74**	40.70

Table 7: Median Surface normal Reconstruction error for 30 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	6.73	11.76	**7.92**	46.01	7.89	**30.48**	14.42	40.42
RPCA	6.63	11.32	8.05	32.71	7.59	33.65	13.46	38.95
Our(MC)	**5.68**	**9.72**	8.55	**31.96**	**6.89**	32.59	**11.69**	**38.22**

Table 8: Median Surface normal Reconstruction error for 40 images

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	6.70	11.80	**7.74**	44.46	7.90	30.89	14.14	39.75
RPCA	6.69	11.44	7.92	34.05	7.64	29.05	13.39	38.84
Our(MC)	**5.58**	**9.83**	8.37	**31.71**	**6.56**	**27.62**	**12.01**	**38.28**

Table 9: Median Surface normal Reconstruction error for all (96) images.
(We set RPCA parameter to 0.5)

Algorithms	cat	buddha	bear	goblet	pot1	reading	cow	harvest
Baseline	6.60	11.58	7.34	43.98	7.94	31.18	13.99	39.55
RPCA	6.32	10.80	**7.19**	**32.27**	6.92	27.08	**11.93**	**38.27**
Our(MC)	**5.34**	**9.14**	7.55	32.68	**5.18**	**27.06**	12.01	38.60

show how to perform this optimization using a truncated nuclear norm and ADMM. Our joint formulation produces better solutions, compared to other methods that use SVD, for fewer images. We have shown promising results compared to baseline approaches using both real and synthetic examples. We also observe that our method can handle certain degrees of model irregularities as it has outperformed RPCA in synthetic examples with specularities generated using the Phong model.

In the future, it will be interesting to apply the idea of Robust PCA to our formulation. We would also like to extend this work to handle more general lighting configurations, e.g., using spherical harmonic approximations to lighting.

References

1. Ackermann J, Goesele M (2015) A survey of photometric stereo techniques. Foundations and Trends® in Computer Graphics and Vision 9(3-4):149–254
2. Alldrin NG, Mallick SP, Kriegman DJ (2007) Resolving the generalized bas-relief ambiguity by entropy minimization. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, pp 1–7
3. Basri R, Jacobs D, Kemelmacher I (2007) Photometric stereo with general, unknown lighting. International Journal of Computer Vision 72(3):239–257
4. Belhumeur PN, Kriegman DJ, Yuille AL (1999) The bas-relief ambiguity. International journal of computer vision 35(1):33–44
5. Boyd S, Parikh N, Chu E, Peleato B, Eckstein J (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning 3(1):1–122
6. Cabral RS, Torre F, Costeira JP, Bernardino A (2011) Matrix completion for multi-label image classification. In: Advances in Neural Information Processing Systems, pp 190–198
7. Cai JF, Candès EJ, Shen Z (2010) A singular value thresholding algorithm for matrix completion. SIAM J on Optimization 20(4):1956–1982
8. Chandraker M, Agarwal S, Kriegman D (2007) Shadowcuts: Photometric stereo with shadows. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, pp 1–8
9. Chandraker MK, Kahl CF, Kriegman DJ (2005) Reflections on the generalized bas-relief ambiguity. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, IEEE, vol 1, pp 788–795
10. Drbohlav O, Chaniler M (2005) Can two specular pixels calibrate photometric stereo? In: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, IEEE, vol 2, pp 1850–1857
11. Favaro P, Papadhimitri T (2012) A closed-form solution to uncalibrated photometric stereo via diffuse maxima. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, pp 821–828
12. Georghiades AS (2003) Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, Ieee, pp 816–823
13. Goldstein T, O'Donoghue B, Setzer S, Baraniuk R (2014) Fast alternating direction optimization methods. SIAM Journal on Imaging Sciences 7(3):1588–1623
14. Hayakawa H (1994) Photometric stereo under a light source with arbitrary motion. JOSA A 11(11):3079–3089
15. Hu Y, Zhang D, Ye J, Li X, He X (2013) Fast and accurate matrix completion via truncated nuclear norm regularization. Pattern Analysis and Machine Intelligence, IEEE Transactions on 35(9):2117–2130
16. Joshi N, Kemelmacher I, Simon I (2015) Photometric stereo dataset. url=`http://courses.cs.washington.edu/courses/cse455/10wi/projects/project4/`
17. Mecca R, Tankus A, Wetzler A, Bruckstein AM (2014) A direct differential approach to photometric stereo with perspective viewing. SIAM Journal on Imaging Sciences 7(2):579–612
18. Oh TH, Kim H, Tai YW, Bazin JC, Kweon IS (2013) Partial sum minimization of singular values in rpca for low-level vision. In: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE, pp 145–152
19. Oh TH, Tai YW, Bazin JC, Kim H, Kweon IS (2016) Partial sum minimization of singular values in robust pca: Algorithm and applications. IEEE transactions on pattern analysis and machine intelligence 38(4):744–758

20. Okabe T, Sato I, Sato Y (2009) Attached shadow coding: Estimating surface normals from shadows under unknown reflectance and lighting conditions. In: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, pp 1693–1700
21. Okatani T, Yoshida T, Deguchi K (2011) Efficient algorithm for low-rank matrix factorization with missing components and performance comparison of latest algorithms. In: Computer Vision (ICCV), 2011 IEEE International Conference on, IEEE, pp 842–849
22. Papadhimitri T, Favaro P (2013) A new perspective on uncalibrated photometric stereo. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1474–1481
23. Phong BT (1975) Illumination for computer generated pictures. Communications of the ACM 18(6):311–317
24. Quéau Y, Lauze F, Durou JD (2015) Solving uncalibrated photometric stereo using total variation. Journal of Mathematical Imaging and Vision 52(1):87–107
25. Shi B, Matsushita Y, Wei Y, Xu C, Tan P (2010) Self-calibrating photometric stereo. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, pp 1118–1125
26. Shi B, Wu Z, Mo Z, Duan D, Yeung SK, Tan P (2016) A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3707–3716
27. Sunkavalli K, Zickler T, Pfister H (2010) Visibility subspaces: Uncalibrated photometric stereo with shadows. In: European Conference on Computer Vision, Springer, pp 251–264
28. Tan P (2014) Phong reflectance model. In: Computer Vision, Springer, pp 592–594
29. Tan P, Mallick SP, Quan L, Kriegman DJ, Zickler T (2007) Isotropy, reciprocity and the generalized bas-relief ambiguity. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, pp 1–8
30. Woodham RJ (1980) Photometric method for determining surface orientation from multiple images. Optical engineering 19(1):191,139–191,139
31. Wu L, Ganesh A, Shi B, Matsushita Y, Wang Y, Ma Y (2011) Robust photometric stereo via low-rank matrix completion and recovery. In: Computer Vision–ACCV 2010, Springer, pp 703–717
32. Xiong Y, Chakrabarti A, Basri R, Gortler SJ, Jacobs DW, Zickler T (2015) From shading to local shape. Pattern Analysis and Machine Intelligence, IEEE Transactions on 37(1):67–79
33. Yuille A, Snow D (1997) Shape and albedo from multiple images using integrability. In: Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on, IEEE, pp 158–164