Robust Photometric Stereo via Dictionary Learning

Andrew J. Wagenmaker, Student Member, IEEE, Brian E. Moore, Student Member, IEEE, and Raj Rao Nadakuditi, Member, IEEE

Abstract—Photometric stereo is a method that seeks to reconstruct the normal vectors of an object from a set of images of the object illuminated under different light sources. While effective in some situations, classical photometric stereo relies on a diffuses surface model that cannot handle objects with complex reflectance patterns, and it is sensitive to non-idealities in the images. In this work, we propose a novel approach to photometric stereo that relies on dictionary learning to produce robust normal vector reconstructions. Specifically, we develop three formulations for applying dictionary learning to photometric stereo. We propose a preprocessing step that utilizes dictionary learning to denoise the images. We also present a model that applies dictionary learning to regularize and reconstruct the normal vectors from the images under the classic Lambertian reflectance model. Finally, we generalize the latter model to explicitly model non-Lambertian objects. We investigate all three approaches through extensive experimentation on synthetic and real benchmark datasets and observe state-of-the-art performance compared to existing robust photometric stereo methods.

Index Terms—Dictionary learning, photometric stereo, sparse representations.

I. INTRODUCTION

PHOTOMETRIC stereo [1] is a method that seeks to reconstruct the normal vectors of an object from a set of images of the object illuminated under different light sources. Concretely, we have images $I_1, \ldots, I_d$ of the threedimensional object and, in each image, the object is illuminated by a (distant) light source with light incident on the object in direction $\ell_1, \ldots, \ell_d \in \mathbb{R}^3$. Given $I_1, \ldots, I_d$ and $\ell_1, \ldots, \ell_d$, the goal is to estimate the normal vector map of the object, which can be numerically integrated to obtain a three-dimensional representation of the object. The appeal of photometric stereo is its simplicity: it requires only a camera and a movable light source to generate a three-dimensional representation of an object.

Since its introduction by Woodham [1], significant work has been performed to increase the generality and robustness of photometric stereo [2]–[15]. This body of work typically seeks to weaken one of two underlying assumptions in Woodham’s original model: that the position of the object relative to the position of the light source is known or that the object follows the Lambertian reflectance model. Works addressing the first assumption generally attempt to either reconstruct the normal vectors of objects whose reflectance properties deviate from the Lambertian model, or they try to develop methods that are robust to corruptions in the observed images. In this work, our focus is primarily on the design of robust photometric stereo algorithms, but we also incorporate a non-Lambertian model for increased generality.

The Lambertian reflectance model states that the intensity of light reflected by a point on a surface is linearly proportional to the inner product of the direction of illumination and the normal vector of the surface at that point [1]. Given a set of images of a Lambertian object illuminated under several (known) lighting directions, a simple system of equations can be solved to determine the normal vector at each point on the surface. In practice, while this is a reasonable model for some objects, the reflectance properties of many real-world objects differ significantly from the Lambertian model. Furthermore, shadows, specularities, and other non-idealities can cause additional deviations from the linear reflectance behavior. Performing classical photometric stereo on such non-Lambertian data typically yields large errors in the estimated normal vectors. As such, developing photometric stereo methods for objects that are inherently non-Lambertian and improving robustness to other imperfections are essential to extending the applicability and accuracy of photometric stereo.

Two primary approaches have found success addressing these problems. Several works assume the Lambertian model is fundamentally correct and seek to account for deviations from the model through explicit outlier removal [8], [11]—often assuming that non-idealities are sparse. While achieving some level of success, these approaches can place overly restrictive assumptions on the data, which may result in falsely rejecting useful data as outliers, and they make no attempt to model the true reflectance properties of objects. In turn, other works propose more complex reflectance models that enable non-Lambertian photometric stereo [13]–[15]. Approaches in this class are able to accurately model a wider range of objects, but they still break down when their modeling assumptions fail. Furthermore, they often fail when the data contains corruptions not accounted for by their reflectance models. In addition to the aforementioned difficulties, state-of-the-art methods in both categories typically rely on a large number of images to accurately construct the normal vectors, which may not be feasible to gather in practice.

In this work, we propose a novel approach to photometric stereo that relies on dictionary learning [26], [27] to robustly handle a wide range of non-idealities in the data. Dictionary learning seeks represent local patches of the data as sparse in normal vectors of the object without any knowledge of the lighting directions [16]–[25]. Works addressing the latter assumption generally attempt to either reconstruct the normal vectors of objects whose reflectance properties deviate from the Lambertian model, or they try to develop methods that are robust to corruptions in the observed images. In this work, our focus is primarily on the design of robust photometric stereo algorithms, but we also incorporate a non-Lambertian model for increased generality.
a learned collection of atoms. Such models effectively act as dynamic regularization that adapts to the underlying structure of the data and removes spurious corruptions. Inspired by recent successes applying dictionary learning to a variety of imaging problems [28], [29], we adopt this methodology to improve the robustness of photometric stereo. Our approach is data-driven and adapts to the underlying structure of the data without imposing additional explicit constraints. Furthermore, we incorporate an existing non-Lambertian reflectance model into our method to better handle non-Lambertian surfaces. In total we present three dictionary learning-based formulations of robust photometric stereo. We investigate the performance of each method in a variety of different scenarios. In particular, we evaluate their performance on the benchmark DiLiGent dataset [30] and their ability to handle general, non-sparse corruptions. This work is an extension of our recent work [31]. Here, we substantially build on the work by performing an extensive numerical study to evaluate the performance of our proposed dictionary learning-based approaches on real and synthetic data, and we extend our methods to incorporate the aforementioned non-Lambertian reflectance model.

The remainder of this work is organized as follows. In Section II we provide a brief overview of related works in photometric stereo. In Section III we carefully define the photometric stereo problem and the non-Lambertian reflectance model we will incorporate into our method. Section IV presents our dictionary learning-based methods and their associated algorithms. Finally, Section V provides an extensive numerical study of the performance of our proposed methods compared to state-of-the-art methods.

II. RELATED WORK

Lambertian photometric stereo was originally proposed by Woodham [1] in 1980. Since then, much work has been done extending it to more general settings where the Lambertian model does not hold exactly. This body of work has typically either taken the approach of treating non-Lambertian effects as outliers or as incorporating non-Lambertian effects into the reflectance model.

A variety of approaches have been proposed to robustly perform photometric stereo via outlier rejection. In general such works assume that the underlying data is inherently Lambertian and seek to identify non-Lambertian effects as outliers—then rejecting them to increase the accuracy of the computed normal vectors. Early works in robust photometric stereo—typically referred to as four source photometric stereo—utilized four images to identify and reject specularities [2], [32], [33]. Since these works, more complex methods have been developed that rely on maximum likelihood estimation [34], expectation maximization [6], and a maximum feasible subsystem framework [35]. Other approaches include a graph cuts based algorithm to identify shadows [36], a method which seeks to transform RGB images so they lie in a two-dimensional subspace invariant to specularities [37], and several methods that utilize RANSAC based algorithms [38], [39].

The most recent works that seek to solve the problem of robust photometric stereo via outlier rejection—and the current state-of-the-art in this category—are those by Wu et al. [8] and Ikehata et al. [11]. Observing that the images collected, if corresponding to a truly Lambertian object, must lie in a three-dimensional subspace, they assume non-Lambertian effects only sparsely affect the images and employ rank minimization based approaches to determine the underlying Lambertian portion of the data.

Regardless of their robustness to outliers, approaches that rely on the Lambertian reflection model as the underlying model of the data are inherently limited in their scope due to the wide variety of non-Lambertian surfaces that exist in the real world. As a result, in addition to these outlier-rejection based methods, a significant amount of work has also been done incorporating more general reflectance models into photometric stereo [40]. In particular, uncalibrated photometric stereo based on the Torrance and Sparrow reflectance model has been proposed [19] as well as calibrated photometric stereo based on the Ward reflectance model [5], [9], [41].

A large amount of work has also been done developing photometric stereo that uses reflectance models based on general reflectance properties exhibited by a wide range of materials. In particular, the property of isotropy has been successfully utilized in a variety of works [10], [12], [42]–[44]. The current state-of-the-art in this category are the works of Shi et al. [15] and Ikehata et al. [14]. Ikehata et al. attempts to model the reflectance function using a sum-of-lobes representation [7], utilizing Bernstein polynomials as the basis of the inverse reflectance function and performing bivariate regression to determine the normal vectors. Shi et al. focuses instead on modeling the low-frequency reflectance component using polynomials of up to order 3 while discarding the high-frequency reflectance component. Of particular interest to this paper is another work by Ikehata et al. [13] that seeks to model the reflectance function as piecewise-linear. We explore this method in more detail in the following section.

In addition to the previously mentioned approaches, a variety of other works have proposed alternate methods for robust photometric stereo [3], [4], [45], [46]. Finally, the recent work of Shi et al. [30] seeks to standardize future work in photometric stereo by introducing an extensive dataset to facilitate future testing and evaluation. Further, they compare a variety of existing approaches on this dataset, providing a benchmark for future work.

III. PROBLEM FORMULATION

A. Basis of Photometric Stereo

The Lambertian reflectance model states that, to an observer, the brightness of a point on a Lambertian surface is independent of the observer’s viewing angle. Surfaces that follow this model are matte in appearance. Consider some image taken of a Lambertian object. The light intensity observed at some point on the surface with pixel coordinates \((x, y)\) then satisfies the following equation

\[
I_{xy} = \rho_{xy} \ell^T n_{xy}
\]

where \(I_{xy}\) is the image intensity at pixel \((x, y)\), \(\ell \in \mathbb{R}^3\) is the direction of the light source incident on the surface, \(|\ell|\) is the
light source intensity, $n_{xy}$ is the normal vector of the surface at $(x, y)$, and $\rho_{xy}$ is the surface albedo at $(x, y)$—a measure of the reflectivity of the surface.

If we fix the position of a camera facing our surface and vary the position of the light source over $d$ unique locations, we can write $d$ equations of the form (1). These can be stacked into the following matrix equation

$$
\begin{bmatrix}
I_{xy}^1 \\
\vdots \\
I_{xy}^d
\end{bmatrix} =
\begin{bmatrix}
\ell_1^T \\
\vdots \\
\ell_d^T
\end{bmatrix}
(\rho_{xy}n_{xy})
$$

(2)

where $I_{xy}^k$ denotes the image intensity at $(x, y)$ in the $k$th image. Assuming each of our $d$ images has dimension $m_1 \times m_2$, equation (2) can be solved $m_1m_2$ times to obtain the normal vector of the object at each point on the surface. We may also combine these $m_1m_2$ equations into a single matrix equation.

Define the following observation matrix

$$
Y = \left[ \mathbf{vec}(I^1) \mid \cdots \mid \mathbf{vec}(I^d) \right] \in \mathbb{R}^{m_1m_2 \times d}
$$

(3)

where $\mathbf{vec}(I^k) \triangleq [I_{11}^k, \cdots, I_{m_1m_2}^k]^T$. Assuming our light source is at infinity and there is no variation in illumination from point to point on our object, using equation (4), we can write

$$
Y = NL
$$

(4)

where $N = [\rho_{11}n_{11} \mid \cdots \mid \rho_{m_1m_2}n_{m_1m_2}]^T \in \mathbb{R}^{m_1m_2 \times 3}$ and $L = [\ell_1 \mid \cdots \mid \ell_d] \in \mathbb{R}^{3 \times d}$.

In order to avoid scaling ambiguity, we assume all light sources are the same strength, constricting each light vector to unit norm. Further, we effectively ignore the albedo term, $\rho_{xy}$, and require that $|\rho_{xy}n_{xy}| = 1$.

Each normal vector $n_{xy}$ contains three unknown components. Given $d \geq 3$ images with corresponding light directions, we can then solve (1) to obtain the normal vector at each point on the object. Once computed, we can integrate the normal vectors to produce a full three-dimensional model of our surface [44].

### B. Deviations From Lambertian Model

While the Lambertian reflectance model is able to approximate the reflectance properties of some surfaces, it is a poor approximation for many real-world objects. Lambertian objects are matte in appearance and any non-matte objects deviate from the Lambertian reflectance model. This includes any object that exhibits specularities—bright points observed when light reflects off shiny objects. Further, even when an object is Lambertian, shadows, both self-cast and those produced by other objects, cause the Lambertian model to break down.

To account for such effects, one could modify (4) to

$$
Y = NL + E
$$

(5)

where $E$ is an error matrix accounting for non-Lambertian effects.

Given this formulation, more than the minimum of 3 images are generally used. A simple, naive approach for obtaining $N$ from (5) would be a least squares solution. This can be formulated as

$$
\min_N \|Y - NL\|_F^2
$$

(6)

which has solution $\hat{N} = YL^\dagger$. Here $\dagger$ denotes the Moore-Penrose pseudoinverse.

Several works apply further constraints to (5)—such as constraining $E$ to be sparse—allowing them to, in cases where their assumptions hold, derive estimates of $N$ more accurate than those obtained by (6) [8], [11]. In Section V, we propose two additional such approaches that assume (5) accurately models our data.

### C. Piecewise Linear Reflectance Model

Regardless of the constraints imposed, equation (5) fundamentally relies on the Lambertian reflectance model—limiting its applicability and cases where it can accurately reconstruct the normal vectors. Recent works have sought to move beyond the Lambertian assumption and utilize more general reflectance models to accurately reconstruct the normal vectors of a wider range of objects [13]–[15]. For the purposes of this work we take particular interest in the model presented in [13], which we briefly summarize here.

A simple extension to the Lambertian model is to assume that, while the image intensity still depends only on the inner product of $\ell$ and $n$, the intensity is now related to this inner product by a nonlinear function. This implies that we modify equation (4) to

$$
I_{xy} = f_{xy}(\ell^T n_{xy})
$$

(7)

for some nonlinear function $f_{xy}$.

Following [13], if we assume the reflectance function at each pixel, $f_{xy}$, is monotonically increasing, a unique inverse of $f_{xy}$ is guaranteed to exist. We can then invert (7) and write

$$
g_{xy}^{-1}(I_{xy}) = g_{xy}(I_{xy}) = \ell^T n_{xy}
$$

(8)

Given some set of lighting vectors and corresponding images, our task is then to simultaneously estimate both $g_{xy}(\cdot)$ and $n_{xy}$ for each pixel. This is a highly underconstrained problem and, to solve this in practice, further constraints must be imposed. A natural possibility is to assume the data behaves in a roughly linear fashion and restrict $g_{xy}(\cdot)$ to be piecewise linear. That is, let

$$
g_{xy}(t) = \sum_{k=1}^{p} a_{g_{xy}}^k t^{b_{g_{xy}}^k}
$$

(9)

where

$$
g_{g_{xy}}^k(t) = \begin{cases} 
0, & 0 \leq t < b_{g_{xy}}^{-1}^k \\
b_{g_{xy}}^{-1}^k - t, & b_{g_{xy}}^{-1}^k \leq t \leq b_{g_{xy}}^k \\
b_{g_{xy}}^k - b_{g_{xy}}^{-1}^k, & b_{g_{xy}}^k \leq t
\end{cases}
$$

(10)

Here $p$ is a design parameter that designates the number of segments present in the piecewise linear portion, $b_{g_{xy}}^k$ are inflection points of $g_{xy}(\cdot)$, and $a_{g_{xy}}^k$ is the slope of each segment. We set $b_{g_{xy}}^0 = 0$ and choose the remaining values of $b_{g_{xy}}^k$ such that, given a set of $d$ images, the values of $b_{g_{xy}}^k$
evenly divide the range of intensity values present at a given pixel, \((x, y)\).

This choice of \(g_{xy}(.)\) further constrains the problem to that of estimating for each pixel a set of slopes \(a_{xy}^1, \ldots, a_{xy}^p\) and a normal vector \(n_{xy}\). We note that, in the case where \(a_{xy}^1 = \ldots = a_{xy}^p\), the model reduces to the Lambertian case.

If we let
\[
a_{xy} = [a_{xy}^1 \ldots a_{xy}^p]^T \in \mathbb{R}^p
\]
and
\[
\tilde{g}_{xy}(t) = [g_{xy}^1(t) \ldots g_{xy}^p(t)]^T \in \mathbb{R}^p
\]
we can rewrite (2) as the vector product
\[
\tilde{g}_{xy}(t)^T a_{xy}
\]
Similarly, (8) can be written as
\[
\tilde{g}_{xy}(I_{xy})^T a_{xy} = \ell^T n_{xy}
\]
Given \(d\) images, let the matrix \(C_{xy} \in \mathbb{R}^{d \times p}\) be the matrix with \(j\)th row \(\tilde{g}_{xy}(I_{xy})^T\). We may then write an equation analogous to (2)
\[
C_{xy} a_{xy} = L^T n_{xy}
\]
This can be solved for \(n_{xy}\) and \(a_{xy}\) to determine the normal vector. To avoid the degenerate solution where \(a_{xy}\) and \(n_{xy}\) are zero vectors, we further constrain \(a_{xy}\) so that \(a_{xy}^1 + \ldots + a_{xy}^p = 1\). In practice we do not explicitly constrain values of \(a_{xy}^1, \ldots, a_{xy}^p\) to be greater than 0.

As in the Lambertian case, this model only accounts for properties—or, more explicitly, that do not follow the piece-wise linear relationship between \(I_{xy}\) and \(\ell^T n_{xy}\)—prevent (16) from holding exactly. We may modify (16) analogously to (3) to yield the following
\[
C_{xy} a_{xy} = L^T n_{xy} + e
\]
Here \(e \in \mathbb{R}^d\) accounts for any corruptions found in the data not captured by the reflectance model. A constrained least squares problem can now be solved to estimate \(n_{xy}\) and \(a_{xy}\)
\[
\min_{n_{xy}, a_{xy}} \|C_{xy} a_{xy} - L^T n_{xy}\|_2^2 \text{ s.t. } \sum_{k=1}^p a_{xy}^k = 1
\]
In practice, when such corruptions are present in the data, applying a more robust approach than simply solving (17) can improve accuracy further. \([13]\) utilizes this reflectance model but assumes the corruptions to the data are sparse. We seek to relax this sparsity assumption while robustly solving (17)—a problem we will return to in the following section.

IV. DICTIONARY LEARNING APPROACHES

Dictionary learning essentially seeks to determine some “dictionary”—a collection of representative or basis elements—that can sparsely represent the data of interest. In general, this allows one to determine some structure present in the data without any a priori knowledge of how the data should be structured. In this section we propose three adaptive dictionary learning algorithms to estimate the normal vectors of our object.

A. Preprocessing of Images through Dictionary Learning (DLPI)

We first propose applying dictionary learning to our data in a preprocessing step performed on the images before reconstructing the normal vectors. Our formulation utilizes dictionary learning to represent our original data matrix \(Y\) as sparse in an adaptive dictionary—thereby removing certain non-idealities that are not represented in the dictionary. Intuitively, this can simply be thought of as applying a denoising step to the input images before using them to reconstruct the normal vectors. This yields the following optimization problem
\[
\min_{v, B, D} \frac{1}{2} \|y - v\|_2^2 + \lambda \left( \sum_{j=1}^c \|P_j v - D b_j\|_2 + \mu^2 \|B\|_0 \right)
\]
\[\text{s.t. } \|B\|_\infty \leq q, \quad \|d_i\|_2 = 1 \forall i.\] (18)

Here \(y = \text{vec}(Y) \in \mathbb{R}^{m_1 m_2 d}\) and \(P_j \in \mathbb{R}^{c_x c_y c_z \times m_1 m_2 d}\) is a patch extraction matrix that extracts vectorized patches of dimensions \(c_x \times c_y \times c_z\) from \(v\) where \(c_x\) and \(c_y\) correspond to the dimensions of the patches extracted from each image and \(c_z\) corresponds to the number of images these patches are extracted from. \(D \in \mathbb{R}^{c_x c_y c_z \times K}\) represents the learned dictionary and has columns \(d_i\) corresponding to the dictionary atoms. While we again represent these as vectors, each column \(d_i\) corresponds to a 3D patch of size \(c_x \times c_y \times c_z\). \(B \in \mathbb{R}^{K \times c}\) is a sparse coding matrix whose columns \(b_j\) define (usually sparse) linear combinations of dictionary atoms used to represent each patch. Also, \(\| \cdot \|_0\) is the familiar \(\ell_0\) (pseudo-)norm, and \(\lambda, \mu \geq 0\) are regularization parameters. Note that \(K\) is a user-defined value specifying the number of atoms we allow in our dictionary and \(c\) is a constant corresponding to the number of dictionary atoms needed to represent our data.

We impose the constraint \(\|B\|_\infty \leq q, \|P_j v\|_2 + \mu^2 \|B\|_0 \leq q\), where \(q\) is typically very large, since \([13]\) is non-coercive with respect to \(B\), but the constraint is inactive in practice \([48]\). Without loss of generality, we also constrain all dictionary atoms—\(d_i\), the columns of \(D\)—to unit-norm. This allows us to avoid scaling ambiguity between \(D\) and \(B\) \([49]\).

Once we have solved (18), we reshape \(v\) back into an \(m_1 m_2 \times d\) matrix and estimate the associated normal vectors using the standard least-squares model \([4]\). While we do not investigate this here, one could also combine the processed images with a different approach to avoid relying on the Lambertian assumption. Henceforth, we refer to this approach as the Dictionary Learning with Preprocessed Images (DLPI) method.

B. Normal Vectors through Dictionary Learning (DLNV)

We next propose modifying \([4]\) by extending the cost function to include an adaptive dictionary learning term applied to the normal vectors. This allows us to jointly solve for the normal map and represent it as locally sparse in some dictionary—resulting in a smoother normal map that tends to be robust to non-idealities. Note that this formulation
implicitly assumes the Lambertian model accurately represents our data. Our optimization problem is

$$\min_{n,B,D} \frac{1}{2} \|y - An\|_2^2 + \lambda \left( \sum_{j=1}^{w} \|P_j n - Db_j\|_2^2 + \mu^2 \|B\|_0 \right)$$

s.t. \(\|B\|_\infty \leq q, \|d_i\|_2 = 1 \forall i. \) (19)

Here \(y = \text{vec}(Y) \in \mathbb{R}^{m_1m_2d}, A = L^T \otimes I \in \mathbb{R}^{m_1m_2 \times 3m_1m_2}\) where \(\otimes\) denotes the Kronecker product and \(I\) is the \(m_1m_2 \times m_1m_2\) identity matrix. Further, \(n = \text{vec}(N) \in \mathbb{R}^{m_3m_3}\) is the vectorized normal vectors. As in the DLPI formulation, \(P_j\) denotes a patch extraction matrix which extracts vectorized patches from \(n\) corresponding to 3D regions of \(N\) of dimensions \(w_x \times w_y \times w_z. D \in \mathbb{R}^{w_xw_yw_z \times K}\) again represents our learned dictionary where columns correspond to individual dictionary atoms and \(B \in \mathbb{R}^{K \times w}\) stores the sparse codes needed to represent each patch as a linear combination of dictionary elements. We impose the same constraints on \(B\) and \(D\) as we imposed in our formulation of DLPI. Henceforth, we refer to this approach as the Dictionary Learning on Normal Vectors (DLNV) method.

C. Non-Lambertian Normal Vectors through Dictionary Learning (PDLNV)

Finally, we present an approach that relies on the non-Lambertian reflectance model presented in Section III-C. Taking [17] as our baseline, we seek to modify it with a dictionary learning term to increase its robustness to corruptions. In particular, we again perform dictionary learning on the normal vectors, regularizing them while attempting to learn structure present in the data that allows us to sparsely represent them in our learned dictionary. This yields the following

$$\min_{n,B,D,a} \frac{1}{2} \sum_{x=1}^{m_1} \sum_{y=1}^{m_2} \|C_{xy}a_{xy} - L^T n_{xy}\|_2^2 + \frac{1}{2} \sum_{x=1}^{m_1} \sum_{y=1}^{m_2} \|1^T a_{xy} - 1\|_2^2 + \lambda \left( \sum_{j=1}^{w} \|P_j n - Db_j\|_2^2 + \mu^2 \|B\|_0 \right)$$

s.t. \(\|B\|_\infty \leq q, \|d_i\|_2 = 1 \forall i. \) (20)

All terms here are defined analogously to their previously given definitions. Note that we relax the constraint given in [17] enforcing that \(a_{xy}^1 + \ldots + a_{xy}^p = 1\) and include it in the optimization. Here, in order to encourage this constraint to hold approximately, we typically set \(\gamma\) to be very large.

This approach can be thought of as a generalization of DLNV. Indeed, if we set \(p = 1\) and let \(\gamma\) approach infinity, (20) reduces to (19). For the purposes of this paper, in order to highlight the differences between models that rely on the Lambertian assumption versus models that rely on more complex reflectance models, we include results from both this approach as well as DLNV. We refer to this approach as Dictionary Learning on Normal Vectors with Piecewise-Linear Reflectance (PDLNV).

D. Algorithms for DLPI and DLNV

We propose solving [18], [19], and (20) via block coordinate descent-type algorithms. We alternate between updating \(n\) or \(v\), respectively, with \((D,B)\) fixed and updating \((D,B)\) with \(n\) or \(v\) fixed. In the case of solving (20) we use a similar strategy but now alternate between updating \(n\), \((D,B)\), and \(a\) while keeping all other quantities fixed.

1) \((D,B)\) Update: The \((D,B)\) update is identical for all three cases. Here we present the update for [18].

We first define \(G \triangleq B^T\) and denote by \(P\) the matrix with columns \(P_jv\). With \(v\) fixed, the optimization problem with respect to \((D,B)\) can then be written as

$$\min_{G,D} \|P - DG^T\|_F^2 + \mu^2 \|G\|_0$$

s.t. \(\|G\|_\infty \leq a, \|d_i\|_2 = 1 \forall i. \) (21)

In order to solve (21), we iterate over all columns \(g_i\) of \(G\) and atoms \(d_i\) of \(D\). For every \(1 \leq i \leq K\), we solve (21) first with respect to \(g_i\), and then with respect to \(d_i\).

We first consider the minimization of (21) with respect to \(g_i\). Define \(E_i \triangleq P - \sum_{k \neq i} d_k g_k^T\) where \(E_i\) is computed using the most recent values of the other dictionary atoms and coefficients. Our problem then becomes

$$\min_{g_i} \|E_i - d_i g_i^T\|_F^2 + \mu^2 \|g_i\|_0$$

s.t. \(\|g_i\|_\infty \leq q \) (22)

The solution to (22) is given by [50]

$$\hat{g}_i = \min \left( \|H_\mu(E_i^T d_i)\|_1, q_1\right) \odot \text{sign} \left( H\mu(E_i^T d_i) \right)$$

(23)

where \(q_1\) is a vector of ones of length \(w\) and \(\min(\cdot, \cdot)\) is the element-wise minimum operator. Further, \(\odot\) denotes the element-wise multiplication and operator and \(H_\mu(\cdot)\) denotes the element-wise hard thresholding operator. This is defined as

$$[H_\mu(y)]_i = \begin{cases} 0, & |y_i| < \mu \\ y_i, & |y_i| \geq \mu \end{cases}$$

(24)

Minimizing (21) with respect to some \(d_i\) while keeping all other variables fixed produces the following problem

$$\min_{d_i} \|E_i - d_i g_i^T\|_F^2$$

s.t. \(\|d_i\|_2 = 1 \) (25)

It is possible to show that the global minimizer of this expression is given by

$$d_i = \begin{cases} \frac{E_i g_i}{\|E_i g_i\|_2}, & \text{if } g_i \neq 0 \\ u_1, & \text{if } g_i = 0 \end{cases}$$

(26)

where \(u_1\) is the first column of a \(c_x c_y c_z \times c_x c_y c_z\) identity matrix. Note that the solution is unique if and only if \(g_i \neq 0 \) [50].

2) \(v\) Update: Solving [18] for \(v\) with \(D\) and \(B\) fixed yields the problem

$$\min_v \frac{1}{2} \|y - v\|_2^2 + \lambda \sum_{j=1}^c \|P_j v - Db_j\|_2^2.$$ (27)

We see that equation (27) is a simple least-squares problem with normal equation

$$(I + 2\lambda \sum_{j=1}^c P_j^T P_j) v = y + 2\lambda \sum_{j=1}^c P_j^T Db_j.$$ (28)
where \( I \) denotes the identity matrix. The matrix pre-multiplying \( v \) in (23) is diagonal, and so its inverse can be cheaply computed, allowing us to efficiently update \( v \).

3) \( n \) update: Similarly, solving (19) for \( n \) with \( D \) and \( B \) fixed yields the problem

\[
\min_n \frac{1}{2} \|y - An\|^2 + \lambda \sum_{j=1}^w \|P_j n - Db_j\|^2. \tag{29}
\]

While (29) is also a least-squares problem, its normal equation cannot be easily inverted due to the presence of the \( A \) matrix. We instead utilize a proximal gradient scheme \([51]\). The cost function in (29) can be written in the form \( f(n) + g(n) \) where \( f(n) = \frac{1}{2} \|y - An\|^2 \) and \( g(n) = \lambda \sum_{j=1}^w \|P_j n - Db_j\|^2 \). The proximal updates thus become

\[
n^{k+1} = \text{prox}_g(n^k - \tau \nabla f(n^k)), \tag{30}
\]

where

\[
\text{prox}_g(y) := \arg\min_x \frac{1}{2} \|y - x\|^2 + \tau g(x). \tag{31}
\]

Define \( \hat{n}^k = n^k - \tau \nabla f(n^k) \). Then (30) and (31) imply that \( n^{k+1} \) satisfies the normal equation

\[
(I + 2\tau \lambda \sum_{j=1}^w P_j^T P_j) n^{k+1} = \hat{n}^k + 2\tau \lambda \sum_{j=1}^w P_j^T Db_j. \tag{32}
\]

As in (28), the matrix multiplying \( n^{k+1} \) in (32) is diagonal and can be efficiently inverted to determine \( n^{k+1} \). Note that in addition to proximal gradient, multiple other iterative strategies could have been utilized to minimize the quadratic objective (29). (20) can be solved in an identical manner—we simply replace \( y \) with the vector

\[
\begin{bmatrix}
C_{11} a_{11} \\
\vdots \\
C_{m_1 m_2} a_{m_1 m_2}
\end{bmatrix} \tag{33}
\]

where we utilize the current values of \( a_{xy} \).

4) \( a \) update: Solving (20) also requires us to update the value of \( a \). Keeping \( n \), \( D \), and \( B \) fixed, we must solve the following optimization problem

\[
\min_a \sum_{x=1}^{m_1} \sum_{y=1}^{m_2} \| C_{xy} a_{xy} - L^T n_{xy} \|^2 + \gamma \|1^T a_{xy} - 1\|^2 \tag{34}
\]

Equivalently, for each \((x, y)\) pair, we can solve the following problem

\[
\min_{a_{xy}} \left\| \begin{bmatrix} C_{xy} \\ \gamma 1^T \end{bmatrix} a_{xy} - \begin{bmatrix} L^T n_{xy} \\ \gamma \end{bmatrix} \right\|^2 \tag{35}
\]

This is simply a least squares problem and has the solution

\[
\hat{a}_{xy} = \begin{bmatrix} C_{xy} \\ \gamma 1^T \end{bmatrix} ^\dagger \begin{bmatrix} L^T n_{xy} \\ \gamma \end{bmatrix} \tag{36}
\]

Since \( C_{xy} \) does not depend on \( n \), \( a \), \( D \), or \( B \), it stays fixed throughout each iteration. We can therefore precompute

\[
\begin{bmatrix} C_{xy} \\ \gamma 1^T \end{bmatrix} ^\dagger \tag{37}
\]

allowing us to update \( a \) via a simple, cheap matrix multiplication.

V. EXAMPLES

We now verify the performance of our methods experimentally. In order to obtain quantitative numerical results, we rely primarily on the DiLiGenT dataset \([30]\). This dataset contains images of a variety of surfaces of different materials and provides the true normal vectors of each object, allowing us to evaluate the performance of each method and determine how accurately they are able to construct the normal vectors. We quantify the error in each computed normal vector by measuring the angular difference between the true normal vector and the computed normal vector.

Our method is evaluated on a variety of cases. For each experiment we compare the results of our method to Wu et al.’s robust PCA (RPCA) approach \([8]\), Ikemoto et al.’s sparse regression (SR) approach \([11]\), and Ikemoto et al.’s constrained bivariate regression (CBR) approach \([14]\). In addition, we also compare against the traditional least squares (LS) approach given by (6).

With the exception of least squares, each of these methods contains one or more tunable parameters that dictate the performance. For each method, we swept the parameters across a wide range of values to find the optimal parameters for each trial. For existing methods, we included any parameters recommended in the respective papers in our sweep. The reported results are the errors produced by the optimal parameter values. In each case, we ran the experiment over multiple noise realizations and averaged the results.

Up to this point in time, the majority of the photometric stereo literature has focused primarily on the problem of reconstructing normal vectors from uncorrupted, generally fairly large datasets such as the DiLiGenT dataset. In cases where additional corruptions have been added, typically they were added in a sparse fashion in order to better align with the modeling assumptions made in developing these various approaches. We seek here to more fully investigate the ability of our approach as well as existing approaches to handle non-sparse corruptions. We add such non-sparse noise to each pixel according to a Poisson distribution. Poisson noise can be used to model corruptions found in low-light images and thus, given that photometric stereo data is often collected in relatively low light, is a relevant noise model to use for this problem.

A pixel-wise shadow mask can be applied to the images as a preprocessing step. Such masks are usually computed by performing a simple thresholding operation on the images, excluding any pixels below a certain value from subsequent computations. While this can improve results in some cases, in practice such a simple operation generally does not capture the complexity of shadows present in the image and often results in useful data being rejected. This is of particular importance when working with small datasets or when the data being considered is severely corrupted, in which case it is important that reconstruction approaches are able to extract any relevant information from all the data presented. Due to this, robust methods should have the capacity to adapt to shadows in images without the use of a shadow mask. As such, all evaluations included here are performed without the use of a shadow mask.
TABLE I: Result on full, uncorrupted DiLiGenT datasets

| Dataset | PDNLV Mean Angular Error (degrees) | DLPI Mean Angular Error (degrees) | DNLV Mean Angular Error (degrees) | CBR Mean Angular Error (degrees) | SR Mean Angular Error (degrees) | RPCA Mean Angular Error (degrees) | LS Mean Angular Error (degrees) | PDNLV Median Angular Error (degrees) | DLPI Median Angular Error (degrees) | DNLV Median Angular Error (degrees) | CBR Median Angular Error (degrees) | SR Median Angular Error (degrees) | RPCA Median Angular Error (degrees) | LS Median Angular Error (degrees) |
|---------|-----------------------------------|-----------------------------------|-----------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| Ball    | 3.60                              | 3.99                              | 3.82                              | 6.78                            | 2.08                            | 3.20                            | 4.10                            | 1.95                              | 2.44                              | 1.85                              | 2.06                            | 2.02                            | 2.02                            | 2.41                             |
| Cat     | 6.40                              | 8.29                              | 7.10                              | 8.05                            | 6.73                            | 7.96                            | 8.44                            | 3.58                              | 6.52                              | 6.15                              | 3.88                            | 5.75                            | 6.03                            | 6.52                             |
| Pot1    | 6.99                              | 8.88                              | 8.67                              | 8.57                            | 7.28                            | 8.81                            | 8.89                            | 3.70                              | 6.64                              | 6.39                              | 4.15                            | 5.29                            | 6.61                            | 6.65                             |
| Bear    | 8.51                              | 8.31                              | 8.32                              | 9.77                            | 6.01                            | 7.89                            | 8.39                            | 6.34                              | 6.12                              | 6.20                              | 7.07                            | 4.30                            | 6.01                            | 6.18                             |
| Pot2    | 10.37                             | 14.57                             | 13.88                             | 10.36                           | 11.98                           | 11.94                           | 14.65                           | 7.84                              | 11.62                             | 12.79                             | 6.95                            | 8.32                            | 10.07                           | 11.60                            |
| Buddha  | 13.56                             | 14.91                             | 14.72                             | 14.90                           | 11.11                           | 13.88                           | 14.92                           | 7.84                              | 10.54                             | 10.33                             | 8.85                            | 7.74                            | 9.25                            | 10.54                            |
| Goblet  | 15.49                             | 18.43                             | 17.69                             | 15.10                           | 15.53                           | 15.14                           | 18.50                           | 11.13                             | 15.75                             | 16.04                             | 9.35                            | 12.23                           | 11.35                           | 15.70                            |
| Reading | 20.28                             | 19.66                             | 19.58                             | 19.39                           | 12.56                           | 17.42                           | 19.80                           | 13.90                             | 12.49                             | 12.51                             | 13.39                           | 7.18                            | 11.64                           | 12.50                            |
| Cow     | 21.30                             | 25.38                             | 17.58                             | 15.08                           | 22.82                           | 11.06                           | 25.60                           | 21.82                             | 26.13                             | 12.30                             | 13.35                           | 21.32                           | 9.38                            | 26.32                            |
| Harvest | 20.89                             | 30.55                             | 27.07                             | 26.93                           | 26.80                           | 25.50                           | 30.62                           | 15.36                             | 25.11                             | 23.34                             | 22.71                           | 19.00                           | 20.32                           | 25.33                            |

A. Evaluation on Uncorrupted DiLiGenT Dataset

We first investigate the performance of our methods on the DiLiGenT dataset. For each object, we use all 96 images present in the original dataset and do not add any additional corruptions to these images. For PDNLV, we set $p = 2$ for each dataset except for Harvest, for which we set $p = 3$. The results of these trials are presented in Table I.

As this table illustrates, PDNLV outperforms all existing approaches on 4 of the 10 objects. In cases where our methods do not outperform existing approaches, with the exception of the Reading and Cow objects, we achieve performance comparable to the best performer. We believe the primary strength of our algorithms are in constructing normal vectors from images much less pristine than the DiLiGenT dataset. However, these results illustrate that our method is still able to perform comparably to or better than methods developed specifically designed to operate on large, clean datasets.

In practical situations, 96 images under varying lighting directions may not be available for an object of interest. As such, it is important that methods are able to handle smaller datasets. Figure I and II illustrate the performance of each method on the uncorrupted DiLiGenT Cat and Harvest datasets with a varying number of images present. Here we randomly sample without replacement the images from the original 96 images, averaging the results across 10 such samplings.

In both of these cases, it is clear that, for nearly every size dataset, PDNLV is able to produce the most accurate reconstructions.

B. Evaluation on Corrupted DiLiGenT Dataset

We next compare the performance of our method against the performance of existing approaches on images corrupted with Poisson noise. We first subsample the DiLiGenT Pot2 dataset to 20 images. We then add random Poisson noise to these images, varying the SNR. The results of this experiment are illustrated in Figure VI.

From this figure it is clear that our dictionary learning based approaches are significantly more robust to high levels of non-sparse corruptions than other approaches. In particular, for SNR values below 10 dB, a dictionary learning based approach outperforms other approaches by 5 to 10 degrees. Further, the error varies significantly less for dictionary learning based approaches than for other approaches—the normal vector reconstructions are much more stable and robust to these errors than those constructed by other approaches.

Figures V and VI illustrate the actual normal vector reconstructions and corresponding error maps produced on the Pot2 dataset containing 20 images at a noise level of 5 dB. As these figures illustrate the dictionary learning based reconstructions seemed to be much less affected by noise than the other approaches. Figure V illustrates the error maps of the normal vectors constructed by PDNLV on the DiLiGenT Cat dataset with varying amounts of noise added. As this figure shows, the errors present in the normal vectors are roughly constant as the noise changes—PDNLV is robust against such corruptions and is not significantly affected by the presence of noise.

C. Evaluation on non-DiLiGenT Datasets

In addition to the DiLiGenT dataset, we also consider the dataset [52]. This dataset contains images of several real objects yet no ground truth normal vectors are available. In order to obtain an estimate of the ground truth normal vectors, we assume the objects present in this dataset follow a truly Lambertian model. While this assumption does not hold exactly, the objects present are matte in appearance and thus nearly Lambertian. To determine an estimate of the ground truth normal vectors, we simply utilize the standard least squares approach on the original uncorrupted data.

When running trials on the DiLiGenT dataset, even when corrupted by noise, a primary difficulty is to simply estimate the surface given that the data is fundamentally non-Lambertian. Thus, previous trials do not necessarily evaluate the ability of each method to estimate a Lambertian surface in the presence of noise. Since the majority of the methods we are investigating fundamentally assume an underlying Lambertian model, we will assume our data is Lambertian, determine our normal vectors based on this assumption, and evaluate the ability of each method to reject corruptions given underlying data that is ideal. Thus, this experiment illustrates the robustness of each method to noise when our data otherwise perfectly follows the modeling assumptions. These results are given in Figure VII.

As this figure illustrates, for high SNR, the average errors approach 0, as we would expect since the modeling assumptions are satisfied. In the high noise case, similar to the performance on the DiLiGenT dataset, the dictionary learning approaches are significantly more robust to the presence of noise.
based approaches are significantly more robust to imperfections than our other existing approaches. Unlike before, DLPI now outperforms both DLNV and PDLNV. This again follows from the fact that we have here assumed the underlying data is Lambertian—an assumption DLPI fundamentally relies on. We see then that in the high noise case, DLPI seems to be the most robust to noise when working with Lambertian objects.

We also evaluate the qualitative performance of our algorithms on this dataset. Figure 8 gives images of the ground truth normal vectors—computed on the uncorrupted data using the least squares approach—as well as the normal vectors computed by each method on data corrupted with Poisson noise with SNR equal to 5 dB.

Figure 9 illustrates the errors associated with each normal vector. As Figure 7 also indicated, the errors present in the normal vectors computed by the dictionary learning based methods are significantly lower than those obtained by other approaches. We note further that, for portions of the cat that are smooth, DLPI is able to effectively eliminate all error while still exhibiting some error in regions with sharper edges.

Finally, Figures 10 provides images of the surfaces computed from the normal vectors using the method outlined in [47]. Qualitatively, we see that the surfaces produced by the dictionary learning approaches are much smoother and, to the eye, seem to be more accurate representations of the actual surface. In contrast, the surfaces computed by other methods, though preserving the general shape, tend to be rather bumpy, compromising the appearance of the true surface. In particular,
we notice that, while smooth, DLNV exhibits certain artifacts visible as a flattened region on the side of the head. DLPI produces an extremely smooth surface though loses definition around the edges while PDLNV, though slightly less smooth, retains much of the sharpness of the ground truth surface.

D. Algorithm Properties

Finally, we investigate how tuning the dictionary learning model can affect the results. Beyond the weighting parameters, multiple additional parameters in the model can be altered. The size of the dictionary atoms—how large the patches extracted from the images or normal vectors must be—can be changes as can the degree to which adjacent patches overlap and what the dictionary atoms are initialized to. In addition, we may vary the number of dictionary atoms present in our dictionary. This allows us to produce over-complete or under-complete dictionaries—dictionaries where there is a smaller or larger number of atoms than degrees of freedom within each atom. When enforcing the sparsity penalty \( \| B \|_0 \), we can also choose to use soft thresholding rather than hard thresholding—

replacing the \( \ell_0 \) norm with the \( \ell_1 \) norm. In the case of PDNLV, we may also alter the value of \( p \) to attempt to capture different surface properties.

In the preceding results, we have used fairly standard dictionary learning parameters commonly found in applications. Specifically, our patches are \( 8 \times 8 \) for DLPI and \( 8 \times 8 \times 3 \) for DLNV and PDNLV. In each case, adjacent patches overlap each other four pixels in each direction of the first two dimensions. We have used a dictionary containing the same number of atoms as degrees of freedom in each atom—64 atoms for DLPI and 192 atoms for DLNV and PDNLV. Dictionary atoms are initialized to columns of the discrete cosine transform matrix. Finally, we use the hard thresholding operator as we originally specified in [18], [19], and [20].

While the problem is non-convex, the algorithm presented in Section IV-D is guaranteed to monotonically decrease the objective function value at each iteration. Figure 11a illustrates the value of the objective function at each iteration for PDNLV with \( p = 2 \). Figure 11b gives the corresponding mean angular error of the normal vectors produced at each iteration. While the cost is guaranteed to decrease at each iteration, angular error can increase. Empirically, we have found, however, that the error does typically decreases with iteration. Figure 11c gives the percentage of values of \( B \) that are nonzero at each iteration. Since the cost function penalizes \( B \) for containing nonzero values, the optimization process often results in \( B \) being relatively sparse.

Each plot in Figure 11 illustrates the performance of PDNLV for several dictionary sizes. Of particular interest is how the size of the dictionary affects the mean angular error. As Figure 11b illustrates, larger dictionaries typically perform better than smaller, undercomplete dictionaries. However, we do not observe a significant boost in performance using an overcomplete versus a complete dictionary. Note that, due to dictionary dimensions of \( 8 \times 8 \times 3 \) used here, a dictionary of size 192 is complete.

We next illustrate the affect changing the value of \( p \) has on the performance of PDNLV. Figure 12 gives the angular error of the computed normal vectors when varying the number of images in the DiLiGenT Pot1 dataset for several values of \( p \). As this figure illustrates, in this case \( p = 2 \) is the optimal value. Further, this plot shows that, when working with a relatively small number of images, \( p = 3 \) in general performs poorly but, as the number of images increases, \( p = 3 \) overtakes the performance of \( p = 1 \). We have found that, in general, setting \( p = 2 \) typically produces optimal results yet for certain cases, such as when working with the DiLiGenT Harvest dataset, \( p = 3 \) does outperform \( p = 2 \).

Figure 13 illustrates the initial and final dictionaries produced by PDNLV. The initial dictionary is simply constructed as a DCT matrix. Each patch here, while originally \( 8 \times 8 \times 3 \)—where the third dimension corresponds to the \( x \), \( y \), and \( z \) components of the normal vector—is plotted as an \( 8 \times 24 \) image. While we see that, due to the initialization strategy used, the initial dictionary contains much uniformity across the \( x \), \( y \), and \( z \) dimensions, the optimization modifies this to ultimately capture different structure in each dimension—illustrated by the patches often exhibiting significant contrast.
between their different components.

VI. CONCLUSION

In this work, we proposed three methods for applying dictionary learning to photometric stereo. Each method seeks to represent some form of the data—either the original images or the reconstructed normal vectors—as sparse with respect to an adaptive dictionary. We showed through extensive numerical studies that these approaches are significantly more robust than existing approaches in the high-noise regime while preserving accuracy in the low-noise regime. Dictionary learning is a general purpose adaptive regularization scheme, and, as such, it could be coupled with other, more complex, reflectance models from the photometric stereo literature to further improve reconstruction quality. We plan to investigate this line of inquiry in future work.

REFERENCES

[1] R. J. Woodham, “Photometric method for determining surface orientation from multiple images,” *Optical Engineering*, vol. 19, no. 1, pp. 191–199, 1980.
[2] S. Barksy and M. Petrou, “The 4-source photometric stereo technique for three-dimensional surfaces in the presence of highlights and shadows,” *IEEE PAMI*, vol. 25, no. 10, pp. 1239–1252, Oct. 2003.
[3] I. Sato, T. Okabe, Q. Yu, and Y. Sato, “Shape reconstruction based on similarity in radiance changes under varying illumination,” in *ICCV*. IEEE, 2007, pp. 1–8.
[4] T. Okabe, I. Sato, and Y. Sato, “Attached shadow coding: Estimating surface normals from shadows under unknown reflectance and lighting conditions,” in *ICCV*. IEEE, 2009, pp. 1693–1700.
[5] D. B. Goldman, B. Curless, A. Hertzmann, and S. M. Seitz, “Shape and spatially-varying brdfs from photometric stereo,” *IEEE PAMI*, vol. 32, no. 6, pp. 1060–1071, 2010.
[6] T.-P. Wu and C.-K. Tang, “Photometric stereo via expectation maximization,” *IEEE PAMI*, vol. 32, no. 3, pp. 546–560, Mar. 2010.
[7] M. Chandraker and R. Ramamoorthi, “What an image reveals about material reflectance,” in *ICCV*. IEEE, 2011, pp. 1076–1083.
[8] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma, “Robust photometric stereo via low-rank matrix completion and recovery,” *ACCV*, pp. 703–717, 2011.
[9] J. Ackerlmann, F. Langguth, S. Fuhrmann, and M. Goesele, “Photometric stereo for outdoor webcams,” in *CVPR*. IEEE, 2012, pp. 262–269.
[10] B. Shi, P. Tan, Y. Matsushita, and K. Ikeuchi, “Elevation angle from reflectance monotonicity: Photometric stereo for general isotropic reflectances,” *ECCV*, pp. 455–468, 2012.
[11] S. Ikehata, D. Wipf, Y. Matsushita, and K. Aizawa, “Robust photometric stereo using sparse regression,” in *CVPR*. 2012, pp. 318–325.
[12] M. Chandraker, J. Bai, and R. Ramamoorthi, “On differential photometric reconstruction for unknown, isotropic BRDFs,” *IEEE PAMI*, vol. 35, no. 12, pp. 2941–2955, 2013.
Mean Angular Error (degrees) | Cost Function Value | % Sparsity
---|---|---
10 | 11 | 12 | 13
10 | 11 | 10 | 15
11 | 12 | 13 | 20
14 | 25 | 14 | 25
15 | 30 | 15 | 30

Fig. 11: Cost, Angular Error, and Sparsity of Normal Vectors on DiLiGenT Cat Dataset with 20 Images, SNR = 20dB, and $p = 2$ for Several Dictionary Sizes

Fig. 12: Sweeping Number of Images in DiLiGenT Pot1 Dataset, Varying Value of $p$

Fig. 13: Initial and Final Dictionaries for Full, Uncorrupted DiLiGenT Pot1 Dataset with $p = 2$

[13] S. Ikehata, D. Wipf, Y. Matsushita, and K. Aizawa, “Photometric stereo using sparse bayesian regression for general diffuse surfaces,” *IEEE PAMI*, vol. 36, no. 9, pp. 1816–1831, 2014.

[14] S. Ikehata and K. Aizawa, “Photometric stereo using constrained bi-variate regression for general isotropic surfaces,” in *CVPR*, 2014, pp. 2179–2186.

[15] B. Shi, P. Tan, Y. Matsushita, and K. Ikeuchi, “Bi-polynomial modeling of low-frequency reflectances,” *IEEE PAMI*, vol. 36, no. 6, pp. 1078–1091, June 2014.

[16] H. Hayakawa, “Photometric stereo under a light source with arbitrary motion,” *JOSA A*, vol. 11, no. 11, pp. 3079–3089, 1994.

[17] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille, “The bas-relief ambiguity,” *International Journal of Computer Vision*, vol. 35, no. 1, pp. 33–44, Nov. 1999.

[18] A. L. Yuille, D. Snow, R. Epstein, and P. N. Belhumeur, “Determining generative models of objects under varying illumination: Shape and albedo from multiple images using SVD and integrability,” *International Journal of Computer Vision*, vol. 35, no. 3, pp. 203–222, 1999.

[19] A. S. Georghiades, “Incorporating the torrance and sparrow model of reflectance in uncalibrated photometric stereo,” in *ICCV*, vol. 2, 2003, pp. 816–823.

[20] M. K. Chandraker, F. Kahl, and D. J. Kriegman, “Reflections on the generalized bas-relief ambiguity,” in *CVPR*, 2005, pp. 788–795.

[21] N. G. Aldrdrin, S. P. Mallick, and D. J. Kriegman, “Resolving the generalized bas-relief ambiguity by entropy minimization,” in *CVPR*, June 2007, pp. 1–7.

[22] B. Shi, Y. Matsushita, Y. Wei, C. Xu, and P. Tan, “Self-calibrating photometric stereo,” in *CVPR*, June 2010, pp. 1118–1125.

[23] P. Favaro, “A closed-form solution to uncalibrated photometric stereo via diffuse maxima,” in *CVPR*, 2012, pp. 821–828.

[24] Z. Wu and P. Tan, “Calibrating photometric stereo by holistic reflectance symmetry analysis,” in *CVPR*, 2013, pp. 1498–1505.

[25] Y. Queau, F. Lauze, and J.-D. Durou, “Solving uncalibrated photometric stereo using total variation,” *Journal of Mathematical Imaging and Vision*, vol. 52, no. 1, pp. 87–107, 2015.

[26] M. Elad and M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. on Image Proc.*, vol. 15, no. 12, pp. 3736–3745, 2006.

[27] M. Aharon, M. Elad, and A. Bruckstein, “k-svd: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. on Signal Proc.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[28] S. Ravishankar and Y. Bresler, “MR image reconstruction from highly undersampled k-space data by dictionary learning,” *IEEE Trans. on Med. Imag.*, vol. 30, no. 5, pp. 1028–1041, 2011.

[29] S. Ravishankar, B. E. Moore, R. R. Nadakuditi, and J. A. Fessler, “LASSI: A low-rank and adaptive sparse signal model for highly accelerated dynamic imaging,” in *IVMSP Workshop*, 2016, pp. 1–5.

[30] B. Shi, Z. Wu, Z. Mo, D. Duan, S.-K. Yeung, and P. Tan, “A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo,” in *CVPR*, 2016.

[31] A. J. Wagenmaker, B. E. Moore, and R. R. Nadakuditi, “Robust photometric stereo using learned image and gradient dictionaries,” in *ICIP*, IEEE, 2017.

[32] F. N. Coleman and R. Jain, “Obtaining 3-dimensional shape of textured and specular surfaces using four-source photometry,” *Computer graphics and image processing*, vol. 18, no. 4, pp. 309–328, 1982.

[33] F. Solomon and K. Ikeuchi, “Extracting the shape and roughness of specular lobe objects using four light photometric stereo,” *IEEE PAMI*, vol. 18, no. 4, pp. 449–454, 1996.

[34] F. Verbiest and L. V. Gool, “Photometric stereo with coherent outlier handling and confidence estimation,” in *CVPR*, June 2008, pp. 1–8.

[35] C. Yu, Y. S. Yeo, and S. W. Lee, “Photometric stereo from maximum feasible lambertian reflections,” in *ECCV*, 2010, pp. 115–126.
[36] M. Chandraker, S. Agarwal, and D. Kriegman, “Shadowcuts: Photometric stereo with shadows,” in CVPR, June 2007, pp. 1–8.

[37] T. Zickler, S. P. Mallick, D. J. Kriegman, and P. N. Belhumeur, “Color subspaces as photometric invariants,” International Journal of Computer Vision, vol. 79, no. 1, pp. 13–30, 2008.

[38] K. Sunkavalli, T. Zickler, and H. Pfister, “Visibility subspaces: Uncalibrated photometric stereo with shadows,” ECCV, pp. 251–264, 2010.

[39] Y. Mukaigawa, Y. Ishii, and T. Shakunaga, “Analysis of photometric factors based on photometric linearization,” JOSA A, vol. 24, no. 10, pp. 3326–3334, 2007.

[40] M. Oren and S. K. Nayar, “Generalization of the lambertian model and implications for machine vision,” International Journal of Computer Vision, vol. 14, no. 3, pp. 227–251, 1995.

[41] H.-S. Chung and J. Jia, “Efficient photometric stereo on glossy surfaces with wide specular lobes,” in CVPR, June 2008, pp. 1–8.

[42] N. G. Alldrin and D. J. Kriegman, “Toward reconstructing surfaces with arbitrary isotropic reflectance: A stratified photometric stereo approach,” in ICCV, 2007, pp. 1–8.

[43] N. Alldrin, T. Zickler, and D. Kriegman, “Photometric stereo with non-parametric and spatially-varying reflectance,” in CVPR, June 2008, pp. 1–8.

[44] T. Higo, Y. Matsushita, and K. Ikeuchi, “Consensus photometric stereo,” in CVPR, June 2010, pp. 1157–1164.

[45] A. Hertzmann and S. M. Seitz, “Example-based photometric stereo: Shape reconstruction with general, varying BRDFs,” IEEE PAMI, vol. 27, no. 8, pp. 1254–1264, Aug. 2005.

[46] M. Holroyd, J. Lawrence, G. Humphreys, and T. Zickler, “A photometric approach for estimating normals and tangents,” ACM Trans. on Graphics, vol. 27, no. 5, p. 133, 2008.

[47] T. Simchony, R. Chellappa, and M. Shao, “Direct analytical methods for solving poisson equations in computer vision problems,” IEEE PAMI, vol. 12, no. 5, pp. 435–446, May 1990.

[48] S. Ravishankar, R. R. Nadakuditi, and J. A. Fessler, “Efficient sum of outer products dictionary learning (SOUP-DIL) - the \( \ell_0 \) method,” arXiv preprint arXiv:1511.08842, 2015.

[49] R. Gribonval and K. Schnass, “Dictionary identification–sparse matrix-factorization via \( \ell_1 \) -minimization,” IEEE Trans. on Inform. Theory, vol. 56, no. 7, pp. 3523–3539, 2010.

[50] S. Ravishankar, R. R. Nadakuditi, and J. A. Fessler, “Efficient sum of sparse outer products dictionary learning (SOUP-DIL),” CoRR, vol. abs/1511.06333, 2015.

[51] N. Parikh and S. Boyd, “Proximal algorithms,” Found. Trends Optim., vol. 1, no. 3, pp. 127–239, Jan. 2014.

[52] Y. Xiong, A. Chakrabarti, R. Basti, S. J. Gortler, D. W. Jacobs, and T. E. Zickler, “From shading to local shape,” IEEE PAMI, vol. 37, no. 1, pp. 67–79, 2015.