Guided Robust Matte-Model Fitting for Accelerating Multi-light Reflectance Processing Techniques

Ruggero Pintus\textsuperscript{1}
raggio.pintus@crs4.it

Andrea Giachetti\textsuperscript{2}
andrea.giachetti@univr.it

Giovanni Pintore\textsuperscript{1}
giovanni.pintore@crs4.it

Enrico Gobbetti\textsuperscript{1}
enrico.gobbetti@crs4.it

\textsuperscript{1} CRS4, Visual Computing Group
Pula (CA), Italy

\textsuperscript{2} University of Verona, CS Dept.
Verona, Italy

Abstract

The generation of a basic matte model is at the core of many multi-light reflectance processing approaches, such as Photometric Stereo or Reflectance Transformation Imaging. To recover information on objects’ shape and appearance, the matte model is used directly or combined with specialized methods for modeling high-frequency behaviors. Multivariate robust regression offers a general solution to reliably extract the matte component when source data is heavily contaminated by shadows, inter-reflections, specularity, or noise. However, robust multivariate modeling is usually very slow. In this paper, we accelerate robust fitting by drastically reducing the number of tested candidate solutions using a guided approach. Our method propagates already known solutions to nearby pixels using a similarity-driven flood-fill strategy, and exploits this knowledge to order possible candidate solutions and to determine convergence conditions. The method has been tested on objects with a variety of reflectance behaviors, showing state-of-the-art accuracy with respect to current solutions, and a significant speed-up without accuracy reduction with respect to multivariate robust regression.

1 Introduction

Multi-light reflectance acquisition and processing techniques, such as Photometric Stereo (PS) \cite{47}, Reflectance Transformation Imaging (RTI) \cite{28}, and Polynomial Texture Maps (PTM) \cite{26}, aim at characterizing surfaces by observing them from a fixed point of view under different lighting conditions.

The generation of a matte model from image measurements is at the core of many successful approaches, which use it directly, or augment it with specialized methods for modeling high-frequency components (see Sec. 2). Such recovered low-frequency information, typically obtained by fitting low-order polynomials or Hemispherical Harmonics (HSH) to the captured data, is used for a variety of applications, including relighting\cite{15,26}, enhanced

© 2017. The copyright of this document resides with its authors. It may be distributed unchanged freely in print or electronic forms.
visualization[25], normal/albedo extraction[10], as well as material classification and segmentation [1, 22].

Reliably fitting the matte model to measurements in the presence of non-Lambertian phenomena (e.g., shadows, anisotropic behaviors, gloss, specularity) is easily prone to errors, and a variety of methods, from heuristic trimming of outliers to robust fitting procedures, have been proposed (see Sec. 2). The most general solutions are based on multivariate robust regression [10, 54], which is able to automatically separate inlier and outlier measurements according to the fitted models, to identify matte and non-matte behavior, and to provide a more reliable diffuse interpolation at non-sampled locations due to reduced risks of over-fitting. However, the highest accuracy of robust algorithms comes at the price of increasing computational costs, due to the need to evaluate a large number of hypotheses. This makes those solutions impractical in most scenarios, especially as the number of parameters increases [25].

Our contribution harnesses the capacity of robust statistics to be resistant to highly contaminated data, but exploits a computational framework that provides a considerable speed-up while keeping the fitting error comparable to current slow robust estimators, by using a guided sampling approach to drastically reduce the number of computed fitting trials. We first show how the efficiency of a Least-Median of Squares (LMS) robust regression can be improved by providing a fitting tolerance threshold and by associating to each measurement a weight proportional to the a-priori likelihood that the measurement is part of the solution. This additional information provides a means for likely generating good candidate solutions early-on and for recognizing an optimal candidate solution as soon as it is generated. We then show how we can generate the required information for such a guided algorithm by propagating the information from already computed pixels using a similarity-driven region growing strategy implemented using multi-core parallelism.

While guided sampling approaches have been successfully proposed to speed-up computation in many Computer Vision areas, see Sec. 2, this is to the best of our knowledge the first time that an optimization of robust fitting is proposed for the specific case of parameter estimation from MLICs. In particular, we introduce in this context a specific guided sampling approach based on appearance profile similarity, as well as parallelizable techniques to propagate results from already known pixels to unknown ones. As demonstrated in Sec. 4, the proposed approach, evaluated both on synthetic and real-world data provides a significant speed-up without accuracy reduction, and can be employed as a building block in RTI, PTM or PS pipelines. Moreover, we show how generic robust fitting approaches (baseline or accelerated), without domain specific modifications, are generally on-par with specifically tailored state-of-the-art PS techniques, and offer the additional possibility of being applicable, without modification, to other estimation techniques, such as higher order polynomial fitting for PTM computation. While the method has been implemented, tested and validated by using a LMS solver, it is possible to apply our strategy to many other iterative robust regressions, such as M-estimators or iteratively re-weighted Least Squares (LS). Moreover, the algorithm is fully data-driven, and does not need user-defined, tuned parameters.

2 Related work

Computing surface properties from multi-light data is a very old and still very active research area [16, 25, 26, 47]. We discuss here only the approaches strongly related to ours, and, in particular, works that aim at fitting a low-frequency model to reflectance measurements.
For a wider coverage, we refer the reader to established surveys in surface reflectance capture [39], multi-light computational frameworks [1], digital modeling of material appearance [9], and geometric analysis in cultural heritage [30].

The simplest technique is the classical Lambertian PS approach [47], which assumes image intensity proportional to the cosine between the surface normal and the light direction, and exploits it to recover albedo and normals for multiple images. PTM [26] is presented as a fitting-based generalization of the simplest PS, and extends the basic Lambertian model with a 6-term polynomial; two of its coefficients are the two components of the normal vector (Lambertian term), while the others try to capture the remaining low-frequency signals. A linear LS regression is used to fit the model. Subsequent works presented a wide range of follow-ups to RTI-based surface characterization [11, 17, 28, 46], and proposed alternatives to define and represent surface properties, such as spherical and hemispherical harmonics [29], bivariate Bernstein polynomials [15], bi-polynomial functions [35], and discrete modal decomposition [31]. Unfortunately, all those methods cannot cope well with the presence of non-diffuse phenomena such as gloss, highlights, specularity, or shadows. The underlying computational framework is strongly affected by those leverage points, and exhibits high error in the resulting interpolation, albedo and normal maps.

Many techniques have been proposed to detect/remove outliers from the fitting computation. The first attempts have been made in the field of Photometric Stereo. Since three inliers are enough to compute the normal maps in the diffuse case, four- or five-light setups are used to exclude the lights that generates significantly distorted outcome [1, 6, 13, 28]. In the case of N lights (i.e., N-dimensional appearance profile), Wenger et al. [45] heuristically eliminate the half darkest measurements and the brightest 10% before applying LS. Conversely, Willems et al. [46] gradually prune unreliable values through iterative normal estimations, while Yuille and Snow [52] rely on the integrability constraint. Monotonicity is another reflectance property exploited to deal with non-Lambertian diffuse behaviors [18, 34]. Some works exploit the difference between the light and object chromaticity, by exploiting the dichromatic reflection model [23, 50]. Component based modeling [22] is another more general way to separate reflectance terms by decomposing input into a local and global component. Others handle outliers within a rank minimization [28] or matrix factorization frameworks [21]. Recent techniques employ statistical models as labeling problems coupled with shadow graphs [6, 21], maximum-likelihood (ML) [7], or Markov random fields (MRFs) [8, 49], as well as non-parametric data-driven approximation of object reflectance [1]. Robust regression has proven to be a general powerful approach for RTI computational problems [14, 15, 36]. Ikehata et al. [20] present a hierarchical Bayesian approximation to deal with non-Lambertian corruptions, and to estimate reliable surface normals. Zhang and Drew [54] propose to use a LMS approach to solve for a 6-term polynomial diffuse component. Their robust method automatically discards outliers (up to 50% of input measurements), obtains a reliable interpolation, and retrieves a good estimation of albedo and normals. Once they compute a reliable matte component, they can isolate the remaining high-frequency signal, and model it with Radial Basis Functions (RBFs). Although some of the previous approaches can provide a good appearance representation, some important issues remain unsolved. The methods that throw away a fixed or proportional number of values with conservative heuristics do not ensure both the removal of leverage points or an adequate level of Signal-to-Noise Ratio (SNR) [1, 6, 13, 21, 21, 36]. Others have complex rationales based on too constrained assumptions [28], initial guesses [14], or on a-priori knowledge [54]. Most of them have also to address the trade-off between accuracy and a heavy computational load [6, 23]. Previously proposed robust methods [14, 23] are very
general, since they do not impose any prior knowledge or heuristics about outlier distribution, but, since they achieve robustness and generality by blindly applying a classic robust fitting method [32] independently for each pixel, they require very high computational times even in parallel settings.

For this reason, we investigated possible strategies to reduce the computational complexity of robust fitting methods. Our solution follows the general idea of estimation by guided sampling, which has proven effective in many areas of Computer Vision, in particular for increasing the chance to pick inlier samples at each iteration of RANSAC-style estimators [3, 23, 35, 49]. The common idea underpinning guided sampling is to exploit prior knowledge to increase the density of samples in the regions where the correctness probability of the hypothesis is higher, rather than sampling the search space uniformly [41]. With an accelerated hypothesis generation [5], it is thus possible to reach a low residual error and an acceptable convergence within a reduced amount of iterations.

Our guided sampling solution introduces a speed-up strategy that harnesses spatial consistency, appearance profile similarity, and data-driven estimation of outlier distribution among neighboring pixels. To the best of our knowledge, this is the first algorithm that effectively exploits information from past pixels to drive robust per-pixel computation in the field of MLIC processing. Although tested with LMS, the proposed rationale is general enough to be easily adapted to increase computational efficiency of other, general iterative robust estimators applied to RTI processing pipeline. Moreover, while we validate our method by fitting polynomial bases, the theoretical underpinning of our solution can be generally used with other types of basis functions, e.g., HSH [4, 48].

3 Method

Multi-light reflectance data is acquired by taking from the same pixel-registered viewpoint a set of photographic images of an object, with each image illuminated from a different direction. As a result, a calibrated processing pipeline receives as input a so-called per-pixel appearance profile, i.e., a list of pairs that couple a measured value with a light direction and intensity. Surface shape and diffuse appearance can then be characterized for each pixel by converting this information to shape and appearance descriptors. In this work, we focus on matte model extraction, in which the set of multi-light measurements is fitted to an analytical low-frequency model (e.g., a polynomial). Our technique is based on two main pillars. The first (Sec. 3.2) is the modification of the standard robust regression method (Sec. 3.1) in order to provide additional parameters that allow for an early termination of the iterative process, and a resulting speed up of the atomic, per-pixel fitting coefficient computation. The second contribution (sec. 3.3) is a global strategy to automatically compute a data-driven sparse initialization of those parameters, and a region-growing procedure to propagate that information to the entire image domain. As we will see, the structure of the proposed algorithm is highly parallelizable, and can be implemented as efficiently as current pixel-independent solutions on multi-core machines.

3.1 Standard per-pixel robust estimator

The Least Median of Squares (LMS) estimator [32], which ensures up to 50% breakdown point (half input measurements can be outliers) [32], has already proved to be capable to provide excellent results for matte-model extraction in general settings [11, 12, 25]. It is an
We estimate a robust standard deviation as the median of squared residuals for each profile element, take the best 50% of them (assumed to be inliers), and launch a refinement step with a more overdetermined system. The presence of Gaussian noise is also an issue for the efficiency of LMS. After the robust iterations terminate a further refinement step is typically added to improve the solution. As proposed by Rousseeuw [32], we use this information to compute a least-squares procedure for all the inliers.

It should be noted that, since the solver used in each iteration takes the minimal number of unknown coefficients in our fitting model. For each subset $J$, a generic solver (e.g., LS) is used to find the corresponding solution $p_J$. Then, the median value $M_J$ is obtained as $M_J = med \, r_i^2 (p_J)$. The final solution will be the $p_J$ that corresponds to the minimal among all $M_J$'s.

The number of trials $m$ that ensures convergence of the algorithm is computed as $m = log (1 - P) / log [1 - (1 - \varepsilon)^p]$, where $P$ is the probability that at least one of the $m$ subsets contains only inliers, and $\varepsilon$ is the percentage of outliers expected in the input data. This is the main reason of the heavy computational times. Since we want to ensure high accuracy, we need to set $P \approx 1$, and if we consider the highest allowed fraction of outliers, we have to set $\varepsilon = 0.5$ (50% breakdown point). We always use those values for our experiments. With those numbers, as $p$ slightly increases (e.g., 6-term PTM polynomial), $m$ tends to rapidly become very high; for instance, over 300 trials per pixel are required to have a probability of 99% of generating a non-contaminated base when up to 50% outliers might be present.

### Algorithm 1: Standard vs. Guided LMS algorithm.

**Standard Least Median of Squares**

**Input:**
- $AP$: appearance profile
- $\varepsilon$: fraction of outliers
- $P$: probability of picking at least an inlier subset

**Output:** Solution $S$

**begin**

1. Compute number of trials $nTrials(\varepsilon, P)$
2. $sAP_p = \text{Random sampling of an } AP\text{'s subset of } p \text{ cardinality}$
3. Compute fitting coefficients from $sAP_p$
4. Compute fitting coefficients from elements with the best half residuals (Refinement)
5. Evaluate the median residual $M_J = med \, r_i^2$
6. Update solution $S$ if $M_J$ is less than current minimal residual

**while** $J < nTrials$

1. Compute inliers for $S$ with $r_i^2 \leq (2.5\sigma)^2$
2. Compute final fit $S$ using all the inliers

**return** $S$

**Guided Least Median of Squares**

**Input:**
- $AP$: appearance profile
- $wAP$: appearance profile weights
- $th$: residual threshold
- $\varepsilon$: fraction of outliers
- $P$: probability of picking at least an inlier subset

**Output:** Solution $S$

**begin**

1. Compute number of trials $nTrials(\varepsilon, P)$
2. $sAP_p = \text{Weighted random sampling of an } AP\text{'s subset of } p \text{ cardinality}$
3. Compute fitting coefficients from $sAP_p$
4. Compute fitting coefficients from elements with the best half residuals (Refinement)
5. Evaluate the median residual $M_J = med \, r_i^2$
6. Update solution $S$ if $M_J$ is less than current minimal residual

**while** $(J < nTrials) \text{ or } (M_J < th)$

1. Compute inliers for $S$ with $r_i^2 \leq (2.5\sigma)^2$
2. Compute final fit $S$ using all the inliers

**return** $S$
3.2 Guided per-pixel robust estimator

We modify that standard pipeline in order to increase efficiency but keeping a similar accuracy (see Algorithm 1 right, modifications highlighted in red). Typical LMS uses a Monte Carlo technique to draw the $m$ subsets in no particular order, and considers a uniform distribution of the likelihood that an appearance profile element was an inlier or an outlier. It is therefore in general impossible to reduce the number of iterations for early convergence.

In our case, we provide an additional list of weights that rank the elements by a given inlier probability (we will see in Sec. 3.3 how we obtain these probabilities). Now, each time we need to obtain a subset from the input data, we employ a weighted random picking, that ensures that the probability of picking an item is proportional to its weight. Such a weighted selection can be efficiently implemented using techniques employed in Reservoir sampling [12]. The basic idea is, at each time a subset has to be selected, to assign to each potential candidate $i$ a key $r^{1/w_i}$ where $r$ is a random number. By partially sorting the items, and select the top $p$, we can generate a sample at each trial. Since such a guided weighted selection favors good candidate solutions (according to a-priori probabilities encoded in weights), it is likely that the inlier set is identified in the first few iterations. Thus, a strategy in which the process is stopped as soon as the median residual $M_J$ is less than a given residual threshold likely decreases the computational cost.

Algorithm 2: Global fitting

\begin{algorithm}
\textbf{Input:}
- A\textit{Parray}: nxm 2D-array of appearance profiles
- $M$: number of sparse seed pixels
\textbf{Output:}
- nxm 2D-array of fitting coefficients
\begin{algorithmic}
\State Compute similarity map of $I$
\State Select a sparse set of $M$ seed pixels $S$
\For {pixel $p \in S$ do in parallel}
\State Compute fitting coefficients with $th_p = 0$ and uniform weights
\EndFor
\State Compute the residual threshold $th = \text{avg}_p (r_p + 2.5\sigma_p)^2$
\Do
\State Select candidate pixel set $C$
\For {pixel $c \in C$ do in parallel}
\State Compute fitting coefficients with $th_c = th$ and weights from the most similar, already processed, neighbor of $c$
\EndFor
\EndWhile
\State \Return nxm 2D-array of fitting coefficients
\end{algorithmic}
\end{algorithm}

3.3 Parallel guided estimation

The per-pixel robust estimator presented in Sec. 3.2 is employed on a global basis to convey a fitting solution for the entire image domain (see Alg. 2). In particular, we employ a region-growing approach to propagate already computed solutions to neighbor pixels, and to exploit this data-driven information to determine convergence conditions and anticipate process termination.

At the beginning of the procedure, we do not have any information about both expected residual and profile weights, neither globally nor on a per-pixel basis. For this reason, we select a small amount of evenly distributed random pixels across the image domain, and for those corresponding appearance profiles we compute a standard robust fitting, recording the solution, together with the fitting residual and the residual error for each light. This is slow,
but feasible when applied only on very few seed pixels. For the rest of the pixels, computed using our accelerated guided solution, we set a convergence threshold equal to the median of seed residuals. We then compute the solution for all pixels using a region-growing approach, which computes, in multiple passes, new pixel solutions only in the neighborhood of already computed ones. The nature of this procedure is highly suitable for parallel execution by selecting and computing all the pixels at a given iteration in parallel.

Each iteration of the region-growing pipeline starts by selecting as candidate pixels for processing the spatial neighbors of already computed pixels that exhibit a large similarity with them. To favor a similarity driven propagation, only the top 50% of most similar pixels are retained at each propagation step. The basic idea of this similarity-driven flood-fill approach is that neighboring pixels having a similar profile have a high probability of exhibiting a similar illumination behavior. Thus, the weights required for weighted random sampling can be derived from the known errors associated to each light in the most similar neighbor. In practice, we set the weight to be proportional to the residuals \( r_i^2 \) for each appearance profile element \( i \), remapped so that the minimum residual has a weight of 1, and the maximum residual has a weight of \( 1/255 \). These weights, together with the estimated error threshold, are used to control the guided pixel estimator.

In order to quickly compute the most similar neighboring pixels, a similarity map, depending only on input data, is computed in a pre-processing step. There is a vast amount of methods to measure similarity between appearance profiles \[24\]. Here we adopt a straightforward correlation-based measurement, that proved to behave well in our framework. Anyhow, the proposed pipeline is independent from this choice, and more fancy metrics can be employed.

4 Results

The proposed computational framework has been implemented on Linux using C++. We test the performance of our algorithm by using the data and the evaluation results of the DiLiGenT benchmark \[36\] (Fig. 1), which is a photometric stereo image dataset with calibrated directional lights. The dataset contains ten objects of varying reflectance properties. For each object, images with resolution of 612x512 depict the model illuminated from 96 different calibrated lighting directions. Ground-truth normal maps aligned at sub-pixel precisions are provided. All the benchmarks of our method have been executed on a PC with a Intel Core i7-4790K CPU (4 cores, 8 threads, 4.00GHz, 8MB cache) and 8GB RAM.

![Figure 1: Sample images from the DiLiGenT benchmark \[36\]](image-url)
4.1 Lambertian model fitting

The first test is the robust fitting of a Lambertian model aimed at the computation of the surface normals. This makes it possible to test the accuracy of the proposed approach with respect to the available ground-truth data in the DiLiGenT database, as well as to measure the performance in a worst-case situation for our general matte-model fitting method, given that only three parameters have to be derived from the measures. Although our aim is to propose a general strategy for guided robust fitting in the context of MLIC, we present here results and comparisons in the PS case, since it is a well researched subproblem with available benchmark data. Nevertheless, it should be noted that, while PS methods can exploit field based knowledge (monotonicity, symmetries, differential properties), we do not make any adaptation to our fitter.

![Figure 2: Error statistics. Comparison of our results with the state-of-the-art techniques surveyed in Shi et al. [36]. (Left) Distribution of the median angular error of all tested methods. The thin line depicts the minimum and maximum error interval, the box represents the 1\textsuperscript{st} to the 3\textsuperscript{rd} quartile range, while the central mark (white and black dot) is the median of all angular errors. The red dot is the error of the classic PS, while the error of our method is shown as a green dot. (Right) Median angular error for all approaches (our method highlighted in red).](image)

We compare our error statistics with those presented in the DiLiGenT benchmark, which are the classic photometric stereo [47] as well as some non-Lambertian methods with state-of-the-art performances [2, 16, 18, 19, 20, 34, 35, 48], and with the classic robust fitting [32]. In Figure 2 we show how our method, that uses a robust statistics technique to automatically separate Lambertian from non-Lambertian behavior, provides state-of-the-art results in normals estimation for all the datasets. Both the standard full per-pixel LMS estimation and our accelerated results are within the most accurate normal fitting methods. As noted by Shi et al. [36], particularly complicated BRDFs (Cow), spatially-varying materials (Goblet), and concave shapes (Reading, Harvest) remain challenging types of objects. Methods tailored for general isotropic reflectance (e.g., Shi et al. [34]) work well for some complicated data (e.g., Cow), but have bad error behavior for easier objects (e.g., Bear or Buddha).

In order to evaluate the performance, for each dataset we estimate the Lambertian model...
using the classic robust fitting and our guided version. In Table 1, we report the computational times together with the number of LS solve iterations, and we present the average, median, $1^{st}$ and $3^{rd}$ quartile of the angular error. We show how our method is capable of providing a relevant speed-up (an average of $\sim 10x$) without a significant loss in normal map accuracy. For display purposes, we show in Fig. 3 a visual comparison between the ground truth normal map of the dataset Buddha and those computed by the classic PS algorithm [47], the classic robust approach [32] and the proposed method. Again it provides a visual cue on how our speed-up strategy, compared to the classic robust method, does not affect the final reconstruction result. Low speed-ups are related to complex, still challenging objects, i.e., Cow and Harvest. Although our self-tuning adaptive method slightly increases the error for PS compared to the classic robust fitting, the increase is only $\sim 5\%$ on average, and results remain within the error range of specifically designed PS techniques, as shown in Fig 2. Our future work will concentrate on improving the estimation of the residual threshold.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Time</th>
<th># Solve</th>
<th>Avg.</th>
<th>Med.</th>
<th>1st Qr</th>
<th>3rd Qr</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ball</td>
<td>2.8s/0.2s</td>
<td>2.5M/122K</td>
<td>2.0/2.1</td>
<td>2.1/2.1</td>
<td>1.5/1.6</td>
<td>2.6/2.6</td>
<td>$\sim 14x$</td>
</tr>
<tr>
<td>Cat</td>
<td>8.3s/0.5s</td>
<td>7.2M/345K</td>
<td>6.4/6.7</td>
<td>5.7/5.9</td>
<td>3.7/3.8</td>
<td>7.9/8.6</td>
<td>$\sim 16x$</td>
</tr>
<tr>
<td>Pot1</td>
<td>10.2s/0.7s</td>
<td>9.2M/481K</td>
<td>7.4/8.0</td>
<td>5.3/6.0</td>
<td>3.2/3.4</td>
<td>8.9/10.1</td>
<td>$\sim 14x$</td>
</tr>
<tr>
<td>Bear</td>
<td>7.7s/0.7s</td>
<td>6.6M/463K</td>
<td>5.5/5.5</td>
<td>4.2/4.4</td>
<td>2.5/2.6</td>
<td>6.7/7.0</td>
<td>$\sim 10x$</td>
</tr>
<tr>
<td>Pot2</td>
<td>5.9s/0.8s</td>
<td>5.6M/547K</td>
<td>11.8/12.7</td>
<td>9.6/10.7</td>
<td>5.5/5.6</td>
<td>16.4/18.7</td>
<td>$\sim 7x$</td>
</tr>
<tr>
<td>Buddha</td>
<td>8.3s/0.7s</td>
<td>7.1M/449K</td>
<td>9.0/9.4</td>
<td>7.3/7.7</td>
<td>4.5/4.6</td>
<td>10.9/12.0</td>
<td>$\sim 11x$</td>
</tr>
<tr>
<td>Goblet</td>
<td>4.7s/0.6s</td>
<td>4.2M/357K</td>
<td>12.9/14.3</td>
<td>11.2/11.9</td>
<td>6.8/7.1</td>
<td>16.7/20.0</td>
<td>$\sim 8x$</td>
</tr>
<tr>
<td>Reading</td>
<td>4.5s/0.7s</td>
<td>4.4M/461K</td>
<td>12.8/13.3</td>
<td>7.2/7.4</td>
<td>4.2/4.3</td>
<td>14.7/16.1</td>
<td>$\sim 7x$</td>
</tr>
<tr>
<td>Cow</td>
<td>4.4s/1.1s</td>
<td>4.2M/721K</td>
<td>21.3/24.0</td>
<td>21.9/26.1</td>
<td>13.2/14.9</td>
<td>29.2/33.4</td>
<td>$\sim 4x$</td>
</tr>
<tr>
<td>Harvest</td>
<td>9.3s/2.6s</td>
<td>8.8M/1.9M</td>
<td>24.3/25.2</td>
<td>18.5/19.6</td>
<td>8.0/8.6</td>
<td>34.5/35.9</td>
<td>$\sim 4x$</td>
</tr>
</tbody>
</table>

Table 1: **Lambertian Fitting.** Time and error statistics of the Lambertian Fitting with three unknown coefficients. Each cell compares the classic Robust method with our proposed Guided approach (R/G). The speed-up is $\sim 10x$ on average.

Figure 3: **Normal Maps.** Visual comparison between ground truth normal map (b) and those computed by the classic PS (c), the classic robust method (d), and our approach (e). We provide a significant speed-up ($\sim 11x$) with a preserved reconstruction quality.

### 4.2 Matte-model estimation for polynomial texture maps

Beside photometric stereo (or normal map) computation, our method is more general, since it aims at robustly fitting an appearance analytical model with a custom number of parameters. Testing the provided speed-up in the case of a model with only three coefficients is the worst case for us, not in terms of error comparison, but in terms of gain in computational time; this is because the more coefficients we have, the more the classic robust algorithm is...
computationally heavy, and the more our algorithm can prove to be more efficient by providing a higher speed-up for it. For this reason, in Table 4(a) we show the time statistics related to the commonly used 6-coefficient polynomial fitting [10], together with the number of LS solves performed during the classic and guided iterative routine. In this case the speed-up is greater than $\sim 20x$ for most of all the datasets, and $\sim 30x$ on average. In Figure 4(b) and Figure 4(c) we visually compare the matte model relighting of the Buddha model obtained with the 6-coefficient polynomial computed respectively by the classic robust and our guided approach; we produce a very similar outcome while significantly breaking down the computational time.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Time</th>
<th># Solve</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ball</td>
<td>48.4s/0.8s</td>
<td>21.3M/311K</td>
<td>$\sim 60x$</td>
</tr>
<tr>
<td>Cat</td>
<td>137.6s/2.3s</td>
<td>60.9M/891K</td>
<td>$\sim 60x$</td>
</tr>
<tr>
<td>Pot1</td>
<td>173.8s/2.8s</td>
<td>78.2M/1.1M</td>
<td>$\sim 63x$</td>
</tr>
<tr>
<td>Bear</td>
<td>129.5s/3.3s</td>
<td>55.8M/1.3M</td>
<td>$\sim 39x$</td>
</tr>
<tr>
<td>Pot2</td>
<td>104.0s/3.7s</td>
<td>47.9M/1.4M</td>
<td>$\sim 28x$</td>
</tr>
<tr>
<td>Buddha</td>
<td>138.9s/4.0s</td>
<td>60.7M/1.5M</td>
<td>$\sim 35x$</td>
</tr>
<tr>
<td>Goblet</td>
<td>80.5s/4.1s</td>
<td>35.4M/1.5M</td>
<td>$\sim 19x$</td>
</tr>
<tr>
<td>Reading</td>
<td>77.8s/3.9s</td>
<td>36.6M/1.5M</td>
<td>$\sim 20x$</td>
</tr>
<tr>
<td>Cow</td>
<td>77.5s/16.1s</td>
<td>35.6M/6.1M</td>
<td>$\sim 5x$</td>
</tr>
<tr>
<td>Harvest</td>
<td>159.9s/25.5s</td>
<td>74.6M/10.5M</td>
<td>$\sim 6x$</td>
</tr>
</tbody>
</table>

Figure 4: 6-Coefficient Fitting. Time statistics of the 6-coefficient polynomial fitting(a). Each cell compares the classic Robust method with our proposed Guided approach (R/G). The fitted coefficients from the robust and the guided algorithms produce the same relighting results (b)(c), while providing a $\sim 35x$ speed-up.

5 Conclusions

We have presented a new, guided multivariate robust regression to extract the matte component of a spatially-varying material from a multi-light data. We have proved how our similarity-driven flood-fill strategy is capable of exploiting knowledge of already computed pixels to accelerate per-pixel convergence and to produce a significant speed-up (on average greater than $\sim 10x$) of the overall computation. We have presented a series of testbeds with different reflectance behaviors, and proved that our acceleration strategy maintains state-of-the-art performances. Open problems and challenges still remain, mostly related to complex BRDFs. Future works will include the study of different fitting models beside polynomials (e.g., non-linear models such as rational functions), as well as an investigation on different strategies to select initial seeds, and to compute a better estimation of convergence thresholds to further increase speed-up while decreasing exit residuals.

Acknowledgments. This work was partially supported by the Scan4Reco project funded by European Union’s Horizon 2020 Framework Programme for Research and Innovation under grant agreement no 665091. We also acknowledge the contribution of Sardinian Regional Authorities under projects VIGEC and Vis&VideoLab.
References


