Retrieving Multiple Light Sources in the Presence of Specular Reflections and Texture

Pascal Lagger and Pascal Fua

Computer Vision Laboratory
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland

Abstract

Recovering multiple point light sources from a sparse set of photographs in which objects of unknown texture can move is challenging. This is because both diffuse and specular reflections appear to slide across surfaces, which is a well known physical fact. What is seldom demonstrated, however, is that it can be taken advantage of to address the light-source recovery problem. In this paper, we therefore show that, if approximate 3D models of the moving objects are available or can be computed from the images, we can solve the problem without any a priori constraints on the number of sources, on their color, or on the surface albedos.

Our approach involves finding local maxima in individual images, checking them for consistency across images, retaining the apparently specular ones, and having them vote in a Hough-like scheme for potential light source directions. The precise directions of the sources and their relative power are then obtained by optimizing a standard lighting model. As a byproduct we also obtain an estimate of various material parameters such as the unlighted texture and specular properties.

We show that the resulting algorithm can operate in presence of arbitrary textures and an unknown number of light sources of possibly different unknown colors. We also estimate its accuracy using ground-truth data.

Email address: {pascal.lagger,pascal.fua}@epfl.ch (Pascal Lagger and Pascal Fua).

1 Introduction

Let us consider the following scenario. We are given a set of images acquired by a fixed camera of a scene in which rigid objects may move between exposures. Assuming that no additional information is provided, we can use commercially available software packages to reconstruct the geometry of the moving objects and to estimate their 3D poses. By contrast, recovering the number and direction of the possibly multiple light sources and the object textures remains a difficult problem. This is because, for any given surface patch projected in several images, none of the photometric angles are preserved between views. This implies that both diffuse and specular reflections move across the surface.

The above scenario is highly relevant to many real-world applications such as de-lighting and relighting scene objects or adding new objects that blend correctly in the scene. For example, in the remastered versions of the *Star Wars* movie, new synthetic ships and robots were added to scenes that had been filmed a long time ago using real props [7]. Since both the original props and lighting parameters are probably long lost, a technique able to recover those parameters automatically from the old images of the moving props could have saved a great deal of time and money.

In this paper, we present an approach to recovering multiple point light sources out of a sparse set of registered photographs, provided that approximate 3D models of some moving scene objects are available or can be computed. Our approach involves finding local gray-level maxima in several images and deciding whether or not they represent specularities. The detected specularities are then used to count how many point light sources are in the scene and estimate their location. The precise directions of the sources, their relative power and their color are then obtained by optimizing a standard lighting model. As a byproduct, we also obtain an estimate of various material parameters such as the unlighted texture and specular properties.

It is well known that lighting maxima glide across the surface of a moving object according to the normals, whereas texture maxima tend to be detected at the same surface location in many images [17]. However, this has not been explicitly exploited to recover multiple light sources without making strong assumptions about object texture by any previously published technique we are aware of. The originality of our method is therefore to take advantage of this physical fact to distinguish between lighting and texture maxima and to implement a light-source detection scheme that relies upon consistency across images. Although, the current detection algorithm is not designed to deal with very extended light sources, it *is* designed to handle sources that are not at infinity and are not truly point-light-sources.

As shown in Figs. 1 and 2, the resulting method can operate in presence of arbitrary textures and multiple light sources of possibly different unknown colors. This
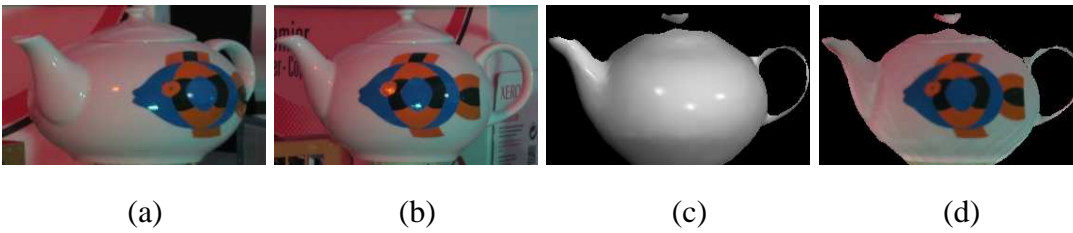
(a) (b) (c) (d)

Fig. 1. Recovering the number of sources and their directions in a scene lighted by three different light sources of unknown and different colors. To see this, we suggest that the reader view these images in color. (a,b) Two of nine input photographs. (c) Shaded view of the 3D model using the recovered light sources. Note that the specularities appear at the correct places. (d) The recovered albedo map is free of specular artifacts.
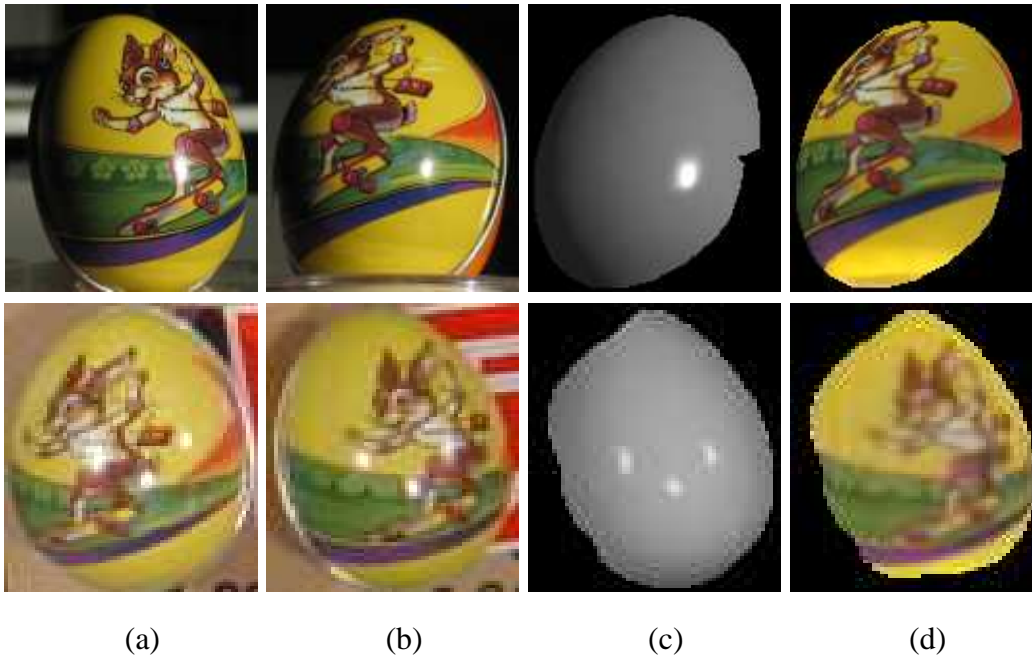

(a) (b) (c) (d)

Fig. 2. Lighting parameters recovery in presence of texture. The first row shows the recovery for one light source from *high* resolution images. The second row shows the recovery for three light sources from *low* resolution images. (a,b) Two out of nine input images. (c) Using the recovered lighting parameters and the perspective of (b) yield a shaded view in which the specularity is very similar to the true one. (d) The recovered albedo map is free of specular artifacts.

unknown chromaticity completely rules out the use of methods based on the dichromatic model [9,8,23,25,13,32]. Similarly there are almost no shadows and no obviously visible critical boundaries, which would surely handicap methods [27,12] that require them. In short, given a set of images containing moving objects, our method makes it possible to recover the lighting parameters under less restrictive assumptions than the state-of-the-art methods discussed above.

We require specularities to be present in several registered images and a 3D model, which can be relatively rough. For many applications, this is not particularly onerous given the prevalence of video cameras and the robustness and accuracy of cur-

rent structure-from-motion techniques. Either manual or automated methods could be used to build the required models in the case of the set of images depicted by Figs. 1 and 2 despite some strong lighting effects. Here, the fact that our approach can handle textures is essential since many of these geometric reconstruction techniques rely on it. To prove the validity and practicality of this approach, we will present results obtained using 3-D models produced using standard vision-based registration and reconstruction techniques.

In the remainder of the paper, we first review some of the most representative approaches to light source recovery that have been proposed with a view to understanding what the kind of assumptions they make. We then present the light source estimation algorithm and describe the refinement process. The results section shows the ability of the method to handle multiple light source recovery and discuss its accuracy.

2 Related Work

The numerous difficulties encountered in direct reconstruction of the lighting environment have encouraged researchers to find alternatives that do not require explicitly knowing the lighting parameters and, yet, are suitable for specific applications [5,2]. However, many other applications in fields such as Mixed Reality, Object Recognition, or Texture Synthesis, can greatly benefit from this knowledge.

In particular, specularities have bedeviled Computer Vision algorithms for many years. Many researchers have attempted either to handle them by introducing calibration objects [29,22,10,3,6] or by controlling the image acquisition process [15,4], or to eliminate them by treating them as statistical outliers [14,27]. Others have attempted to actually use them as information sources.

Since the method we present here falls in the latter class, we briefly review below some of the most popular approaches to doing this. In the process, we hope to convince the reader that using inter-image consistency allows us to relax many of the assumptions on texture, light source colors, and object motion, that state-of-the-art techniques make.

2.1 Specularities in Single Images

Many algorithms that can reliably detect specularities in single images use color imagery and are based on the dichromatic model, which represents the light reflected by an object as a sum of a diffuse contribution whose color is the same as the diffuse material and a specular contribution of whose color is the same as the sources.

4

The works presented in [9,8,23] separate images of homogeneous dielectric objects into a perfectly diffuse and a purely specular image. This kind of separation is very useful as it permits to work independently on the diffuse and specular parameters. In particular knowledge of the specular contribution of the lighting is key to determining the lighting environment, since specularities are strongly affected by the slightest lighting variation. But in general, dichromatic model based separation algorithms are unable to deal with texture because they rely on finding the separate clusters produced by each different albedos in a color space. Furthermore they are only suitable for dielectric materials. The methods presented in [25,13,32] are interesting exceptions, which overcome the texture limitation by using local information and produce impressive results from a single image.

However, this class of techniques requires the material chromaticity to be different from that of the lighting and all sources to be of the same color, except in very specific cases where it is possible to segment the contributions of individual sources. To the best of our knowledge, [32] is the only exception that can handle two different colors for the sources, but with the limitation that they should be known beforehand. These methods would therefore be difficult to apply to the image of Fig. 1, in which the teapot is lit by three sources of very different and unknown colors.
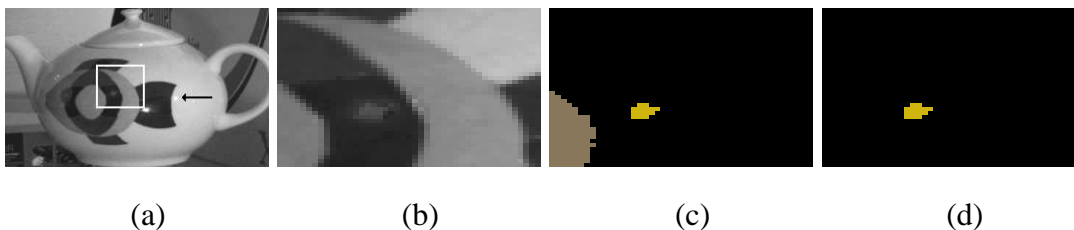

(a) (b) (c) (d)

Fig. 3. Interpreting intensity maxima. (a) The white rectangle encompasses an area of the teapot that contains both a specularity that appears as a white dot on a black area at the bottom and a diffuse bright white area in the upper right corner. (b) A close-up view of the corresponding 8-bit image window. The gray levels within the specularity are all below 80 while the gray levels in the upper right corner are above 200. (c) Our algorithm correctly labels the specular pixels. It discards the ones in the bright white area because, even though their intensities are high, they are only the tail end of the maximum intensity area toward which the arrow in (a) points. A texture maximum corresponding to the center of the fish is also detected and appears in the lower left corner. (d) By enforcing consistency across images, the texture maximum is discarded, leaving only the specularity.

An alternative to the use of color is to use shadows and critical boundaries. For example, in [12], specularities are detected as local maxima and eliminated if their intensity falls below a global threshold. Critical boundaries are then used to select the meaningful ones. In practice, as shown in Fig. 3, an appropriate threshold may not be easy to find. Furthermore, in cases such as the ones depicted by Figs. 1 and 2 where there are either no shadows or detecting critical boundaries is made difficult by the presence of a complex texture, the performance of the algorithm is bound to decrease.

In the case of static scenes filmed using a moving camera, specularities have been successfully used to recover the light sources [16,31]. In that case, the diffuse lighting maxima do *not* glide across the surface and the problem is comparatively easier than when objects can move. The detection is based on the fact that the intensity variations between images can only be caused by specularities.

More sophisticated models that explicitly represent specularities have been proposed. For example, the signal-processing framework introduced in [1,19,20] describes the reflection operator as a convolution and formulates the recovery of a BRDF and lighting distribution as a deconvolution. In addition to providing a mathematical tool to analyze the well-posedness and conditioning of inverse-rendering problems, the method can recover complete BRDFs, thus handling extended light sources, environmental lighting and specular materials. It has, however, only been demonstrated in cases where the albedo is either constant or known *a priori*, which is a severe limitation.

It has recently been shown that specularities produce a flow field in video sequences that can be distinguished from that produced by diffuse surfaces [21]. However, this has only been demonstrated in a case where the flow can be reliably obtained, which is not the case when dealing with sparse sets of images.

3   Method

It is generally accepted that, once the specularities have been detected, counting and positioning the light sources is comparatively easy. However, this detection is often hard and the simple thresholding techniques that are often reported in the literature cannot be expected to work very well in general. For example, in the textured scene of Fig. 3 that was lit by multiple light-sources, a diffuse area of high albedo is brighter than a specular one of low albedo. In such a case, a simple threshold cannot be used to extract specular gray-level maxima while eliminating others.

In this work, we overcome this problem and reliably detect specularities by explicitly using the fact that they glide over moving surfaces in a predictable fashion. This lets us distinguish between non-specular maxima and specular ones and take advantage of the latter to count the number of light-sources and to estimate the lighting parameters. Although the physical facts about the gliding behavior of specularities are well known, to the best of our knowledge, they have not been exploited in this manner before. This is key to detecting low intensity specularities, such as those of Fig. 3.

<center>(a)   (b)</center>

Fig. 4. Two images of the setup we used to evaluate the accuracy of our algorithm. The three target objects, a white ball, a teapot, and an Easter egg were placed on a wheeled table. The table was moved after the acquisition of each image without disturbing the objects. In the background, two of the three light sources can be observed. Note that they are neither perfect point light source nor particularly distant. When acquiring the teapot images of Fig. 1, we used red and blue filters to alter the color of the sources.

Given these lighting parameters estimates, we refine the directions of the sources and their relative powers by fitting the Ward [28] lighting model to gray levels at all locations in all images where the projected 3D model gives us normal vectors. In addition, this yields values for the albedos and the specular coefficients. We chose the Ward model because it is one of the simplest models that, unlike the Phong model [18], is based on physical measures. An alternative would have been to use an even more accurate model, such as the Torrance-Sparrow one [26]. However, such models have a much larger number of parameters, making them more difficult to instantiate with the imagery at our disposal.

The theory behind the method presented here is that of point light sources. This implies that the ambiguity between lighting distribution and material roughness is not solved. However, our specularity detector is designed to handle small area light sources such as those depicted by Fig. 4, which are neither perfect point light sources nor at infinity. Note that we do not make assumptions about the presence or absence of texture and do not require any foreknowledge of the albedos. Since our method is based on grayscale images, it is insensitive to the fact that the light sources may be of different and unknown colors, which sets it apart from most color-based techniques that place restrictive assumptions on the color of the lighting. However, as will be discussed at the end of the section, we can also use color images to estimate the chromaticity of the light sources.

3.1 Detecting, Counting, and Localizing Multiple Light Sources

One important property of our approach is that in practice we only require 3D models of limited precision, such as those that can be derived from the images them-

<center>7</center>
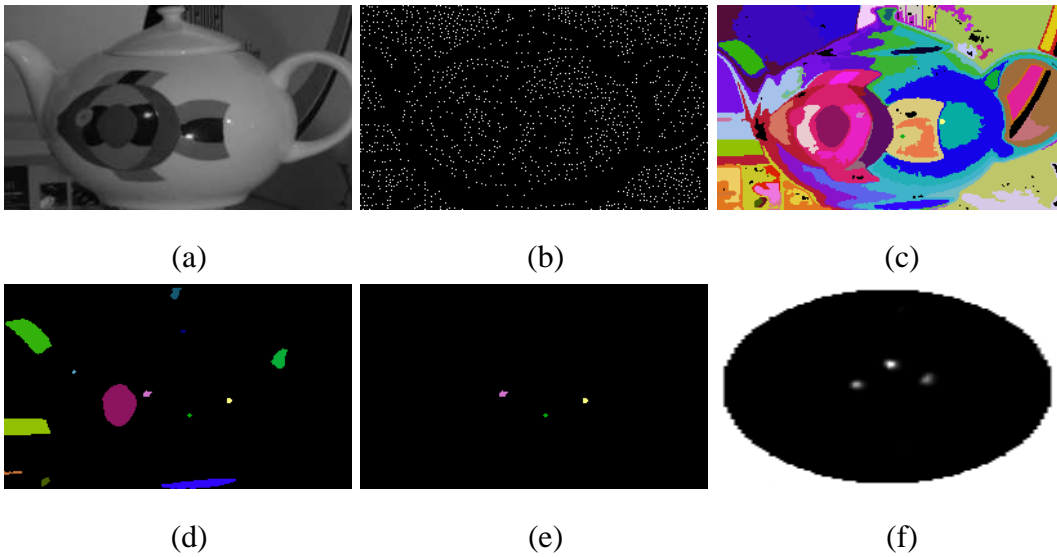
|(a)|(b)|(c)|
|(d)|(e)|(f)|

Fig. 5. Specularity detection workflow. (a) The input image is converted to grayscale. (b) The maxima are detected and neighboring ones are merged. (c) The maxima are grown by including neighboring pixels of lower intensity to form regions. (d) Only the regions that are brighter than their neighbors are retained. (e) Enforcing consistency across images lets us discard non-specular regions. (f) The remaining specular regions vote in an accumulator yielding maxima that corresponds to the number and direction of the light sources.

selves. Since the precise specular locations are very sensitive even to the smallest shape or registration inaccuracies, we seek to detect them not as single points but as specular areas in which individual points can vote for a potential light source direction. This is designed to ensure that the point with the correct normal will be included in the detected area and that points with almost correct normals will also vote, thus providing additional support for the retained directions. This also provides robustness in the presence of high-frequency textures because it ensures that an extended increase of intensity has been detected around the specular area, which is characteristic of lighting maxima. These considerations led us to develop the three-step algorithm depicted by Fig. 5. It involves:

(1) Detecting locally maximal-intensity areas in grayscale images (Fig. 5(b,c,d)).
(2) Using consistency across images to classify these maxima as either texture or lighting ones (Fig. 5(e)).
(3) Having the lighting maxima vote into a Hough-style accumulator to estimate the number of light sources and their directions (Fig. 5(f)).

Each of these step are described in more detail below.

3.1.1 Detecting Local Maxima and Estimating their Local Support

For this step we use grayscale images—obtained by converting the color images if necessary—that we smooth using a Gaussian mask of variance 1.0 to filter out
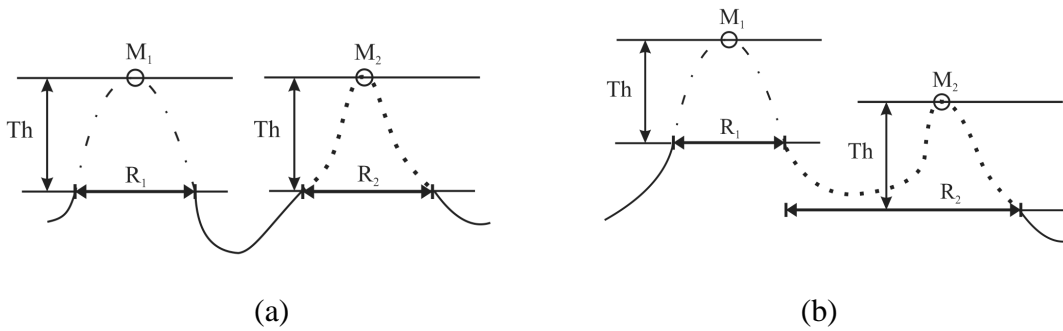
8

(a)   (b)

Fig. 6. Maxima filtering in the case of a 1D image. (a) There are two maxima (M1 and M2) of approximately equal intensity. Our approach to growing areas of support around them results in two separate regions (R1,R2), which are retained as potential specularities. The in between region, depicted as a continuous line, is discarded. (b) The intensity of maximum M1 is larger than that of M2. Retaining all pixels whose difference in intensity with the corresponding maximum is below a predefined value results in two adjacent regions. Since they are adjacent, R2 the region corresponding to the smallest intensity maximum is discarded.

some of the noise. We then select all the pixels of greater or equal intensity than their eight-connected neighbors. In general, this results in several neighboring local maxima of equal intensity. Among these, we select one at random and discard the others, which still leaves a very large number of local maxima, as shown in Fig. 5(b).

To estimate the local support of these maxima, we sequentially consider them, starting with the brightest ones. We grow areas around them by including all the neighboring pixels whose difference in intensity with the corresponding maximum remains below a predefined value. This means that we do *not* use a global intensity threshold to select potentially specular pixels. Instead, we use a local threshold that adapts to the gray level of the maxima. In practice, the algorithm is not very sensitive to the value we use as a cutoff. This is due to the fact that specular pixels are generally of significantly higher intensity than neighboring diffuse ones since specular reflection is both additive and of high energy. The value must simply be high enough to allow some intensity variation due to the underlying texture during the area growing, while being small enough to prevent the absorption of neighboring regions clearly presenting different intensities. In all the examples presented in this paper, we deal with 8-bit images and the threshold is set to 25. During this area growing process, weaker maxima may be absorbed into areas corresponding to stronger ones. They are then removed from the list of maxima to be considered further. Fig. 6 illustrates this process in the case of a 1D image.

This results in a set of regions such as those depicted by Fig. 5(c). Among these regions, we only retain those that correspond to peaks of higher intensity than those of their neighbors, such as those of Fig. 5(d).

The remaining maxima are the ones that will be used as input to the voting scheme

9

described in the following subsection. Note that diffuse maxima can also be detected by our method. However, they will be subsequently discarded by the voting scheme because they do not follow the same consistency rules as the specular ones. At this stage, the fact that materials may have isotropic or anisotropic specular properties does not matter. The same remark applies to plastic versus metallic objects. These distinctions will only play a role for parameter recovery of the lighting model, as discussed in Section 3.2.

The main limitation of the approach presented here is that the distance between light sources we can resolve is bound to increase with surface roughness, which tends to generate broader specularity peaks to the point where they may end up merging.

3.1.2   Checking for Consistency across Images

After having processed the individual images in order to find maximal intensity areas, it becomes possible to classify these areas as either texture or lighting maxima. Here, we assume that the sources are far enough from each other to produce separated specularities on the object. The required distance between the sources is dependent on the surface roughness of the material, the curvature of the object and the image resolution. For instance with a highly-shiny, non-planar object, we can deal with sources separated by a few centimeters.

The consistency check is based on two important observations. First, for a moving object lighting maxima are traveling along the object surface, according to its normals. Second, texture maxima tend to be detected at the same surface location in several images. We exploit this as follows. The 3D model and the camera registration parameters are first used to establish correspondences between the maximal areas in the images and their mesh locations. This allows to compute surface normal and line of sight for each pixel of these areas. If a maximum region, instead of moving with respect to the mesh, stays at the same mesh location even though the object rotates, the algorithm labels it a texture maximum. It may happen that a real specularity coincides with a texture maximum and is therefore discarded by the algorithm. However, if the source that creates it produces other specularities on other images, this will only have a very limited impact. In the particular case of a translating planar object, the approach will keep on working as long as the lines of sight from the camera to the object patches are not close of being parallel throughout the sequence.

3.1.3   Voting for Candidate Light-Source Directions

Specularities are located at places where an object surface behaves like a mirror so that the viewer sees a reflection of the light source. In practice, the mirror angle does not exactly match the specular peak angle, but it is generally very close [26].
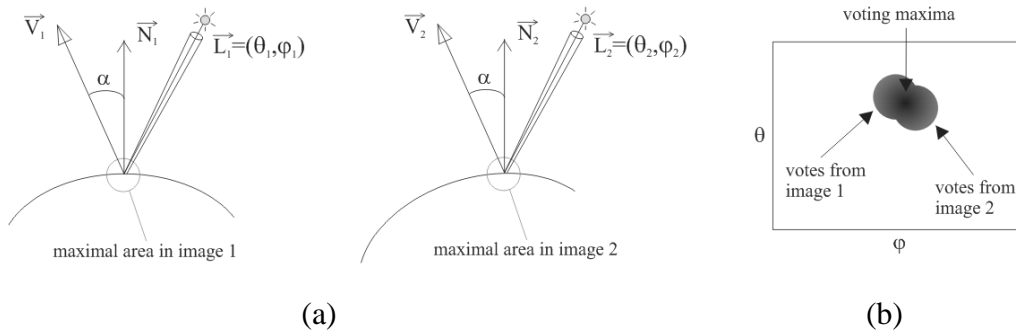
(a)   (b)

Fig. 7. Voting scheme. (a) For each pixel that belongs to a maximal area in any image, the lighting direction L is computed using the line of sight V and the normal N. (b) The direction L estimated for each pixel represents a vote in a 2D accumulator.

(a)   (b)   (c)

Fig. 8. Extended sources detection. (a,b) Two input images of a helmet lighted by two extended light sources, one being more so than the other. The arrows point toward the main specularities. (c) The accumulator obtained after applying the method of Section 3.1. The widest peak corresponds to the most extended source. In our current scheme, the brightest pixel of this peak will represent the source.

Therefore, given a potentially specular pixel, we compute the reflected line of sight mirrored off the 3D model, as shown in Fig. 7(a). Recall that the image registration parameters are known. It is therefore straightforward to express a vector in a coordinate system common to each of the input images. In practice, we write the mirrored line's orientation in terms of its (ϕ, θ) spherical coordinates, which we use to increment a 2D orientation accumulator such as the one of Fig. 5(f). In practice, we perform this operation for all pixels belonging to maximum intensity areas that have passed the consistency test of Section 3.1.2. The combination of having multiple viewpoint information and allowing specular areas to vote for a cone of directions results in robustness to geometrical and registration errors. It also provides some robustness to non-strictly punctual sources. The resulting number of peaks in the accumulator gives us the number of sources and we take the location of each one to be the initial estimate of the corresponding light source's orientation. In the case of the teapot, this procedure produces the three distinct peaks that can be seen in Fig. 5(f), which coincides with the three light sources used to lit the scene.

If a somewhat extended light source was present in the scene, the algorithm would

detect the related specularities and retain them for voting. As shown in Fig. 8, an extended peak would be present in the accumulator. However, only one direction, corresponding to the highest numbers of vote inside the region, would be retained to represent the source.

3.2 Light Source Refinement and Parameter Estimation

Once the number of sources and their directions have been recovered, these directions can be refined and the material properties estimated by minimizing the difference between the actual image gray-level intensities and those predicted by the lighting model. Thus, we minimize

$$\sum_{ij} V_{ij} \left(I_{ij} - \left\{ amb + \sum_{l}^{nbLights} p_l (N_{ij} \cdot L_l) \left[\rho_{d_j} + spec_{ijl} \right] \right\} \right)^2 , \tag{1}$$

where i is the image number, j is the index of a 3D point on the surface, V_{ij} is the visibility function —equal to 1 if the projection point j is visible in the image i and zero otherwise—, I_{ij} is the gray level of point j in image i, amb the ambient term, ρ_{d_j} the diffuse albedo of point j, N_{ij} the normal and p_l the power of the point light source L_l. The $spec_{ijl}$ specular term is defined as

$$spec_{ijl} = \frac{\rho_s}{\sigma^2} \frac{1}{\sqrt{(N_{ij} \cdot E_{ij})(N_{ij} \cdot L_l)}} \exp(\frac{-\tan^2(H_{ijl} \cdot N_{ij})}{\sigma^2}) ,$$

where ρ_s is the specular albedo, σ the specular exponent and H_{ijl} the halfway vector between the light vector L_l and the viewing direction E_{ij}. We initialize these parameters as follows.

It is well known that simultaneously recovering the diffuse and specular parameters as well as the source position is difficult, even when using a lighting model with relatively few parameters [28], instead of the full BRDF. We overcome this problem as follows. The initial albedo values are estimated by robustly averaging the gray levels of the individual surface points in the various images. The initial power of the sources is set with respect to the relative ratio of the specular intensity they create. To ensure that the information provided by the specularities is taken into account, we set the specular parameters ρ_s and σ to initial values that are sufficiently large to produce strong specularities. We then minimize a first time the objective function of Eq. 1, while keeping ρ_s and σ fixed, which yields improved values for all the non-specular parameters. We then release ρ_s and σ and minimize again. The specular parameters can therefore change. However, since we use the same value for ρ_s and σ across the whole object, we are effectively assuming that its specular properties

12

are initially unknown but constant. This assumption is reasonable considering most specular object and it greatly constraints the optimization.

Loss of accuracy may occur if important sources did not produce any specularities on the observed images. Our optimization scheme will use ambient lighting to compensate for the missing sources, thus degrading the estimates of the parameters of the actually detected sources. Because sources directions estimated in section 3.1.3 are reasonably accurate, we can mitigate this problem by introducing a penalty term that prevents the source directions from moving by more than a few degrees from their initial positions. This penalty term also acts as a regularization term that stabilizes the computation.

To achieve robustness to image noise, we minimize the expression of Eq. 1 using a RANSAC type approach. We draw random samples of surface points to perform the computation and only retain those yielding parameters that correctly describe the whole data set. These parameters are then used to filter the outliers out and to perform a new minimization that takes all the inliers into account. The minimization itself is performed in several stages. We first refine the diffuse albedos and the locations of the sources before also adjusting the specular albedo plus roughness and the light-source power parameters.

Color images are processed as described above, using a grayscale version of these images. The results are then used as an initialization to optimize the energy function of Eq. 1 for each channel independently. In presence of dielectric materials, estimating the power of each sources by averaging the intensity of the specularities they create allows for a good estimation of the sources color. For metallic materials, comparing the respective specular intensities in each channel also provides the relative difference of power between the sources and can be used to initialize the parameter p_l of Eq. 1, although the real sources chromaticity is not likely to be recovered. We then re-estimate the power of each source in each channel by optimizing Eq. 1 only with respect to the p_l parameters. Finally, we start over the optimization of the lighting parameters with fixed p_l's. This process is repeated until convergence is reached and yields an unlighted texture map and precise lighting parameters for each channel. Note that it remains an indetermination about the absolute color of the albedo, since we cannot recover the absolute color of the illuminant without a process similar to the white balancing of cameras.

The proposed optimization approach assumes strict point light sources, which implies that specularities produced by extended light sources cannot be explained. A more sophisticated approach should be able to take advantage of the specularity detection step to identify this kind of sources and include them in the photometric optimization. Such an approach will be investigated in future work.

4 Results

To demonstrate the performance of our algorithm, we use three distinct objects, each of them with very different properties:

- The shiny teapot of Fig. 1 we have used to illustrate its workflow.
- The highly textured Easter egg of Fig. 2.
- A human face for which both a precise 3D model and a rougher one are available.

In all these examples, we used the same parameter settings. The only thing that changes is the 3D model and calibration matrices we feed to our system. Different acquisition methods for these inputs are tested and we demonstrate that useful results can be obtained from vision based reconstructed geometric data.

We conclude this section by showing that the lighting model we recover can be used to convincingly add virtual objects into the scene, relight them, and have them cast plausible shadows. In other words, our approach is very suitable for practical Augmented Reality applications.

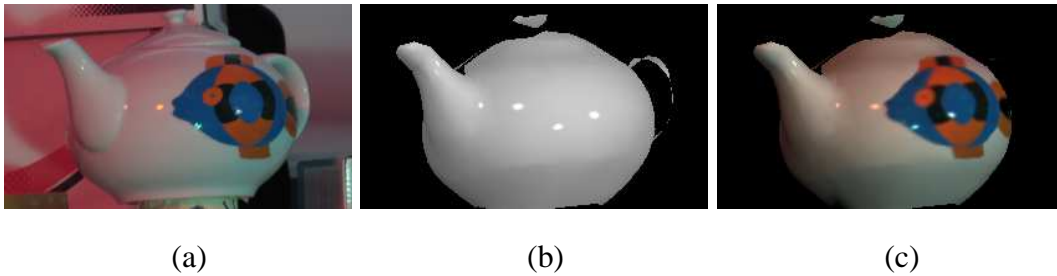| (a) | (b) | (c) |

Fig. 9. New view and corresponding synthetic images. (a) A new view of the object that has *not* been used for recovery. (b) Synthetic view with specularities at the correct places on the side and spout of the teapot which suggest an accurate lighting recovery. (c) Synthetic view using both lighting and albedo parameters.

4.1 Shiny and Partially Textured Teapot

In this example the 3D model has been acquired with a range scanner and the images manually registered using a commercial software. The results of Fig. 1 were obtained using nine images whose specularities are not particularly easy to exploit because they tend to fall on textured areas. They were acquired using the set-up of Fig. 4 by moving the table on which the teapot sits between exposures. Note that the light sources are not true point light sources and are positioned between 1.0 and 2.0 meters from the object, as opposed to being at infinity. When acquiring these images, we used red and blue filters to make their color different from each other, and to create a situation in which the use of dichromatic-based algorithms would

be inappropriate. In short, this example involves many of the difficulties that tend to create problems for state-of-the-art techniques.

As shown in Fig. 1(c), the algorithm correctly estimates the number and orientation of the light sources, which allows us to synthesize images with the right shading and correct location of the specularities. To emphasize this point, Fig. 9(a) shows a tenth image that was *not* used during the computation and demonstrate our ability to produce the corresponding synthetic image, shown without texture in Fig. 9(b) and with it in Fig. 9(c). Note that the specularities synthesized on the spout and on the side of the teapot all fall exactly at the right places. This indicates a very precise recovery since specularities are notoriously sensitive to the exact source positions.

Light source	Error after detection	Error after fitting	Fitting improvement	Light source	Error after detection	Error after fitting	Fitting improvement
1	1.82	1.43	0.39	1	4.27	4.54	-0.27
2	1.18	0.72	0.46	2	2.73	0.18	2.55
3	2.64	2.50	0.14	3	5.33	2.06	3.27
(a)				(b)			

Table 1

Accuracy of the estimated light source directions in the teapot case. (a) Using the full model of Fig. 1(c). (b) Using the partial model of Fig. 10(a). For each light source, we compare the estimates to those obtained using a white sphere, first after the voting scheme of Section 3.1.3 and then after the refinement step of Section 3.2. The numbers we give are differences between directions expressed in degrees.
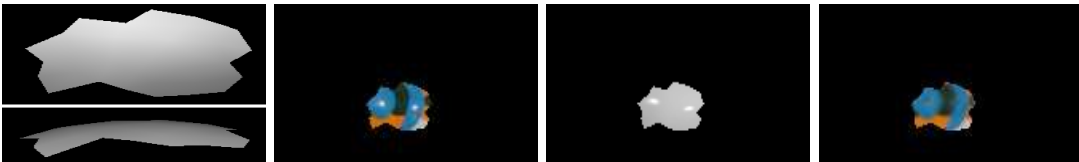
(a)   (b)   (c)   (d)

Fig. 10. Using only a partial 3D model of the teapot, instead of the complete one. (a) Front and side views of the partial 3D model. It does not have much curvature and includes only a small range of normals. (b) Projected model into one of the teapot images, which covers only a very reduced portion of the teapot's surface. (c) Shaded view of the partial 3D model using the recovered light sources. Note that the specularities still appear at the correct places. (d) The recovered albedo map is free of specular artifacts.

To quantify the accuracy of our estimates, we placed a white sphere on the same table as the teapot. It behaves almost like a mirror, making it very easy to precisely locate the specularities it produces. We used them to accurately estimate the locations of the light sources using a 3D spherical model, which is very easy to register to the actual sphere, thus making these estimates highly believable. We therefore treat them as a baseline against which we compare results obtained using other objects. In Table 1 (a), we give the error in the recovery of the orientations for each

15

one of the three light sources, first after the voting scheme of Section 3.1.3 and, second, after the refinement step of Section 3.2. This yields errors ranging from 0.72 to 2.50 degrees. Note that light source number 2 is the one recovered most accurately, presumably because it produces more specularities that fall on parts of the images where the angle between the normal and the light source direction is small.

To test the applicability of our approach to 3D surfaces that only exhibit a narrow cone of normals either because the object is relatively planar or because it has only been partially reconstructed, we ran again our algorithm on the same nine images but using only the uncomplete and very limited 3D teapot model of Fig. 10. In Table 1(b), we again compare the results to those obtained using the white sphere. Even though far fewer image pixels can be exploited, the algorithm still returns a plausible result but, unsurprisingly, becomes less accurate on average.

Note that the improvement obtained by the refinement step of Section 3.2 is much more significant in this case than it was when using the full model. This makes perfect sense: since fewer pixels are available, it is all the more important to exploit the information they can provide as fully as possible. What can seem somewhat more surprising is that, even though the accuracy degrades very noticeably for source number 1, it actually improves slightly for sources number 2 and 3. The accuracy loss can be attributed to the fact that fewer of the specularities caused by source number 1 are processed. By contrast, the improvement stems from the fact that the partial model we use happens to incorporate the most reliable normals of the reconstructed model.

In any event, this case is interesting because it clearly shows that we can derive useful results by only building a 3D model of a relatively small and textured part of the object, which is easy to do using structure-from-motion techniques. The only limitation, as the loss of accuracy for the source number 1 suggests, is that the models projections must cover a sufficient number of specularities of each source.

4.2   Highly Textured Industrially Painted Egg

Fig. 2 depicts a small industrially-painted Easter egg, which includes both a medium-sized area of constant albedo and highly textured areas allowing for automatic calibration. In the upper row, the egg is lit by a single directional source and the images are high-resolution ones. The images in the lower row were acquired by putting the egg on the same table as the teapot and are therefore also lit by the three same sources. They are of much lower resolution because we needed to have the teapot, the white sphere, and the egg, all in the same image. In both cases, our algorithm correctly identifies the specular highlights and recovers the appropriate number of light sources. This lets us synthesize the de-lighted pictures of the last column of Fig. 2, from which the highlights have been completely eliminated. Of course, the top one is of higher quality than the bottom one, but this only reflects the difference

in resolution between the input images.

Light source	Error after detection	Error after fitting	Fitting improvement
1	1.65	1.44	0.21
2	0.62	0.46	0.16
3	0.60	0.58	0.02

Table 2
Accuracy of the estimated light source directions in the Easter egg case. The numbers we give are differences between directions computed in the same manner as those of Table. 1

As in Section 4.1, in the three light-sources case, we compare the estimates against those obtained using the white sphere. Table 2 summarizes the results, which are even more accurate than those obtained with the teapot. In short, whether using the spherical ball, the Easter egg, the full or the partial teapot, we obtain results that are *all* consistent with one another, which gives strong support for the reliability of our method. Furthermore, even if the absolute precision of the recovery may degrade if the 3D model is not accurate or complete enough, the result nevertheless remains consistent with the input data. This allows artifact-free texture recovery, as shown in Figs. 1 and 2, and correct new view synthesis, as shown in Fig. 9.

4.3   Human Face

Fig. 11 depicts results obtained using images of a human face, which was lit by a single directional source. This example complements the other two because skin, although potentially specular, tends be rougher and to produce broader intensity peaks than the industrial objects we have been showing so far. Furthermore, it is more difficult to produce a metrically accurate 3D model.

We used input meshes at two different resolutions to perform our computation and check its sensitivity to the quality of the model. One is a high resolution laser scan and the second is a much lower-resolution generic face model that has been fitted to the images using a structure from motion technique. Note that the algorithm identifies the same areas as being specular even though the shape of the two 3D models differs substantially. In particular, the chin and the cheekbone of the low-resolution model of Fig. 11(g) are very inaccurate. Nevertheless, the algorithm is robust enough to accommodate these errors.

In Fig. 12(b), we blacked out the background pixels of the face image of Fig. 12(a), which was *not* used to recover the lighting parameters and albedos. In Fig. 12(c), we show an image of the face synthesized using these light sources and albedos and seen from the same viewpoint. Note that both in the real and synthetic views, half of the face is in shadow and that the specularities are similarly located.

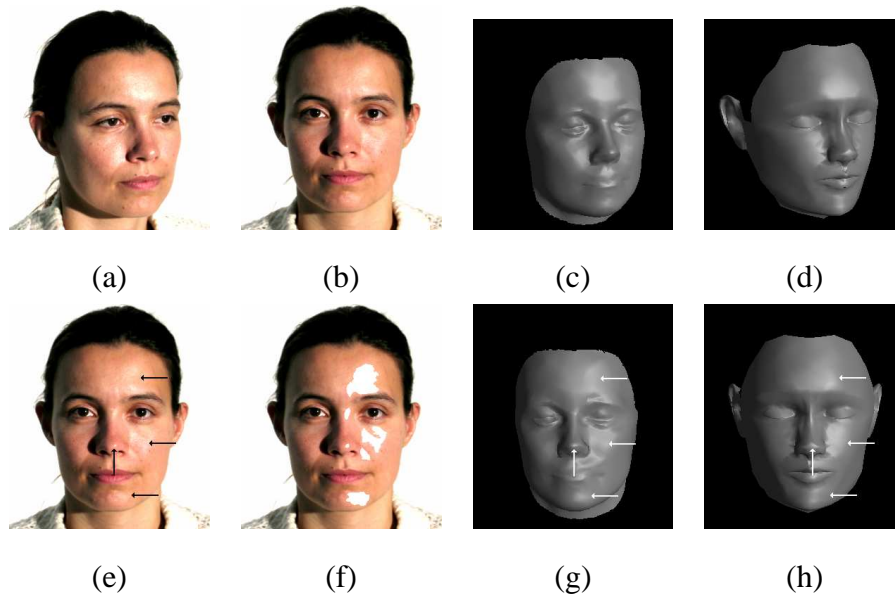(a)　　　　(b)　　　　(c)　　　　(d)

(e)　　　　(f)　　　　(g)　　　　(h)

Fig. 11. Insensitivity to the quality of the model. (a,b) Two out of nine input images. (c) The high-resolution scan. (d) A much lower resolution model computed using a structure from motion technique. (e) The arrows point towards the main specularities. (f) Specular areas detected using the 3D scan overlaid in white. (g) Shaded view, using the parameters recovered with the high resolution 3D scan. Note that the synthetic specularities appear at the right places. (h) Shaded view, using the parameters recovered with the low resolution model, to be compared to (g). Note that the specularities still appear at the right places.
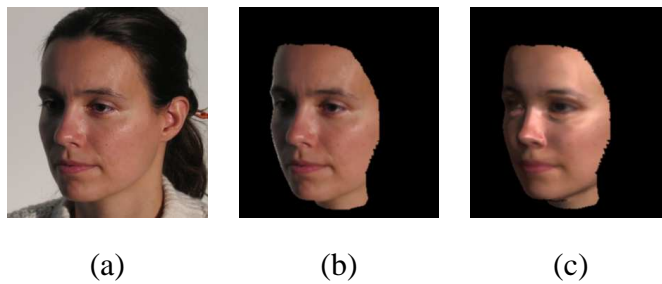

(a)　　　　　　(b)　　　　　　(c)

Fig. 12. Synthesizing a new view of the face. (a) An image that was *not* used to recover the lighting parameters. (b) The same image in which the background pixels have been blacked-out. (c) An image synthesized using the recovered light sources and albedos. Note that the same parts of the face are in shadow in the both the real and synthetic images and that the specularities are similarly located.

4.4 Augmented Reality

Figs. 13 and 14 demonstrate the effectiveness of our approach for Augmented Reality purposes. We fixed the Easter egg of Section 4.2 on top of a box whose orientation with respect to the camera our system automatically computes by first extracting the textured pattern in its center [11], computing the corresponding homography, and estimating the extrinsic camera parameters from it [24,30]. Since the position of the egg with respect to the box does not change, that also gives us its
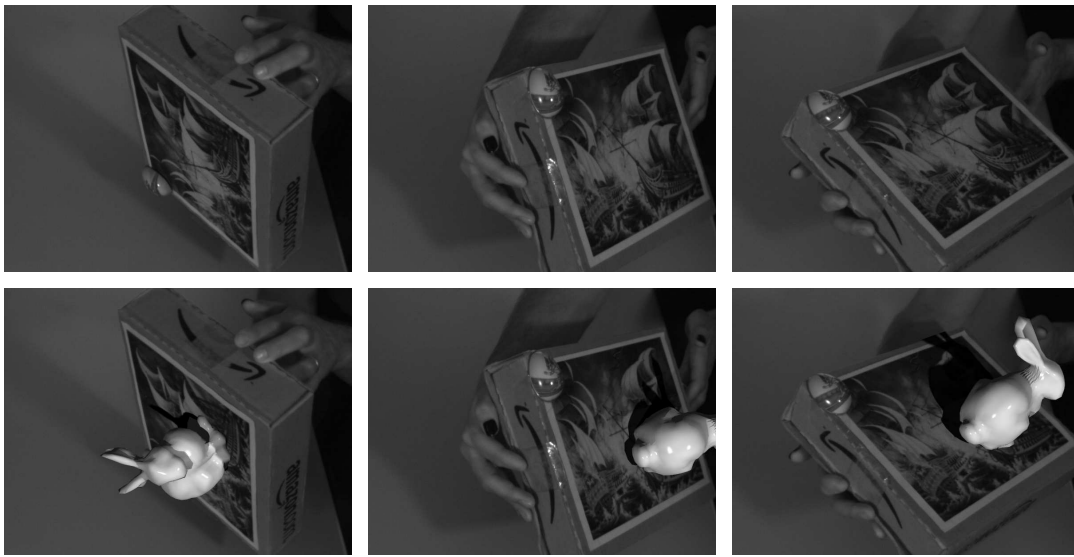
18

Fig. 13. Adding and relighting a virtual object in a real scene. First line presents three of the ten images that were used to recover the number of light sources and their orientation. Second line shows corresponding augmented images where a virtual rabbit is added on the box and lighted using the recovered light sources. Note that it produces realistic highlights and cast shadows.

Fig. 14. The synthetic rabbit has been added to the other images of the video *without* re-computing the light sources. The result remains realistic and we provide the corresponding video as supplementary material.

orientation and lets us retrieve the number of the light sources and their position. To this end, we used ten images from the video sequences, three of which are depicted by the first line of Fig. 13. As shown in the bottom line, we can then add a virtual rabbit using the estimated pose and relight it using the recovered light sources.

Furthermore, since we can automatically estimate the position of the box in all

frames [11], it becomes easy to add the rabbit into the other images of the video without recomputing the light sources. This produces synthetics images such as those of Fig. 14 and a complete augmented video that we provide as supplementary material. Note that the shiny virtual rabbit generates highlights at the right places and casts realistic shadows on the real object. This behavior is critical for a convincing illusion as can be seen in our synthetic video.

5 Conclusion

We have shown that, given a set of images and a 3D model of a moving object, which may be neither very precise nor complete, we can accurately recover the number and direction of multiple light sources of potentially different colors, even when the object is highly textured.

This is achieved by explicitly using the fact that specularities glide over moving surfaces in a predictable fashion, which is an original approach even though the basic underlying physical facts are well known. This lets us distinguish specular areas from those surrounding other kinds of image intensity maxima. As a result, we can accurately recover lighting parameters and produce unlighted texture maps by removing specular artifacts, even in situations where other state-of-the-art techniques are not applicable.

In future work, to further increase robustness, we will explore ways to track the maximum intensity areas in video sequences to exploit spatio-temporal constraints. We will also incorporate shadows that we currently ignore into our estimation framework.

References

[1] Ronen Basri and David Jacobs. Lambertian reflectance and linear subspaces. In *International Conference on Computer Vision*, 2001.

[2] E. Beauchesne and S. Roy. Automatic Relighting of Overlapping Textures of a 3D Model. In *Conference on Computer Vision and Pattern Recognition*, pages 166–176, 2003.

[3] S. Boivin and A. Gagalowicz. Image-based rendering of diffuse, specular and glossy surfaces from a single image. In *ACM SIGGRAPH*, pages 107–116. ACM Press, 2001.

[4] D., R. T. Tan, K. Hara, and K. Ikeuchi. Polarization-based Inverse Rendering from a Single View. In *International Conference on Computer Vision*, pages 982–987, 2003.

[5] R. Dror, E. Adelson, and A. Willsky. Recognition of surface reflectance properties from a single image under unknown real-world illumination. In *A. S. Proceedings of the SPIE 4299: Human Vision and Electronic Imaging IV*, 2001.

[6] A. Hertzmann and S. M. Seitz. Shape and materials by example: A photometric stereo approach. In *Conference on Computer Vision and Pattern Recognition*, pages 533–540, 2003.

[7] http://www.thedigitalbits.com/reviews3/starwarschanges.html.

[8] A. Koschan K. Schluens. Global And Local Highlight Analysis In Color Images. In *Computer Graphics and Image Processing*, 2000.

[9] G. J. Klinker, S. Shafer, and T. Kanade. The measurement of highlights in color images. *International Journal of Computer Vision*, 2(1):7–32, June 1988.

[10] H. P. A. Lensch, M. Goesele, J. Kautz, W. Heidrich, and H.-P. Seidel. Image-based reconstruction of spatially varying materials. In *Eurographics Rendering Workshop*, pages 103–114. Springer-Verlag, 2001.

[11] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006. In press.

[12] Y. Li, S. Lin, H. Lu, and H.-Y. Shum. Multiple-cue illumination estimation in textured scenes. In *International Conference on Computer Vision*, page 1366. IEEE Computer Society, 2003.

[13] Satya P. Mallick, Todd Zickler, Peter N. Belhumeur, and David J. Kriegman. Specularity removal in images and videos: A pde approach. In *European Conference on Computer Vision*, volume 1, pages 550–563, 2006.

[14] S. R. Marschner and D. P. Greenberg. Inverse lighting for photography. In *Proceedings of the Fifth Color Imaging Conference, Society for Imaging Science and Technology*, 1997.

[15] S. K. Nayar, X.-S. Fang, and T. Boult. Separation of reflection components using color and polarization. *IJCV*, 21(3):163–186, 1997.

[16] K. Nishino, Z. Zhang, and K. Ikeuchi. Determining Reflectance Parameters and Illumination Distribution from a Sparse Set of Images for View-Dependent Image Synthesis. In *International Conference on Computer Vision*, pages 599–606, Vancouver, Canada, July 2001.

[17] M. Oren and S.K. Nayar. A Theory of Specular Surface Geometry. *International Journal of Computer Vision*, 24(2):105–124, 1996.

[18] B.T. Phong. Illumination for Computer Generated Pictures. *Comunications of the ACM*, 18(6), June 1975.

[19] R Ramamoorthi and P. Hanrahan. A signal processing framework for inverse rendering. In *SIGGRAPH*, pages 117–128, 2001.

[20] R. Ramamoorthi and P. Hanrahan. A signal-processing framework for reflection. *ACM Trans. Graph.*, 23(4):1004–1042, 2004.

[21] S. Roth and M.J. Black. Specular Flow and the Recovery of Surface Structure. In *Conference on Computer Vision and Pattern Recognition*, New York, 2006.

[22] Y. Sato, M. Wheeler, and K. Ikeuchi. Object shape and reflectance modeling from observation. In *ACM SIGGRAPH*, pages 379–387, August 1997.

[23] K. Schluens and M. Teschner. Analysis of 2d color spaces for highlight elimination in 3d shape reconstruction. In *Asian Conference on Computer Vision*, pages 801 – 805, 1995.

[24] P. Sturm and S. Maybank. On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications. In *Conference on Computer Vision and Pattern Recognition*, pages 432–437, June 1999.

[25] R.T. Tan and K. Ikeuchi. Iterative framework for separating reflection components of textured surface using a single image. In *CREST03*, pages 67–81, 2003.

[26] K.E. Torrance and E.M. Sparrow. Theory for Off-Specular Reflection from Roughened Surfaces. *Journal of the Optical Society of America*, 67(9), 1967.

[27] Y. Wang and D. Samaras. Estimation of multiple directional light sources for synthesis of mixed reality images. In *Pacific Conference on Computer Graphics and Applications*, pages 38–47, 2002.

[28] G.J. Ward. Measuring and modeling anisotropic reflection. In *ACM SIGGRAPH*, pages 265–272, 1992.

[29] Y. Yu, P. E. Debevec, J. Malik, and T. Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *ACM SIGGRAPH*, pages 215–224, 1999.

[30] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334, 2000.

[31] W. Zhou and C. Kambhamettu. Estimation of illuminant direction and intensity of multiple light sources. In *European Conference on Computer Vision*, page IV: 206 ff., 2002.

[32] T. Zickler, S.P. Mallick, D.J. Kriegman, and P. N. Belhumeur. Specularity removal in images and videos: A pde approach. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2000–2010, 2006.