Multiple Light Sources Recovery in the Presence of Specularities and Texture

Pascal Lagger and Pascal Fua
Computer Vision Laboratory
Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
{pascal.lagger,pascal.fua}@epfl.ch

EPFL Technical Report IC/2005/036

Abstract

Recovering the light sources from images in which the objects can move is difficult both because the specularities travel along the surfaces and because diffuse reflections do not remain constant.
In this paper, we show that this can nevertheless be done without making the very restrictive assumptions that are usually made provided that 3D models of some scene objects are available or can be computed. Our approach involves following local gray-level maxima and deciding whether or not they represent specularities. The true specularities are then used to count how many light sources are in the scene, compute their precise location and estimate the material properties.
The resulting algorithm can operate in presence of arbitrary textures and multiple light sources of possibly different unknown colors.

1. Introduction

The recovery of lighting parameters such as the number of light sources, their directions and intensities, or the reflectance parameters of illuminated materials from image sequences has been a very challenging and active research area for many years. However, this remains an open problem in the presence of specularities because they are difficult to detect reliably in the general case. In this paper, we focus on the difficult case where we have several images of moving objects, which means that the specularities travel along the surface and that diffuse reflections cannot be expected to remain constant. The approach we propose relies on explicitly detecting the specular areas in the images and using them to compute the number and location of the light sources. Unlike earlier approaches, we do not need to make strong assumptions about either the lighting conditions or the nature of the scene materials.

Using polarizing filters [13, 4] has been shown to be very effective, but is only practical if one has full control over the image acquisition process. Similarly, excellent results have been demonstrated [23, 17, 10, 2, 6] in cases where calibration objects, such as reflective spheres, can be introduced in the environment before the actual images are acquired. However, the situation is much more complicated when controlling the environment or the image acquisition process is not an option and algorithms have to recover the illumination parameters directly from the appearance of objects naturally present in the scene.

Until recently, a majority of approaches to this problem relied on the assumption that the reflectance propriety of the scene was Lambertian [12, 21] and that the albedos were uniform. This is of course extremely restrictive, especially in the presence of specularities, and newer approaches have relaxed these assumptions. For example, in [11], specularities are handled by simple thresholding but this only works if they correspond to the brightest image pixels, which is not always true. In [14], the scene is static but the camera moves so that intensity variations between images can be attributed to specularities, which ceases to be true as soon as scene objects can move. Color has also been extensively used and a whole class of methods relies on the dichromatic model [9] to separate specular and diffuse reflection components. Here the limitations come from the fact that the materials must be dielectric and of different chromaticity than the single colored light sources. Furthermore, to the best of our knowledge, the approach presented in [19] is the only one of this type that can handle texture.

In short, most state-of-the-art approaches that do not rely on engineering the environment make strong assumptions either on the lighting conditions or on the material characteristics of illuminated surfaces. Here we propose an approach that goes much further in relaxing the assumptions by emulating the ability of the human eye to follow local

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

Figure 1: Recovering 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 images. (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.

intensity maxima, when looking at a moving object, and deciding whether or not they truly represent specularities. We then use them to decide how many light sources there are and where they are. We work with image sequences and assume that we are given more or less precise 3D models of some scene objects. The local maxima of intensity are detected using a mean shift approach that allows us to keep only the ones with sufficient support, which is much more effective than simple thresholding and works even in the presence of texture. We then use a voting scheme that lets the detected areas cast their vote for likely light-source orientations. Enforcing consistency on these votes then lets us classify them as either texture or specular maxima and to decide on the number and orientation of the light sources, that can then be refined using an optimization algorithm. Figs.1,2 illustrate the ability of our approach to identify specularities whose intensity may be relatively weak and to use them to recover multiple light sources, lighting parameters and albedo map.

Our algorithm can operate in presence of arbitrary textures and multiple light sources of possibly different unknown colors. The input images can be either color or grayscale and we can handle materials of the same chromaticity as the illuminant.

The limitations of our method comes from the fact that we require 3D models of the objects we use. However, as will be shown below, models do not have to be particulary accurate. They can therefore easily be constructed from the image sequences themselves either manually using commercially available software [7] or automatically using one of the many structure-from-motion techniques that have been developed. Here, the fact that our approach can handle texture is essential since many of these geometric reconstruction techniques rely on it.

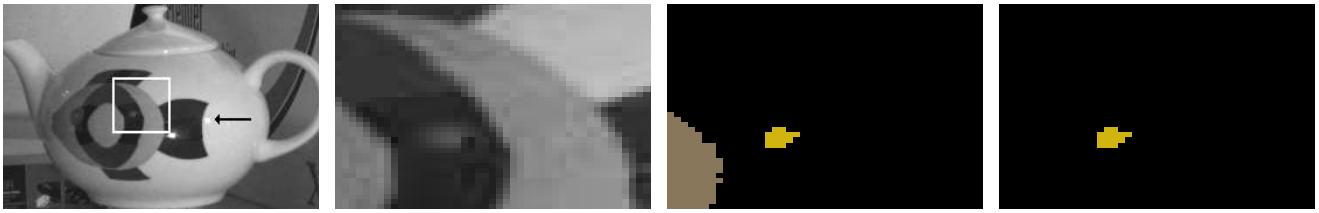
2. Related Work

For many years, algorithms designed to handle multiple light sources have treated specularities as outliers. For example, in inverse lighting [12] it is assumed that the reflectance is Lambertian and constant. The illuminant is modeled as a linear combination of a basis of predefined lights. There is a linear relationship between the coefficients of the linear combination and the pixel values, which makes it possible to recover the former from a single image. [21] proposes a method to find multiple sources for a perfectly Lambertian non-textured object using critical boundary detection that represent lighting discontinuity between the objects areas where a different number of sources are visible. The methods discussed above use a 3D model as input. However, this is not an absolute requirement. As shown in [20], it is possible to recover the azimuthal angle of a single source from two images of random Lambertian surfaces of spatially varying albedo without a mesh.

More recently, models that explicitly represent specularities have been proposed. For example the signal processing framework introduced in [15, 16] describes the reflection operator as a convolution and formulates the recovery of a BRDF and lighting distribution as a deconvolution. Aside from giving a mathematical tool to analyze the well-posedness and conditioning of inverse rendering problems, the method is able to recover a complete BRDF, thus handling specular material, and the lighting environment. The limitation is that it only works for static scenes. It is thus unable to resolve the lighting-texture ambiguity, implying a restriction to objects of constant or known albedo.

In some approaches such as [11], the specularities are explicitly used in conjunction with the critical boundary method and shadow detection allows lighting reconstruction in presence of texture. However, the specularities are detected using a simple threshold. As a result the algorithm can be fooled by texture maxima, especially because the approach does not enforce multiple image consistency. [14] also uses detected specularities in the case of static scenes and moving cameras. The detection is based on the fact that the intensity variations between images are caused only by specularities, which ceases to be true as soon as scene objects can move.

An interesting class of algorithms relies on the dichromatic model that represents the lighting of an object as a



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

Figure 2: 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.

sum of a diffuse contribution of the same color as the diffuse material and a specular contribution of the same color as the sources. [9, 8, 18] separate images of homogenous dielectric objects into a perfectly diffuse and a purely specular image. This kind of separation is very useful as it lets us work independently on the diffuse and specular parameters. 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. Although the method presented in [19] is an interesting exception, which overcomes the texture limitation by using local information and produces impressive results from a single image, it is only applicable when there is a single known illuminant color. Generally speaking, the material chromaticity must be different from the lighting chromaticity and all sources must be of the same color except in very specific cases where it is possible to segment the contributions of individual sources.

The many difficulties encountered in direct reconstruction of the lighting environment have encouraged researchers to find alternatives such as the one presented in [5] for recognition of surface reflectance properties using a statistical classification. Another approach [1] compensates for the lighting difference of multiple views of the same object without reconstructing the lighting distribution.

Nevertheless, many applications, such as Augmented Reality ones, require explicit recovery of light sources and material properties. We will therefore present here an approach that solves this problem with fewer restrictions than existing techniques.

3. Method

Our approach involves three steps. We start by detecting locally maximal intensity areas in grayscale images. Then we use consistency across images to classify them as texture or lighting maxima. The latter are used in a Hough style voting scheme to count the number of light sources and estimate their directions. As a final step, we refine the positions of the recovered sources and compute material properties using an optimization algorithm. The final result includes the number of light sources and the accurate recovery of the lighting parameters. We also obtain an unlighted texture map.

Because the detection process uses grayscale images, our method is robust to the fact that the light sources may be of different colors. To the best of our knowledge, none of the existing color based technique can guarantee this. It also gives us the ability to work with the widest possible range of existing images.

3.1. Mean shift Based Maxima Area Detection

When there is no texture, image intensity maxima can be diffuse, specular, or both. Usually specular maxima are brighter than diffuse ones. However, in the presence of texture, a diffuse area of large albedo may appear brighter than a specular area of smaller diffuse albedo. Fig. 2 illustrates this fact and shows that a simple threshold approach is not sufficient for specularity detection.

Detecting specularities in the general case therefore requires a more elaborate approach than simple detection of global intensity maxima; the detection algorithm must be able to detect locally maximal regions even in presence of texture.

A key idea is to think of a gray-level image as the discretization of a density function. It then becomes possible to use the mean shift technique [3], which is very effec-

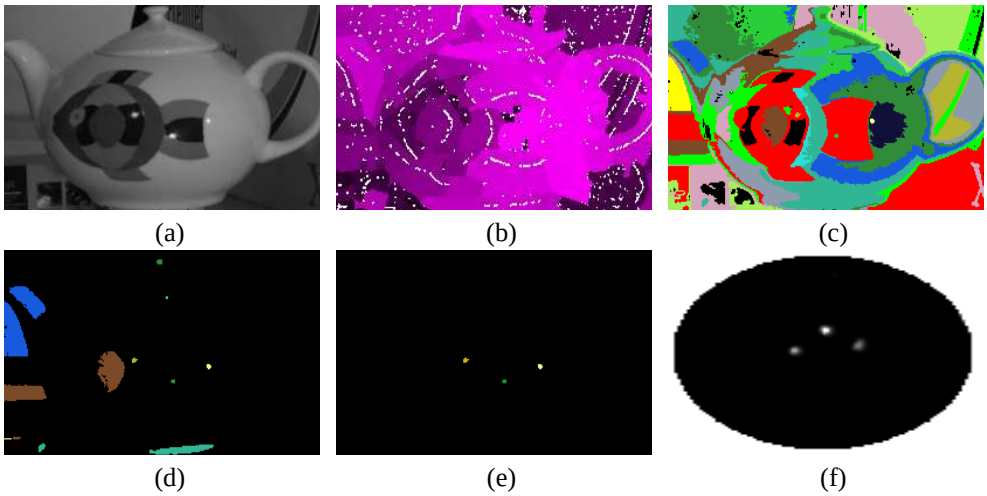


Figure 3: Specularity detection workflow. (a) The input image is converted to grayscale. (b) Our mean shift based maxima area detector detects many modes shown in white. (c) The neighboring modes are merged and then grown toward neighboring pixels of lower intensity. (d) Only the region that are brighter 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 orientation of the light sources.

tive for finding local maxima of density, called stationary points or modes, and to attach each data point to the closest mode. A brief reminder of the theoretical background can be found in Appendix. In our case the dimension d of the density function is two and since the intensity is treated as a counter of the number of data at location x, the mean shift vector that always points toward the increase of density is expressed as

$$m_{h,G}(x) = \frac{\sum_{i=1}^{n} x_i \cdot I(x_i) \cdot g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} I(x_i) \cdot g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x \quad , \qquad (1)$$

where $I(x)$ is the gray-level at x, and g(x) is the profile of the kernel G(x) of size h. In practice we can use either an Epanechnikov or Gaussian kernel without noticeable effect on the quality of the final result. However, the size of the kernel can strongly affect the quality of the detection. A small spread kernel allows a precise localization of maxima but prevents close distinct maxima from merging. In presence of texture this behavior is not suitable because the detection of too many local texture maxima will complicate the detection of the lighting maxima. By contrast a broader kernel does not localize maxima very accurately but accounts for the global shape of a lighting maximum even in the presence of texture and the result will be a maximum supported by a reasonable amount of neighboring pixels. Note that the kernel size can easily be tuned according to the material we are analyzing; for instance, in presence of relatively constant albedo and shiny material one can use a narrower kernel than in the case of a highly textured scene.

The algorithm begins by converting the color images to grayscale and apply a light gaussian smoothing to filter the noise. The mean shift vector is then evaluated for each pixel using a fixed size window. Each window is then moved according to the mean shift vector of Eq. 1. The operation is repeated until all windows reach a stationary point. Pixels where one or more windows converged are stationary points and the starting pixels are attached to their stationary points.

As shown in Fig.3 (b), the mean shift step usually finds a high number of maxima, even after the local maxima of a high frequency texture have been filtered out. We have therefore implemented a filtering algorithm that retains only the most significant and makes them grow as shown in Fig.3 (c) in order to detect maximal areas, we proceed as follows.

The algorithm iteratively takes the brightest stationary point and grows an area around it by including all the neighboring pixels for which the difference between the pixel's intensity and the intensity of the stationary point remains under a fixed threshold. During this process, other stationary points can be absorbed and they are removed from the list of stationary points to be considered further. This results in a list of regions such as the one depicted by Fig.3 (c). The most significant ones are those that contain pixels of higher intensity than their neighbors and we retain only those, as shown in Fig.3 (d). These maximum intensity areas are the key to lighting reconstruction.

These areas are locally maxima and therefore allow for the detection of low intensity specularity such as the one in Fig. 2, which cannot be achieved by a simple global threshold. Another important point is also that these maximal ar-

eas are supported by a whole neighborhood. This kind of support is valuable for robustness to high frequency texture because it means that a global increase of intensity has been detected around the maximum area which is characteristic of lighting maxima.

As far as light detection is concerned, the fact that the materials may have isotropic or anisotropic specular properties does not interfere. The same remark applies to plastic and metallic objects. These distinction will only play a role for parameter recovery of the lighting model.

Note that, in theory, a diffuse maximum could also be detected by this method. However, this very rarely happens because specular maxima tend to be much brighter than diffuse ones. Furthermore, since cameras are not usually orthographic, even if a diffuse maximum is detected, it will be discarded by the voting scheme described below because it does not follow the same consistency rules as a specular one.

3.2. Multiple Light Source Direction Estimation

Now that individual images have been processed to find the maximal intensity areas, it becomes possible to classify these areas as either texture or lighting maxima. Here, we make the assumption that the sources are far enough from each other to produce separated specularities on the object. For now, only directional point sources are considered, but to some extent small area sources can also be detected.

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 specular angle is not exactly the same as the specular peak, but it is generally very close. The main idea is to identify the specular areas and make the lines of sight reflect on the mesh at their location to produce cones of vectors that will point toward the source location. In theory, if the object to analyze were a perfectly reflecting sphere, it would be possible to recover the complete lighting sphere, with the exception of the lighting direction opposite to the camera. But, in practice, the quality of the model, the precision of the registration and the resolution of the capturing device are limited. For these reason we limit ourselves to normals that are within seventy degrees of the viewing direction. This yields a maximum recovery of about eighty percents of the lighting sphere. The unrecovered portion which is actually behind the object is of little impact on the object's appearance.

We exploit the fact that on a moving object the lighting maxima are travelling across the object surface according to the normals while texture maxima tend to be detected at the same surface location for many images. The 3D mesh and the camera registration are first used to establish correspondences between the maximal areas in the images and their locations on the mesh. This lets us 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 therefore be discarded by the algorithm but, if the source that creates it produce other specularities on other images, this will not prevent the source recovery.

Candidate directions of light sources are found by computing the mirror reflected lines of sight on the mesh surfaces where the brightest pixels lie. These directions are expressed in the same coordinate system and the algorithm discards areas that suggest a source direction supported by a too small number of images. For the remaining areas, the reflected vectors are transformed to their 2D spherical coordinates and used to increment a 2D orientation accumulator such as shown in Fig.3 (f). Then a simple mean shift step determine the number of sources and their directions.

3.3. Light Source Refinement and Parameter Estimation

Once the number of sources and their directions have been recovered, we refine these directions and compute the material's properties by minimizing the difference between the actual image intensities and those predicted by the lighting model. It is well known that recovering the diffuse and specular parameters and the position of the sources at the same time is difficult, even if we don't consider a full BRDF function but only a reflection model with few parameters such as Ward's [22]. However, in our case, the initial light source estimates are dependable and we can use them to constrain the problem.

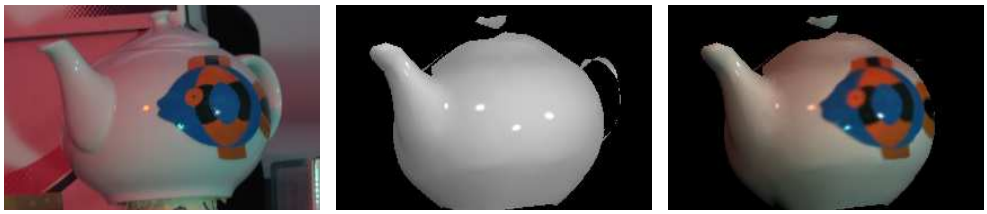
In practice, we minimize

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

where i is the image number, j is the index of a 3D point on the surface, I_{ij} is the gray level of point j in image i, amb the ambient term, ad_j the diffuse albedo of point j, and p_l the power of the source L_l. $spec_{ijl}$, the specular term, is taken to be

$$spec_{ijl} = \frac{as}{e^2} \frac{1}{\sqrt{(N_{ij} \cdot V_{ij})(N_{ij} \cdot L_l)}} \exp\left(\frac{-\tan^2(H_{ijl} \cdot N_{ij})}{e^2}\right)$$

where as is the specular albedo, e the specular exponent and H_{ijl} the halfway vector between the light vector L_l and the viewing direction V_{ij}. To reduce the number of free parameters, we fix *a priori* the values of the specular albedo and exponent. During the minimization, we prevent the source directions from moving more than a few degrees from their initial positions by introducing an additional penalty term.



(a) (b) (c)

Figure 4: 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 right places on the side and spout of the teapot suggest an accurate lighting recovery. (c) Synthetic view using both lighting and albedo parameters.

In practice, we compute initial values for these parameters as follows. The initial albedo values are estimated by robustly averaging the gray level of the individual surface points in the various images. For dielectric materials, we estimate the colors of the sources by averaging the colors of the specularities, which produces an estimate of the initial sources power in each channel. For metallic materials, comparing the respective specular intensities also yields the relative difference of power between the sources and can be used to initialize the parameter p_l of Eq. 2. To exploit the information provided by the specularities, we set the initial specular parameters to values that produce noticeable specularities.

To achieve robustness to image noise, we minimize the expression of Eq. 2 unsing a RANSAC type of approach. We draw random samples of surface points to perform the computation and retain only the one that yields parameters that best describe the whole data set. These parameters are then used to filter out the outliers and to perform a new minimization that takes all the inliers into account. The minimization itself is performed in stages. We first refine the diffuse albedos and the locations of the sources before also adjusting the specular albedo and the light source power parameters.

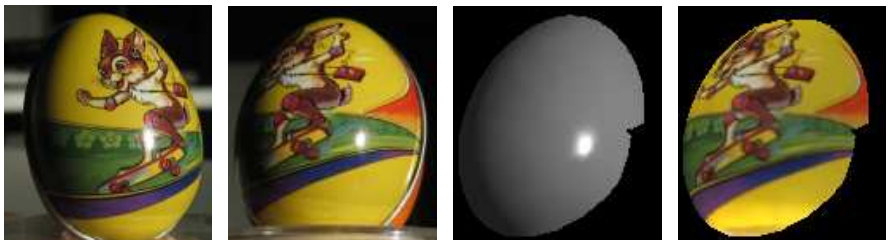
In the case of color images, we proceed as described above on a grayscale version of the images and use the results as an initialization for optimizing the energy function of Eq. 2 in each channel independently. Then we re-estimate the sources power in each channel, optimizing Eq. 2 only with respect to the p_l parameters, then we start over the optimization of the lighting parameters with fixed p_l's. This process is pursued until convergence is reached and we finally end up with an unlighted texture map and precise lighting parameters for each channel. Note that an indetermination about the absolute color of the albedo remains since we cannot recover the absolute color of the illuminant without a process similar to the white balancing of cameras.

4. Results

We have demonstrated the algorithm workflow using nine images of the teapot of Figs.1,2,3 that was lit by three source of different colors. As shown in Fig.1 (c), the algorithm correctly estimates the number and orientation of the light sources, which lets us synthetize images with the right shading and correct location of the specularities. To emphasize this point, in Fig.4 (a) we show a tenth image that was *not* used during the computation and demonstrate that we are able to produce the corresponding synthetic image, shown without texture in Fig.4(b). Note that the specularities both on the spout and on the side of the teapot fall exactly at the right places. This indicates a very precise recovery since specularities are notoriously sensitive to the exact source positions. Note also that the specularities in the original images fell on highly textured areas, which had no adverse effect on the results thanks to our ability to discard texture maxima and retain only specular ones.

Fig.5 (a,b) depicts a small industrially painted Easter egg lit by a single directional source. The egg includes both highly textured areas and a medium sized area of constant albedo. From a sequence of such frames, the specularity detection algorithm was able to find the specular highlight in each image and to discard texture maxima. The synthetic image (d) has been generated by texture mapping the 3D model with the recovered unlighted texture. Note that the specularity has been completely eliminated.

Fig.6 depicts results from pictures of a human face. In this case the original scene was lit by a single directional source and 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. In particular the chin and the area of the cheek bones of the fitted model of Fig.6 (g) are very inaccurate, but the algorithm is robust enough to accommodate these errors.



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

Figure 5: Lighting parameters recovery in presence of texture. (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.

5. Conclusion

We have shown that, given a set of images and a rough 3D model of a moving object, we can recover the number and orientation of multiple light sources of potentially different colors even when the object is textured. As a result, we can produce unlighted texture maps by removing specular artifacts. This is achieved by following local gray-level maxima using a mean shift algorithm and deciding whether or not they represent specularities. The true specularities are then used to decide how many light sources there are and where they are.

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.

6. Appendix

We present here a brief reminder of the mean shift technique. A detailed study can be found in [3]. In order to find the maxima of a distribution, we start from the multivariate kernel density estimator that may be written as

$$\hat{f}\left(x\right) = \frac{1}{n}\sum_{i=1}^{n} K_H\left(\frac{x-x_i}{h}\right) \ ,$$

where n is number of data points and K(x) a kernel with its dxd bandwidth matrix H. Without developing the details, we consider radially symmetric kernel that satisfies

$$K(x) = c_{k,d} \cdot k(\|x\|^2) \ ,$$

where k(x) is an univariate kernel, called the profile of K, and $c_{k,d}$ is the volume of a d-dimensional sphere volume. We can rewrite the estimator equation

$$\hat{f}_h\left(x\right) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right) = \frac{c_{k,d}}{nh^d}\sum_{i=1}^{n} k\left(\left\|\frac{x-x_i}{h}\right\|^2\right) \ ,$$

where n is number of data points, d represent the data dimension and h is the single parameter for the bandwidth. After deriving this expression, and defining $g(x) = -k'(x)$, it can be found that

$$m_{h,G}(x) = \frac{1}{2}h^2 c\frac{\hat{\nabla} f_{h,K}(x)}{\hat{f}_{h,G}(x)} = \frac{\sum_{i=1}^{n} x_i \cdot g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x \ ,$$

This equation signifies that $m_{h,G}(x)$, the mean shift, is a vector in the direction of $f_{h,k}(x)$'s gradient and normalized by the density estimated in (x) with the kernel G(x). This desirable normalization implies that the mean shift will have a large norm in low density area and a small one in region of high density which is useful for a quick and precise localization of the maxima.

To find the mode of a distribution in the neighborhood of a starting sample point, a d-dimensional window is placed on the sample point and the mean shift vector is evaluated. The window is next moved according to this vector and the mean shift is reevaluated. This is repeated iteratively until convergence.

References

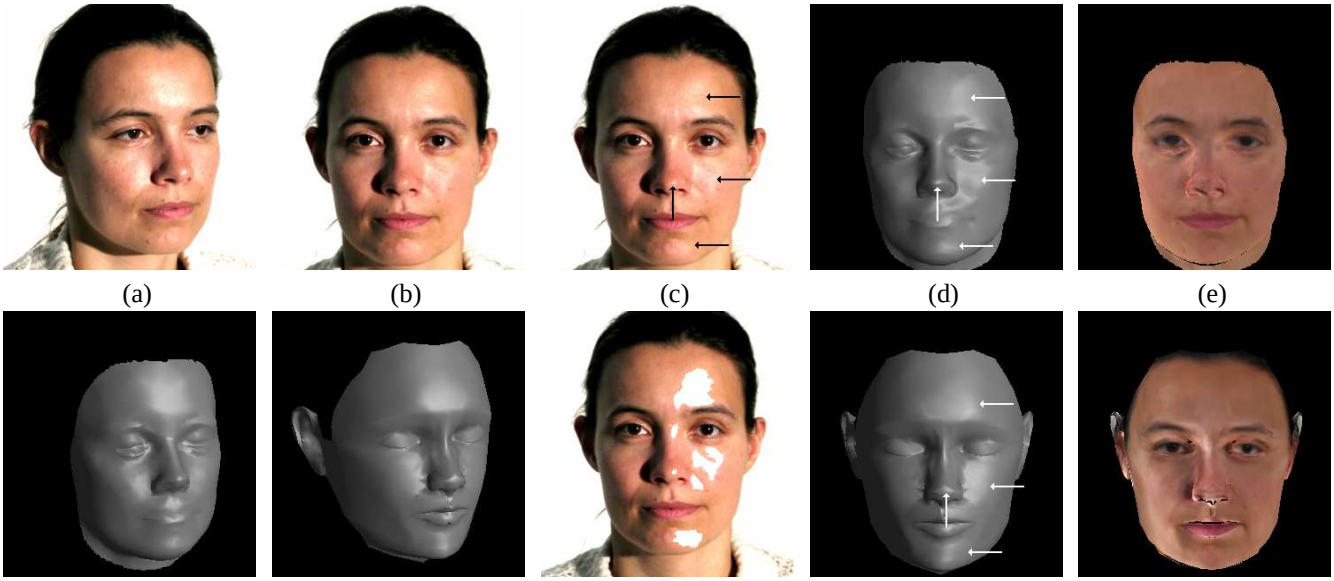
[1] E. Beauchesne and S. Roy. Automatic relighting of overlapping textures of a 3d model. In *CVPR*, pages 166–176, 2003.

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

[3] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, 2002.

[4] D., R. T. Tan, K. Hara, and K. Ikeuchi. Polarization-based inverse rendering from a single view. In *ICCV*, 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, 2001.



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

(f) (g) (h) (i) (j)

Figure 6: Insensitivity to the quality of the model. (a,b) Two out of nine input images. (c) The arrows point towards the main specularities. (d) Shaded view, using the recovered parameters, of the high resolution 3D scan used in the recovery process. Note that the specularities appear at the right places. (e) Recovered albedo map. It is very uniform, except on the left side of the face because we do not explicitly take shadows into account. (f) The high-resolution scan. (g) A much lower resolution model computed using a structure from motion technique. (h) Specular areas detected using the 3D scan overlaid in white. (i,j) Shaded model and albedo map, recovered using the low resolution model, to be compared to (d,e). Note that the specularities appear at the right places.

[6] A. Hertzmann and S. M. Seitz. Shape and materials by example: A photometric stereo approach. In *CVPR*, pages 533–540, 2003.

[7] Realviz: ImageModeler. Copyright 2005 realviz s.a.

[8] A. Koschan K. Schluens. Global and local highlight analysis in color images. 2000.

[9] G. J. Klinker, S. Shafer, and T. Kanade. The measurement of highlights in color images. *IJCV*.

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

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

[12] S. R. Marschner and D. P. Greenberg. Inverse lighting for photography. In *CIC*, 1997.

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

[14] 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 *ICCV*, pages 599–606, 2001.

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

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

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

[18] K. Schluens and M. Teschner. Analysis of 2d color spaces for highlight elimination in 3d shape reconstruction. 1995.

[19] R. T. Tan and K. Ikeuchi. Separating reflection components of textured surfaces using a single image. In *ICCV*, pages 870–877, 2003.

[20] M. Varma and A. Zisserman. Estimating illumination direction from textured images. In *CVPR04*, pages I: 179–186, 2004.

[21] Y. Wang and D. Samaras. Estimation of multiple directional light sources for synthesis of mixed reality images. In *PCCGA*, pages 38–47, 2002.

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

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