

# Multiple-cue Illumination Estimation in Textured Scenes

Yuanzhen Li<sup>†‡</sup>

Stephen Lin<sup>†</sup>

Hanqing Lu<sup>‡</sup>

Heung-Yeung Shum<sup>†</sup>

<sup>†</sup>Microsoft Research, Asia\*

<sup>‡</sup>National Laboratory of Pattern Recognition, Chinese Academy of Sciences

## Abstract

*In this paper, we present a method that integrates cues from shading, shadow and specular reflections for estimating directional illumination in a textured scene. Texture poses a problem for lighting estimation, since texture edges can be mistaken for changes in illumination condition, and unknown variations in albedo make reflectance model fitting impractical. Unlike previous works which all assume known or uniform reflectance, our method can deal with the effects of textures by capitalizing on physical consistencies that exist among the lighting cues. Since scene textures do not exhibit such coherence, we use this property to minimize the influence of texture on illumination direction estimation. For the recovered light source directions, a technique for estimating their intensities in the presence of texture is also proposed.*

## 1 Introduction

The appearance of objects depends greatly on illumination conditions. Since substantial image variation can result from shading, shadows and highlights, there has been much research on dealing with such lighting effects, such as in face recognition where variations in appearance are generally greater for changes in illumination than for changes in face identity [5]. Because of the significant effect of lighting, it is often helpful to know the lighting conditions of a scene so that an image can be more accurately analyzed.

Recovery of illumination conditions is also important for computer graphics applications, such as inserting correctly-shaded virtual objects into augmented reality systems [9] and lighting reproduction for compositing actors into video footage [2]. While these graphics methods introduce special devices into a scene to capture the lighting distribution, estimation of illumination in a general image has proven to be a challenge.

---

\*Correspondence email: stvelin@microsoft.com

## 1.1 Previous works

Previous approaches for illumination estimation have obtained information from either shading, shadows or specular reflections. Most methods are based on shading, and many of these focus on recovering the direction of a single light source [18, 8]. To deal with the more common scenario of multiple illuminants, several methods have been proposed. Hough and Ahuja [4] solve a set of linear equations for intensities of sampled light directions. Yang and Yuille [15] use image intensities and surface normals at occluding boundaries to constrain illuminant directions. Ramamoorthi and Hanrahan [7] compute a low-frequency illumination distribution from a deconvolution of reflectance and lighting. Zhang and Yang [17] estimate lighting directions from critical points that have surface normals perpendicular to an illuminant direction. Based on this, Wang and Samaras [14] segmented images into regions of uniform lighting and then performed estimation by recursive least-squares fitting of the Lambertian reflectance model to these regions.

Cast shadows can also provide important information about light directions and intensities. Sato et al. [10, 12, 11] utilized brightness values within shadows to solve a system of equations for light source intensities at sampled directions.

Specularity-based methods typically rely upon a calibration sphere inserted into the scene, where specularity positions and the mirror reflection property give illuminant directions. Debevec [1] uses a mirrored ball to capture real-world illumination environments. Powell et al. [6] use three mirrored spheres at known relative positions to triangulate light source locations. Zhou and Kambhampettu [19] capture a stereo image pair of a sphere that exhibits both diffuse and specular reflection, where specularities are used to triangulate light positions and diffuse reflections provide information on light intensities.

These previous methods have shown some success in lighting estimation, but they also are restricted by some shortcomings. In shading-based estimation, critical points

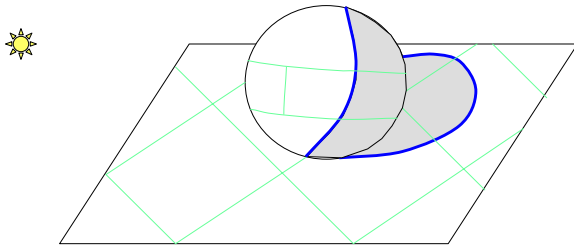


Figure 1. Consistency of shading critical points and shadow contours (blue) with respect to a illuminant direction. Such consistency does not exist between texture edges (green) on the sphere and on the planar surface.

and image intensity features are often difficult to detect accurately because their subtle appearance can be masked by image noise. For shadow techniques, shadows that result from frontal illumination tend to be occluded by the object, and frontal lighting is of most importance in image appearance. Methods based on specularities all require the use of a special calibration object, so they cannot be used for general images.

A major limitation of all these previous methods is that known and/or uniform reflectance of the surfaces is an essential condition. The presence of texture would disrupt the detection of critical points in shading. For shadow-based algorithms, texture would either distort brightness values or be mistaken for shadow edges. Because of this reflectance assumption, these methods are unsuitable for many real scenes.

## 1.2 Overview

Consideration of only a single illumination cue does not take full advantage of the rich information present in an image. In this paper, we propose to integrate shading, shadow and specular information in a single framework for estimating multiple illuminant directions. One reason for combination is that each individual cue is susceptible to error, so a consensus of these independent cues provides a more robust estimation. They also complement each other in that shading and specularities offer more reliable performance for frontal illumination, while shadows yield better clues about anterior lighting.

Another reason for integration is that we can exploit the physical consistencies that exist among these cues to avoid the confounding effects of scene texture. Fig. 1 illustrates this concept for shading and shadows. The curves drawn in green represent texture edges, and the curves drawn in blue denote lighting edges from critical points and shadows. For the given illuminant direction, there is a contour of critical

points (critical boundary) on the sphere and a cast shadow from this sphere onto the planar surface. A relationship between shading and shadows is that the line defined by the light source direction through a critical point must include a shadow edge point. In other words, there is a correspondence between critical points and shadow edge points with respect to the light direction. Such consistency does not exist among texture edges because there is no physical relationship between the texture on the sphere and the texture on the plane. We take advantage of this property to filter out texture effects from our illuminant estimation.

From Fig. 1, we can furthermore notice that some consistency also must exist within each cue. For shading, a subset of critical points on a critical boundary implies that the other points on the critical boundary should be present. A similar observation can be made about cast shadows. In our proposed method, we use this characteristic to help evaluate the presence of light sources, and also to reject texture edges from the estimation process.

Our method uses this distinction between lighting edges and texture edges to determine the light intensities of the estimated directions. From these edges, our technique locates pixels of the same texture color that have different lighting conditions. With these pixels a relationship between the two lighting conditions is formed without estimating texture albedos, and a system of these constraints is used to solve for the light intensities. A related approach is that of [11], where an additional image is captured without the occluding object, and is used to cancel out the effect of albedo in the image with the occluding object. An additional image without the occluding object, though, is rarely available for general canned images. Rather than using a correspondence between two registered images to deal with albedo, our method relates certain pixels within a single image to estimate illumination intensities in the presence of textures.

The primary contributions of this work towards illumination estimation are as follows:

- Integration of shading, shadows and specular reflections in a single framework.
- Consideration of data consistency within each cue.
- Utilization of consistency among cues to distinguish lighting changes from texture edges in a scene.
- Texture-independent estimation of light source intensities.

We assume that texture edges in the scene are not too densely distributed, since this would effectively mask all the shadow and shading information, and lead to numerous coincidental correspondences between critical points and shadow edges. Like many previous methods, we also assume the lights to be distant point sources and that diffuse reflectance is Lambertian.

There exists one recent work [13] that combines shading and shadows by adding the shadow-based estimation of [12] into the shading algorithm of [14]. In this approach, texture is not addressed, and consistency is only lightly considered by removing weak critical points for illumination directions processed by the shadow algorithm. Another related work attempts to distinguish shadow boundaries from reflectance edges by projecting image chromaticity values along the direction of the Planckian locus [3]. In this method, the chromaticity-space projection will mistakenly map together different texture colors that lie along the same Planckian locus, and some results indicate that shadow edges are reduced in magnitude but not eliminated.

In our algorithm, we take as input a single image that is annotated with partial geometry of the scene. As in [10, 12, 14] or [16], this geometry can be provided using an interactive modeling tool. From this information, we evaluate the consistency of shading, shadows and specular reflections as described in Section 2, and from their integrated information, the illuminant directions are estimated. The relative intensities of these illuminants are then computed using the technique described in Section 3. Experimental results are given in Section 4, followed by a discussion of our algorithm in Section 5.

## 2 Multiple-cue integration

The approach we take for integrating cues is based on physical consistency. For a potential illuminant direction, the existence of a light source is more likely if it is supported by more than one cue, such as when corresponding critical points, shadow edges and specularities are present. Additionally, evidence is more convincing if within a cue, support for the light direction is more complete. A full critical boundary or shadow contour provides greater validation of a light direction than a few scattered points. Attention to consistency both among cues and within each cue can lead to more robustness and can also avoid the problems presented by texture edges, which do not exhibit such coherence.

Our method first determines the expected positions of critical points, shading edges and specularities for hypothetical lighting directions sampled from a tessellated hemisphere. For a direction  $L$ , let  $D(L)$  denote the set of expected unoccluded shadow edges,  $C(L)$  be the set of critical point positions that correspond to the elements in  $D(L)$ , and  $S(L)$  be the expected specular peaks determined using the mirror reflection law. These sets are all computed from the scene geometry, and expressed in the 3D global coordinate frame. Note that some points in  $C(L)$  may be occluded in the image.

After computing the expected positions of the three cues for a given light direction, our method then checks whether

these cues are present in the image at these predicted locations. Shadow boundary points in the image are computed by Canny edge detection. For each point  $x_D$  in  $D(L)$ , we set  $P_D(x_D) = 1$  if the distance of its image projection from a detected edge falls below a threshold, which allows for some inexactness due to geometric inaccuracies. Otherwise,  $P_D(x_D) = 0$ . Since texture edges are also included in the edge detection results, they will introduce error into  $P_D$  that needs to be discounted by consistency measures.

To determine whether a critical point exists at an expected location  $x_C$ , we apply the critical point detection technique of [14] in the perpendicular direction to the critical boundary at this point. A benefit of discerning critical points with respect to a light direction  $L$ , instead of solving for them independently, is that a critical boundary can be computed, and the intensity profile features that characterize critical points are more pronounced along the perpendicular direction to this boundary. We set  $P_C(x_C) = 1$  for each unoccluded point  $x_C$  in  $C(L)$  whose image projection is detected as a critical point; otherwise,  $P_C(x_C) = 0$ . For occluded points  $x_C$ , we set  $P_C(x_C) = 1$ , so that they will not be considered inconsistent with the presence of other visible cues. A modification we add to the method of [14] is that our algorithm disregards points where intensity profile parameters cannot be accurately fit. Such cases occur in local windows of dense texture where intensity profiles are useless for critical point detection. Many texture edges will nevertheless be detected as critical points, and critical points found on surface normal discontinuities can possibly result from shading differences instead of an actual critical boundary.

Specularity peaks are simply computed as pixels whose intensity exceeds a threshold and is the local maximum within a 5x5 window centered on the pixel. We set  $P_S(x_S) = 1$  for a point  $x_S$  in  $S(L)$  if its image projection lies within a certain distance of a specular peak pixel; otherwise,  $P_S(x_S) = 0$ . Since some objects in the image may be composed of non-specular material, intra-cue consistency is not considered for the specularity cue. As a result, if  $P_S(x_S) = 1$  for any  $x_S$  in  $S(L)$ , we consider the specularity cue to support light direction  $L$ , and we denote this by  $Sp(L) = 1$ . Otherwise,  $Sp(L) = 0$ .

With these detected cues, inter-cue consistency for  $L$  is then expressed as

$$\sum_{x_D \in D(L)} [P_C(x'_D) \wedge P_D(x_D)] \vee [P_D(x_D) \wedge Sp(L)],$$

where  $x'_D$  denotes the point  $x_C$  that corresponds to the shadow point  $x_D$ . For a given light direction, this quantity measures the consistent presence of shadows and at least one other cue. Our method does not require consistency of all three cues because some objects exhibit little specular reflection, and critical points may become too subtle to detect

when there are numerous light sources present.

To incorporate intra-cue consistency, we account for how much of a shadow contour or critical boundary is present in the image. This is done by normalizing the above equation by the total number of expected points  $|D(L)|$  on the shadow contour. With the normalized values, the prospect of  $L$  containing an illumination source is measured as

$$\gamma(L) = \frac{1}{|D(L)|} \sum_{x_D \in D(L)} \frac{[P_C(x'_D) \wedge P_D(x_D)]}{\vee [P_D(x_D) \wedge Sp(L)]}. \quad (1)$$

For values of  $\gamma(L)$  that are a local maximum and exceed a given threshold,  $L$  is estimated as a light direction. To refine an estimated light source direction, our method samples more finely within the pencil of neighboring light directions on the tessellated hemisphere, and then finds the sample with the largest value of  $\gamma$ .

This integration based on consistency greatly reduces texture effects, since texture edges in shading and in shadows are generally inconsistent in position with respect to lighting and geometry. Some coincidental consistencies due to texture will often be present, but they typically do not affect the estimation results since they are too infrequent and random in location to exhibit a high degree of intra-cue consistency. With the estimated light directions, the corresponding lighting edges can be computed, then subtracted from the original set of critical points and shadow edges to give a set of estimated texture edges.

### 3 Light intensity estimation

After estimating the illuminant directions, the intensities of the light sources are computed in a manner that avoids the effects of texture. In our technique, we first divide the image into regions of uniform lighting conditions. For each image pixel, its visibility with respect to each light direction can be computed from the geometry. Pixels with the same visibility conditions are grouped together such that each group is a connected component in the image and is connected geometrically in the scene. Let  $S(R)$  denote the set of light sources visible from region  $R$ . For each region  $R$ , let the set of adjacent regions be represented by  $\Lambda(R)$ , where adjacency is determined from the geometry.

For each pair of adjacent regions, we find several pairs of pixels between the two regions such that each pair has a similar reflectance value. From the set of all selected pixel pairs, a system of equations is formed to solve for the light intensities. In our implementation, we utilize the Lambertian model for diffuse reflectance, so we locate pixel pairs that have the same albedo. Specifically, we first sample points along the boundary between each pair of adjacent regions, and take these sampled points as seed points. For each seed point, a circular window is grown such that it lies

within the two regions and its pixels do not lie within a certain distance from the estimated texture edges. This window contains two sets of pixels with the same texture color but with different illumination conditions. To reduce susceptibility to slight errors in illuminant directions, we discard from the window all pixels within a given buffer zone surrounding the region boundary.

Our method then determines the pixel of median intensity in each of the two sets. With this pair of pixels  $(x_{R_1}, x_{R_2})$ , the lighting conditions of the two regions  $R_1, R_2$  can be related. All the selected pixel pairs from all adjacent regions forms a set  $\Omega$ .

From the lighting additivity property, we can express the color of a pixel  $x$  in region  $R$  as

$$I(x) = \rho(x) \sum_{i \in S(R)} l_i L_i \cdot N(x) \quad (2)$$

where  $\rho$  is the albedo,  $N$  is the surface normal, and  $l_i$  is the light intensity from direction  $L_i$ . With this, we can form a system of equations for light intensities  $l_1, l_2, \dots, l_k$ , which are solved by least-squares minimization:

$$\arg \min_{l_2, l_3, \dots, l_k} \sum_{(x_{R_1}, x_{R_2}) \in \Omega} \left[ \frac{\sum_{i \in S(R_1)} l_i L_i \cdot N(x_{R_1})}{\sum_{j \in S(R_2)} l_j L_j \cdot N(x_{R_2})} - \frac{I(x_{R_1})}{I(x_{R_2})} \right]^2 \quad (3)$$

where  $l_1 = 1$ , and  $k$  is the number of light directions. This quotient of Lambertian equations effectively avoids dependence on the unknown albedo values. Note that each  $R, G, B$  color channel can be computed separately in this way, so that illuminant colors could be recovered as well. The intensity of  $L_1$  is set to 1 because only relative intensity values can be computed from the system of equations. This arises from an inherent ambiguity between illumination intensity and albedo magnitude, as seen in Eq.(2).

The magnitude of the minimized error in (3) is an indicator of errors in the estimated light directions. To improve the direction estimates, we perturb the light directions in an effort to reduce the error in (3). In this manner, both the illuminant directions and intensities can be refined.

### 4 Experimental Results

With our proposed algorithm, we estimated the illuminant directions for different images. The number of sampled illuminant directions is 1800, and for each estimated direction, 72 more finely sampled directions are used for more precise estimation. The lights we used are all white sources, and for all images we set the  $\gamma$  threshold to 0.6. The distance threshold for computing  $P_D(x_D)$  is 8 pixels for coarse sampling and 2 pixels for fine sampling. For light intensity estimation, we sample the lighting boundary at every 15 points, and the buffer zone width from the lighting boundary and the texture edges is 5.

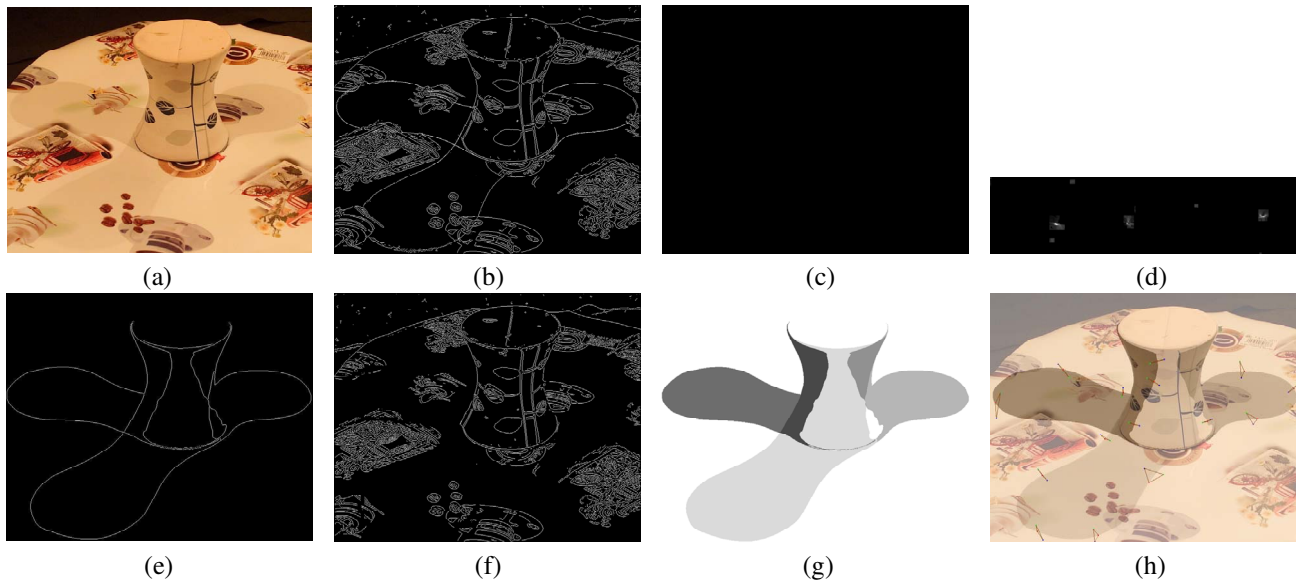


Figure 2. Illumination estimation for a textured non-specular vase on a textured plane. (a) original image; (b) Canny edges; (c) detected specularities; (d) map of  $\gamma$  with respect to spherical angles of  $L$ ; (e) illumination edges for estimated light directions; (f) remaining texture edges; (g) regions of uniform lighting condition; (h) pixel pairs for light intensity estimation (marked on the blended image of the (a) and (g)).

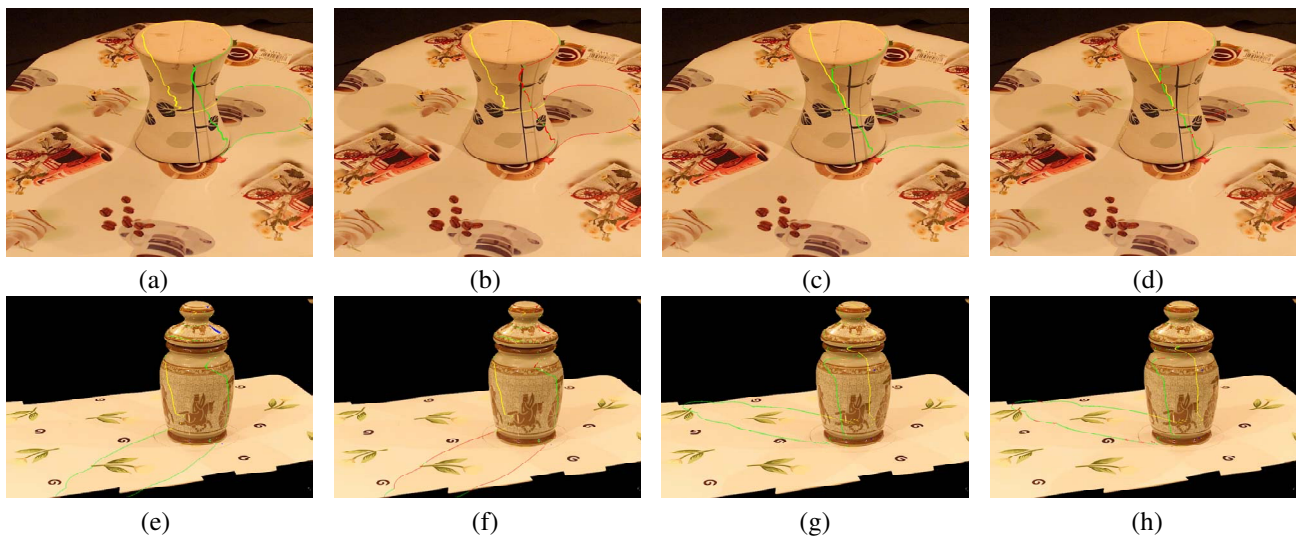


Figure 3. Intermediate integration results. (a,e) expected lighting edges for eventually estimated light directions (green: visible, yellow: occluded, blue: specular); (b,f) detected lighting edges and specularities (red); (c-d,g-h) expected and detected edges for light directions that are not estimated.

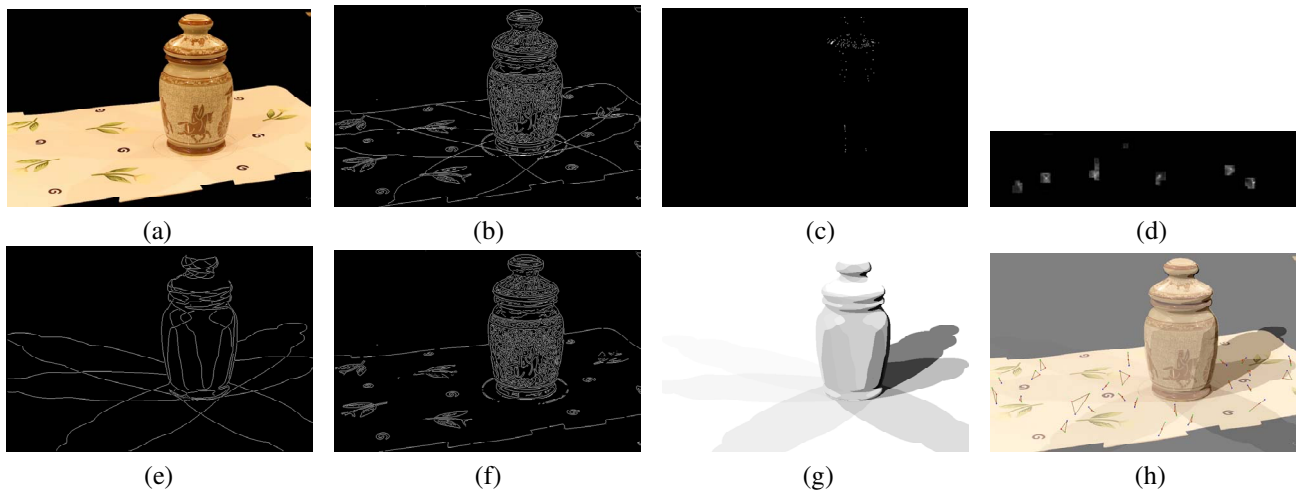


Figure 4. Illumination estimation for a textured specular bottle on a textured plane. (a-h) same as in Fig.2.

Light source	Ground truth				Estimated			
	x	y	z	intensity	x	y	z	intensity
1	0.2111	0.7935	0.5707	1.0000	0.2038	0.7882	0.5807	1.0000
2	0.5499	-0.4581	0.6984	0.9081	0.5495	-0.4450	0.7071	0.9234
3	-0.7925	0.1991	0.5764	0.8709	-0.7882	0.2038	0.5807	0.8852

Table 1. Ground truth comparison of light estimation for vase image

Light source	Ground truth				Estimated			
	x	y	z	intensity	x	y	z	intensity
1	0.5549	-0.6680	0.4959	1.0000	0.5536	-0.6716	0.4924	1.0000
2	0.1640	-0.6824	0.7123	0.9769	0.1636	-0.6815	0.7132	0.9655
3	-0.7544	-0.2699	0.5984	0.9330	-0.7528	-0.2666	0.6018	0.9113
4	-0.4455	0.6567	0.6084	0.9707	-0.4436	0.6577	0.6088	0.9821
5	0.3232	0.7617	0.5616	0.9853	0.3201	0.7540	0.5736	0.9607
6	0.7321	0.5055	0.4567	1.1001	0.7320	0.4937	0.4695	1.1352

Table 2. Ground truth comparison of light estimation for specular bottle image

The first image, displayed in Fig.2(a), is of a textured vase lying on a textured planar surface. There are three light sources present. The shadow edge points found by Canny edge detection are shown in (b). The specular cue is not present in this scene, as shown in (c), so we depend on the consistency between shading and shadows for our estimation. After integrating shading and shadows using our algorithm, the support of each lighting direction is depicted in (d), where the horizontal direction represents the azimuth angle and the vertical direction denotes the elevation angle. From this, our method determines three illuminants which have critical points and shadow edges exhibited in (e). Removing the illumination edge points from the Canny results leaves the texture edge points shown in (f). With the estimated directions, the image can be partitioned into regions of uniform lighting condition illustrated in (g). With the pixel pairs marked in (h), where each pixel pair and the corresponding seed point on the lighting edge are connected by a triangle, the illuminant intensities are computed.

Experiments were also done for an image of a specular object lying on a textured surface, illuminated by six light sources. The original image and the results are displayed in Fig.4. In this scene, estimation of critical points cannot be done reliably because the dense textures prevent accurate fitting of intensity profiles. Since most shading information is consequently disregarded, and the burden of proof is shifted to the consistency between shadows and specularities. For instances where some points are incorrectly detected as specular, such as some bright textures on the lid, our method relies on intra-cue consistency of the shadow edges to avoid errors in illumination direction estimates.

Some intermediate results of the integration process are exhibited in Fig.3. For one of the eventually estimated light directions, we show the expected critical points, shadow edges, and specular peaks in (a) and (e), where the visible illumination edge points are marked in green, the occluded ones are in yellow, and the specular peaks in blue. (b) and (f) give the corresponding integration results, where red denotes points on the expected shadow edges, critical points and specular peaks that are detected. As can be seen for the given light directions, much consistency exists and the integration results provide strong evidence. We also show examples for light directions that are ultimately not estimated. In (c) and (g), the expected lighting edges for this direction are drawn, and in (d) and (h) the expected lighting edges that are detected are marked by red as in (b) and (f). The false detections in (d) and (h) do not exhibit much inter-cue or intra-cue consistency, so these two light directions are not strongly supported.

Table 1 and Table 2 list comparisons of the estimated directions and intensities with ground truth obtained from high dynamic range images of a mirrored sphere. The light direction is expressed as a unit vector in a coordinate system

centered on the sphere. Intensity values are given relative to light source 1.

Although our experiments have been done for single objects, multiple objects would provide additional data. In the case of multiple objects, it would be assumed that they do not interact with each other, such as by light occlusion or having overlapping shadows.

## 5 Discussion

We note that cue integration does not apply for some cases of backlighting, since only cast shadow information may be available. In this case, illumination estimation could potentially be performed based on consistency within the shadow cue. For most computer vision purposes though, the backlighting distribution is of far less importance than frontal lighting, since it has relatively little impact on object appearance. We also note that although some cast shadows are occluded for frontal lighting, this does not affect the consistency calculations of critical points on the visible object surface, because their corresponding shadow edges lie to the sides of the object, not behind it.

Detection of low-level features such as edges and critical points can be affected by sensor noise and imaging conditions. Despite this obstacle, we base our method on illumination edges because textures significantly complicate lighting estimation based on image intensities. To obtain illumination information from image colors, the reflectances over the textured surfaces need to be recovered, and this remains a challenging research problem. The positions of lighting edges, however, are independent of scene reflectances and can be used for illuminant direction estimation without regard of reflectance properties.

Our algorithm does not deal with extended or diffuse lighting. While it would be useful to handle both textures and continuous lighting distributions together, this presents a major challenge given our scenario of only a single image with no assumptions about scene reflectance. Previous methods that address continuous lighting require additional images or restrictive assumptions on the scene. We purposely do not use more than one image or knowledge of the reflectance, since this information is not available for general canned images. The combination of continuous lighting and scene textures is nevertheless an important problem, and we plan to investigate in future work possible ways to make this problem more manageable.

In this paper, we have presented a method for illumination estimation that integrates different cues for greater robustness and for handling scene texture. Consistency among cues and within each cue is the basic principle behind our integration framework. To reduce the influence of texture on the illumination estimation process, our algorithm exploits the physical coherence between shadow

edges and critical points that texture edges do not have. From estimated lighting directions and texture edges, a system of constraints is formulated for solving the lighting intensities in a texture-independent manner. Although the presence of continuously varying texture could render the shading or shadow cue useless, the ability of our method to handle a fair amount of texture significantly increases the applicability of illumination estimation methods, which have not addressed texture until now.

## References

- [1] P. Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proc. ACM SIGGRAPH*, pages 189–198, 1998.
- [2] P. Debevec, A. Wenger, C. Tchou, A. Gardner, J. Waese, and T. Hawkins. A lighting reproduction approach to live-action compositing. *ACM Trans. on Graphics*, 21:547–556, July 2002.
- [3] G. D. Finlayson, S. D. Hordley, and M. S. Drew. Removing shadows from images. In *Proc. European Conf. on Computer Vision, LNCS 2353*, pages 823–836, 2002.
- [4] D. R. Hougen and N. Ahuja. Estimation of the light source distribution and its use in integrated shape recovery from stereo and shading. In *Proc. Intl. Conf. on Computer Vision*, pages 148–155, 1993.
- [5] Y. Moses, Y. Adini, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. In *Proc. European Conf. on Computer Vision*, pages 286–296, 1994.
- [6] M. W. Powell, S. Sarkar, and D. Goldgof. A simple strategy for calibrating the geometry of light sources. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23:1022–1027, 2001.
- [7] R. Ramamoorthi and P. Hanrahan. A signal-processing framework for inverse rendering. In *Proc. ACM SIGGRAPH*, pages 117–128, 2001.
- [8] D. Samaras and D. Metaxas. Coupled lighting direction and shape estimation from single images. In *Proc. Intl. Conf. on Computer Vision*, pages 868–874, 1999.
- [9] I. Sato, Y. Sato, and K. Ikeuchi. Acquiring a radiance distribution to superimpose virtual objects onto a real scene. *IEEE Trans. on Visualization and Computer Graphics*, 5:1–12, 1999.
- [10] I. Sato, Y. Sato, and K. Ikeuchi. Illumination distribution from brightness in shadows: Adaptive estimation of illumination distribution with unknown reflectance properties in shadow regions. In *Proc. Intl. Conf. on Computer Vision*, pages 875–883, 1999.
- [11] I. Sato, Y. Sato, and K. Ikeuchi. Illumination distribution from shadows. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 306–312, 1999.
- [12] I. Sato, Y. Sato, and K. Ikeuchi. Stability issues in recovering illumination distribution from brightness in shadows. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages II:400–407, 2001.
- [13] Y. Wang and D. Samaras. Estimation of multiple directional light sources for synthesis of mixed reality images. In *Proc. Pacific Graphics*, pages 38–47, 2002.
- [14] Y. Wang and D. Samaras. Estimation of multiple illuminants from a single image of arbitrary known geometry. In *Proc. European Conf. on Computer Vision, LNCS 2352*, pages 272–288, 2002.
- [15] Y. Yang and A. L. Yuille. Sources from shading. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 534–539, 1991.
- [16] L. Zhang, G. Dugas-Phocion, J.-S. Samson, and S. M. Seitz. Single view modeling of free-form scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages I:990–997, 2001.
- [17] Y. Zhang and Y.-H. Yang. Multiple illuminant direction detection with application to image synthesis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23:915–920, 2001.
- [18] Q. Zheng and R. Chellappa. Estimation of illuminant direction, albedo, and shape from shading. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:680–702, 1991.
- [19] W. Zhou and C. Kambhamettu. Estimation of illuminant direction and intensity of multiple light sources. In *Proc. European Conf. on Computer Vision, LNCS 2353*, pages 206–220, 2002.