

Light source detection in photographs

Jorge Lopez-Moreno¹, Sunil Hadap², Erik Reinhard³ and Diego Gutierrez¹
¹ Universidad de Zaragoza ² Adobe Systems Inc ³ University of Bristol

Abstract

Common tasks related to image processing or augmented reality include rendering new objects into existing images, or matching objects with unknown illumination. To facilitate such algorithms, it is often necessary to infer from which directions a scene was illuminated, even if only a photograph is available. For this purpose, we present a novel light source detection algorithm that, contrary to the current state-of-the-art, is able to detect multiple light sources with sufficient accuracy. 3D measures are not required, only the input image and a very small amount of unskilled user interaction.

Categories and Subject Descriptors (according to ACM CCS): Computing Methodologies [I.3.7]: Computer Graphics—3D Graphics; Computing Methodologies [I.4.10]: Image Processing and Computer Vision—Image Representation

1. Introduction

This paper deals with the problem of obtaining the positions and relative intensities of light sources in a scene, given only a photograph as input. This is generally a difficult and under-constrained problem, even if only a single light source illuminates the depicted environment.

Traditionally light probes are used to acquire the lighting data. A light probe is an object of known 3D shape and BRDF properties (Bidirectional Reflectance Distribution Function; description of the reflectance properties of the material) which is positioned in the scene when the image is captured. Unfortunately, in several cases this technique is not applicable: e.g. paintings, photographs taken under uncontrolled conditions, etc. Instead, it would be possible to use any object in the image if geometry information were available, allowing light source positions or directions to be estimated [GHH01]. Conversely, if the light source is known, the 3D geometry can be approximately recovered, an ill-posed problem known as shape-from-shading [ZTCS99].

However, we are interested in the problem of light source recovery without the benefit of *any* geometric prior. To this end, we first carry out a psychophysical experiment to quantify the accuracy with which humans can generally detect light sources. The results of this experiment are then used to validate the results of our light detection algorithm, both numerical and perceptually. We then use any existing ob-

ject in the image as a de-facto light probe. We have found that assuming a globally convex shape for such light probe is sufficient to reconstruct light directions. The user only needs to provide its silhouette in the image, a task similar or simpler than other existing image editing applications [OCDD01, LHE*07]. We then analyze the information in the contour and the gradients contained in the shape to infer the light directions and relative intensities.

Real environments are likely to contain multiple light sources. We found that in practice finding up to four sources, combined to provide similar illumination as in the image, suffices for most situations. This keeps the dimensionality of the solution manageable, in a way similar to professionally lit environments, which are usually lit by a three-light setup. Additionally, although we do assume in principle that the chosen light probe is Lambertian, we will show that this is not a strong requirement.

We believe that by analyzing lighting consistency between images our algorithm can help improve several types of applications, such as Photo Clip Art [LHE*07], Interactive Digital Photomontage [ADA*04] or Photo Tourism [SSS06].

2. Previous Work

The computation of light source directions from images is an ill-posed problem, with many possible solutions leading to the same observed image. As a result, assumptions about

the environment are made, known geometry must be present in the scene, or extra information must be captured to change the problem into a solvable one.

For detecting single light sources, by itself a strong assumption on the environment, a local analysis of the surface and image derivatives may be used to estimate the direction of the light source [Pen82, BH85, LR89]. Alternatively, occluding contours within a single object [Hor86, NE01] or texturing [KP03, VZ04] provide clues as to where the light is coming from.

To detect multiple lights, the environment could be photographed along with the aforementioned lightprobe: a calibration object of known size and shape. For instance, a Lambertian sphere could be employed, and subsequently analysed for multiple light source directions [HA93, ZY01], or alternatively multiple specular spheres can be triangulated for the same purpose [PSG01, LF06]. Combinations of Lambertian and specular spheres have also been used [ZK02]. Finally, reflections of a human eye can be analysed to detect light sources [NN04].

Another way to overcome the under-constrained nature of the problem is to use a range camera to record geometry, allowing light sources to be inferred from the combination of photograph and range data [MG97]. Known geometry can be used to the same effect [WS02, SSI99]. In contrast, our approach is free of previous restrictions, e.g.: there is no need for a calibration object or known geometry. Furthermore, we do not require shadows being cast on nearby objects, nor is any camera information needed.

3. Light Detection

Consider a typical input image as depicted in Figure 1a. The problem at hand is to estimate the number of illumination sources, their dominant directions and the relative intensities. We propose to use any object in the image as a virtual light-probe, as long as it covers a reasonable area in the image. The user provides the outline defining the object, typically with the aid of a smart selection tool [WAC07]. We do not assume any restrictions on the shape, the color or any other peculiarities for the object.

3.1. Assumptions and Overview

To achieve a reasonable solution, we rely on the particular characteristics of human vision. In estimating illumination, the human visual system tends to ignore the local shape variations, and treats the object as a globally convex geometry [LB01]. We also leverage the tendency of the human visual system to perceive objects correctly as long as the illumination is locally consistent [OCS05]. Further, we observe that humans are surprisingly good at estimating back-lighting using cues from shadows [KvDP04]. Based on these assumptions, we devise a three-step algorithm as follows.

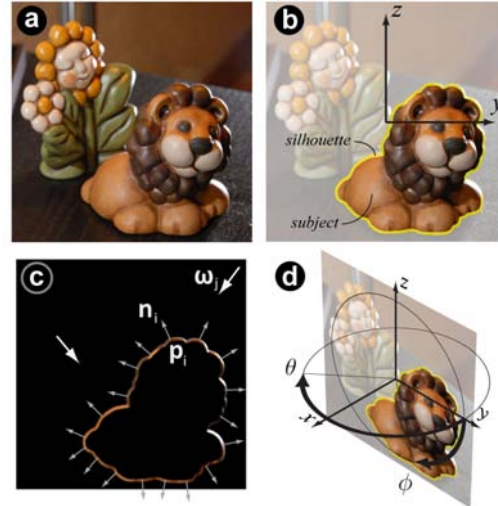


Figure 1: a) Input Image b) Object c) Silhouette Normals d) Coordinate System

1. To estimate the number of lights N and their respective azimuth coordinates $\phi_i, i = 1 \dots N$; we analyze the intensity variation along the silhouette of the object. We assume that the surface normals of the object at the silhouette lie in the image plane [Hor86]. Using the silhouette normal assumption and the nominal diffuse lighting equation, we can accurately predict the azimuth coordinate ϕ of the individual lights. The number of lights and their relative intensities are estimated in an iterative fashion.
2. We use the globally convex assumption to estimate the zenith angles $\theta_i, i = 1 \dots N$ and relative intensities I_i . For each light detected in the first step, we sweep the image from the silhouette to the interior along the azimuth direction, looking for maxima in the shading. The corresponding shape normal at the maxima \bar{n}_i is indicative of the direction of the light and thus the zenith angle θ_i . To robustly handle local non-convexities and back-lighting we detect and use shadows. Following Khan et al. [KRFB06], we differentiate the relatively high frequency variations of the luminance due to albedo (texture), from the low frequency variations of luminance due to shading, using bilateral filtering.
3. By analyzing median intensity in the shadow areas, we estimate the ambient light intensity.

Each of these steps are explained in detail in the following sections.

However, we start by defining the coordinate system used. As depicted in Figure 1d, the image plane is assumed to be aligned with the y - z plane, whereas the x axis points out of the image plane. The origin lies at the center of the image. We also set a polar coordinate system such that the equator

is aligned with the image plane and the axis is aligned with x axis. Thus the direction of a light is uniquely identified by the azimuth angle ϕ and the zenith angle θ .

3.2. Estimating Azimuth Angles

We assume that the normals at the silhouette lie in the image plane. We further assume that there are N discrete lights, each being either a directional light or a far away point light (we estimate N below). Thus each light is uniquely characterized by its unknown luminance L_j and unknown unit direction ω_j , $j = 1 \dots N$. To analyze the intensity variation of the silhouette pixels, we assume a nominal Lambertian surface. Consider all pixels $\{\mathbf{p}_i\}$ that belong to the silhouette. Let \mathbf{n}_i be the normal and L_i^v be the known luminance of the object at point \mathbf{p}_i :

$$L_i^v = \sum_{j=1}^N \Omega_{ij} L_j \quad (1)$$

$$\Omega_{ij} = \Omega(\mathbf{n}_i, \omega_j) = \begin{cases} 0 & \text{if } \mathbf{n}_i \cdot \omega_j < 0, \\ K_i^d \mathbf{n}_i \cdot \omega_j & \text{if } \mathbf{n}_i \cdot \omega_j \geq 0 \end{cases}$$

where Ω represents the occlusion function of pixel i in the viewing direction and K_i^d is the unknown diffuse reflectivity or albedo. We encode the normals, which are in the y - z plane, as polar coordinates $\phi_i^n \rightarrow \mathbf{n}_i = [0, \sin(\phi_i^n), \cos(\phi_i^n)]^T$, $0 \leq \phi_i^n \leq 2\pi$.

To estimate the lights' azimuth angles ϕ_j^l , we use a k-means clustering algorithm. In traditional k-means clustering algorithms, each data point belongs to a certain cluster, and affects the centroid of only that cluster. Unfortunately, a silhouette pixel may be illuminated by more than one light. Thus, we should not partition the pixels into exclusive clusters. Instead, we devise a partial voting scheme based on the occlusion function to form the "fuzzy" clusters, and simultaneously compute the corresponding centroids as the lighting directions, as outlined in Algorithm 1.

We go through the list of pixels sorted by luminance (line 7) to perform the normal voting. Notice that each silhouette normal ϕ_i^n votes all the N light clusters (lines 10 to 16), according to their luminances L_j^v . However, each normal only partially votes to each light cluster according to the occlusion function (line 12). For that, the individual occlusion function with respect to each light direction Ω_{ij} is normalized with the aggregate of occlusion functions $\Omega_i^\oplus = \sum_j \Omega(\mathbf{n}_i, \omega_j)$.

We repeat the voting process (line 7 to 17) until we converge on the light azimuth angles ϕ^l (line 6 and 18). The choice of the initial guess (line 3) for the azimuth angles is important to ensure a speedy and effective convergence. We assign the azimuth of the brightest pixel's normal ϕ_1^n to the first light ϕ_1^l . For the successive lights, we set the azimuth angles to $\phi_1^l + 2\pi(j-1)/N$.

For the estimation of the number of lights N , our approach

Algorithm 1 Contour Voting - N lights

Require: $\mathbf{L}^v \equiv \{L_j^v\}$ {discrete luminances}
Require: $\mathbf{n} \equiv \{\mathbf{n}_i\}$ {silhouette normals}
Require: $\phi^n \equiv \{\phi_i^n\}$ {azimuth coordinates of the normals}

- 1: sort($\mathbf{L}^v, \mathbf{n}, \phi^n$) {sort by decreasing luminances}
- 2: $\phi^l \equiv \{\phi_j^l\} \mid j \in [1 \dots N]$ {azimuth coordinates of the lights}
- 3: seed(ϕ^l)
- 4: $\alpha^\oplus \equiv \{\alpha_j^\oplus\} \mid j \in [1 \dots N]$ {aggregate of weights per light}
- 5: $\alpha^\oplus \leftarrow \mathbf{0}$
- 6: **repeat**
- 7: **for all** $L_i^v \in \mathbf{L}^v$ **do**
- 8: $\omega_j \leftarrow [0, \sin(\phi_j^l), \cos(\phi_j^l)]^T$ {current direction}
- 9: $\Omega_i^\oplus \leftarrow \sum_j \Omega(\mathbf{n}_i, \omega_j)$ {total occlusion weight}
- 10: **for all** $j \in [1 \dots N]$ **do**
- 11: $\omega_j \leftarrow [0, \sin(\phi_j^l), \cos(\phi_j^l)]^T$ {current direction}
- 12: $\alpha_{ij} \leftarrow L_i^v \Omega(\mathbf{n}_i, \omega_j) / \Omega_i^\oplus$ {weight of normal i }
- 13: $\phi_j^l \leftarrow \alpha_j^\oplus \phi_j^l + \alpha_{ij} \phi_i^n$ {update direction}
- 14: $\alpha_j^\oplus \leftarrow \alpha_j^\oplus + \alpha_{ij}$
- 15: $\phi_j^l \leftarrow \phi_j^l / \alpha_j^\oplus$
- 16: **end for**
- 17: **end for**
- 18: **until** convergence(ϕ^l)
- 19: **return** ϕ^l

will subsequently increase the number of lights $N = 1..i$ until either the error is below a given tolerance or the added light source does not improve the result. In practice, we found that the number of iterations is usually below $N = 4$. This is due to the quantization associated with the image's finite bit-depth. As the number of opposed lights increases, the variation in the shading over the surface decreases and becomes rather constant.

Although the proposed voting method has built-in resistance to local variations in albedo due to its search of global tendencies, ultimately the results are biased if the points in the contour form large clusters with very different luminance values as the first image of Figure 2a demonstrates.

It is possible to reduce this bias, in a second pass, as follows. Once we have a set of N centroids (light directions), we go through all the voting pixels assigned to each k-group, corresponding to a light direction. We then check that the dot product of the normal and the estimated light direction yields a luminance value equal to the original luminance of the pixel, fractioned by its occlusion function. If not, we force the fractional albedo of the pixel to be coherent with the fractional luminance of the brightest pixel in the group. Then we repeat the contour voting algorithm. This correction in albedo values usually produces small shifts (10 to 20

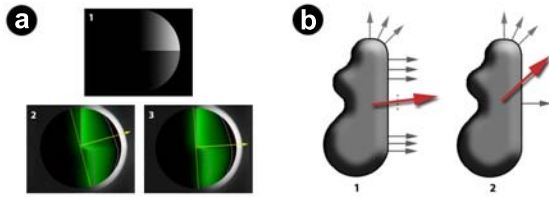


Figure 2: *a*¹) Sphere with a change in the albedo *a*²) Initial biased estimation due to higher albedo *a*³) Corrected light direction estimate *b*¹) An estimate incorrectly biased due to the geometry of the silhouette *b*²) The correct result, after eliminating multiple normals.

degrees) in the directions in the case of extreme albedo variations (Figure 2a).

As in other previous approaches based on contour analysis [YY91, VY94, NE01], the first step will fail if the light is situated around the x axis, e.g.: $\theta \approx \pi/2$. In this case there is no variation in luminances due to shading. This would result in erroneous estimation of azimuth angles. However, the final direction of the light would be estimated accurately in the second step when we analyze the shading in the interior.

Finally, we correct the potential bias along the direction stemming from the geometry of the silhouette. As depicted in Figure 2b, a significant number of silhouette normals point in y axis, biasing the resultant light towards that direction. We propose to correct this by eliminating multiple normals. We choose a set of discrete normal directions ϕ_i^n and distribute all the silhouette normals into bins. Then we compute the average of luminances for each bin \bar{L}_i and use this set of silhouette normals and luminances instead.

3.3. Estimating Zenith Angles and Intensities

To estimate zenith angles $\{\theta_j\}$ accurately, we disambiguate the luminance variations due to shading from the variations due to texture, which are relatively high in frequency. We use bilateral filtering to remove high frequencies, while keeping lower frequency content which is typically attributed to shading [KRFB06].

Then, for each light detected in the previous step, marching in the light's direction $\omega_j = \omega(\phi_j^l)$ from the silhouette to the interior, we analyze the luminances. As the pixels are lit by multiple lights, this directional derivative of the luminance $\omega_j \cdot \nabla L^v$ is the main indicator of the shading due to a particular light j aligned to its direction. There are two cases of luminance variations in the interior.

Case 1: If the directional derivative $\omega_j \cdot \nabla L^v$ is positive at the silhouette, the light is towards the camera from the image ($\theta \geq 0$). In this case, the luminances will continue to

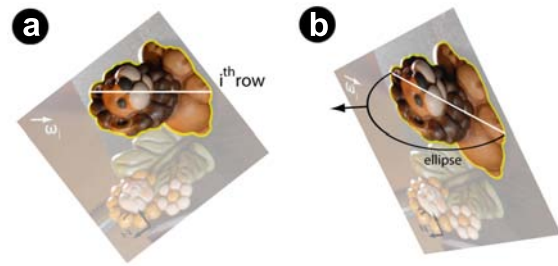


Figure 3: Estimating zenith angle *a*) Scanning in lights direction for highlight or shadow *b*) Ellipsoidal geometry

increase as we march along the direction of the light, to reach the first local maximum. We denote this point as \mathbf{p}_j^{hi} . At this point the surface normal points in the direction of the light, e.g.: $\theta_j = \theta^n(\mathbf{p}_j^{hi})$. We ignore all the pixels thereafter as the geometry might be self-occluding or under the influence of another light.

Case 2: At the silhouette, if the directional derivative is negative, this is an indication of back lighting ($\theta < 0$). The luminances will successively decrease as we march along the light direction to reach a singularity. This point is the first self-shadow point \mathbf{p}_j^{lo} , and is marked by either a change of sign in the gradient of the directional derivative $\omega_j \cdot \nabla L^v$ or a zero value of its luminance L^v . A change of sign will be produced when the contribution to the luminance value at that point by a second light is greater than the contribution of L^v . At this point, the surface normal is perpendicular to the light direction, e.g.: $\theta_j - \theta^n(\mathbf{p}_j^{lo}) = \pi/2, \theta_j < 0$.

To estimate the normal at each point we cannot rely on shape-from-shading due to the overlapping of multiple lights. It is not possible to know a priori which combination of light sources is contributing to a certain point. Good solutions for estimate a valid normal at points \mathbf{p}_j^{hi} or \mathbf{p}_j^{lo} in arbitrary images do not exist [ZTCS99].

Furthermore, this is complicated if two given points on the surface of the object are lit by a different and unknown number of light sources. Wang et al. [WS02] determined the number of lights, but they could perform this thanks to accurate knowledge of 3D depth and normals. Instead, we revert once more to our global convexity assumption and fit an ellipse along the scanline: one of the axis is given by the intersection of such scanline and the silhouette; the other axis will approximate the object convexity and is a user parameter. By default, both axes are equal (in fact defining a circumference). The surface normal is subsequently assumed to be the normal of the ellipse at the point under consideration.

We could start marching along the light direction from a brightest silhouette point that corresponds to the light. However, in order to minimize the influence of albedo variations,

we scan the light direction from multiple silhouette points. One way to realize this scheme is to rotate the image such that the light direction $\omega(\phi_j^l)$ is aligned with the y-axis and the light on the left, see Figure 3. Then we simply scan each raster line i , starting from the silhouette boundary on left into the interior. We detect the set of points $\{\mathbf{p}_{ij}^{hi}\}$ or $\{\mathbf{p}_{ij}^{lo}\}$, corresponding zenith angles $\{\theta_{ij}\}$ and luminances L_{ij}^v . Thus for the light j , the resultant zenith angle is the weighted sum:

$$\theta_j = \frac{\sum_i L_{ij}^v \theta_{ij}}{\sum_i L_{ij}^v} \quad (2)$$

By using two objects in the image as light probes and repeating the process for the second one, we can approximate the position of a point light source, e.g.: a source that is not infinitely far away. Given that the directions computed for both objects cannot be absolutely precise (we assume directional lights), there will be no intersection point. We simply place the light source halfway between the points d_1 and d_2 defining its minimum distance.

Once we have estimates of the light directions, estimating the relative intensities is fairly straight forward. For each light j , we compute the total sum of luminances normalized by the occlusion function of the light, over all the pixels i of the contour of the object. The intensity of the light I_j is proportional to:

$$I_j \propto \sum_i L_{ij}^v / \Omega(\mathbf{n}_i, \omega_j) \quad (3)$$

Any potentially remaining light sources will be treated by our algorithm as ambient illumination which we will explain next.

3.4. Ambient Illumination

The shading contribution of the ambient light is assumed to be constant for all pixels and we can therefore estimate its intensity by analyzing pixels in the shadow regions. We already have detected the shadow lines in the previous step. The region bounded by these shadow lines is determined to be a shadow region. We average the set of samples along these boundaries. This ambient intensity estimate is also relative to the previously detected lights.

4. Results

We have tested our algorithm on several images with controlled (known) light configurations, in order to measure the errors of our light detection. The images include varied configurations (see Figure 4): Apple1, Apple2 and Apple3 show a relatively simple geometry under very different lighting schemes (with one or two light sources, plus ambient light). The Guitar and Quilt images show much more complex scenes lit by three and two light sources respectively. The light directions returned by our algorithm show

errors usually below 20 degrees for the more restrictive azimuth angle θ . This error range is discussed in the next section. Even for the zenith angle ϕ , only the second light in the Quilt scene returned a larger error due to the bouncing of that light in the surface on the left. Table 1 shows all the data for the input images shown in Figure 4: for each light source present in the scene, we show the real measured locations of the light sources, the results output by our algorithm and the corresponding absolute error. The light probe used in the first three images is the apple; for the other two, we used the head of the guitar player and the Scottish quilt.



Figure 4: Input images for the error analysis of Table 1. From left to right: Apple1, Apple2 and Apple3, Guitar and Quilt.

		Light 1		Light 2		Light 3	
		ϕ	θ	ϕ	θ	ϕ	θ
Apple1	R	-15.00	40.00	165.00	-40.00	-	-
	A	5.71	35.31	162.25	-64.03	-	-
	E	20.71	4.69	2.75	24.03	-	-
Apple2	R	90.00	-70.00	-	-	-	-
	A	94.54	-65.70	-	-	-	-
	E	4.54	4.3	-	-	-	-
Apple3	R	180.00	0.00	0.00	0.00	-	-
	A	168.50	14.48	0.0	11.31	-	-
	E	12.50	14.48	0.00	11.31	-	-
Guitar	R	180.00	10.00	30.00	-45.00	260.00	45.00
	A	185.71	29.66	25.64	-49.19	272.29	41.48
	E	5.71	19.66	4.36	4.19	12.29	3.16
Quilt	R	10.00	-35.00	120.00	-10.00	-	-
	A	24.70	-51.79	162.25	4.74	-	-
	E	14.70	16.79	42.25	14.74	-	-

Table 1: Real measured light directions (R), value returned by our algorithm (A) and absolute error (E) for the zenith ϕ and azimuth θ angles in the scenes depicted in Figure 4.

We have further tested our algorithm on uncontrolled images, depicting scenes with unknown illuminations and varying degrees of diffuse-directional lighting ratios. Given that we obviously cannot provide error measures in those cases,

we provide visual validation of the results by rendering a synthetic object with the lighting scheme returned by our algorithm. Figure 5, left, shows the original image and an untextured version of the 3D objects to be rendered. The image on the right shows the results of illuminating the 3D objects with the output returned by our algorithm. The chosen light probe was one of the mushrooms. Figure 6 shows additional examples of uncontrolled input images with synthetic objects rendered into them, using the head of the doll and the whole human figure as light probes respectively. Note how our system is robust enough even if the light probe is composed of multiple objects with very different BRDFs (such as the skin, glasses and hair in the doll image). Shadows cast onto the original images are generated in the usual way by defining synthetic planes at approximately the right locations when placing the synthetic objects.



Figure 5: Rendering synthetic objects into the images. Left, top: original input image. Left, bottom: 3D models lit according to the output of our light detection algorithm. Right: final result, with the 3D models textured and inserted in the image.



Figure 6: Additional examples of synthetic objects rendered into images using the results of our algorithm. Left: synthetic teapot. Right: synthetic cone.

5. Discussion and Future Work

We have presented a novel light detection algorithm from single images, which only requires the silhouette of any object in the image as additional user input. Our method yields a result in less than 4 seconds using a 512x512 version of the original image. Although it works in smaller resolutions, higher resolutions have small effect on its accuracy. It may seem that the average error of our method is too high in comparison with previous works in the field, however in opposition to those works we are not limited to detecting just



Figure 7: Spheres rendered with information from the Guitar image in Figure 4. Left: using the image as an environment map. Middle: using the real measured data. Right: using the results of our algorithm. Our algorithm provides a much better solution if the light sources are not present in the original image.

one light and no knowledge of the actual 3D geometry is required.

Additionally, this error (around 15 degrees in average) might be below the human perception sensitivity as suggested by the work of [OCS05] and shown in our rendered images. However the accuracy of human vision in light detection has not been measured yet and it should be quantified by psychophysical experiments in a future research.

We have shown good results both with controlled lighting environments (where the light positions were measured and thus numerical data could be compared), and uncontrolled settings (with free images downloaded from the internet, and rendering synthetic objects with the results of our algorithm). Our algorithm could potentially help photographers mimic a given lighting scheme inspired by any other shot, for which a reduced set of light directions (namely the typical three-light setup made up of key, fill and rim lights) is preferable.

It could be argued that given that humans are not particularly good at detecting light sources, simpler algorithms that approximate light sources could be employed as well. For instance, in the context of rendering synthetic objects into existing images, one of the most popular recent approaches is to build an environment map from the image. Whilst this approach would provide reasonable results in certain cases (as shown in [KRFB06]), it would fail if the main light sources are actually outside the image. One such example would be the Guitar image in Figure 4. If we were to render an object into the image, it would appear unrealistically dark. Figure 7 shows a sphere rendered with the actual measured lights for that scene, compared to the results from rendering with an environment map and using the lights detected by our algorithm.

Several existing applications can benefit from our system, specifically those based on combining pictures from an existing stack to create novel images. This kind of appli-

cations are gaining popularity due to, among other factors, the current existence of huge databases and their accessibility through the internet. Some examples include Photo Clip Art [LHE*07], Interactive Digital Photomontage [ADA*04] or Photo Tourism [SSS06].

We assume global convexity for the chosen de-facto light probes in the images. Whilst this is true for most objects, the algorithm will return wrong values if a concave object is chosen instead. Our algorithm will also fail in the presence of purely reflective or transparent (refractive) objects chosen as light probes, which break our assumption about shading. In these cases, an approach similar to [NN04] may be more suitable, although previous knowledge about the geometry of the objects in the image would be needed. As future work, we would like to address these cases.

6. Acknowledgements

This research was partially funded by a generous gift from Adobe Systems Inc and the Spanish Ministry of Science and Technology (TIN2007-63025).

References

- [ADA*04] AGARWALA A., DONTCHEVA M., AGRAWALA M., DRUCKER S., COLBURN A., CURLESS B., SALESIN D., COHEN M.: Interactive digital photomontage. *ACM Transactions on Graphics* 23, 3 (2004), 294–302.
- [BH85] BROOKS M., HORN B.: Shape and source from shading. In *Proc. Int. Joint Conf. Artificial Intell.* (1985), pp. 932–936.
- [GHH01] GIBSON S., HOWARD T., HUBBOLD R.: Flexible image-based photometric reconstruction using virtual light sources. *Computer Graphics Forum* 19, 3 (2001), C203–C214.
- [HA93] HOUGEN D., AHUJA N.: Estimation of the light source distribution and its use in integrated shape recovery from stereo shading. In *ICCV* (1993), pp. 29–34.
- [Hor86] HORN B.: *Robot Vision*. McGraw-Hill, 1986.
- [KP03] KOENDERINK J. J., PONT S. C.: Irradiation direction from texture. *Journal of the Optical Society of America* 20, 10 (2003), 1875–1882.
- [KRFB06] KHAN E. A., REINHARD E., FLEMING R., BÜLTHOFF H.: Image-based material editing. *ACM Transactions on Graphics (SIGGRAPH 2006)* 25, 3 (2006), 654–663.
- [KvDP04] KOENDERINK J. J., VAN DOORN A. J., PONT S. C.: Light direction from shaded random gaussian surfaces. *Perception* 33, 12 (2004), 1405–1420.
- [LB01] LANGER M. S., BÜLTHOFF H. H.: A prior for global convexity in local shape-from-shading. *Perception* 30 (2001), 403–410.
- [LF06] LAGGER P., FUA P.: Using specularities to recover multiple light sources in the presence of texture. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 587–590.
- [LHE*07] LALONDE J.-F., HOIEM D., EFROS A. A., ROTHER C., WINN J., CRIMINISI A.: Photo clip art. *ACM Transactions on Graphics (SIGGRAPH 2007)* 26, 3 (August 2007). See the project webpage at <http://graphics.cs.cmu.edu/projects/photoclipart>.
- [LR89] LEE C., ROSENFELD A.: Improved methods of estimated shape from shading using the light source coordinate system. In *Shape from shading*, Horn B., Brooks M., (Eds.). MIT Press, 1989, pp. 323–569.
- [MG97] MARSCHNER S. R., GREENBERG D. P.: Inverse lighting for photography. In *Fifth IST/SID Color Imaging Conference* (1997), pp. 262–265.
- [NE01] NILLIUS P., EKLUNDH J.-O.: Automatic estimation of the projected light source direction. In *CVPR* (2001), pp. I:1076–1083.
- [NN04] NISHINO K., NAYAR S. K.: Eyes for relighting. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)* 23, 3 (2004), 704–711.
- [OCDD01] OH B. M., CHEN M., DORSEY J., DURAND F.: Image-based modeling and photo editing. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer Graphics and Interactive Techniques* (2001), 433–442, (Ed.).
- [OCS05] OSTROVSKY Y., CAVANAGH P., SINHA P.: Perceiving illumination inconsistencies in scenes. *Perception* 34 (2005), 1301–1314.
- [Pen82] PENTLAND A.: Finding the illuminant direction. *Journal of the Optical Society of America A* 72, 4 (1982), 448–455.
- [PSG01] POWELL M., SARKAR S., GOLDFOS D.: A simple strategy for calibrating the geometry of light sources. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, 9 (2001), 1022–1027.
- [SSI99] SATO I., SATO Y., IKEUCHI K.: Illumination distribution from brightness in shadows: Adaptive estimation of illumination distribution with unknown reflectance properties in shadow regions. In *ICCV* (2) (1999), pp. 875–882.
- [SSS06] SNAVELY N., SEITZ S. M., SZELISKI R.: Photo tourism: Exploring photo collections in 3d. *ACM Transactions on Graphics* 25, 3 (2006), 835–846.
- [VY94] VEGA E., YANG Y.-H.: Default shape theory: with the application to the computation of the direction of the light source. *Journal of the Optical Society of America A* 60 (1994), 285–299.
- [VZ04] VARMA M., ZISSERMAN A.: Estimating illumination direction from textured images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC* (June 2004), vol. 1, pp. 179–186.
- [WAC07] WANG J., AGRAWALA M., COHEN M.: Soft scissors: An interactive tool for realtime high quality matting. *ACM Transactions on Graphics* 26, 3 (2007), 9.
- [WS02] WANG Y., SAMARAS D.: Estimation of multiple illuminants from a single image of arbitrary known geometry. In *ECCV02* (2002), vol. 3, pp. 272–288.
- [YY91] YANG Y., YUILLE A.: Sources from shading. In *Computer Vision and Pattern Recognition* (1991), pp. 534–539.
- [ZK02] ZHOU W., KAMBHAMETTU C.: Estimation of illuminant direction and intensity of multiple light sources. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV* (London, UK, 2002), Springer-Verlag, pp. 206–220.
- [ZTCS99] ZHANG R., TSAI P., CRYER J., SHAH M.: Shape from shading: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 8 (1999), 690–706.
- [ZY01] ZHANG Y., YANG Y.-H.: Multiple illuminant direction detection with application to image synthesis. *IEEE Trans. Pattern Anal. Mach. Intell.* 23, 8 (2001), 915–920.