

Short Papers

Multiple Light Source Detection

Christos-Savvas Bouganis, *Member, IEEE*, and
Mike Brookes, *Member, IEEE*

Abstract—This paper presents the V2R algorithm, a novel method for multiple light source detection using a Lambertian sphere as a calibration object. The algorithm segments the image of the sphere into regions that are each illuminated by a single virtual light and subtracts the virtual lights of adjacent regions to estimate the light source vectors. The algorithm uses all pixels within a region to form a robust estimate of the corresponding virtual light. The circumstances under which the light source detection problem lacks a unique solution are discussed in detail and the way in which the V2R algorithm resolves the ambiguity is explained. The V2R algorithm includes novel procedures for identifying the critical lines that bound the regions, for estimating the light source vectors, and for identifying opposite light pairs. Experiments are performed on synthetic and real images and the performance of the V2R algorithm is compared to that of a recent algorithm from the literature. The experimental results demonstrate that the proposed algorithm is robust and that it gives substantially improved accuracy.

Index Terms—Computer vision, illuminant detection, Lambertian sphere limitations, image synthesis.

1 INTRODUCTION

THE problem of estimating illuminant directions and intensities from an image arises in a number of areas of computer vision including Augmented Reality and Shape from Shading. Early research on illuminant estimation addressed the case of a single point light source [1], [2], [3]. More recently, researchers have developed techniques for estimating multiple point sources and several approaches to this problem can be found in the literature. One approach exploits the shadows that an object of known shape casts onto a second object of known shape and reflectance [4], while others make use of the reflections in one or more mirror-like spherical test objects [5], [6]. A common approach [7], [8], [9], [10], [11], also followed here, is to use a Lambertian sphere of known size and constant albedo as a calibration object in order to detect multiple illuminants which are assumed to be distant point sources.

This paper describes and evaluates a novel algorithm for detecting multiple light sources using a spherical Lambertian test object and also presents a detailed analysis of the fundamental limitations of such algorithms. The algorithm is straightforward to extend to an arbitrary convex test object of known albedo and camera aspect.

2 PROBLEM FORMULATION AND DEFINITIONS

2.1 Sphere Geometry

This paper is concerned with determining the directions and intensities of the light sources illuminating a Lambertian sphere with unit albedo. We choose the coordinate system so that the sphere is of unit radius with its center at the origin. We assume that the camera position is known and that its center of projection lies on the positive z-axis at $z = c^{-1}$ with the image

plane at $z = 0$. The case $c = 0$ corresponds to orthogonal projection with the camera at infinity. A point p on the sphere with coordinates $[p_x \ p_y \ p_z]^T$ is projected onto the image plane at $[x \ y]^T = (1 - cp_z)^{-1}[p_x \ p_y]^T$.

A plane through the origin with unit normal $\mathbf{u} = [u_x \ u_y \ u_z]^T$ will intersect the sphere in a great circle of unit radius that is projected onto the image plane as a partially visible ellipse having the equation

$$u_z^2(x^2 + y^2 - 1) + (1 - c^2)(u_x x + u_y y)^2 + 2cu_z(u_x x + u_y y) = 0. \quad (1)$$

If $|u_z| > 1 - c^2$, then no part of the great circle will be visible to the camera. When $u_z = 0$, the ellipse reduces to a straight line through the origin with $u_x x + u_y y = 0$.

2.2 Sphere Illumination

When the sphere is illuminated by a point light source at infinity, the image intensity at a point p on its surface is given by

$$e_p = \max(\mathbf{s}\mathbf{n}_p^T \mathbf{u}, 0) = \max(\mathbf{n}_p^T \mathbf{s}, 0) = \frac{1}{2} \mathbf{n}_p^T \mathbf{s} + \frac{1}{2} |\mathbf{n}_p^T \mathbf{s}|, \quad (2)$$

where the unit column vectors \mathbf{u} and $\mathbf{n}_p = [p_x \ p_y \ p_z]^T$ denote, respectively, the direction of the light and the surface normal at p and s denotes the light intensity [12]. The *source vector*, $\mathbf{s} = s\mathbf{u}$, combines the direction and intensity of the light source into a single quantity. A negative value of s corresponds to a light in the direction $-\mathbf{u}$. The points satisfying $\mathbf{n}_p^T \mathbf{u} = 0$ form the boundary between the illuminated and unilluminated portions of the sphere. This *critical line* forms a great circle on the sphere and is projected onto the image as a segment of an ellipse.

With M source vectors, $\mathbf{s}_1, \dots, \mathbf{s}_M$, (2) becomes

$$e_p = \sum_{m=1}^M \max(\mathbf{n}_p^T \mathbf{s}_m, 0) = \mathbf{n}_p^T \sum_{m: \mathbf{n}_p^T \mathbf{s}_m > 0} \mathbf{s}_m. \quad (3)$$

The summation in (3) is performed over all the light sources that illuminate p , namely, those satisfying $\mathbf{n}_p^T \mathbf{s}_m > 0$. We may partition the image into a finite number of regions, R_1, \dots, R_K , such that all the pixels within a region are illuminated by the same set of light sources; these regions are bounded by the critical lines associated with the light sources. If L_k is the set of light sources that illuminate the pixels within R_k , then for all $p \in R_k$,

$$e_p = \mathbf{n}_p^T \sum_{m \in L_k} \mathbf{s}_m = \mathbf{n}_p^T \mathbf{v}_k, \quad (4)$$

where \mathbf{v}_k is the *virtual light* associated with region R_k .

2.3 Opposite Lights

Two lights that are exactly opposite each other will share the same critical line and, so, the number of regions in the sphere image will be the same as if only one of the lights were present. The pair of *opposite lights* may be represented as $\mathbf{s}_1 = s_1 \mathbf{u}$ and $\mathbf{s}_2 = s_2 \mathbf{u}$ where $s_1 s_2 < 0$. From (2), the combined effect on the intensity at p can be expressed as

$$e_p = \frac{1}{2} (s_1 + s_2) \mathbf{n}_p^T \mathbf{u} + \frac{1}{2} |(s_1 - s_2) \mathbf{n}_p^T \mathbf{u}| = \frac{1}{2} d \mathbf{n}_p^T \mathbf{s} + \frac{1}{2} |\mathbf{n}_p^T \mathbf{s}|, \quad (5)$$

where $\mathbf{s} = (s_1 - s_2) \mathbf{u}$ and $d = (s_1 + s_2) / (s_1 - s_2)$. Thus, we may describe the effect of the pair of opposite lights by combining a single source vector \mathbf{s} with a parameter d that lies in the range ± 1 and that characterizes the relative strength of the two lights. The sum of the opposite light intensities is $|\mathbf{s}| = |s_1| + |s_2|$ and the extreme cases of $d = \pm 1$ correspond to single lights at $\mathbf{s}_1 = \mathbf{s}$ and $\mathbf{s}_2 = -\mathbf{s}$, respectively, in which case (2) may be regarded as a special case of (5).

• The authors are with the EEE Department, Imperial College, Exhibition Rd., London SW7 2BT, UK. E-mail: {mike.brookes, ccb98}@imperial.ac.uk.

Manuscript received 22 May 2002; revised 17 May 2003; accepted 18 Sept. 2003. Recommended for acceptance by D. Kriegman.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 116607.

2.4 Fundamental Limitations

It is not always possible to determine all the light sources unambiguously from the sphere image. There are two specific sources of ambiguity: undetectable lights and opposite light pairs.

The critical line from a light source for which $|u_z|^2 > 1 - c^2$ lies on a part of the sphere invisible to the camera; such a light source is said to be *undetectable* [10]. If the z component of an undetectable source vector \mathbf{s} is negative, then the light is hidden behind the sphere and does not affect the image. If the z component is positive, then the light is approximately in line with the camera and illuminates all regions in the image. Following (4), the effect of all such visible undetectable lights may be combined into a single virtual light, \mathbf{v}_0 , the *camera light*.

The second source of ambiguity arises if the source vectors, \mathbf{s}_m , form a linearly dependent set. This can arise with as few as two visible lights together with an undetectable light behind the sphere and is always the case if there are four or more light source directions. A special case of this ambiguity was identified in [10], but the treatment given here is more general. We first form a matrix $\mathbf{S} = [\mathbf{s}_1 \ \mathbf{s}_2 \ \dots \ \mathbf{s}_M]$ from the distinct source vectors. Each column of \mathbf{S} represents either the direction and intensity of a single light or, as in (5), the direction and intensity sum of a pair of opposite lights.

Following (5), we may express the intensity at p due to all M source vectors as

$$e_p = \frac{1}{2} \mathbf{n}_p^T \mathbf{S} \mathbf{d} + \frac{1}{2} \sum_{m=1}^M |\mathbf{n}_p^T \mathbf{s}_m|. \quad (6)$$

In (6), each element d_m of the vector \mathbf{d} multiplies the corresponding column of \mathbf{S} which contains the source vector \mathbf{s}_m . If \mathbf{s}_m corresponds to a single light, then the corresponding d_m will equal either $+1$ or -1 . If, on the other hand, \mathbf{s}_m corresponds to an opposite pair of lights, then d_m will lie in the range ± 1 according to the ratio of the light intensities as described in Section 2.3 above. If the nullity of \mathbf{S} is J , we can find an $M \times J$ matrix, \mathbf{H} , whose columns form a basis for the null space of \mathbf{S} and that satisfies $\mathbf{S}\mathbf{H} = \mathbf{0}$. Since \mathbf{S} has only three rows, we will always have $J \geq M - 3$.

Given an arbitrary vector \mathbf{g} of dimension J , we can define an alternative set of coefficients as $\mathbf{d}' = \mathbf{d} + \mathbf{H}\mathbf{g}$ and we see that

$$\mathbf{n}_p^T \mathbf{S} \mathbf{d}' = \mathbf{n}_p^T \mathbf{S} (\mathbf{d} + \mathbf{H}\mathbf{g}) = \mathbf{n}_p^T \mathbf{S} \mathbf{d} + \mathbf{n}_p^T \mathbf{S} \mathbf{H} \mathbf{g} = \mathbf{n}_p^T \mathbf{S} \mathbf{d}. \quad (7)$$

Thus, the intensity at p given by (6) is unchanged by the replacement of \mathbf{d} by \mathbf{d}' . To represent a feasible set of coefficients, each component of \mathbf{d}' must however lie in the range ± 1 . For each of the M rows, \mathbf{h}_m^T of \mathbf{H} , this imposes two linear constraints on the vector \mathbf{g} of the form

$$-1 \leq d_m + \mathbf{h}_m^T \mathbf{g} \leq +1. \quad (8)$$

These $2M$ constraints restrict \mathbf{g} to lie within a J -dimensional polytope that includes the origin.

Each point \mathbf{g} within the polytope corresponds to a different configuration of lights that could have generated the observed image. We resolve this ambiguity by selecting the configuration that contains the fewest visible lights. This will always correspond to one of the polytope vertices since moving away from a vertex always increases the number of opposite light pairs. Also, it cannot include more than three pairs of opposite lights since, at a vertex, the directions of all remaining opposite pairs must be linearly independent. In some situations, there may be more than one feasible configuration having the fewest number of visible lights and there is then no unique best choice. In particular, if a subset of the \mathbf{s}_m sum to zero, then the corresponding d_m may be negated without affecting the illumination or the number of lights required; under these circumstances, an ambiguity will arise even if all opposite light pairs are excluded as in [10].

3 RELATED WORK

The detection of multiple light sources was initially addressed by Yang and Yuille in [7]. Their algorithm is based on the observation that, along a segment of the boundary that lies within a single region, the image intensity varies sinusoidally with the angle of the boundary normal for a smooth convex Lambertian object. Hougen and Ahuja [8] begin by assuming a large number of predefined light directions and then perform a least squares or nonnegative least squares minimization to find which combination of these lights can reconstruct the intensities on a grid of image points. A similar approach can be found in Marschner and Greenberg's work [13]. Hougen and Ahuja's method was evaluated by Yang [9] who reported that only the nonnegative least squares technique produced usable results. He reports that, although his algorithm gives good results for image synthesis, the detected directions of the light sources do not necessarily converge to the correct values. This result is unsurprising in view of the fundamental limitations discussed in Section 2.4 above.

More recently, Zhang and Yang [10], [11] have proposed a scheme for detecting multiple light sources based on identifying the critical lines on a Lambertian sphere. They observed that, within a single region, the intensity function (4) along a great circle on the sphere is a cosine wave whose phase and amplitude depend on the virtual light \mathbf{v}_k . They detect the critical lines from discontinuities in these cosine waves and find each light direction as the normal to the plane containing a critical line. The light intensities are found by minimizing the reconstruction error

$$F(\mathbf{u}_0, s_0, \dots, s_M) = \sum_{p=1}^P \left(e_p - s_0 \mathbf{n}_p^T \mathbf{u}_0 - \sum_{m=1}^M \max(s_m \mathbf{n}_p^T \mathbf{u}_m, 0) \right)^2, \quad (9)$$

where P is the total number of image pixels. The algorithm cannot detect pairs of opposite lights since it assumes that each detectable light generates a distinct critical line. We have evaluated the algorithm of [11] and found that, although it works well in some cases, the minimization of (9) can converge to an incorrect local minimum. The final solution reached depends on the choice of initial signs for the $\{s_m\}$ and the only way to guarantee finding the correct minimum is to try all 2^M possible combinations.

Recently, Ramamoorthi and Hanrahan [14] and Basri and Jacobs [15] have shown that a Lambertian surface attenuates the high frequency components of the illumination and that the first nine spherical harmonics capture most of the appearance of the image. Figs. 1a and 1b show the absolute error when nine spherical harmonics are used to approximate the intensity of a Lambertian sphere illuminated by one and three lights, respectively. It can be seen that the errors are concentrated at the gradient discontinuities that exist along the critical lines. Although the mean square error reduces rapidly as the number of harmonics is increased, the peak error along the critical lines falls more slowly because of Gibb's phenomenon. Fig. 1c shows the absolute error for three lights when using 225 coefficients with additive white noise at 40dB SNR. Despite the attenuation of the high order harmonics by the Lambertian surface, the critical lines are still apparent above the noise floor and, by integrating over many pixels, the techniques described in this paper make it possible to identify the critical lines and the source vectors reliably.

4 VIRTUAL-TO-REAL ALGORITHM (V2R)

4.1 Overview

In this section, we describe a novel algorithm, the Virtual-to-Real algorithm (V2R), for finding the source vectors. The basis of the algorithm is that, if two regions R_1 and R_2 are adjacent on the image, with \mathbf{v}_1 and \mathbf{v}_2 the corresponding virtual lights, then we see from (4) and (5) that $\mathbf{v}_1 - \mathbf{v}_2 = \mathbf{s}_m$, where $|s_m|$ is either the intensity of the single light or the sum of the intensities of the opposite pair of lights that correspond to the critical line. Wang and Samaras

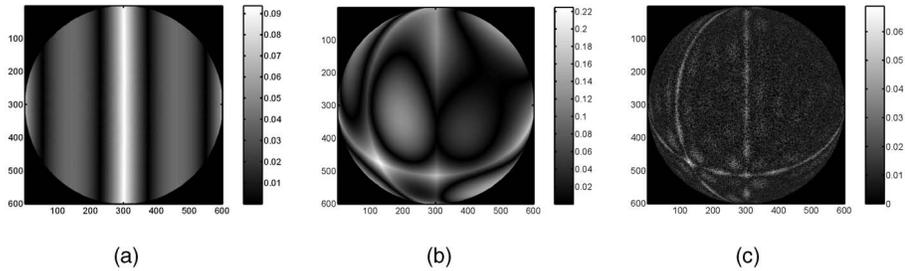


Fig. 1. The absolute error when the intensity of a Lambertian sphere is approximated by a finite number of spherical harmonics using (a) one light and nine harmonics, (b) three lights and nine harmonics, and (c) three lights and 225 harmonics with 40dB SNR.

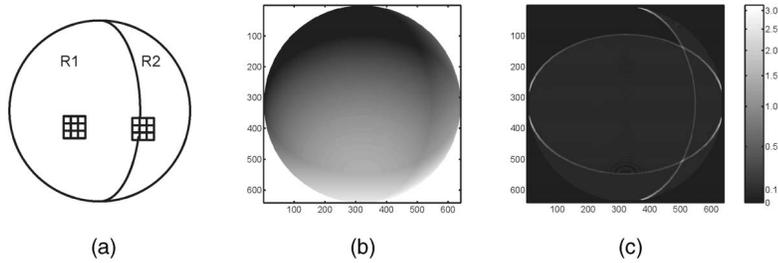


Fig. 2. (a) Shows the windows W_p used in (10) when identifying critical points. The window that straddles a critical line will give a high value of f_p in (11) and, so, the corresponding point p will be identified as a critical point. (b) Shows a sphere illuminated with three lights at $[0 \ 1 \ -1]$, $[0 \ 1 \ 1]$, and $[1 \ 0 \ -1]$. (c) Shows, for each image pixel, the value of f_p from (11) when using a window size of $w = 9$. Critical points correspond to high values of f_p .

[16], [17] have independently proposed an algorithm that is also based on this principle, but they use different approaches for region segmentation and real light determination and do not perform ambiguity resolution.

The V2R algorithm begins by determining the critical lines in the image, dividing the image into regions, and then calculating the virtual light \mathbf{v}_k for each region. By subtracting the virtual lights from adjacent regions, the real lights are found to within a sign ambiguity that is resolved by an enumeration search. A final steepest descent stage is then used to refine the estimated source vectors \mathbf{s}_m . In our work, all images are quantized to 256 intensity levels.

4.2 Detailed Description of Algorithm

4.2.1 Region Segmentation

The first stage of the V2R algorithm is to identify the elliptical critical lines on the image and to use them to segment the sphere into regions. The procedure is illustrated by the two square windows shown in Fig. 2a. One window lies entirely within region R_1 and the intensities of the pixels within it will satisfy (4) with a single virtual light \mathbf{v}_1 . However, the other window straddles a critical line and will not satisfy (4) for any value of \mathbf{v} since it contains pixels from two different regions. For each pixel p in the image, we construct a window W_p of size $w \times w$ centered on p . We can estimate the corresponding virtual light, \mathbf{v} , as the least squares solution, $\hat{\mathbf{v}}$, to the overdetermined set of equations

$$\mathbf{N}_{W_p} \mathbf{v} = \mathbf{e}_{W_p} \Rightarrow \hat{\mathbf{v}} = (\mathbf{N}_{W_p}^T \mathbf{N}_{W_p})^{-1} \mathbf{N}_{W_p}^T \mathbf{e}_{W_p}, \quad (10)$$

where $\mathbf{N}_{W_p} = [\mathbf{n}_\alpha \ \mathbf{n}_\beta \ \dots \ \mathbf{n}_\omega]^T$ and $\mathbf{e}_{W_p} = [e_\alpha \ e_\beta \ \dots \ e_\omega]^T$ for $\{\alpha, \beta, \dots, \omega\} \in W_p$. The components of \mathbf{e}_{W_p} are the image intensities at each pixel of the window W_p , while the rows of \mathbf{N}_{W_p} contain the corresponding surface normals. The matrix \mathbf{N}_{W_p} is independent of the image intensities and it is always full rank if $w > 1$. The resultant mean square error f_p is [18]

$$f_p = \frac{1}{w^2} (\mathbf{N} \hat{\mathbf{v}} - \mathbf{e})^T (\mathbf{N} \hat{\mathbf{v}} - \mathbf{e}) = \frac{\det([\mathbf{e} \ \mathbf{N}]^T [\mathbf{e} \ \mathbf{N}])}{w^2 \det(\mathbf{N}^T \mathbf{N})}, \quad (11)$$

where the subscripts W_p have been omitted for clarity. A pixel p is identified as a critical point if f_p exceeds a threshold T whose value is discussed in Section 4.3. Fig. 2b shows an image of a sphere

illuminated by three lights and Fig. 2c shows how f_p varies over the sphere's surface. From the figure, it is clear that, when the window straddles a critical line, the value of f_p rises to many times its value within the region. We apply dilation and erosion operators to remove isolated pixels and to reduce the thickness of the detected critical lines. We group the detected critical pixels into elliptical projections of great circles using the Hough Transform [19] for constrained ellipse detection. The search space is two-dimensional since the three free parameters $u_x, u_y,$ and u_z in (1) are reduced to two by the constraint $|\mathbf{u}| = 1$.

4.2.2 Real Light Identification

For each region R_k bounded by the critical lines identified in the previous stage, a virtual light is estimated as the least squares solution to an overdetermined set of equations similar to (10), but modified to include a bias term b_k . This bias term accounts for any isotropic background illumination or any DC offset arising from the camera or quantization process and we have found that its inclusion improves the estimation of the virtual light \mathbf{v}_k . For each region k , the algorithm therefore finds the least squares solution to the augmented equations

$$\mathbf{N}_{R_k} \mathbf{v}_k + \mathbf{1} b_k = \mathbf{e}_{R_k}, \quad (12)$$

where $\mathbf{1}$ denotes a column vector containing 1 in each position and \mathbf{N}_{R_k} and \mathbf{e}_{R_k} are defined as for (10) but now include all pixels within the region R_k . Note that a bias term should not be used for critical point determination in (10) as it reduces the difference in mean square error f_p between critical and noncritical pixels. A global bias B is now determined as a weighted average of the region bias estimates b_k as

$$B = \frac{\sum_{k=1}^K |R_k|^2 b_k}{\sum_{k=1}^K |R_k|^2},$$

where $|R_k|$ is the number of pixels in region k . The choice of weights penalizes the bias estimates from small regions since these are based on fewer data pixels most of which lie close to a critical line. A final estimate for the virtual light in each region is now found as the least squares solution to

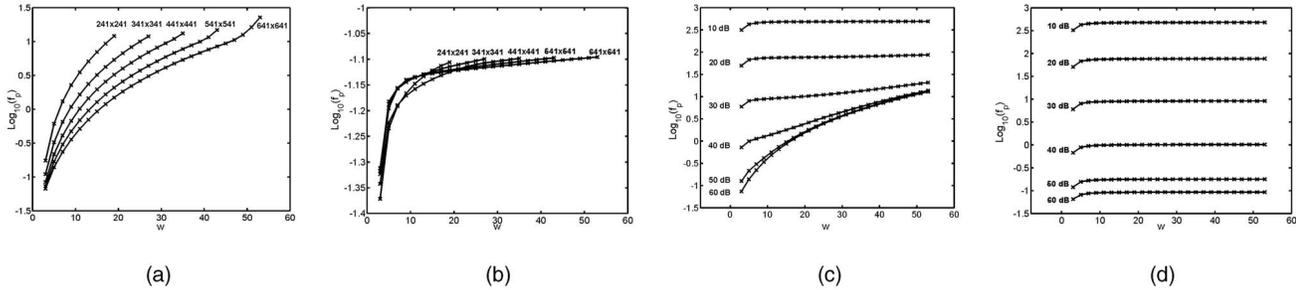


Fig. 3. (a) and (b) Plot $\log_{10}(f_p)$ versus window size w for image resolutions between 241×241 and 641×641 pixels for (a) a critical pixel and (b) a noncritical pixel. (c) and (d) Plot the same quantities for SNRs between 10dB and 60dB using an image resolution of 641×641 pixels for (c) a critical pixel and (d) a noncritical pixel.

$$\mathbf{N}_{R_k} \mathbf{v}_k = \mathbf{e}_{R_k} - B. \quad (13)$$

It can be shown that, if the region boundaries are correct, the mean square error norm in the estimate of \mathbf{v}_k is given by

$$c \text{ trace} \left(\left(\mathbf{N}_{R_k}^T \mathbf{N}_{R_k} \right)^{-1} \right) \approx 6.25cD^2 |R_k|^{-2}, \quad (14)$$

where the second expression was determined by fitting an empirical formula to values for rectangular regions of the sphere image. In (14), c is the mean square error of pixel intensity values, D is the diameter of the sphere image in pixels, and $|R_k|$ the number of pixels in region R_k .

Having found the virtual lights \mathbf{v}_k for each region, the source vector \mathbf{s}_m associated with each critical line can be found to within a sign ambiguity by taking the difference between pairs of adjacent regions that are separated by the line. There will, in general, be several pairs of adjacent regions that are separated by a particular critical line and, so, these will give rise to several estimates of the real light. We form a weighted average of these estimates using $(|R_j||R_k|)^2$ as the weight for the estimate from adjacent regions R_j and R_k because we find this gives slightly better results than the $(|R_j|^{-2} + |R_k|^{-2})^{-1}$ implied by (14) under the assumption that the individual errors are independent.

The final step of this stage is to resolve opposite light pairs and determine the intensity of the camera light v_0 whose direction $\mathbf{u}_0 = [0 \ 0 \ 1]^T$. The purpose of this virtual light is to account for any real lights whose critical lines are not visible on the image as discussed in Section 2. We wish to determine the M direction indicators d_m in the range ± 1 and the camera virtual light intensity $v_0 > 0$ that minimize the reconstruction error over the entire image given by

$$F(v_0, d_1, d_2, \dots, d_M) = \sum_{p=1}^P \left(e_p - B - v_0 \mathbf{n}_p^T \mathbf{u}_0 - \frac{1}{2} \sum_{m=1}^M |\mathbf{n}_p^T \mathbf{s}_m| - \frac{1}{2} \sum_{m=1}^M d_m \mathbf{n}_p^T \mathbf{s}_m \right)^2, \quad (15)$$

where P is the total number of image pixels. The problem of minimizing (15) is an example of quadratic programming. In order to resolve the ambiguities defined in Section 2.4, we find the minimum by enumerating all possible constraint combinations. From Section 2.4, we need only consider combinations in which at most three of the $\{d_m\}$ are unconstrained and the remainder are set to ± 1 . The intensity v_0 can also be unconstrained or else set to 0 and it can be shown that this gives a total of $2^M(M^3 + 3M^2 + 20M + 48)/24$ combinations. For each feasible constraint combination, the algorithm determines the values of the unconstrained variables that minimize (15) and rejects the solution if any variables lie outside their permitted bounds. Of the accepted combinations, the one that minimizes F is selected. To eliminate unnecessary opposite light pairs (see Section 2.4), a penalty may be added to F that is proportional to the number of visible lights. The final stage of the algorithm is to refine the estimates for the lights \mathbf{s}_m . Using a steepest descent algorithm, we minimize F in (15) with

respect to both the \mathbf{s}_m and the camera light $\mathbf{v}_0 = v_0 \mathbf{u}_0$ while keeping all the $\{d_m\}$ fixed. This refinement stage normally results in a significant reduction in F .

4.3 Parameter Selection

4.3.1 Window Size and Critical Point Threshold

The identification of critical pixels using (11) works by fitting a virtual light to the data in a small $w \times w$ window centered on each image pixel. The use of a large window makes critical lines easier to detect by increasing the difference in f_p between critical and noncritical pixels, but may mean that the algorithm is unable to resolve two critical lines that are close together. Figs. 3a and 3b show how the value of $\log_{10}(f_p)$ from (11) varies with the window size, w , for a critical and a noncritical pixel for five different image sizes. Figs. 3c and 3d show how $\log_{10}(f_p)$ varies with the window size, w , for a critical and a noncritical pixel for different signal-to-noise ratios. This demonstrates that, by selecting a sufficient large window size, the critical points can be reliably detected as those that exceed a threshold T . Fig. 4 shows, as a function of T , the percentage of critical pixels detected within five pixels together with the overall percentage of pixels detected for a 641×641 image with three lights. We have found that a window size of $w = 9$ gives good results for a 641×641 pixel image having an SNR ≥ 40 dB and that w may be reduced for lower image resolutions but must be increased for worse SNRs. From Fig. 4, it can be seen that the choice of T does not have to be precise, in this case, any value in the range 0.1 to 0.4 will correctly identify the critical points. In practice, we select T adaptively such that 5 percent of the image pixels are designated as critical.

4.3.2 Critical Line Limit

When using the Hough Transform to group the detected critical pixels into critical lines, we carry on grouping critical pixels into critical lines until the number of remaining critical pixels falls below a limit. If this limit is set too high, we may detect too few critical lines and consequently fail to detect the corresponding light sources. On the other hand, if the limit is set too low, we may detect lines that do not actually exist. The latter is a less serious problem as the subsequent processing will either assign a very low intensity to the corresponding source or else two lights will converge to the same

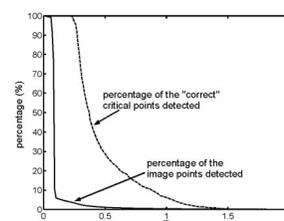


Fig. 4. The figure shows how the overall percentage of pixels identified as critical points and the percentage of the actual critical pixels identified varies as the threshold T is varied between 0 and 2.

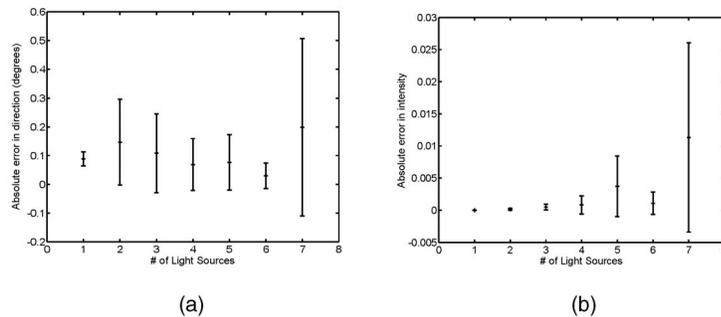


Fig. 5. Performance of the V2R algorithm for up to seven source vectors. (a) Shows the mean error in the light direction estimation, while (b) shows the mean error in the intensity estimation. Each point on the graph is the average of 10 trials: The bars indicate the standard deviation and show that the performance varies substantially for different light configurations.

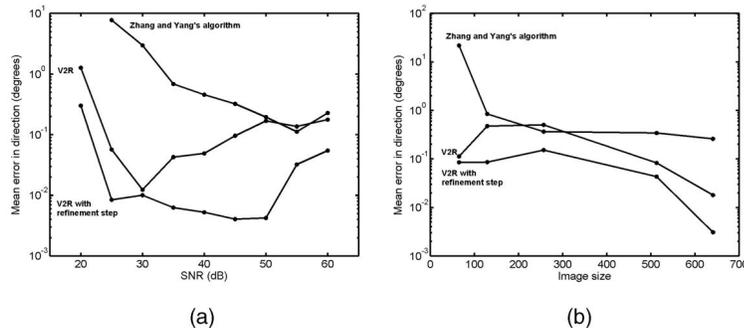


Fig. 6. The graphs plot the mean error in the direction of the source vectors versus (a) the signal-to-noise ratio for an image resolution of 641×641 and (b) the image resolution for a fixed SNR of 60dB.

TABLE 1
Results for the Case of a Sphere with Seven Lights

	Zhang and Yang's alg.	Virtual-to-Real alg.
Maximum absolute difference of the images	3.82×10^{-2}	8.15×10^{-4}
Mean square error of the images	3.05×10^{-4}	2.71×10^{-7}
Mean error of the illum. in the direction (degrees)	1.64×10^{-1}	7.60×10^{-3}
Mean error of the illuminants in the intensity	1.49×10^{-2}	2.96×10^{-4}

The comparison is performed against the original unquantized image.

direction. We have found that a threshold of about 150 pixels gives good results for an image resolution of 641×641 pixels.

5 PERFORMANCE EVALUATION

The proposed algorithm has been evaluated with synthetic images in which a sphere of known position and diameter is illuminated using between one and seven light sources. The source directions are chosen randomly and their intensities are chosen randomly in the range 0.1 to 1. The sphere diameter is 641 pixels and pixel intensities are scaled to a peak image intensity of 255 and then quantized to integer values. Fig. 5 shows the corresponding mean error of the direction and the intensity of the estimated source vectors for 10 trials for each configuration. The bars indicate the standard deviation. These graphs indicate that the errors remain very small as the number of light sources is increased, but that they vary substantially for different light configurations.

We have also implemented Zhang and Yang's algorithm from [11] and compared its performance with the V2R algorithm. The parameters for Zhang and Yang's algorithm were calculated as proposed in [10], [11] and the recommended refinement for the detection of the critical points was applied in all cases. In these tests, we used a sphere illuminated with three sources as in Fig. 2. Fig. 6 shows how the mean error in the direction estimation of the source vectors varies with the signal-to-noise ratio and the image

resolution. In Zhang and Yang's algorithm, the critical lines can only be determined to an accuracy defined by the resolution of the Hough transform. This imposes a lower bound on the direction estimation error which, as can be seen in Fig. 6b, is approached for high image resolutions. There is no such bound for the V2R algorithm and, for high image resolutions, the error is proportional to D^{-2} , as indicated by (14). Although all the algorithms achieve good results with this example, it is clear that the V2R algorithm with its refinement step is the best of the algorithms by a substantial margin.

We attempted to compare the V2R algorithm with Zhang and Yang's algorithm for a test image with seven light sources using an image resolution of 641×641 pixels. However, Zhang and Yang's algorithm failed to detect the correct lights since the minimization of the reconstruction error was converging to a wrong solution. To overcome this, we performed an exhaustive search and found the minimum of the reconstruction error for each of the 2^7 possible sign combinations of the $\{s_m\}$. With this modification, Zhang and Yang's algorithm correctly found the light vectors, but, as is shown in Table 1, the V2R algorithm gives better performance.

We have performed experiments with real data using a ping-pong ball as a calibration object and a desk lamp to approximate point light sources. No camera calibration was performed and no corrections were made for the nonuniform albedo of the ping-pong ball. We found that, with fewer than four lights, the algorithm always

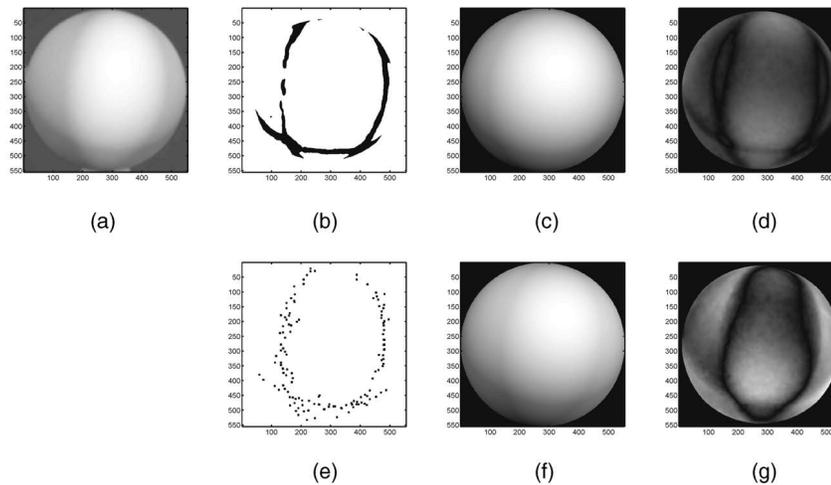


Fig. 7. Real sphere image (a) original, (b) detected critical points using V2R, (c) generated image using the detected light sources from V2R, (d) absolute error image between (a) and (c). (e) Shows the detected critical points using Zhang and Yang's algorithm, (f) is the corresponding reconstructed image, and (g) is the absolute error image between (a) and (f). Images (d) and (g) are scaled to a maximum value of 55.

found the correct light positions with typical errors of a few degrees and a worst-case error of 10 degrees. With four lights, the algorithm always found the correct critical lines, but sometimes converged to the wrong light positions due to poor signal-to-noise ratio. Fig. 7 shows an example of the original image, the detected critical pixels, and reconstructed image for a case of three lights for the V2R and Zhang and Yang's algorithms. Comparing Figs. 7b and 7e, we see that the detected critical points are denser and more compact for the V2R algorithm than for Zhang and Yang's algorithm. The resultant reconstruction errors are shown in Figs. 7d and 7g and are significantly lower for the V2R algorithm than for Zhang and Yang's algorithm with mean square errors of 174 and 472, respectively.

6 CONCLUSIONS

In this paper, we have addressed the problem of identifying multiple light source vectors using a Lambertian sphere as a calibration object. We have introduced a novel procedure, the V2R algorithm, for solving this problem and have demonstrated, using synthesized and real images, that it performs substantially better than the algorithm described in [10] and [11] even when the image size is small or the signal-to-noise ratio is poor. This robustness arises in part because, in contrast to previous approaches, the V2R algorithm uses all the pixels in the image to estimate the source vectors rather than a one-dimensional subset of the pixels. We have identified the circumstances under which the problem lacks a unique solution, have characterized the set of feasible solutions that exist in this case, and have shown how this inherent ambiguity can be resolved. The V2R algorithm will give correct results for these cases and is able to identify pairs of opposite lights where these are required. The V2R algorithm will identify any constant bias in the image intensities and includes a final refinement step that results in a significant improvement in the accuracy of the detected lights.

The V2R algorithm can be used as an initial step in Shape from Shading applications, to compensate for illumination variability, and to enable the synthesis of correctly shaded objects for Augmented Reality applications. Future work will aim to extend the algorithm to include the identification of close and extended light sources.

ACKNOWLEDGMENTS

The authors would like to thank the associate editor-in-chief and the anonymous referees of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* for their helpful comments which have substantially improved this paper.

REFERENCES

- [1] A.P. Pentland, "Finding the Illuminant Direction," *J. Optical Soc. of Am.*, vol. 72, pp. 448-455, 1982.
- [2] W. Chojnacki, M.J. Brooks, and D. Gibbins, "Revisiting Pentland's Estimator of Light-Source Direction," *J. Optical Soc. of Am. A—Optics Image Science and Vision*, vol. 11, no. 1, pp. 118-124, Jan. 1994.
- [3] Q. Zheng and R. Chellappa, "Estimation of Illuminant Direction, Albedo, and Shape from Shading," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 680-702, July 1991.
- [4] I. Sato, Y. Sato, and K. Ikeuchi, "Illumination Distribution from Shadows," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 306-312, 1999.
- [5] P.E. Debevec, "Rendering Synthetic Objects into Real Scenes: Bridge Traditional and Image-Based Graphics with Global Illumination and High Dynamic Range Photography," *Proc. SIGGRAPH '98 Conf.*, pp. 189-198, July 1998.
- [6] M.W. Powell, S. Sarkar, and D. Goldgof, "A Simple Strategy for Calibrating the Geometry of Light Sources," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 1022-1027, Sept. 2001.
- [7] Y. Yang and A. Yuille, "Sources From Shading," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, pp. 534-539, 1991.
- [8] D.H. Hougen and N. Ahuja, "Estimation of the Light Source Distribution and Its Use in Integrated Shape Recovery from Stereo and Shading," *Proc. IEEE Fourth Int'l Conf. Computer Vision*, pp. 148-155, May 1993.
- [9] Y.-H. Yang, "An Experimental Study of Light Source Determination for Computer Graphics," *Vision Interface*, pp. 271-278, 1998.
- [10] Y. Zhang and Y.-H. Yang, "Illuminant Direction Determination for Multiple Light Sources," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 269-276, 2000.
- [11] Y. Zhang and Y.-H. Yang, "Multiple Illuminant Direction Detection with Application to Image Synthesis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 915-920, Aug. 2001.
- [12] B.K.P. Horn, *Robot Vision*. The MIT Electrical Eng. and Computer Science Series, The MIT Press, 1986.
- [13] S.R. Marschner and D.P. Greenberg, "Inverse Lighting for Photography," *Proc. Fifth Color Imaging Conf.*, 1997.
- [14] R. Ramamoorthi and P. Hanrahan, "On the Relationship between Radiance and Irradiance: Determining the Illumination from Images of a Convex Lambertian Object," *J. Optical Soc. of Am.*, vol. 18, no. 10, pp. 2448-2459, 2001.
- [15] R. Basri and D.W. Jacobs, "Lambertian Reflectance and Linear Subspaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218-233, Feb. 2003.
- [16] Y. Wang and D. Samaras, "Estimation of Multiple Illuminants from a Single Image of Arbitrary Known Geometry," *Proc. European Conf. Computer Vision*, vol. 3, pp. 272-288, 2002.
- [17] Y. Wang and D. Samaras, "Estimation of Multiple Directional Light Sources for Synthesis of Mixed Reality Images," *Proc. Pacific Graphics Conf.*, pp. 38-47, 2002.
- [18] F.R. Gantmacher, *The Theory of Matrices*. A.M.S. Chelsea Publishing, 1959.
- [19] P.V.C. Hough, *Methods and Means for Recognising Complex Patterns*, US Patent 3 069 654, Dec. 1962.