Towards Accurate Recovery of Shape from Shading under Diffuse Lighting

A. James Stewart

Department of Computer Science
University of Toronto
Toronto, M5S 1A4, Canada
jstewart@dgp.utoronto.ca

Michael S. Langer

Computer Science Division
NEC Research Institute
Princeton, NJ 08540
langer@research.nj.nec.com

Abstract

We present a surface radiance model for diffuse lighting that incorporates shadows, interreflections, and surface orientation. We show that, for smooth surfaces, the model is an excellent approximation of the radiosity equation. We present a new data structure and algorithm that uses this model to compute shape–from–shading under diffuse lighting. The algorithm was tested on both synthetic and real images, and performs more accurately than the only previous algorithm for this problem. Various causes of error are discussed, including approximation errors in image modelling, poor local constraints at the image boundary, and ill-conditioning of the problem itself.

1 Introduction

The classical formulation of the shape–from–shading problem, both in human psychophysics [14] and in computer vision [3] has been to assume that surface radiance is entirely determined by the surface orientation relative to a point light source at infinity. Shadows and interreflections are usually ignored. When the light source is diffuse, however, this model is not applicable, and a new approach is needed. This is the situation that we address.

We begin by considering two examples to illustrate the difference between point–source–at–infinity and diffuse lighting. The first is a scene consisting of a convex object with a matte surface resting on a ground plane, and illuminated from above by a uniform hemispheric source (see Figure 1). It is argued in [4] that, because of self–shadowing, the radiance of the object is determined by the surface normal. While this is true for the object, it is *untrue* for the ground plane, whose radiance varies continuously (because of cast shadows) without any change in surface normal. This ground plane shading effect is commonly used by cartoonists to depict contact. Children as young as three years have been shown to be sensitive to this cue [15].

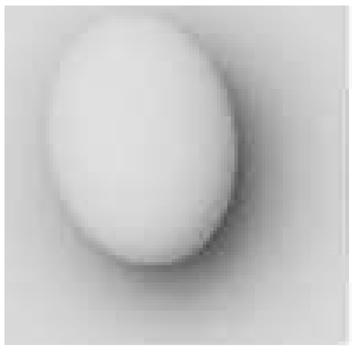

Figure 1: A computer graphics rendered image of an ellipsoid resting on the ground and illuminated by a uniform diffuse source.

A second example is a spherical concavity excavated from a ground plane. A surprising result is that when shadowing, surface orientation, and interreflection effects are all modelled, surface radiance is *constant* within the concavity [9]. In particular, a spherical concavity under diffuse lighting appears the same as a planar surface under point-source-at-infinity lighting. The two examples illustrate that, under diffuse lighting, surface normal variations are neither necessary nor sufficient for radiance variations.

Diffuse lighting has received little attention in vision research. This is unfortunate since diffuse lighting conditions are so common. For example, an overcast or blue sky, the sky at dawn or dusk, or the walls and ceiling of a room all act as diffuse light sources. A key difficulty in addressing the shape–from–shading–

under–diffuse–lighting problem is that the combined effects of surface orientation, shadows, and interreflections are quite complex in general. Recently, it was shown by Langer and Zucker [7, 8] that shape could be computed from shading under diffuse lighting by assuming that surface radiance depended primarily on the solid angle of the source that is visible from the surface. In this paper, we generalize that approach by using a more accurate model of interreflections and surface orientation effects. As we will see, this allows us to more accurately recover shape–from–shading.

2 Global Illumination Model

We address the following situation. A matte surface with albedo, ρ, is illuminated by a uniform hemispheric source of radiance, R_{src}. The surface radiance, $R(\mathbf{x})$, is then determined by the radiosity equation, which accounts for shadows, surface orientation, and interreflection effects. It is a standard model in radiation heat transfer [12] and computer graphics [1], and is written

$$\begin{aligned} R(\mathbf{x}) \quad &= \quad \frac{\rho}{\pi} \int_{\mathcal{V}(\mathbf{x})} R_{src} \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} d\Omega \\ &+ \quad \frac{\rho}{\pi} \int_{\mathcal{H}(\mathbf{x}) \setminus \mathcal{V}(\mathbf{x})} R(\Pi(\mathbf{x}, \mathbf{u})) \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} d\Omega \qquad (1) \end{aligned}$$

where: $\mathbf{x}$ is a surface point; $\mathbf{N}(\mathbf{x})$ is the surface normal; $\mathcal{H}(\mathbf{x}) = \{\mathbf{u} : \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} > 0\}$ is the hemisphere of outgoing unit vectors; $\mathcal{V}(\mathbf{x})$ is the set of unit directions in which the diffuse source is visible from $\mathbf{x}$; $d\Omega$ is an infinitesimal solid angle; $\Pi(\mathbf{x}, \mathbf{u})$ is the surface point visible from $\mathbf{x}$ in direction $\mathbf{u}$ (Π denotes "projection").

Shape–from–shading could be computed exactly by inverting the radiosity equation. An algorithm for doing so using three images taken under point–source–at–infinity lighting and without shadows was presented in [10]. We consider a different situation in which only a single image is available and the light source is diffuse. For this situation, we seek an approximation to the radiosity equation which allows us to recover an accurate approximation to the surface shape. It is conceivable that a *post hoc* iterative method such as in [10] could provide an exact solution, although this is beyond the scope of the paper.

2.1 Interreflections

Recall the example of the convex object on a ground plane. Observe that the bottom of the object tends to be dark because of self–shadowing and also because it is illuminated by darker points on the ground plane. Similarly, points on the top of the object tend to be bright both because little shadowing occurs *and* because the interreflections that do occur are from brighter points on the ground plane. A similar observation holds for a smooth depth map: from a hilltop, one sees mostly other hilltops, while within a valley, one mostly sees the valley.

It follows from these observations that, *under diffuse lighting and constant albedo, points on a surface tend to be illuminated by other points having similar radiance.* This idea is formalized as follows. Given Equation (1), we approximate the right hand side by replacing the incoming radiance, $R(\Pi(\mathbf{x}, \mathbf{u}))$, from other surfaces by the outgoing radiance, $R(\mathbf{x})$, from $\mathbf{x}$ itself. An algebraic manipulation immediately yields an **interreflection model:**

$$R_1(\mathbf{x}) \;\equiv\; \frac{\rho R_{src} \, \frac{1}{\pi} \int_{\mathcal{V}(\mathbf{x})} \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} \; d\Omega}{1 \;-\; \rho \, (1 - \frac{1}{\pi} \int_{\mathcal{V}(\mathbf{x})} \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} \; d\Omega \,)} \,. \quad (2)$$

Note that interreflections are modelled as a non-linear transformation on the direct illumination component. This model is new and will be the basis of a new, accurate shape–from–shading algorithm.

Observe that the interreflection model, R_1, is exact for the case of the spherical concavity (recall Section 1), since surface radiance is constant within the concavity. That is, each surface point sees other surface points of identical radiance. The model is also exact for a matte surface of arbitrary geometry in the limit of $\rho \to 0$ since the interreflection component vanishes, as well as in the limit of $\rho \to 1$ since $R(\mathbf{x}) \to R_{src}$. For intermediate albedos, such as $\rho = 0.5$, we address how well R_1 approximates the radiosity equation by rendering surfaces using computer graphics techniques.[1] Figure 2 shows a rendered image of a smooth depth map with $\rho = 0.5$ along with a scatter plot of $R(\mathbf{x})$ versus $R_1(\mathbf{x})$. Note the high correlation between the two. Similar plots were observed for the other smooth depth maps tested.

To appreciate the accuracy of the R_1 model, consider an alternative model, in which surface normal effects are ignored. Such a model was the basis of a previous algorithm for computing shape–from–shading under diffuse lighting [7, 8] and may be obtained by replacing the $\mathbf{N}(\mathbf{x}) \cdot \mathbf{u}$ expression in Equation (2) by its average value, 0.5, over the hemisphere, yielding

$$R_2(\mathbf{x}) \;\equiv\; \frac{\rho R_{src} \, \frac{1}{2\pi} \int_{\mathcal{V}(\mathbf{x})} d\Omega}{1 \;-\; \rho \, (1 - \frac{1}{2\pi} \int_{\mathcal{V}(\mathbf{x})} d\Omega \,)} \,. \quad (3)$$

[1] The rendering algorithm we use is an improved version of [6]. We have verified the accuracy of the rendering algorithm by rendering surfaces for which an analytic solution to the radiosity equation is known (e.g. a spherical concavity) and found the RMS error to be about 2%.

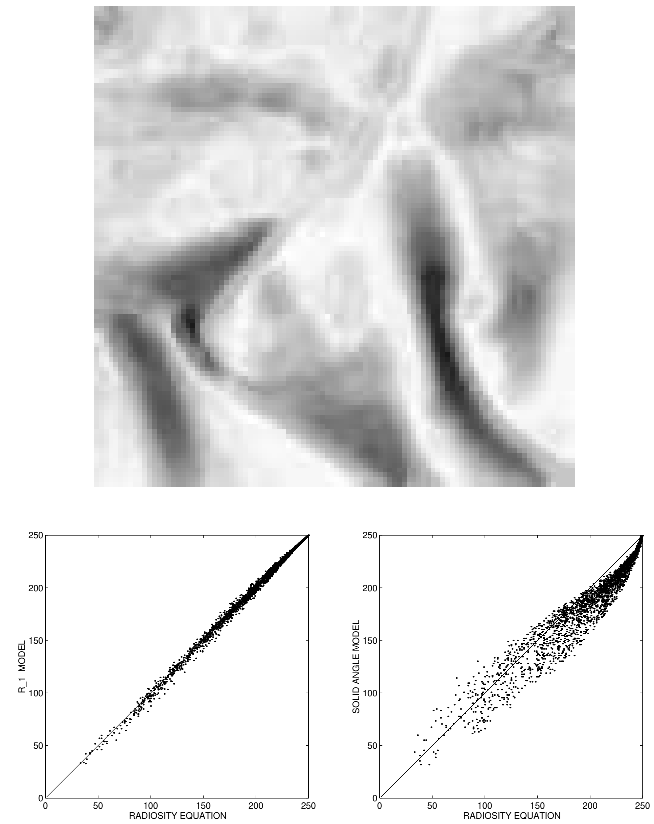

Figure 2: *Above:* A synthetic 100×100 smooth depth map was rendered using the radiosity equation $R(\mathbf{x})$ with $\rho = 0.5$. *Left:* The scatter plot compares $R_1(\mathbf{x})$ to $R(\mathbf{x})$. The R_1 model has a mean error of 10^{-3} % and standard deviation of 1%. *Right:* The scatter plot compares $R_2(\mathbf{x})$ to $R(\mathbf{x})$. The R_2 model has a mean error of 4.9% and standard deviation 4.8%.

We refer to this as the **solid angle model** since the term, $\int_{\mathcal{V}(\mathbf{x})} d\Omega$, is the solid angle of the visible source. The right scatter plot of Figure 2 compares $R(\mathbf{x})$ and $R_2(\mathbf{x})$. Note that the correlation is not as good as with the $R_1(\mathbf{x})$ model.

Finally, we address a situation in which the R_1 model begins to break down. Figure 3 shows a scatter plot of $R_1(\mathbf{x})$ versus $R(\mathbf{x})$ for the rendered scene shown in Figure 1. Two clusters in the scatter plot appear, corresponding to points on the ground plane and ellipsoid respectively. The R_1 model underestimates the radiance on the ellipsoid, but overestimates the radiance on the ground plane. This asymmetry is due to the fact that the ellipsoid may receive one bounce interreflections from the ground whereas the ground may receive only two (or more) bounce interreflections from the ellipsoid [5]. Since radiance decreases with each bounce, the interreflected radiance received by the ground plane is less that that received by the ellipsoid. Thus each point on the ground tends to see points on the ellipsoid that are darker than itself, so the R_1 model overestimates the radiance of points visible from the ground. (Similar clusters occur when the solid angle model is used.) Observe that despite this higher order effect, the mean error of the R_1 model remains quite small.

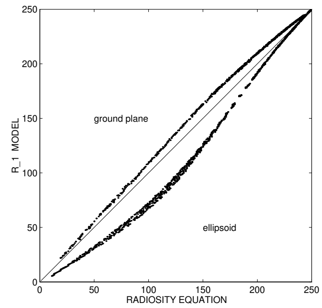

Figure 3: A scatter plot comparing $R_1(\mathbf{x})$ to $R(\mathbf{x})$ for the image of Figure 1. The mean and standard deviation of the modelling errors, $R(\mathbf{x}) - R_1(\mathbf{x})$, are 1.6 % and 5.4 %.

2.2 Viewpoint assumptions

Suppose we are given a single grey-level image $I(x, y)$. To interpret the image intensities as due to shading under diffuse lighting, we make a number of viewpoint assumptions: the surface seen in the image is matte with known albedo, ρ; the image is formed under orthographic projection; the depth map $z(x, y)$ is a continuous, single–valued function of (x, y); the source is a uniform hemisphere at infinity, centered about the line of sight, with its equator at depth zero; shadows are cast only by surfaces visible in the image (in particular, the brightest surface point sees the entire source.) Ignoring non-linearities of the sensor response, we obtain the image intensity model:

$$I(x,y) = \frac{I_{max} \ \frac{1}{\pi} \ \int_{\mathcal{V}(\mathbf{x})} \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} \ d\Omega}{1 \ - \ \rho \ (\ 1 - \frac{1}{\pi} \ \int_{\mathcal{V}(\mathbf{x})} \mathbf{N}(\mathbf{x}) \cdot \mathbf{u} \ d\Omega \ )} \quad (4)$$

where I_{max} is the maximum intensity in the image.

In the following section, we will introduce a algorithm for solving Equation (4): That is, given an image $I(x, y)$, we compute a surface that satisfies Equation (4).

3 New Algorithm

The only previous algorithm for recovering shape–from–shading under diffuse lighting is due to Langer and Zucker (LZ) [7, 8] and is based on a solid angle model of image intensity (recall Equation 3). Our algorithm has three advantages over the LZ algorithm. First, instead of using a coarsely sampled hemicube to represent the 2-D region of the visible source, we represent the 1-D boundary of the visible source. This boundary is called the **skyline**. From the viewing assumptions that led to Equation 4, it follows that a skyline is a single-valued function of the azimuth angle.[2]

Second, rather than computing each pixel's skyline at each discrete depth, we compute it once and update it at only those depths at which a qualitative change occurs. As the depth of a pixel increases, closer surfaces obscure more distant surfaces (see Figure 4). A **visual event** [2, 11, 13] occurs when one surface obscures another, at which point the skyline must be updated.

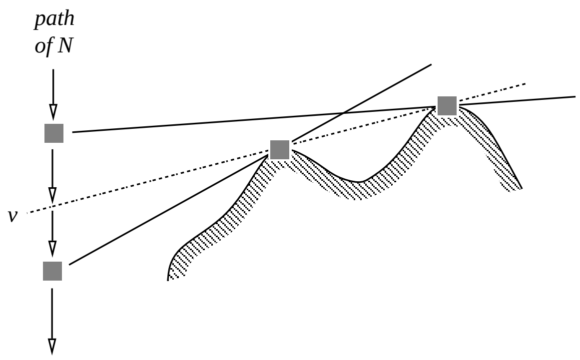

Figure 4: As the depth of a node $\mathcal{N}$ increases, the surface defining the skyline of $\mathcal{N}$ changes. A **visual event** occurs at the point of change (v in the figure). The higher node appears on the skyline when $\mathcal{N}$ is above v and the lower node appears when $\mathcal{N}$ is below.

Third, we treat depth as a continuum. This allows us to estimate the surface normal by fitting a plane through positions of any neighboring pixels for which the depth is known. (While the LZ algorithm could, in principle, use an arbitrarily fine sampling of depth, this would entail prohibitively large space and time costs.) Estimating the normal allows us to use Equation (4) to decide the depth of a pixel, rather than relying on a solid angle model.

[2]This is not always the case, of course. The ellipsoid of Figure 1 is a counterexample. We will see later that errors in the computed depth map for the ellipsoid can be attributed partially to this assumption.

The basic idea of our algorithm is as follows. As in the LZ algorithm, each pixel is associated with a **node** $\mathcal{N}(x, y)$ in space which has the same (x, y) value as the pixel and has a depth value z which increases monotonically from zero as the algorithm progresses. Each node has an initial depth of zero. Over the course of the algorithm, each node descends until it reaches the surface depth z that satisfies Equation 4. At this depth, a **surface event** is said to occur and the node changes status from a **free node** to a **surface node**; thereafter, the depth of the node remains fixed.

The algorithm maintains a priority queue of visual and surface events, where the priority of an event is the depth at which it occurs. For each pixel (x, y) of the image, the highest visual *or* surface event of the node $\mathcal{N}(x, y)$ is stored in the queue. The algorithm proceeds by removing and processing events from the queue in order of increasing depth. Initially, nodes that correspond to the pixels of maximum intensity are inserted into the queue as surface events of depth zero. The processing of an event may cause more events to be inserted into the queue, so the size of the queue varies over time. When the queue becomes empty, all nodes will have a depth z that satisfies Equation (4).

3.1 Visual Events

The skyline of a node is divided by azimuth angle into a number of equal–size sectors. There are typically 16 sectors, each spanning $\frac{\pi}{8}$ radians. The algorithm approximates the skyline in each sector with a constant elevation angle, which is the highest elevation angle of the surface nodes visible in that sector. (We implicitly assume that each surface node spans the entire sector.) Accuracy can be traded for speed by increasing the number of sectors. See Figure 5.

Consider one sector of the skyline of a free node $\mathcal{N}$. As the depth of $\mathcal{N}$ increases, closer surface nodes in the sector obscure more distant surface nodes, since surface nodes remain fixed while $\mathcal{N}$ descends and since each surface node is assumed to span the whole sector. When the surface node of highest elevation within a sector becomes obscured, a visual event occurs and the skyline of $\mathcal{N}$ undergoes a qualitative change and must be updated (recall Figure 4).

To facilitate this update, each sector of $\mathcal{N}$ stores the surface nodes that can potentially appear on the skyline, sorted by increasing horizontal distance. The nodes in this list form a convex chain, as shown in Figure 6. Clearly, no node below the convex chain can appear on the skyline since it will be obscured by a closer node. As the depth of $\mathcal{N}$ increases, each node on the convex chain appears on the skyline in turn, in order of decreasing horizontal distance.

If a sector contains at least two surface nodes on its convex chain, the highest visual event is computed in a straightforward manner: Let z_1 be the depth of the most distant node on the chain. Let d_1 be its horizontal distance from $\mathcal{N}$. Define z_2 and d_2 similarly for the next–most–distant node on the chain. Then the visual event occurs at depth $z_2 + d_2\ (z_1 - z_2)/(d_2 - d_1)$. Each sector stores the depth of its highest visual event.

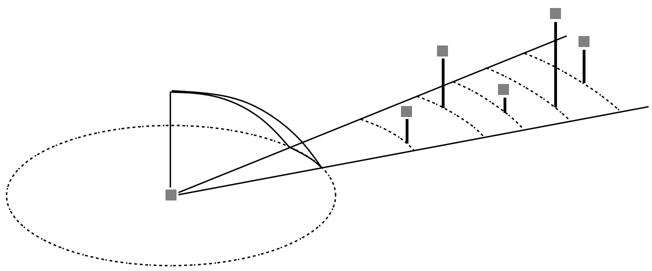

Figure 5: The skyline around a node is divided into sectors. The skyline within each sector is approximated by a constant elevation angle, which is the highest elevation angle of all surface nodes in that sector. Surface nodes are assumed to span the whole sector. Nodes are represented with grey squares in the figure.

3.2 Surface Events and Normal Estimation

A surface event occurs when a node $\mathcal{N}(x, y)$ reaches a depth z that satisfies

$$\int_{\mathcal{V}(x,y,z)} \mathbf{N}(x, y, z) \cdot \mathbf{u}\ d\Omega \ = \ \frac{\pi\ (1-\rho)\ I(x, y)}{I_{max} - \rho I(x, y)}. \quad (5)$$

In this section we discuss how to determine z, given the skyline around $\mathcal{N}(x, y)$. The right side of Equation (5) is be computed once for each (x, y). Assume, for the moment, that we can compute the integral on the left side for any particular value of z. Then the solution of the equation can be determined by binary search, as follows. Let z_{min} be a lower bound on z and let z_{max} be an upper bound. Let $z_{mid} = (z_{min} + z_{max})/2$. Evaluate the integral in Equation (5) at z_{mid}. If this value is greater than the value of the right side assign z_{mid} to z_{min}; otherwise assign z_{mid} to z_{max}. Continue until the difference between z_{min} and z_{max} is below some threshold (we use 10^{-4}).

We next discuss how to compute the integral on the left side of Equation (5), given a depth z and a

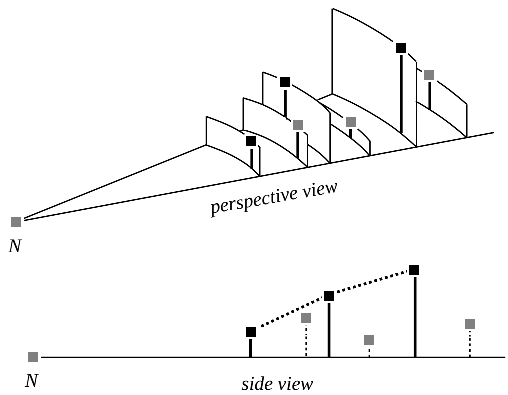

Figure 6: Each sector of $\mathcal{N}$ stores a convex chain of nodes that can potentially appear on the skyline. Nodes below the chain cannot appear on the skyline and are not stored, while nodes on the chain appear on the skyline in order of decreasing horizontal distance as $\mathcal{N}$ descends. In the figure, the black nodes are on the convex chain.

node $\mathcal{N}(x, y)$ for which the skyline is known (recall that the skyline in each sector of $\mathcal{N}(x, y)$ is defined by the surface node of greatest elevation in that sector). First, the surface normal $\mathbf{N}(x, y, z)$ is estimated by considering the nodes in the 8–neighborhood of (x, y). For each *pair* of neighbors that are surface nodes (that is, their final depths are already known), we determine the plane that passes through these nodes and the point (x, y, z). The normals of all such planes are averaged to yield the estimate for $\mathbf{N}(x, y, z)$. If only one adjacent surface node exists, we use the "most upward pointing" normal to the line passing through (x, y, z) and the node (that is, the normal of most negative z component). If no surface nodes exist, we use $(0, 0, -1)$ as the normal.

Given the normal, the integral is computed by summing its value over each sector of the node. Its value in each sector has a closed–form solution which is a function of elevation angle. Lookup tables are used for efficiency. Further details of the algorithm will be presented in a subsequent paper.

4 Results

We now compare the performance of the old (LZ) and new algorithms on several test images, each of size 100×100 pixels. For each image, we show a cross–section of the real depth map and the depth maps computed by the old and new algorithms. For the rendered images, the albedo is 0.5 and each cross–

section is taken horizontally across row 50. For the real image in Section 4.4, the albedo is 0.7, and two cross sections are taken.

4.1 Hemispherical Concavity

Figure 7 shows a grey–level image of a hemispherical concavity excavated from the ground. The image has just two grey–levels (recall Section 1). Note the smooth, accurate depth map that results from the new algorithm, which represents depth as a continuum.

To understand why the new algorithm performs better, consider for example the deepest point in the concavity. The visible source is directly overhead at this point and, thus, the surface radiance is greater than what the solid angle model dictates. (The solid angle model approximates $\mathbf{N} \cdot \mathbf{u}$ by a constant, 0.5). As a result, the old (LZ) algorithm, which is based on the solid angle model, incorrectly attributes the higher radiance to a greater solid angle, that is, to a shallower concavity.

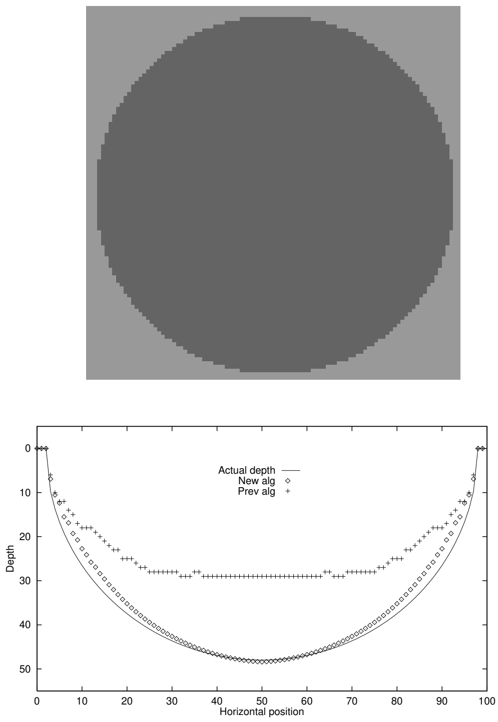

Figure 7: An analytically computed image of a spherical concavity.

4.2 Ellipsoid on plane

Figure 8 shows the ellipsoid from Figure 1, now viewed from directly above. Both algorithms fail to correctly infer the depth of the ground; the reason is interesting. First, both algorithms underestimate the direct illumination of points on the ground because they mistakenly assume that the depth map is a single–valued function of (x, y). In other words, they assume that the ellipsoid has a curtain hanging straight down from its edge. Since light rays from the source pass beneath the ellipsoid, points on the ground receive *more* direct illumination than either algorithm can account for. At the same time, the algorithms overestimate the radiance coming from the dark underside of the ellipsoid (recall Figure 3). The net result of these competing errors is clearly different for the two algorithms.

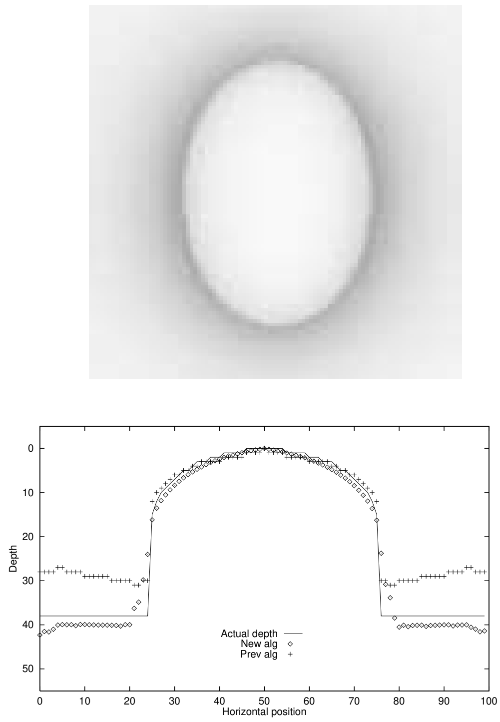

Figure 8: A rendered image of a ellipsoid sitting on a plane.

4.3 Smooth depth map

Figure 9 shows the results of the algorithm on the image of Figure 2. Observe that both algorithms underestimate the depth of central hilltop. This is due to a subtle ill–conditioning property of the shape–from–shading–under–diffuse–lighting problem. Small differences in image intensity of the brightest points in the image can correspond to relatively large differences in depth.

For example, over the entire surface, the depths range from 3 to 39 and the intensities range from 32 to 250. Pixel $(x, y) = (15, 4)$, which is a local intensity maximum, has depth 4 and intensity 249, while pixel $(x, y) = (90, 71)$, also a local intensity maximum, has depth 13 and intensity 247. In this case, a 0.9% difference in intensity corresponds to a 25% difference in depth. Such a small intensity difference is typically lost in the image noise.

Next observe that both algorithms overestimate the depth at the right boundary of the image. Because there is a local intensity maximum on the right boundary, both algorithms assume the surface is horizontal. (Since both algorithms construct the depth map in order of increasing depth, the darker pixels below the local maximum are not processed until the local maximum is fixed. In particular, in the new algorithm, the deeper pixels cannot contribute to the normal estimation.) From the cross–section, it is clear that the normal should be pointing leftward. Because of this error, the surface receives less light than the algorithms expect, so the algorithms infer a smaller solid angle of the visible source and hence a greater depth.

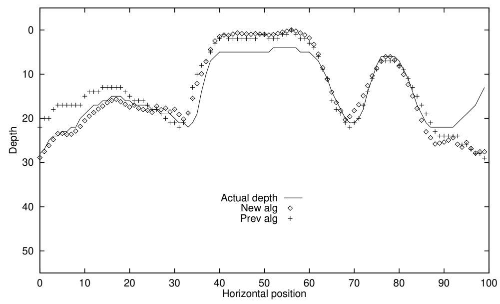

Figure 9: Surface computed from image of Figure 2.

4.4 Real Image

Finally, we compare the old and new algorithms on a real image. Figure 10 shows two eggs resting on a ground plane, viewed from above by a Sony CCD video camera. The eggs and ground were painted grey and matte with albedo 0.7. (The albedo was measured by comparing the image intensity of the grey background with the intensity of a white piece of paper.) The scene and camera were surrounded by a white sheet which acted as a hemispheric diffuse light source. Calibrated depth maps were obtained directly from the images using the fact that eggs have cylindrical symmetry. Pixel noise was reduced to one grey–level out of 255 by averaging over multiple images. A range of camera apertures was used to verify the linearity of the sensor response. Vignetting effects near the image boundary were removed by dividing the image intensities by those of a second image containing only the grey ground.

Horizontal slices through the actual and computed depth maps are shown for rows 36 and 75. Both algorithms recovered the shape of the egg quite well. (Errors were due in part to surface roughening, glossiness, and pigmentation that arose through contact of the eggs with other surfaces.) Observe that the new algorithm was able to recover the height of the ground plane more accurately than the old algorithm.

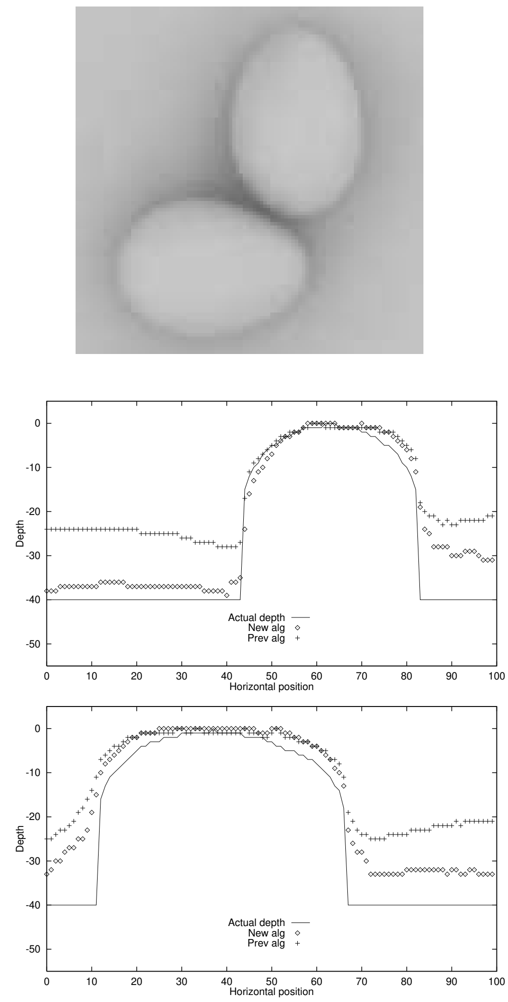

Figure 10: A real image of two eggs resting on a ground plane, viewed from above. Both eggs and ground were painted grey with $\rho = 0.7$. Horizontal slices through the actual and computed depth maps are shown for rows 36 and 75.

4.5 Summary of Experiments

Table 1 summarizes the mean and root–mean–square (RMS) errors for each of the images discussed above. For each image, errors are reported for each algorithm. Error is measured in pixel–width units. The true depth maps were between 35 and 50 units deep at their deepest point. The data show clearly that the new algorithm performs better than the old one. The improvements are modest, however, and suggest that the three types of errors we have discussed – namely, approximations in the image formation model, poor local constraints at the image boundary, and ill-conditioning of the problem itself – provide an ultimate bound on performance.

Table 1: Error Measurements

Scene	Algorithm	Mean Error	RMS Error
Concavity	New	1.5	1.7
	LZ	7.6	6.1
Ellipsoid	New	-0.7	3.4
	LZ	7.7	5.4
Smooth	New	-2.5	4.7
	LZ	-0.1	5.5
Two Eggs	New	6.0	6.0
	LZ	12.1	7.9

Acknowledgements

James Stewart's work is supported by the Information Technology Research Centre of Ontario, the Natural Sciences and Engineering Research Council of Canada, and the University of Toronto. Michael Langer's work was partially supported by grants from NSERC and AFOSR, while he was a graduate student and postdoctoral fellow at the Center for Intelligent Machines, McGill University. The authors would like to thank David Waltz and the reviewers for helpful comments.

References

[1] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice.* Addison-Wesley, Reading, Mass., second edition, 1990.

[2] Z. Gigus and J. Malik. Computing the aspect graph for the line drawings of polyhedral objects. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 12(2), February 1990.

[3] B. K. P. Horn. *The Psychology of Computer Vision*, chapter Obtaining Shape from Shading Information. McGraw-Hill, New York, 1975.

[4] B. K. P. Horn and R. W. Sjoberg. Calculating the reflectance map. *Applied Optics*, 18(11):1770–1779, 1979.

[5] J. J. Kœnderink and A. J. van Doorn. Geometrical modes as a general method to treat diffuse interreflections in radiometry. *Journal of the Optical Society of America*, 73(6):843–850, June 1983.

[6] M. S. Langer, P. Breton, and S. W. Zucker. Massively parallel radiosity in the presence of multiple isotropic volume scattering. In *Graphics Interface '95*, pages 103–108, Quebec, Canada, May 1995.

[7] M. S. Langer and S. W. Zucker. Diffuse shading, visibility fields, and the geometry of ambient light. In *Proceedings of the 4th International Conference on Computer Vision*, pages 138–147, Berlin, Germany, 1993.

[8] M. S. Langer and S. W. Zucker. Shape-from-shading on a cloudy day. *Journal of the Optical Society of America A*, 11(2):467–478, 1994.

[9] P. Moon and D. Spencer. *The Photic Field.* MIT Press, Cambridge, Ma., 1981.

[10] S. K. Nayar, K. Ikeuchi, and T. Kanade. Shape from interreflections. *International Journal of Computer Vision*, 6:173–195, 1991.

[11] H. Plantinga and C. Dyer. Visibility, occlusion, and the aspect graph. *International Journal of Computer Vision*, 5(2):137–160, 1990.

[12] R. Siegel and J. Howell. *Thermal Radiation Heat Transfer.* Hemisphere Publ. Corp., 1981.

[13] A. J. Stewart and S. Ghali. Fast computation of shadow boundaries using spatial coherence and backprojections. *SIGGRAPH Proceedings*, pages 231–238, 1994.

[14] J. T. Todd and E. Mingolla. Perception of surface curvature and direction of illumination from patterns of shading. *Journal of Experimental Psychology: Human Perception and Performance*, 9(4):583–595, 1983.

[15] A. Yonas, L. Goldsmith, and J. Hallstrom. Development of sensitivity to information provided by cast shadows in pictures. *Perception*, 7:333–341, 1978.