

Shape from Shading

Shree K. Nayar

Monograph: FPCV-3-3

Module: Reconstruction I

Series: First Principles of Computer Vision

Computer Science, Columbia University

March, 2025

[FPCV Channel](#)

[FPCV Website](#)

Shape from Shading

Shree K. Nayar
Columbia University

Topic: Shape from Shading, Module: Reconstruction I
First Principles of Computer Vision

1

Shape from a Single Image?

Given image I , source direction s and surface reflectance

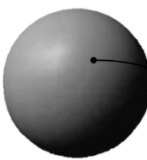
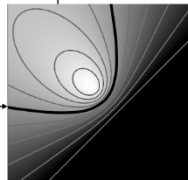


Image I



Reflectance Map $R(p, q)$

Can we estimate surface gradient (p, q) from single intensity? NO

Shape from shading is an under-constrained problem

REVIEW
2

In this lecture, we will discuss the challenging problem of recovering three-dimensional shape information from a single shaded image. We are given an image I of an object, such as the sphere in slide 2, and we know both the direction and the brightness of the light source. We are also given the reflectance properties (BRDF) of the object, which is assumed to be uniform over the object. The source information and the BRDF can be used to compute a reflectance map, $R(p, q)$. This map gives the image intensity corresponding any given surface normal (p, q) .

In shape from shading, we want to go the other way: from an intensity value to the surface normal. Can we estimate the normal (p, q) from a single intensity value? We know the answer to be negative. A given intensity value corresponds to an iso-brightness contour in the reflectance map, which means that an entire family (an infinite number) of (p, q) values would generate the same brightness in the image.

Since the problem of recovering 3D shape from a single shaded image is severely under-constrained, we want to find ways of constraining the problem and then solving it. First, we are going to take a look at human perception of shading. We, humans, are fairly good at perceiving the shape of an object just from its shading. It turns out that the only reason we are able to do this is because we invoke a series of assumptions. To develop a shape from shading algorithm, we will first come up with a reasonable set of assumptions, map those assumptions into mathematical constraints, and then develop an algorithm to recover shape from a single shaded image.

Shape From Shading

Method for recovering 3D shape information from a single image using shading.

Topics:

- (1) Human Perception of Shading
- (2) Shape From Shading Algorithm

3

Human Perception of Shading




Shree K. Nayar
Columbia University

Topic: Shape from Shading, Module: Reconstruction I
First Principles of Computer Vision

4

Shape From Shading in Humans

We seem to perceive shape from a single shaded image

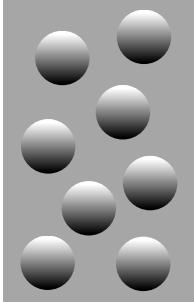
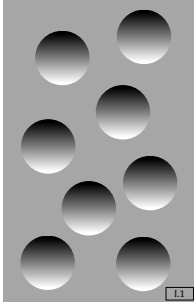
We make many assumptions in doing so

5

Let us take a closer look at human perception of shading. We can easily figure out that the objects shown on the right are a vase, a bunny and a head, respectively. In fact, even if we look at a small piece of the head — the cheek, for example — we would be able to perceive the undulations of the surface from its shading. We have already established that shape from shading is an under-constrained problem. We, humans, are able to come up with interpretations of shape from shading because we invoke strong assumptions. In what follows, we present a few simple experiments conducted by V. S. Ramachandran that reveal some of the assumptions we make.

Shown here are two panels of objects. Most people would agree that the panel on the left is a set of bumps, meaning they are convex and protrude out of the surface. Additionally, most would agree that panel on the right is a set of concavities. What enables us to arrive at these judgements? It turns out that we assume that the light source is above us. This is not unreasonable because the world we live in is lit by the sun, which is above us. In other words, we expect light to arrive at an object from above it, rather than below it. We invoke this assumption to interpret shape of shading. If the lighting is from above, then the objects on the left must be bumps and the ones on the right must be concavities.

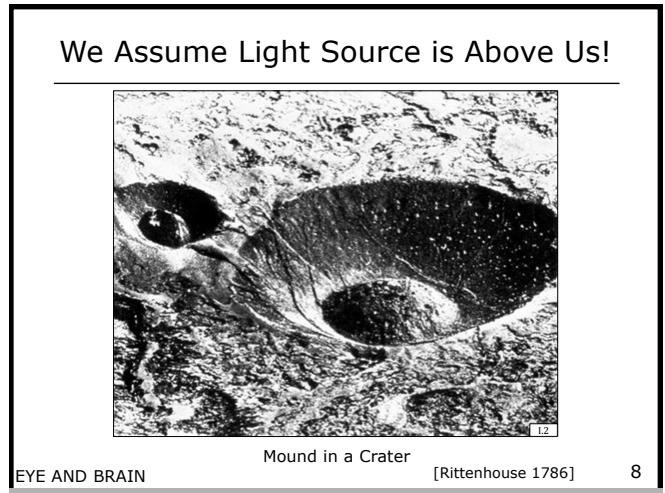
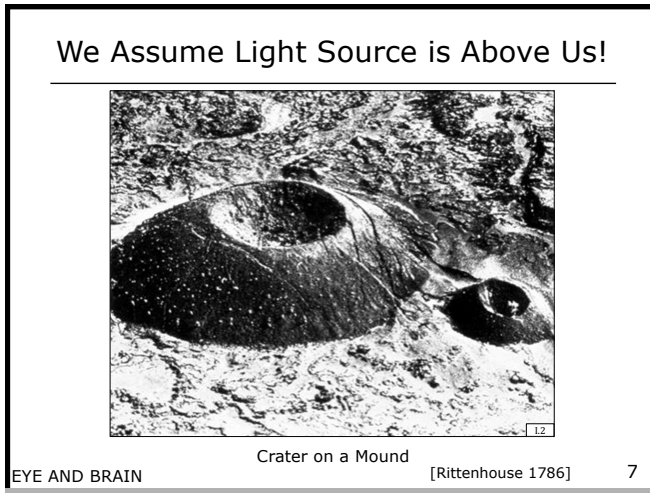
We Assume Light Source is Above Us!

The shaded objects in the left panel are usually seen as convex, whereas those in the right panel are usually seen as concave.

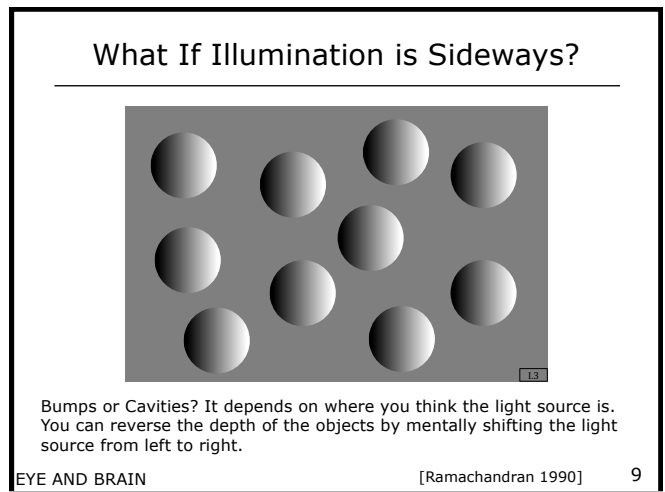
EYE AND BRAIN
[Ramachandran 1990]

6

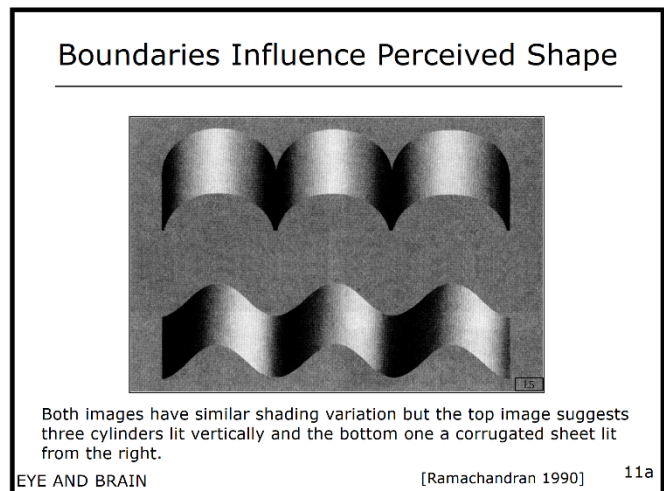
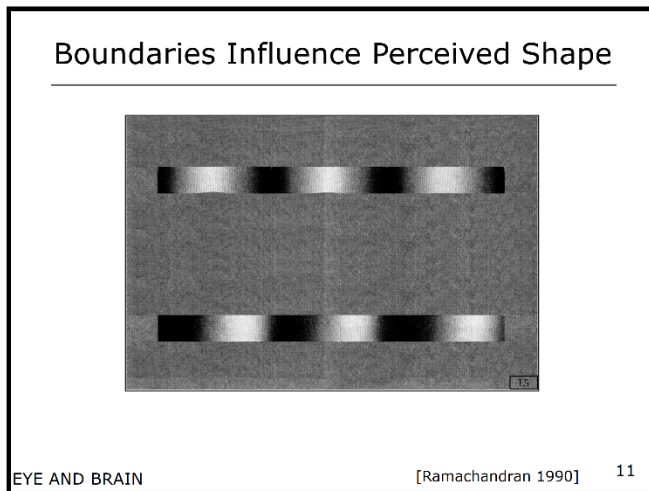
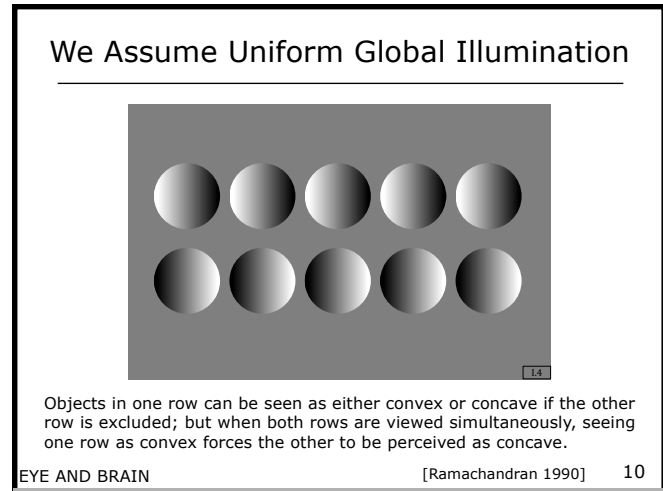


Our assumption that the light source is above us can be further illustrated using this example. We can agree that the left slide is a mound with a crater in the center. If we flip this image upside down, we might instinctively assume that we would see a mound that is upside down with a little crater in the center. However, what we end up seeing (right slide) is a large crater with a little mound in the center. Why does our interpretation of shape change so drastically when the image is simply flipped? It is because both these interpretations are consistent with our assumption that the light falls on the scene from above.

What if the illumination is from the side? In this case, people’s interpretation can go either way—bumps or concavities. If we imagine the light source is on the right side, then we would interpret the objects as bumps. If we now mentally move the light source to the left side, then we can convince ourselves that they are concavities. We do not make a strong assumption regarding lighting from the side, since left lighting and right lighting are equally likely.

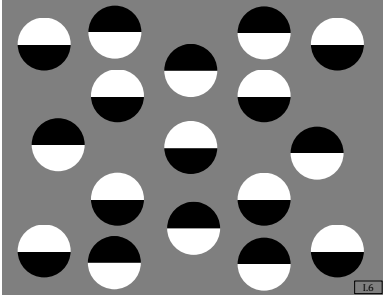


Here, we illustrate the notion of global illumination. If the two rows here were shown one after the other, in each case, our interpretation would depend on whether we assume the lighting to be from the left or from the right. However, when we view the two rows simultaneously, if we assume the top row to be bumps, then we infer the bottom row to be concavities. In other words, we prefer not to consider the lighting on the scene to varying dramatically over the scene. We prefer to assume that the lighting of the entire scene is coming from a single direction.



There is also the issue of boundaries. On the left, we can see two similarly-shaded strips. However, if we give the two strips different boundaries, as shown on the right, we can dramatically change the interpretation of their shapes. For instance, the top strip suggests three cylinders sitting next to each other, lit from above. In contrast, the bottom strip is perceived as a corrugated surface lit from the right. Thus, boundaries are extremely important in the perception of shape from shading.

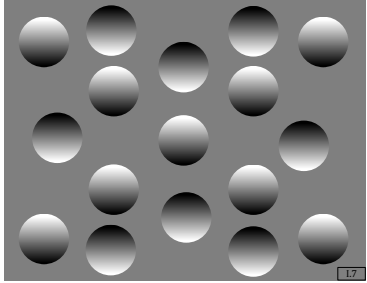
Perceptual Grouping



Perceptual grouping of objects with the identical appearance is difficult to achieve without shading.

EYE AND BRAIN [Ramachandran 1990] 12

We Use Shading for Perceptual Grouping



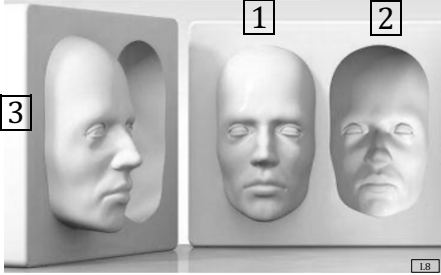
Objects that are lighter on top are usually perceived as convex objects that can be mentally grouped and segregated (to form an X pattern) from the background of concave objects.

EYE AND BRAIN [Ramachandran 1990] 13

Next, let us consider shading in the context of perceptual grouping. The left slide shows several discs with no shading. In terms of appearance, we have two sets of discs. In one set, each disc is black on the top and white on the bottom, and in the other set, each disc is white on the top and black on the bottom. Looking at this image, it is difficult to quickly group together all the discs that have the same appearance. On the right slide one set of discs are replaced with bumps, while the other set is replaced with concavities. In this case, if we stare at the center bump, all the bumps become immediately visible and grouped to form an “X.” It is clear, therefore, that perceptual grouping is aided by shading.

Finally, we have an example here where we override our previous assumption regarding the direction of illumination. Consider the head-on view of this sculpture with two heads. When we look at the head on the left **1**, we perceive it has being lit from above. In the case of the head on the right **2**, it appears to be lit from below. In this case, we are willing to override our assumption that light falls from above because we are familiar with the more or less convex shape of a head. In this case, we are being tricked as the head in **2** is actually a concavity as revealed by the side view of the sculpture shown on the left **3**. In other words, the two sculptures are both lit from the top. In short, while interpreting shaded images, the brain usually assumes that lighting comes from above, but it is willing to reject this assumption when it comes to a familiar shape, such as that of a head.

We Tend to “See” the Familiar



Hollow-Mask interiors lit from above produce an eerie impression of protruding face lit from below. In interpreting shaded images the brain usually assumes light shining from above but here it rejects the assumptions in order to interpret the images as normal, convex objects.

EYE AND BRAIN [Ramachandran 1990] 14

Stereographic Projection

Shree K. Nayar
Columbia University

Topic: Shape from Shading, Module: Reconstruction I
First Principles of Computer Vision

16

Stereographic Projection: fg Space

pq Space

fg Space

Problem:
 p or q is infinite when $\theta = 90^\circ$

$$f = \frac{2p}{1 + \sqrt{p^2 + q^2 + 1}}$$

$$g = \frac{2q}{1 + \sqrt{p^2 + q^2 + 1}}$$

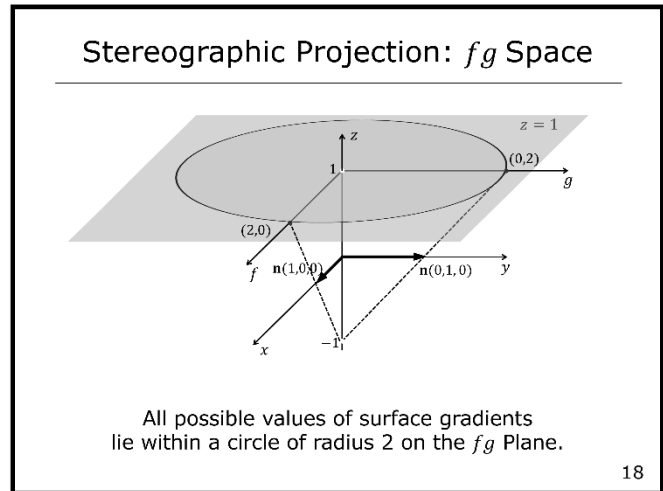
17

Before we develop our shape from shading method, let us revisit our representation of surface orientation. Consider a unit surface normal \mathbf{n} that makes an angle θ with the z -axis, which corresponds to the viewing direction \mathbf{v} of the camera. We already know how to map \mathbf{n} to its gradient space coordinates (p, q) . As discussed in the photometric stereo lecture, we place a plane at $z = 1$ that is parallel to the $x - y$ plane, and label its two axes p and q . To determine the (p, q) value corresponding to the unit normal \mathbf{n} , we extend the normal so that it intersects the $p - q$ plane. The point of intersection is the (p, q) that corresponds to \mathbf{n} .

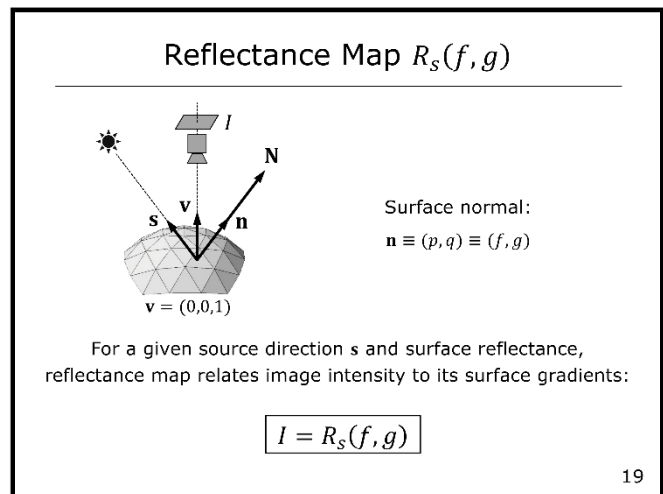
Unfortunately, the $p - q$ space has an undesirable attribute. As θ approaches 90 degrees, p and q can increase in value rapidly, and at 90 degrees, one or both of them can equal infinity. From a computational perspective, we would want the values of p and q to be bounded. To achieve this, we will use what is called the stereographic projection, which we denote as the $f - g$ space. Given the same unit normal \mathbf{n} and the same $z = 1$ plane, we now label the axes of the plane f and g . To go from \mathbf{n} to (f, g) , we draw a line that goes from the point $z = -1$ through the tip of the unit normal. The intersection of this line with the $f - g$ plane gives us the (f, g) corresponding to \mathbf{n} .

What is the relationship between the $f - g$ and the $p - q$ spaces? Consider the triangle formed by the origin of the $f - g$ plane, the intersection point (f, g) , and the point $z = -1$. Then, consider another triangle formed by the tip of \mathbf{n} , the projection of the \mathbf{n} onto the z axis, and the point $z = -1$. These are two similar triangles, which can be used to derive the expressions 1 that relate (f, g) to (p, q) .

Let us discuss the advantage of using the $f - g$ space instead of the $p - q$ space. Consider the unit normal $(0, 1, 0)$ aligned with the y -axis, which has a corresponding (f, g) value of $(0, 2)$. Then, consider the unit normal $(1, 0, 0)$ aligned with the x -axis, which has an (f, g) value of $(2, 0)$. With these two examples, we can convince ourselves that all (f, g) values corresponding to all the normals that lie in the visible (upper) hemisphere, lie within a circle in $f - g$ space with radius 2. Therefore, from a computational perspective, it is advantageous to compute normals in terms of (f, g) rather than (p, q) .



We know that \mathbf{n} is equivalent to (p, q) , which is equivalent to (f, g) . As shown here, we can define our coordinate system such that the viewing direction \mathbf{v} of the camera is aligned with the z -axis. As with photometric stereo, we assume that the light source is far away, such that all points on the object of interest share the same source direction. The reflectance map for a given source and a given surface BRDF can be denoted as R_s , which is a function of f and g . For any normal (f, g) , it gives us the corresponding image brightness I .

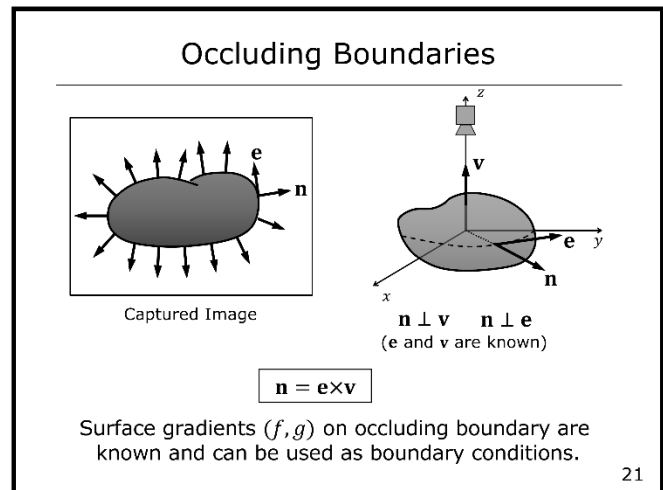


Shape from Shading Algorithm

Shree K. Nayar
Columbia University

Topic: Shape from Shading, Module: Reconstruction I
First Principles of Computer Vision

20



As stated earlier, shape from shading is an under-constrained problem and hence additional assumptions are needed to solve it. Our first assumption is related to occluding boundaries. In slide 21, on the right, we have a curved object, and on the left, we have its image. The boundary of the object in its image is called the occluding boundary—it is where the object curves upon itself, away from the viewing direction. We assume that the camera is distant from the object and hence the viewing direction is the same for all points on the object. If we apply an edge detector to the image, we get edges along the boundary of the object. If we consider a surface normal \mathbf{n} at a point on the occluding boundary, we know it must be perpendicular to the viewing direction \mathbf{v} and to the edge \mathbf{e} computed at the point. Since the normal \mathbf{n} is perpendicular to both \mathbf{e} and \mathbf{v} , which are both known, we can compute \mathbf{n} as the cross product of \mathbf{e} and \mathbf{v} . We will maintain all the normals computed along the occluding boundary as constants when computing the shape of the object from its shading.

The most important constraint we use to formulate shape from shading is called the image irradiance constraint, which states that the intensity value I measured at a location (x, y) should equal the result of plugging its (f, g) value into the reflectance map R_s . We therefore formulate an error e_r , that is the squared difference between the measured intensity value at each pixel and its estimate from the reflectance map, integrated over all the image pixels. We wish to find (f, g) values at all pixels that minimize e_r .

Image Irradiance Constraint

Assumption: Image irradiance (intensity) should equal the reflectance map. That is, $I(x, y) = R_s(f, g)$.

Minimize:

$$e_r = \iint (I(x, y) - R_s(f, g))^2 dx dy$$

Aim: Penalize errors between image irradiance and reflectance map.

22

Since we cannot compute an (f, g) from each measured intensity I , we use a smoothness assumption to relate the (f, g) values of neighboring pixels. That is, we assume that f and g vary slowly over the surface. To this end, we formulate a smoothness error e_s as the sum of the squares of the first derivatives of f and g (i.e., f_x , f_y , g_x and g_y), integrated over the entire image. We want our computed (f, g) values to minimize e_s as well.

Smoothness Constraint

Assumption: Object surface is smooth. That is, the gradient values (f, g) vary slowly.

Minimize:

$$e_s = \iint (f_x^2 + f_y^2) + (g_x^2 + g_y^2) dx dy$$

$$\text{where: } f_x = \frac{\partial f}{\partial x}, f_y = \frac{\partial f}{\partial y}, g_x = \frac{\partial g}{\partial x} \text{ and } g_y = \frac{\partial g}{\partial y}$$

Aim: Penalize rapid changes in f and g during surface estimation.

23

Now, we can formalize the shape from shading problem as finding the surface gradients (f, g) at all image pixels that minimize a total error e that is a weighted sum of the smoothness error e_s and the image irradiance error e_r . The weighting constant λ allows us to control the relative importances of the two errors. For instance, we can use a high value if we know that our image intensity measurements are very accurate. Additionally, in an iterative shape from shading algorithm, we can progressively increase λ and thus relax our smoothness constraint as the (f, g) values approach convergence.

Shape from Shading

Find surface gradients (f, g) at all image points that minimize the function:

$$e = e_s + \lambda e_r$$

where:

- e_s : Smoothness Constraint
- e_r : Image Irradiance Error
- λ : Weight

Known surface gradients (f, g) on
occluding boundary are held constant.

[Ikeuchi 1981] 24

We now present the numerical shape from shading algorithm developed by Ikeuchi and Horn. Since all the constraints and error terms presented above are in the continuous domain, we need to map them to the discrete domain. First, we define the smoothness error at a pixel (i, j) to be $e_{s_{i,j}}$, which includes the first derivatives of f and g computed using finite differences between $f_{i,j}$ and $g_{i,j}$ and their neighboring values. Similarly, we define the image irradiance error at pixel (i, j) to be $e_{r_{i,j}}$, which equals the squared difference between the measured pixel intensity $I_{i,j}$ and the value of the reflectance map for the corresponding $f_{i,j}$ and $g_{i,j}$ values. Finally, we compute the weighted sum of these two error terms per pixel and add them up for all pixels in the image to obtain the final error e . We want to find all the $f_{i,j}$ and $g_{i,j}$ values that minimize e .

Numerical Shape from Shading

Smoothness Error at point (i, j) :

$$e_{s_{i,j}} = \frac{1}{4} \left((f_{i+1,j} - f_{i,j})^2 + (f_{i,j+1} - f_{i,j})^2 + (g_{i+1,j} - g_{i,j})^2 + (g_{i,j+1} - g_{i,j})^2 \right)$$

Image Irradiance Error at point (i, j) :

$$e_{r_{i,j}} = (I_{i,j} - R_s(f_{i,j}, g_{i,j}))^2$$

Find $(f_{i,j}, g_{i,j})$ for all (i, j) that minimizes:

$$e = \sum_i \sum_j (e_{s_{i,j}} + \lambda e_{r_{i,j}})$$

[Ikeuchi 1981] 25

If $f_{k,l}$ and $g_{k,l}$ minimize e , then the derivatives of e with respect to $f_{k,l}$ and $g_{k,l}$ should be equal to zero. Given an image with $N \times N$ pixels, we have N^2 known intensity values. However, we have $2N^2$ unknowns since there are two unknowns (f and g) for each pixel. Fortunately, for any given pixel (i, j) , the values $f_{i,j}$ and $g_{i,j}$ appear in only four terms in the error e — twice in the term that corresponds to pixel (i, j) , once in the term for the pixel above it, and once in the term for the pixel to the left of it.

Numerical Shape from Shading

If $(f_{k,l}, g_{k,l})$ minimizes e , then $\frac{\partial e}{\partial f_{k,l}} = 0$ and $\frac{\partial e}{\partial g_{k,l}} = 0$

Given an image of size $N \times N$, there are $2N^2$ unknowns. ($N^2 f_{i,j}$'s and $N^2 g_{i,j}$'s)

However, note that each $f_{i,j}$ and $g_{i,j}$ appears in 4 terms in $e = \sum_i \sum_j (e_{s_{i,j}} + \lambda e_{r_{i,j}})$:

	×	
×	2×	

26

Therefore, for any given pixel (k, l) , we can write out the four terms of e in which $f_{k,l}$ and $g_{k,l}$ appear, find the derivatives of e with respect to $f_{k,l}$ and $g_{k,l}$, and set them equal to zero. Here, $\bar{f}_{k,l}$ and $\bar{g}_{k,l}$ are the average values of f and g , respectively, computed using the four neighbors (top, bottom, left and right) of pixel (k, l) . Note that $f_{k,l}$ and $g_{k,l}$ appear in both the reflectance map calculation and the calculation of the derivative of the reflectance map. In short, we have two non-linear equations in $f_{k,l}$ and $g_{k,l}$, that do not lend themselves to closed-form solutions for $f_{k,l}$ and $g_{k,l}$.

Numerical Shape from Shading

If $(f_{k,l}, g_{k,l})$ minimizes e , then $\frac{\partial e}{\partial f_{k,l}} = 0$ and $\frac{\partial e}{\partial g_{k,l}} = 0$

Therefore:

Eq 1: $\frac{\partial e}{\partial f_{k,l}} = 2(f_{k,l} - \bar{f}_{k,l}) - 2\lambda(I_{k,l} - R_s(f_{k,l}, g_{k,l})) \frac{\partial R_s}{\partial f} \Big|_{f_{k,l}, g_{k,l}} = 0$

Eq 2: $\frac{\partial e}{\partial g_{k,l}} = 2(g_{k,l} - \bar{g}_{k,l}) - 2\lambda(I_{k,l} - R_s(f_{k,l}, g_{k,l})) \frac{\partial R_s}{\partial g} \Big|_{f_{k,l}, g_{k,l}} = 0$

where $\bar{f}_{k,l}$ and $\bar{g}_{k,l}$ are local averages:

$$\bar{f}_{k,l} = \frac{1}{4}(f_{k+1,l} + f_{k-1,l} + f_{k,l+1} + f_{k,l-1})$$

$$\bar{g}_{k,l} = \frac{1}{4}(g_{k+1,l} + g_{k-1,l} + g_{k,l+1} + g_{k,l-1})$$

27

Since a simple solution to the above equations is not forthcoming, we develop an iterative algorithm to solve for $f_{k,l}$ and $g_{k,l}$. We rewrite the equations with $f_{k,l}$ and $g_{k,l}$ on the left-hand side and use the superscripts (n) and $(n + 1)$ to denote the current and next iteration numbers, respectively. Then, starting with some initial $f_{k,l}$ and $g_{k,l}$ values, we compute the right-hand side of the equations to obtain new $f_{k,l}$ and $g_{k,l}$ values for all pixels in the image. We repeat this iterative process until the $f_{k,l}$ and $g_{k,l}$ values for all (k, l)

Iterative Solution

Update Rule:

$$f_{k,l}^{(n+1)} = \bar{f}_{k,l}^{(n)} + \lambda \left(I_{k,l} - R_s(f_{k,l}^{(n)}, g_{k,l}^{(n)}) \right) \frac{\partial R_s}{\partial f} \Big|_{f_{k,l}^{(n)}, g_{k,l}^{(n)}}$$

$$g_{k,l}^{(n+1)} = \bar{g}_{k,l}^{(n)} + \lambda \left(I_{k,l} - R_s(f_{k,l}^{(n)}, g_{k,l}^{(n)}) \right) \frac{\partial R_s}{\partial g} \Big|_{f_{k,l}^{(n)}, g_{k,l}^{(n)}}$$

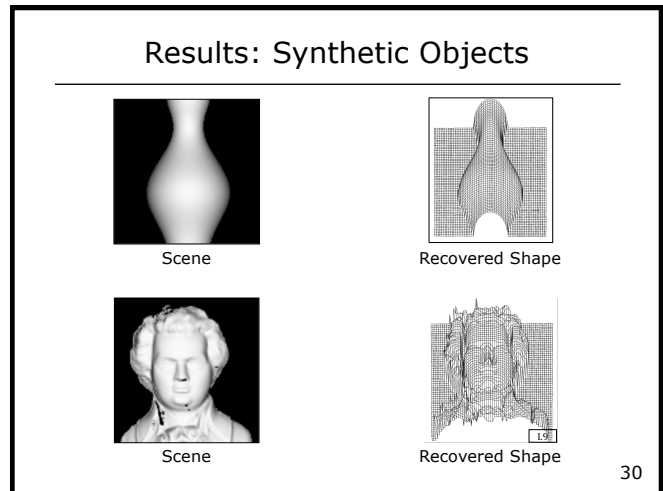
n : iteration

- Use known normals to fix (f, g) values on occluding boundary. Initialize the rest to $(0, 0)$
- Iteratively compute (f, g) until the solution has converged.

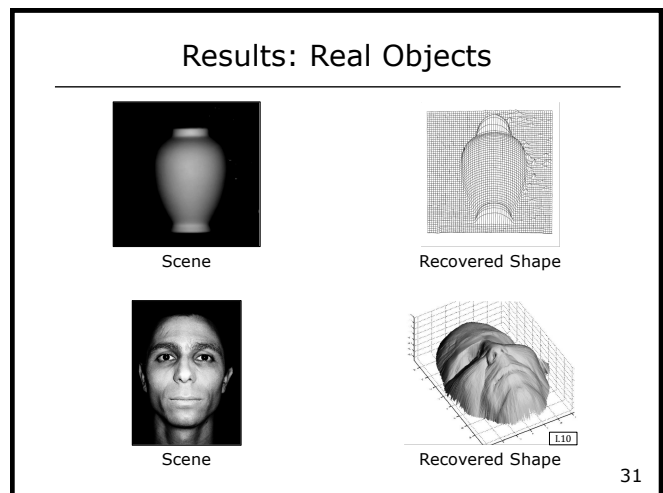
29

converge, that is, all the differences between the $f_{k,l}$ and $g_{k,l}$ values computed in consecutive iterations are below a threshold. Generally, the initial values for $f_{k,l}$ and $g_{k,l}$ are chosen to be 0, except along the occluding boundaries where the values we computed earlier (slide 21) are held constant.

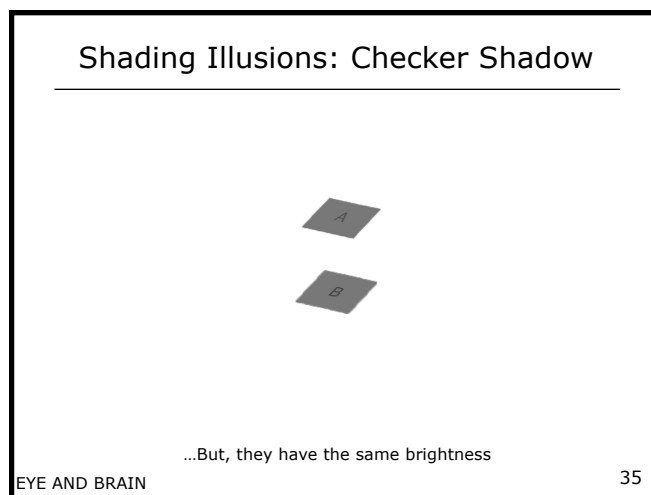
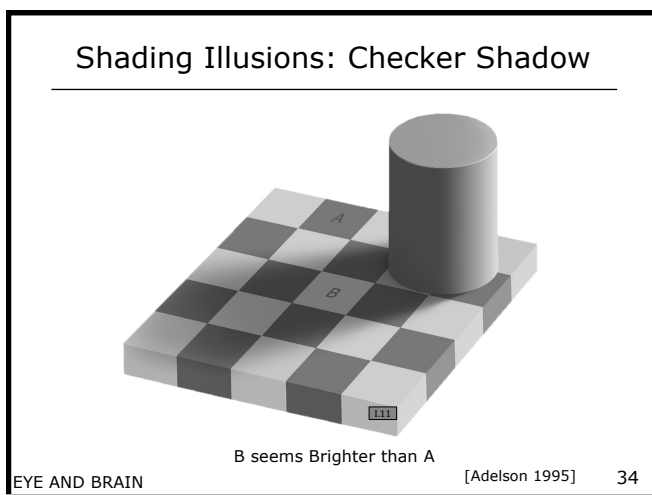
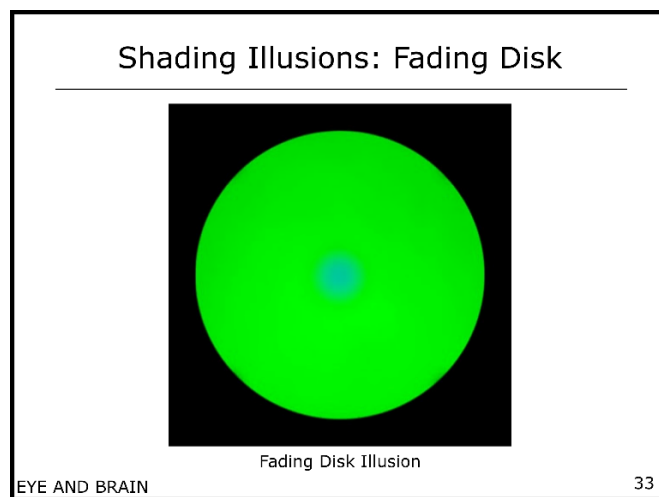
Here are a couple of shape from shading results (right) computed using rendered images of Lambertian objects (left). After applying the shape from shading algorithm, we obtain (f, g) values over the entire image. Then, we apply the technique discussed in the photometric stereo lecture to go from surface normals to a continuous depth map. We see that in the case of the Lambertian vase (top), the estimated shape is fairly accurate. In the case of the bust, however, the results are less accurate around the eyes and the nose. This is because the normals on the actual object vary dramatically in these regions, which violates our smoothness assumption.



Here we show results that use real images. In this case, a more advanced version of the above algorithm is used, which is able to handle the changes in reflectance (albedo) over the surface. In the case of the face, the reflectances of the eyebrows and the eyes are clearly different from that of the skin region. It should be noted that even the most sophisticated shape from shading algorithm can only handle diffuse objects and not highly specular ones, such as polished metals.



We conclude with a few examples related to shading illusions. Here, we see a large green disc with a small fuzzy blueish patch in the center. If we fixate our eyes on any one point in the disc, after a few seconds, the blue patch fades away, and we only see the large green disc. It turns out that the human visual system does not measure the color at each pixel, independently, but instead detects color changes over space and time. Generally, when we look at an object, our eyes move around, or scan, the object. The eye movements cause local changes in color as a function of time. In this example, our eye movements enable us to perceive the subtle change in color between the small blue patch and the larger green disc. However, when we fixate on a single point, the subtle color differences between the small patch and the large disc are no longer detected, causing the patch to vanish.



Here we have the checker shadow illusion created by Adelson. In slide 34, patch A appears much darker than patch B. However, if we look at the two patches when the rest of the scene is hidden (slide 35), we realize that they have exactly the same brightness. What is going on here? Our visual system likes to assume that the illumination of a scene varies gradually. It first estimates this gradual change in illumination (due to the shadow cast by the cylinder in slide 34, for instance), and then normalizes the scene for this varying illumination to perceive the reflectance of each scene point. As a result, despite the fact that the two patches have equal brightness, we are able to perceive A as being made of darker material than B.

References: Textbooks

Robot Vision (Chapter 11)
Horn, B. K. P., MIT Press

37

References: Papers

[Ramachandran 1990] V. S. Ramachandran. "Perceiving shape from shading." Scientific American magazine. 1990.

[Ikeuchi 1981] K. Ikeuchi and B. K. P. Horn. "Numerical Shape from Shading and Occluding Boundaries." Artificial Intelligence, 1981.

38

Image Credits

- I.1 V.S. Ramachandran. Used with permission.
- I.2 Associated Press. WorldWide Photos, 1972.
- I.3 Vilayanur Ramachandran. Used with permission.
- I.4 Vilayanur Ramachandran. Used with permission.
- I.5 Vilayanur Ramachandran. Used with permission.
- I.6 Vilayanur Ramachandran. Used with permission.
- I.7 Vilayanur Ramachandran. Used with permission.
- I.8 Vilayanur Ramachandran. Used with permission.
- I.9 Mubarak Shah. Used with permission.
- I.10 Mubarak Shah. Used with permission.
- I.11 Edward H. Adelson. Used with permission.

39

Acknowledgements: Thanks to Joel Salzman, Kevin Chen, Ayush Sharma and Nikhil Nanda for their help with transcription, editing and proofreading.

References

[Horn 1986] B. K. P. Horn, Robot Vision, MIT Press, 1986.

[Ramachandran 1990] V. S. Ramachandran, Perceiving shape from shading, Scientific American magazine, 1990.

[Ikeuchi 1981] K. Ikeuchi and B. K. P. Horn, Numerical Shape from Shading and Occluding Boundaries, Artificial Intelligence, 1981.