

How Shading Constrains Surface Patches without Knowledge of Light Sources*

Benjamin Kunsberg[†] and Steven W. Zucker[†]

Abstract. Shape-from-shading (SFS) is a classical inverse problem in computer and human vision. This shape reconstruction problem is inherently ill-posed. We show that the isophotes on smooth surfaces with Lambertian reflectance can be directly related to surface properties without consideration of the light source. Using techniques from modern differential geometry, we derive relationships between the curvature of the isophotes and the shape operator for the surface. Neurobiology motivates the geometric approach, and our calculations allow us to characterize the matching local family of surfaces that can result from any given shading patch. We illustrate the local ambiguity in several examples.

Key words. shape from shading, covariant derivatives, orientation fields, isophotes, highlights

AMS subject classifications. 68T45, 92-08, 53B05

DOI. 10.1137/13092647X

1. Introduction. Michelangelo was a master at producing drawings that provide a rich impression of three-dimensionality through his use of shading. The problem of shape-from-shading (SFS) is to invert this procedure: find the surface that best explains a given shaded image. A dilemma arises when we try to explain this in computer vision terms, where research has shown that SFS is ill-posed and underconstrained. The practice in computer vision is to posit a model, for example a Lambertian surface with a known light source, to make the inverse problem better posed. For example, under a light attenuation assumption [36], the problem can even be made well-posed. Returning to perception, however, there is mounting evidence against specifying the light source first, and there is no neurobiological evidence (to our knowledge) that the early visual system is seeking to infer light sources before surfaces.

We seek a way out of this dilemma by asking: *What can be known about the surface without specifying the light source?* As we will show, it is still possible to infer precise constraints on the local surface curvatures when the light source is unknown. We believe this will be helpful in developing a global reconstruction model without ever explicitly defining the light source; for now, we work at just the local level. Below, we sketch a framework for this in a manner that could inform models of how the human visual system solves SFS. To start, this implies an abstraction of the image information into a shading flow, and it leads to a new geometrical approach to the shape inference problem. These steps are expanded below, before deriving our main result: For Lambertian image formation, information about surface curvatures can be calculated from the shading *independently* of any assumption about where the light source(s)

*Received by the editors June 26, 2013; accepted for publication (in revised form) January 13, 2014; published electronically April 24, 2014. This research was supported by NSF, AFOSR, NIH, and the Allen Foundation.

<http://www.siam.org/journals/siims/7-2/92647.html>

[†]Applied Mathematics, Yale University, New Haven, CT 06520-8285 (benjamin.kunsberg@yale.edu, steven.zucker@yale.edu).

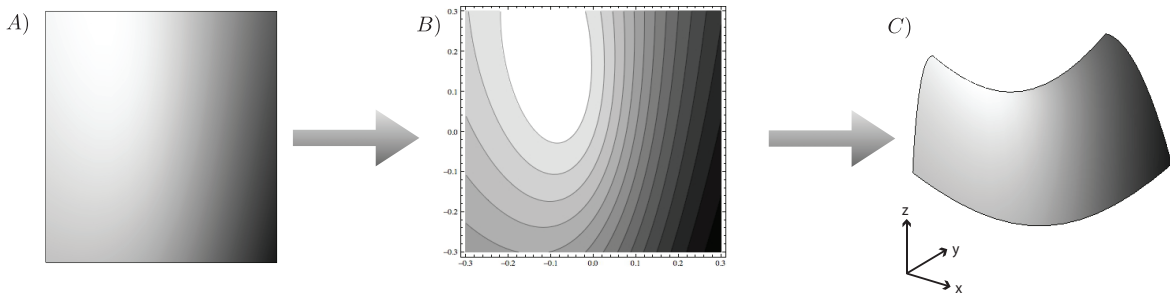


Figure 1. Classical methods attempt to go from pixel values of the image (A) to the surface (C). In this work, we use the intermediate representation of the isophote structure (B). Tangent vectors to these isophotes could be computed by biological V1 models, and comparing the tangent fields allows us to use the mathematical machinery of vector fields.

are located. Our calculations also reveal the ambiguity inherent in geometrical computations when the light source is unknown.

1.1. Neurobiological motivation. Throughout the past few decades, there has been a revolution in neurobiology showing that the visual cortex is organized around orientation. Due to the orientation-tuning of cells of the V1 area of visual cortex [21], we focus on understanding how these sets of orientations could correspond to three-dimensional (3D) surfaces. This suggests what at first glance appears to be a small problem change: instead of seeking the map from images to surfaces (and light sources), the image should first be lifted into an orientation-based representation; see Figure 1. This lift can be accomplished by considering the image isophotes [26, 27]; the lift is then the tangent map to these isophotes. This approach has arisen earlier in computer vision, and is called the shading flow field [5]. A significant body of evidence is accumulating that such orientation-based representations underlie the perception of shape [16, 17], but to our knowledge no one has previously formulated the surface inference problem from it.

Since the shading flow field could be computed in V1 (Figure 2), we have built our approach to Lambertian SFS directly on the information available in visual cortex. Our computations could be implemented by a combination of feedforward and feedback projections, supplemented with the long-range horizontal connections within each visual area. Here we concentrate on developing the math. The calculations are derived in differential-geometric terms, and a crisp curvature structure emerges from the equations; see Figure 3. As such, the latter serves as the foundation of a model for understanding feedforward connections to higher levels (surfaces) from lower levels (flows).

1.2. Local ambiguity. One of the major obstacles to solving SFS is the inherent ambiguity. The equations we will derive will allow us to analyze the local ambiguity in a surface patch when the light source is unknown. We will show that, in the generic second-order surface case, there are four local surfaces that can match the intensity information up to second order. Understanding this ambiguity should guide us in making the correct assumptions when stitching these local patches together.

For many recent reconstruction methods, the ambiguity is reduced immediately using

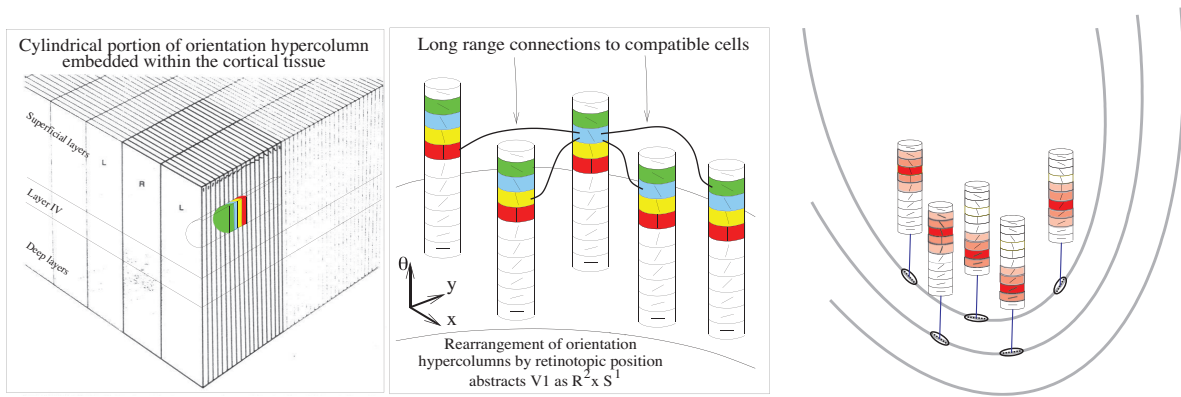


Figure 2. V1 mechanisms applied to the isophote curves result in a shading flow field. (left) Visual cortex contains neurons selective for the local orientation of image patches. In a shading gradient these will respond most strongly to the local isophote orientation, i.e., its tangent. A tangential penetration across V1 yields a column of cells, further described in the next panels. (middle) Abstracting the column of cells over a given (retinotopic) position spans all orientations; different orientations at nearby positions regularize the tangent map to the isophotes and reduce noise in local measurements. (right) Illustrating the lift of the isophotes into a V1 style representation [4]. The mathematical analysis in this paper extends this type of representation to the surface inference problem. As such it could be implemented by similar cortical machinery in higher visual areas.

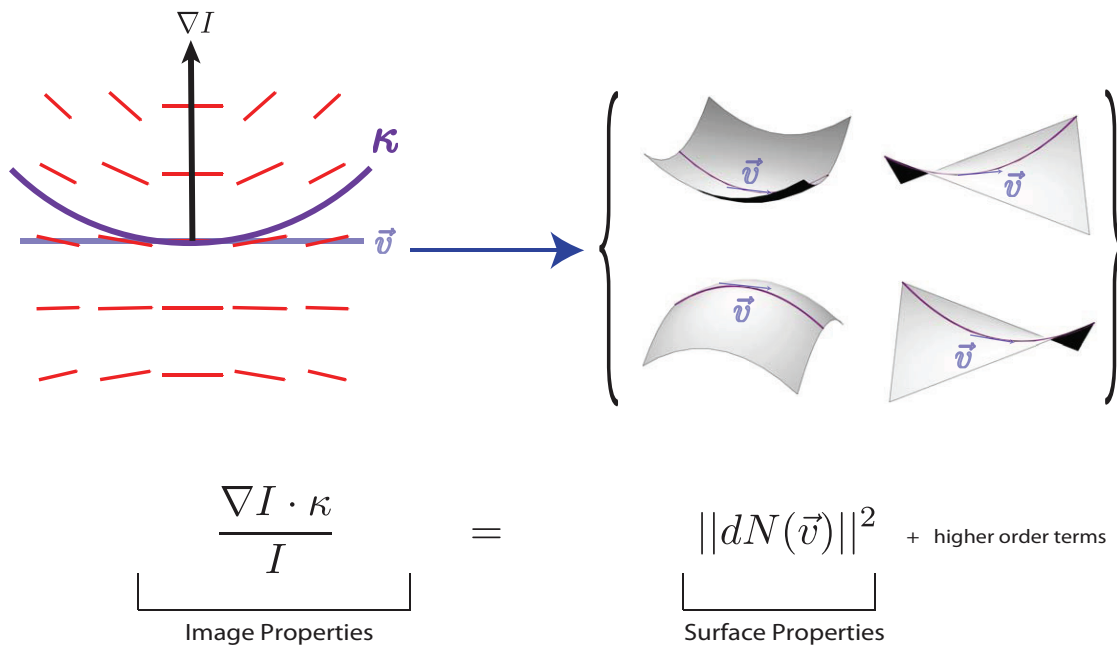


Figure 3. Our main result is that, without knowing the light source, we can still extract surface properties from local image patches. If we assume the surface is locally quadratic, then we get an elegant relation between the curvature of the isophotes (κ), the brightness gradient (∇I), and the curvatures of the surface ($dN(\vec{v})$). However, ambiguity in the possible surface remains; even for a simple situation, up to four surfaces can arise. This example and the general situation are developed fully in section 4.

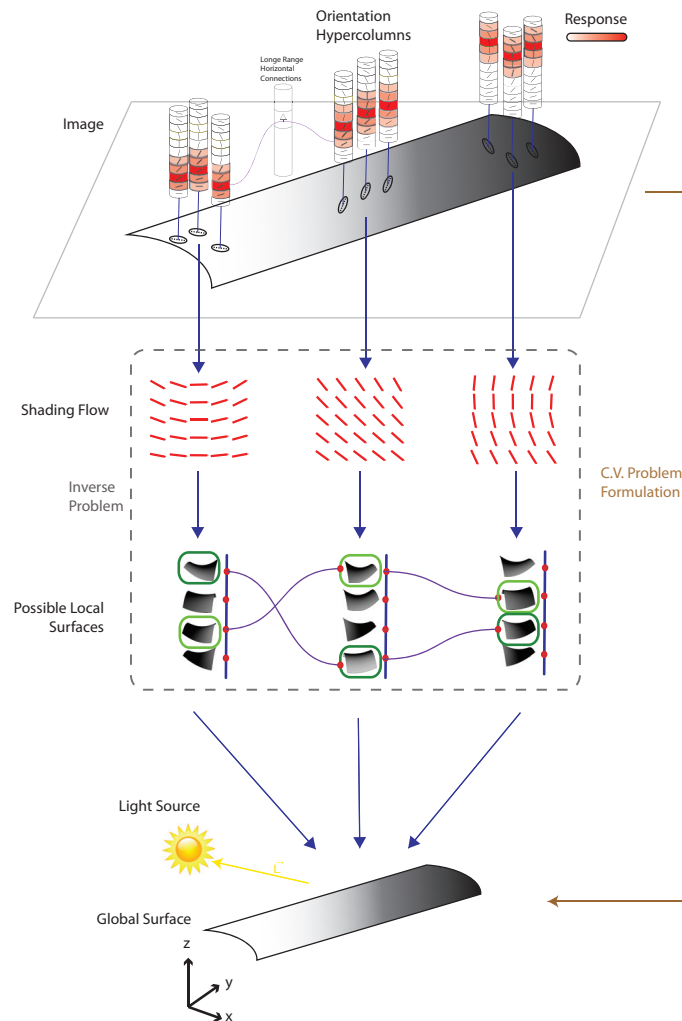


Figure 4. This figure illustrates the “big picture” for our approach to SFS. Instead of inferring surfaces directly from the image, we impose two fiber bundles between them. The first is the lift of the image into the shading flow, and the second defines the fiber of possible surface patches that are consistent with a given patch of shading flow. We do not assume a light source position, but instead will use assumptions on certain local features to restrict the ambiguity. This amounts to finding a section through the bundle of possible local surfaces. The light source position(s) are then an emergent property.

heuristics or various priors [18, 1]. However, we believe that some of these additional constraints are unnecessary and possibly conflicting. By investigating the precise ambiguity of the local surfaces, we hope to arrive at the minimal assumptions that are needed to resolve the SFS problem.

1.3. Overview of approach. We summarize our approach in Figure 4. Rather than attempting to infer a surface directly from the image and (e.g.) global light source priors, we think of the surface as a composite of local patches (charts) [14]. Each patch is described by its (patch of) shading flow, each of which implies a space of (surface patch, light source) pairs.

Much of the formal content of this paper is a way to calculate these pairs.

There are several advantages to this approach. First, it allows us to avoid making hard constraints at the local level. Rather, we seek to allow the local fibers to refine themselves using compatibility metrics. Certain patches near boundaries [38] or highlights are less ambiguous than others and will constrain their nearby patches without explicit constraints. Thus our approach is consistent with Marr's principle of least commitment. Second, the light source positions are essentially an emergent property rather than a prior assumption. Once the surface has been solved for, the light sources are trivial to calculate via the brightness gradient. By not having to estimate the light source, we hope to gain a robustness to the light conditions and remove the error commonly gained from estimating them.

1.4. Comparisons to previous work. Horn's original algorithm [7] solved a first-order partial differential equation (P.D.E). It required a known reflectance map (or very strong conditions on it) and known normals along some closed surface curve. Under constant albedo, the known reflectance map is equivalent to a known light source direction. In contrast, our local method requires none of this information a priori.

We emphasize that Horn's method calculates characteristic curves independently. However, there is information to be exploited among nearby characteristic curves; thus, a method that calculates local surface shape over a 2D image patch (such as ours) will need less initial information. This insight is key to the integral minimization approaches (e.g., [18, 37, 20, 23]). These are often supplemented with a geometric regularization term or require a unit area assumption [18]. We require no such assumption; instead, we will have to parametrize the ambiguity resulting from unknown scene parameters. For more information on these various approaches, see [41].

While integral approaches work over an area, they still directly analyze the image. However, using the shading flow as an intermediate representation provides a different basis for regularization. Rather than focusing on accurate pixel measurement, we focus on accurately estimating orientations to use as input. But in addition to the biological motivation for using the shading flows, there is also very useful mathematical machinery that can be applied once we consider the data as a vector field rather than just pixel values. Therefore, we will think of SFS as a map from vector fields onto "shape space."

There has been much work investigating the conditions when the SFS P.D.E. system is unambiguous [10, 13, 34, 36, 2, 3]. In particular, Prados and Faugeras [36] consider the well-posedness of the SFS problem. In a slightly modified SFS model [36], they show that a unique surface exists and give a provably convergent method towards that unique solution. They require two additional assumptions: a light attenuation factor and the light source at the optical center. To our knowledge, their work represents the first time that a version of an SFS model has been essentially solved. Analysis of their model has also been extended to perspective projection, although ambiguity now returns [6].

A global ambiguity that can result from shading, the bas-relief ambiguity, has been studied by Belhumeur and colleagues [2, 3]. However, the ambiguity depends on a pointwise-defined albedo. In this paper, we will consider a constant albedo on a shading patch, so the ambiguity is reduced to only a concave/convex one. But rather than considering global ambiguity, we study the ambiguity of a shaded image patch in defining a local Taylor surface approximation.

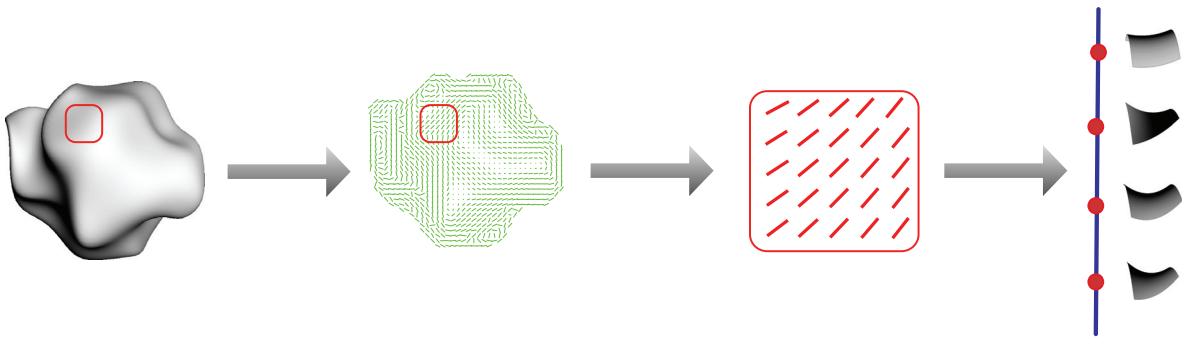


Figure 5. This figure represents the workflow going from image to shading flow to a set of local surfaces. Each surface along the fiber needs a particular light source position to correspond to the given shading flow. In section 4, we solve this particular problem for all second-order surfaces.

This is close in spirit to the line of investigation developed by Koenderink and colleagues [25, 27].

Pentland has also focused on estimates of local SFS [35]. He classified surfaces into general categories (plane, cylinder, saddle, etc.) based on the signs of the second derivatives of the intensity at a point. However, this is a broad classification; there is clearly more information to be used than just the sign. He also considered a simplified “uncalibrated” SFS problem of solving for shape with unknown light source [35]; to make this well-posed, he required a local spherical assumption. In this paper, we also use information contained in the second derivatives of intensity, although we prove the exact correspondence to the surface curvatures rather than make the spherical assumption.

1.5. The shading flow. A smooth surface patch under diffuse Lambertian lighting and orthogonal projection yields smooth image curves of constant brightness. The shading flow derives from these level curves (isophotes) of image intensity $I(x, y)$. To construct the *shading flow field* $V(x, y)$, we simply consider the isophote tangents at each pixel. Of course, these tangents are perpendicular to the image intensity gradient at each point. Our goal is to use this 2D vector field to restrict the family of (surface, light source direction) pairs that could have resulted in the image (Figure 5). In addition, we will use the complementary vector field of brightness gradients. Note that the vector field of brightness gradients (along with some pixel boundary values) can be used to recreate the pixel values of the image via gradient reconstruction.

Due to space limitations, we cannot analyze the shading flow here. For computational work on regularizing and calculating the shading flow, see [5]. We simply remark that (i) it regularizes certain errors due to noise in images, and (ii) it is invariant under some transformations of the albedo and surface [28].

1.6. Problem statement. Consider a small image patch defined by Lambertian reflectance with an unknown parallel light source \vec{L} and unknown albedo ρ . (The analysis is the same if \vec{L} is a linear combination of light sources.) We get the following relation between the apparent brightness of our surface and the scene parameters:

$$I(x, y) = \rho \vec{L} \cdot N(x, y).$$

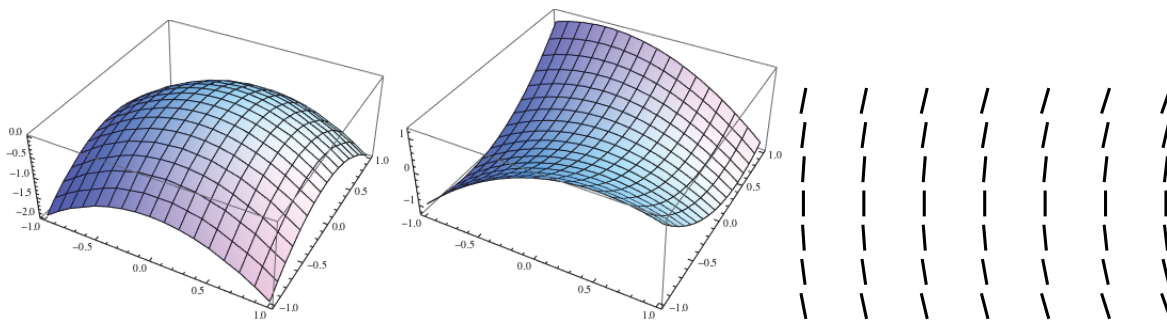


Figure 6. The shading flow field abstracts the image isophotes into a vector field. This has a biological basis (shading flow could be represented in V1 and V2 [21]; surfaces in V4 and IT [40]). We base the surface inference problem on the flow field, and not on the raw image. Here we illustrate that two surfaces (in this instance, a sphere and a saddle) can correspond to the same shading flow, provided the light source changes appropriately. For the sphere, we need a light source at $(1, 0, 1)$; for the saddle, we need a light source at $(1, 1, 1)$. Note that there are two other local surfaces that can also correspond to this flow; apply the concave/convex ambiguity [15, 35] to each surface. It can be proved that, for the problem considered in this paper, this four-fold ambiguity is maximal.

An image patch corresponds to a local surface patch, which by Taylor's theorem we will represent through the map $X(x, y) = (x, y, f(x, y))$ with $f(x, y) = c_0 + c_1x + c_2y + c_3x^2 + c_4xy + c_5y^2 + c_6x^3 + c_7x^2y + c_8xy^2 + c_9y^3$.

Our goal is to understand the derivatives of intensity in terms of the coefficients $\{c_i\}$. It is essential that the order of the Taylor polynomial must be 3, since we shall consider second derivatives of image intensity and intensity is already dependent (via Lambertian lighting) on the first-order derivatives of the surface. Other analyses of SFS consider only second-order Taylor approximations [35, 14].

This leads to the following formulation.

PROBLEM STATEMENT. Assume that a smooth Lambertian surface (representable as a graph of a function) with locally constant albedo and Gaussian curvature $K \neq 0$ is lit from any finite number of unknown directions. Assume that the image is captured through orthogonal projection. Given the shading flow and brightness gradient vector fields, recover the constraints on the surface from the local isophote structure.

Recovering shape from the shading flow will always be ill-posed; see the ambiguity in Figure 6. However, at certain points on the surface, the ambiguities collapse in dimension. For example, along the boundary of a smooth object, the view vector lies in the tangent plane [25]. Along a suggestive contour [9], the dot product of the normal vector and the view vector is at a local minimum in the direction of the viewer. At a highlight under Lambertian shading, the dot product of the light source direction and normal vector are at a local maximum. All these types of points are identifiable in the image and provide additional geometric information that may reduce ambiguity locally. This leads to a plan for reconstruction: First, parametrize the shading ambiguity in the general shading case. Then, locate the points where the shading ambiguity vanishes (or reduces greatly) and solve for the local surface shape at those points. Finally, calculate the more complex regions via a compatibility technique [5] or interpolation. In this paper, we focus on just the first step. Papers corresponding to the latter stages will be forthcoming.

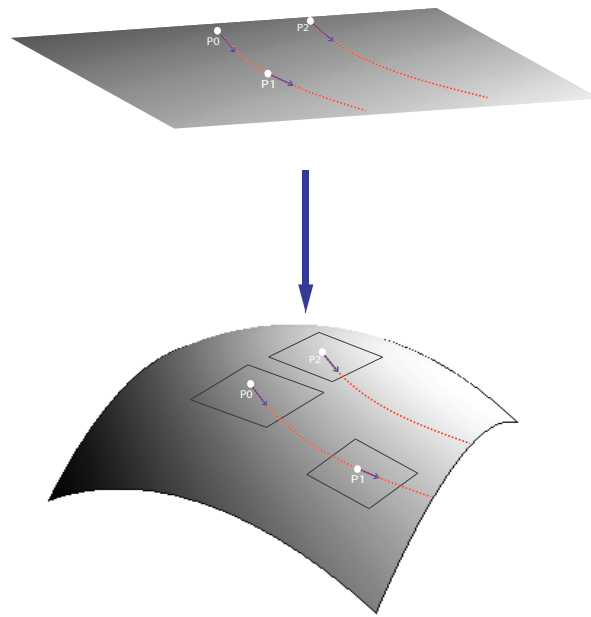


Figure 7. Geometric setting for the SFS flow problem. A surface (bottom) projects orthographically onto the image (top). Isophotes on the surface project to curves in the image. The shading flow is the tangent map to these image contours. When pulled back onto the surface, a tangent vector from the shading flow corresponds to a vector in the tangent plane to the surface. The natural operation on images is to “walk along” the shading flow, either along or across isophotes. We calculate the analogous operations on the surface to understand the conditions on surface curvature that result from changes in the shading flow.

2. Analysis.

2.1. Outline. Our goal is to translate the problem into the local tangent plane and then use the machinery of covariant derivatives and parallel transport to represent image derivatives (see Figure 7) as functions of the surface vector flows. A similar use of this machinery was applied for shape from texture in [19].

1. We write the brightness gradient and isophote as tangent plane conditions between the projected light source and shape operator.
2. We take the covariant derivative of the projected light source and show that it is independent of the direction of the light source.
3. We take the covariant derivative of the isophote condition and separate it into the differentiation of the projected light source and the differentiation of the shape operator dN .
4. We take the covariant derivative of the shape operator applied to the isophote vector. This requires several steps:
 - (a) expansion of the dN operator in terms of the Hessian H ,
 - (b) covariant differentiation of H ,
 - (c) parallel transport of H ,
 - (d) covariant differentiation of dN from covariant differentiation of H .
5. Substitution and algebra.

Table 1
Notation table.

p	A chosen point (x_0, y_0)
$I_p(x, y)$	An image patch centered at p
$\nabla I(x, y)$	The brightness gradient
$S_p(x, y)$	The corresponding (unknown) surface patch
$f(x, y)$	The Taylor approximation at p of S
$\{c_i\}$	The coefficients of the Taylor approximation $f(x, y)$
$T_p(S)$	The tangent plane of S at p
\vec{L}	The light source direction
$\vec{l}_t(p)$	The projection of the L onto the tangent plane
\vec{e}_i	Unit length standard basis vector in the direction of coordinate axis i
$N(x, y)$	The unit normal vector field of S
$V(x, y)$	The vector field of isophote directions at each point (x, y)
$\hat{v} \in T_p(S)$	The image unit length tangent vector in the direction of the isophote at p
$\hat{u} \in T_p(S)$	The image unit length tangent vector in the direction of the brightness gradient at p
$\vec{w} \in T_p(S)$	The tangent vector in direction \vec{w} of unit length in the image, expressed in the surface tangent basis
$\vec{v}[F]$	The directional derivative of a scalar valued function F
$D_{\vec{v}}V$	The directional derivative (in the image) of the vector field V in the direction \vec{v}
$\nabla_{\vec{u}}V$	The covariant derivative (on the surface) of the vector field V in the direction \vec{u}
G	The first fundamental form (also called the metric tensor)
II	The second fundamental form
H	The Hessian
dN	The differential of the Gauss map (also called the shape operator)
$\vec{a} \cdot \vec{b}$	The dot product of \vec{a} and \vec{b} in either \mathbb{R}^2 or \mathbb{R}^3
$\langle \vec{a}, \vec{b} \rangle_G$	The dot product of tangent vectors \vec{a} and \vec{b} on the tangent plane $T_p(S)$ induced by the metric G

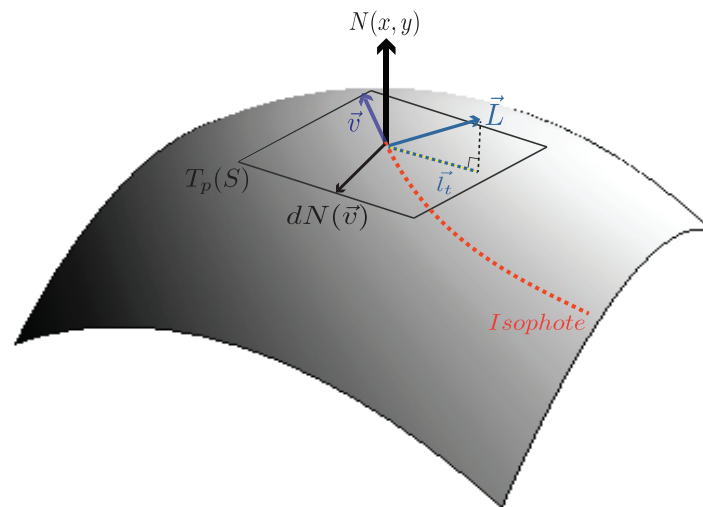


Figure 8. A diagram explaining our defined surface properties.

2.2. Notation. For reference, we define our complete notation in Table 1. Notation will be introduced throughout the analysis. For geometric intuition of important variables, see Figure 8.

We start with an open neighborhood of our image plane $U \in \mathbb{R}^2$ and a map $X : U \rightarrow S \subset \mathbb{R}^3$. We define a shading flow field $V(x, y)$ on U . We normalize $V(x, y)$ to be of unit length in the image, although the corresponding surface tangent vectors have unknown length. We denote the elements of $V(x, y)$ with hat superscripts to avoid confusion: $\hat{v} = V(x_0, y_0)$ for some (x_0, y_0) . The corresponding vectors on the surface tangent plane are defined by the image of \hat{v} under the map composition of the differential $dS : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ and the tangent plane basis change $T : \mathbb{R}^3 \rightarrow T_p(S)$. We will use the vector superscript to denote these surface tangent vectors, e.g., \vec{v} . Due to our Monge patch parametrization, if we choose $\{dN(\vec{e}_1), dN(\vec{e}_2)\}$ (as we do throughout the paper) as the basis for our $T_p(S)$, we will have the same coordinate representation for \vec{v} and \hat{v} .

That is,

$$\vec{v} = T \circ dX(\hat{v}) = \mathbb{1}\hat{v} = \hat{v}.$$

For simplicity, we will abuse the notation slightly by interchanging \hat{v} and \vec{v} when appropriate. We try to use \hat{v} when we want to think about the object as image vectors and \vec{v} as movements on a tangent plane.

2.3. Brightness gradient. We derive the equations for the brightness gradient as a function of the light source and the second fundamental form. Similar derivations (with different notation) appear in [26].

Recall that the differential of the image, dI , is defined as a linear 1-form having as input unit length image vectors \hat{w} and having as output a real number. The output is the change in brightness along a step on the surface using \hat{w} , which can also be computed via dot product with the gradient ∇I . We write

$$\begin{aligned} (2.1a) \quad dI(\hat{w}) &= \nabla I \cdot \hat{w} \\ (2.1b) \quad &= \hat{w} \left[\rho \langle \vec{L}, \vec{N} \rangle \right] \\ (2.1c) \quad &= \rho \left\langle \left(\nabla_{\vec{w}} \vec{L} \right), \vec{N} \right\rangle + \rho \left\langle \vec{L}, \left(\nabla_{\vec{w}} \vec{N} \right) \right\rangle \\ (2.1d) \quad &= 0 + \rho \langle \vec{L}, dN(\vec{w}) \rangle \\ (2.1e) \quad &= \rho \langle \vec{l}_t, dN(\vec{w}) \rangle_G \\ &= -\rho \vec{l}_t^T II \vec{w}, \end{aligned}$$

where the first term in (2.1c) is zero because the light source is fixed. To understand the last line, recall two things. First, $dN = -G^{-1}II$, where G is the first fundamental form and II is the second fundamental form [11]. Second, computation of a dot product on a tangent plane is induced by the metric G . Thus, $\langle \vec{v}_1, \vec{v}_2 \rangle_G = \vec{v}_1^T G \vec{v}_2$. This calculation shows that, once dN is solved for, it is trivial to calculate the direction of the light source.

Proposition 2.1. *The brightness gradient ∇I can be expressed as the vector $-\rho \vec{l}_t^T II$.*

Along an isophote surface curve $\alpha(t)$, the brightness is constant. Writing $\vec{v} = \alpha'(0)$, we have $\rho \langle \vec{l}_t, dN(\vec{v}) \rangle_G = -\rho \vec{l}^T II \vec{v} = \nabla I \cdot \hat{v} = 0$. Thus, we conclude the following.

Proposition 2.2. *Each isophote tangent vector \vec{v} on S is a function of the normal curvatures and light source and is defined by the equation*

$$\langle \vec{l}_t, dN(\vec{v}) \rangle_G = 0.$$

In addition, we calculate each component of the brightness gradient via dot product with \vec{e}_i :

$$(2.2) \quad I_x = \rho \langle \vec{l}_t, dN(\vec{e}_1) \rangle_G,$$

$$(2.3) \quad I_y = \rho \langle \vec{l}_t, dN(\vec{e}_2) \rangle_G.$$

2.4. Covariant derivative of projected light source. One of the major advantages of our approach is that we do not assume a known light source direction. In fact, using the covariant derivative described below, we can calculate the change in the projected light source vector without knowing its precise location!

Remark 1. Here, we briefly remark on the use of covariant derivatives for surfaces in \mathbb{R}^3 . We consider “movements” in the image plane and sync them with “movements” through the tangent plane bundle on the surface. The problem is that the image plane vectors are on a flat surface, whereas the vectors on the surface tangent planes “live” in different tangent spaces: The surface tangent planes are all different orientations of \mathbb{R}^2 in \mathbb{R}^3 . Thus, to calculate derivatives via limits of differences, we need to “parallel transport” nearby vectors to a common tangent plane. The covariant derivative achieves this. For our purposes, we think of the covariant derivative in two ways. The first definition, which we use in this section, is the expression as the composition of a derivative operator in \mathbb{R}^3 and a projection operator onto a tangent plane. This is an *extrinsic definition*—it is a definition that requires use of the ambient space. The second definition, which we use in Appendix A.2, will be in terms of parallel transport.

We exploit the structure in \vec{l}_t : It is the result of a projection from a fixed vector \vec{L} down into the tangent plane $T_p(S)$. Thus, the change in \vec{l}_t just results from changes in the tangent plane, which are dependent on the surface curvatures and not on \vec{L} . Importantly, we avoid having to represent \vec{L} in our calculations by considering only its projected changes. We now show this rigorously.

Lemma 2.3. *The covariant derivative of the projected light source is dependent on the position of the light source only through the observed intensity. Thus,*

$$\nabla_{\vec{u}} \vec{l}_t = - \left(\frac{I(p_0)}{\rho} \right) dN(\vec{u}).$$

Proof. Let Π_{p_0} be the projection operator taking a vector in 3-space onto the tangent plane of S at p_0 . Recall that the covariant derivative of a tangent vector can be expressed as

the composition of a derivative operator and Π_{p_0} :

$$(2.4a) \quad \nabla_{\vec{u}} \vec{l}_t = \Pi_{p_0} \left(\frac{d\vec{l}_t}{dt} \right)$$

$$(2.4b) \quad = \Pi_{p_0} \left(\frac{d}{dt} (\vec{L} - (\vec{L} \cdot \vec{N}) \vec{N}) \right)$$

$$(2.4c) \quad = \Pi_{p_0} \left(\frac{d\vec{L}}{dt} - \frac{d}{dt} [(\vec{L} \cdot \vec{N}) \vec{N}] \right)$$

$$(2.4d) \quad = \Pi_{p_0} \left(0 - \frac{d}{dt} [\vec{L} \cdot \vec{N}] \vec{N} - (\vec{L} \cdot \vec{N}) \frac{d\vec{N}}{dt} \right)$$

$$(2.4e) \quad = \Pi_{p_0} \left(- \left[\frac{d\vec{L}}{dt} \cdot \vec{N} + \vec{L} \cdot \frac{d\vec{N}}{dt} \right] \vec{N} - (\vec{L} \cdot \vec{N}) dN(\vec{u}) \right)$$

$$(2.4f) \quad = -(\vec{L} \cdot \vec{N}) dN(\vec{u})$$

$$(2.4g) \quad = - \left(\frac{I(p_0)}{\rho} \right) dN(\vec{u}).$$

To go from (2.4e) to (2.4f), we note that the first term is just a scalar multiplied by \vec{N} , whereas the second term lies completely in the tangent plane. Thus, when we apply the projection operator Π_{p_0} , the first term is lost and the second term is kept unaffected. ■

The fact that this change in the projected light source does not contain an \vec{L} allows us to remove the light source dependence from the second derivatives of intensity.

2.5. Covariant derivative of the isophote condition. We now use the changes in the brightness gradient and the isophote directions to restrict our surface parameters. Let \hat{v} be the unit length image vector in the direction of the isophote at an arbitrary point p . Let \hat{u} be the unit length image vector in the direction of the brightness gradient at p . In the image, $\hat{v} \perp \hat{u}$, but the projected vectors on the surface \vec{v} and \vec{u} may not be orthogonal (w.r.t. \mathbb{R}^3) on the tangent plane at p .

Considering these particular changes in \hat{u} and \hat{v} is equivalent to choosing a basis. This will result in solving for equations of the three second derivatives $\{D_{\hat{v}}\hat{v}, D_{\hat{u}}\hat{v}, D_{\hat{u}}\hat{u}\}$, although we could have considered the changes in $\{I_x, I_y\}$ and instead solved for $\{I_{xx}, I_{xy}, I_{yy}\}$. However, the equations simplify when choosing the basis defined by the isophote and brightness gradient.

To emphasize the conceptual picture, we will derive the $D_{\hat{v}}\hat{v}$ equation here and save the equations in the general case for the appendix. We start by first calculating I_v and then taking the directional derivative of I_v in the direction \hat{v} . From section 2.3, we can write

$$(2.5) \quad 0 = I_v = \nabla I \cdot \hat{v} = \rho \langle \vec{l}_t, dN(\vec{v}) \rangle_G.$$

Applying the scalar directional derivative with respect to \hat{v} on both sides and using the result

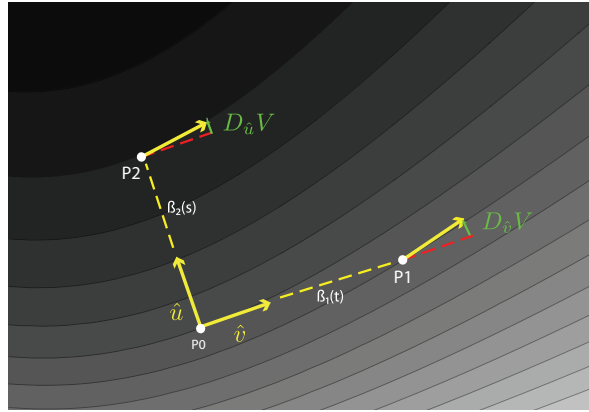


Figure 9. A diagram explaining our use of the shading flow field. As we move on $\beta_1(t)$ in the direction of the image isophote \hat{v} from P_0 to P_1 , the flow field $V(x, y)$ changes by $D_{\hat{v}}V$. Similarly, we may move in direction \hat{u} along $\beta_2(s)$, which is perpendicular (in the image) to the isophote. Then, our flow field changes by $D_{\hat{u}}V$. We want to relate these changes in closed form to the curvatures of the surface.

from (2.4f) (see Figure 9), we get

$$(2.6a) \quad 0 = \hat{v} \left[\rho \langle \vec{l}_t, dN(\vec{v}) \rangle_G \right]$$

$$(2.6b) \quad = \rho \langle \nabla_{\vec{v}} \vec{l}_t, dN(\vec{v}) \rangle_G + \rho \langle \vec{l}_t, \nabla_{\vec{v}}(dN(\vec{v})) \rangle_G$$

$$(2.6c) \quad = \rho \langle -(\vec{L} \cdot \vec{N}) dN(\vec{v}), dN(\vec{v}) \rangle_G + \rho \langle \vec{l}_t, \nabla_{\vec{v}}(dN(\vec{v})) \rangle_G$$

$$(2.6d) \quad = -I \langle dN(\vec{v}), dN(\vec{v}) \rangle_G + \rho \langle \vec{l}_t, \nabla_{\vec{v}}(dN(\vec{v})) \rangle_G.$$

We now unpack $\langle \vec{l}_t, \nabla_{\vec{v}}(dN(\vec{v})) \rangle_G$, which requires a technical computation using parallel transport and tensor algebra. On first reading, some readers may wish to skip directly to the main results in section 3.

2.6. Covariant derivative of the shape operator dN . We expand the rightmost term of (2.6d). This will also expand into several terms; we will analyze each separately. Although $dN(v)$ is an unknown vector, we want to understand its covariant derivative $\nabla_{\vec{u}}(dN(\vec{v}))$ as a function of surface changes $\nabla_{\vec{u}}(dN)$ and isophote changes $\nabla_{\vec{u}}\vec{v}$. We apply the chain rule to get

$$(2.7) \quad \nabla_{\vec{v}}(dN(\vec{v})) = (\nabla_{\vec{v}}dN)(\vec{v}) + dN(\nabla_{\vec{v}}\vec{v}).$$

We now focus on the first term of this expression.

2.6.1. Expansion of dN in terms of the Hessian H . Note that dN is a $(1, 1)$ tensor, and thus we need to be careful when taking its covariant derivative. Recall that the matrix representation of dN is $-G^{-1}II$ and that raising and lowering the tensor characteristic commutes with covariant differentiation:

$$(2.8a) \quad (\nabla_{\vec{v}}dN)(\vec{v}) = -(\nabla_{\vec{v}}G^{-1}II)(\vec{v})$$

$$(2.8b) \quad = -(G^{-1}\nabla_{\vec{v}}II)(\vec{v}).$$

We now write each of the normal components as n_i , so $\vec{N} = \{n_1, n_2, n_3\}$. Note that, due to our Monge patch representation of $S(x, y)$, $\{\vec{f}_{xx}, \vec{f}_{xy}, \vec{f}_{yy}\}$ are nonzero only in their third component, e.g., $\vec{S}_{xx} = \{0, 0, f_{xx}\}$. Recall the definition of the second fundamental form II :

$$(2.9a) \quad II = \begin{bmatrix} \vec{N} \cdot \vec{f}_{xx} & \vec{N} \cdot \vec{f}_{xy} \\ \vec{N} \cdot \vec{f}_{xy} & \vec{N} \cdot \vec{f}_{yy} \end{bmatrix}$$

$$(2.9b) \quad = \begin{bmatrix} n_3 f_{xx} & n_3 f_{xy} \\ n_3 f_{xy} & n_3 f_{yy} \end{bmatrix}$$

$$(2.9c) \quad = n_3 \begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix}.$$

For notational convenience, we use the Hessian $H = \begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix}$. Substituting into (2.8b) and using the appropriate product rule, we obtain

$$(2.10a) \quad (\nabla_{\vec{v}} dN)(\vec{v}) = -G^{-1} \nabla_{\vec{v}} (n_3 H)(\vec{v})$$

$$(2.10b) \quad = -n_3 G^{-1} \nabla_{\vec{v}} (H)(\vec{v}) - \vec{v}[n_3] G^{-1} H(\vec{v}),$$

where $\vec{v}[n_3]$ denotes the directional derivative of the third component of \vec{N} in the direction \vec{v} . Now, we consider the dot product with \vec{l}_t , as in (2.6d). The second term of (2.10b) will be exactly zero because $H = n_3 II$ and $\vec{l}_t^T II \vec{v} = 0$.

$$(2.11a) \quad \langle \vec{l}_t, (\nabla_{\vec{v}} dN)(\vec{v}) \rangle_G = \left\langle \vec{l}_t, -n_3 G^{-1} \nabla_{\vec{v}} (H)(\vec{v}) - \vec{v}[n_3] G^{-1} H(\vec{v}) \right\rangle_G$$

$$(2.11b) \quad = -n_3 (\nabla_{\vec{v}} H)(\vec{v}, \vec{l}_t) - \vec{v}[n_3] H(\vec{v}, \vec{l}_t)$$

$$(2.11c) \quad = -n_3 (\nabla_{\vec{v}} H)(\vec{v}, \vec{l}_t).$$

For (2.11b), we replaced $\langle \vec{l}_t, \nabla_{\vec{v}} H(\vec{v}) \rangle$ by $(\nabla_{\vec{v}} H)(\vec{v}, \vec{l}_t)$. This is appropriate tensor notation, as $\nabla_{\vec{v}} H$ is a $(0, 2)$ tensor. It remains to calculate $\nabla_{\vec{v}} H$.

Proposition 2.4. *Let $\beta(s)$ be a curve with $\vec{v} = \beta(0)$. Let τ_s be the parallel transport operator, and define, for an arbitrary tangent vector \vec{w} , the parallel transport adjustment vector $\vec{T}_w \in \mathbb{R}^2$.*

$$(2.12) \quad \vec{T}_w = \lim_{s \rightarrow 0} \frac{\tau_s^{-1}(\vec{w}(\beta(s))) - \vec{w}(\beta(0))}{s}.$$

Then, for a pair of arbitrary tangent vectors \vec{w}_1 and \vec{w}_2 ,

$$(2.13) \quad (\nabla_{\vec{v}} H)(\vec{w}_1, \vec{w}_2) = \vec{T}_{w_2}^T H \vec{w}_1 + \vec{w}_2^T H \vec{T}_{w_1} - \vec{w}_2^T \vec{v}[H] \vec{w}_1.$$

Proof. As this is a dense calculation, we have left the proof to Appendix A.

2.7. Covariant differentiation of dN from covariant differentiation of H . Now, we substitute (2.13) into (2.11c) to finish the expansion of the initial equation (2.6d):

$$(2.14a) \quad \langle \vec{l}_t, (\nabla_{\vec{v}} dN)(\vec{v}) \rangle_G = -n_3 (\nabla_{\vec{v}} H)(\vec{v}, \vec{l}_t)$$

$$(2.14b) \quad = -\vec{T}_{l_t}^T II \vec{v} - \vec{l}_t^T II \vec{T}_v + n_3 \vec{l}_t^T (\vec{v}[H]) \vec{v}$$

$$(2.14c) \quad = -\vec{l}_t^T II \vec{T}_v + n_3 \vec{l}_t^T (\vec{v}[H]) \vec{v},$$

where the first term of (2.14b) is proportional to $\vec{l}_t I I v$, which is 0 by Proposition 2.2. Now, provided the Gaussian curvature $K \neq 0$, we can invert the Hessian. Thus, $\rho n_3 \vec{l}_t^T = \rho n_3 \vec{l}_t^T H H^{-1} = \rho \vec{l}_t^T I I H^{-1} = -(\nabla I)^T H^{-1}$, so

$$\rho \langle \vec{l}_t, (\nabla_{\vec{v}} dN)(\vec{v}) \rangle_G = -\rho \vec{l}_t^T I I \vec{T}_v - (\nabla I)^T H^{-1} \vec{v} [H] \vec{v}.$$

2.8. Putting it all together. Substituting the above equation into (2.7), we get

$$(2.15a) \quad \rho \langle \vec{l}_t, (\nabla_{\vec{v}} dN)(\vec{v}) \rangle_G = \rho \langle \vec{l}_t, (\nabla_{\vec{v}} dN)(\vec{v}) + dN(\nabla_{\vec{v}} \vec{v}) \rangle_G$$

$$(2.15b) \quad = -\rho \vec{l}_t^T I I \vec{T}_v - (\nabla I)^T H^{-1} \vec{v} [H] \vec{v} + \rho \langle \vec{l}_t, dN(\nabla_{\vec{v}} \vec{v}) \rangle_G$$

$$(2.15c) \quad = -\rho \vec{l}_t^T I I \vec{T}_v - (\nabla I)^T H^{-1} \vec{v} [H] \vec{v} + \rho \langle \vec{l}_t, dN(D_{\hat{v}} \hat{v} - \vec{T}_v) \rangle_G$$

$$(2.15d) \quad = -(\nabla I)^T H^{-1} \vec{v} [H] \vec{v} - (\rho \vec{l}_t^T I I) \cdot D_{\hat{v}} \hat{v}$$

$$(2.15e) \quad = -(\nabla I)^T H^{-1} \vec{v} [H] \vec{v} + \nabla I \cdot D_{\hat{v}} \hat{v},$$

where we have used the fact that a covariant derivative is the sum of the changes due to parallel transport and the coordinate changes $D_{\hat{v}} \hat{v}$ along the curve $\beta(s)$ in (2.15c). This is the shading flow change, as seen in Figure 9.

Plugging into (2.6d) and rearranging, we get

$$\nabla I \cdot D_{\hat{v}} \hat{v} = I \|dN(\vec{v})\|^2 - (\nabla I)^T H^{-1} (\vec{v} [H]) \vec{v}.$$

3. Shading equations. We have now computed the directional derivative of the vector \hat{v} in the direction \hat{v} . For an arbitrary point p , let \hat{u} be the image vector of unit length in the direction of the brightness gradient. Then, we can repeat this calculation for the directional derivative of \hat{v} in the direction \hat{u} . In addition, we can calculate the directional derivative of the vector \hat{u} in the direction \hat{u} . (Both of these proofs are similar to the one above and are left to Appendix A.) This gives us a total of three equations equating the second-order intensity information (as represented in vector derivative form on the left-hand side) directly to surface properties.

Theorem 3.1. *For any point p in the image plane, let $\{\hat{u}, \hat{v}\}$ be the local image basis defined by the brightness gradient and isophote. Let I be the intensity, ∇I be the brightness gradient, $f(x, y)$ be the height function, H be the Hessian, and dN be the shape operator. Then, the following equations hold, regardless of the light source direction, as long as $\det(dN) \neq 0$:*

$$(3.1) \quad \nabla I \cdot D_{\hat{v}} \hat{v} = I \|dN(\vec{v})\|^2 - (\nabla I)^T H^{-1} (\vec{v} [H]) \vec{v},$$

$$(3.2) \quad \nabla I \cdot D_{\hat{u}} \hat{v} = I \langle dN(\vec{v}), dN(\vec{u}) \rangle_G + \|\nabla I\| \frac{\nabla f^T I I(\vec{v})}{\sqrt{1 + \|\nabla f\|^2}} - (\nabla I)^T H^{-1} (\vec{u} [H]) \vec{v},$$

$$(3.3) \quad I_{uu} = -I \|dN(\vec{u})\|^2 - 2 \|\nabla I\| \frac{\nabla f^T I I(\vec{u})}{\sqrt{1 + \|\nabla f\|^2}} + (\nabla I)^T H^{-1} (\vec{u} [H]) \vec{u}.$$

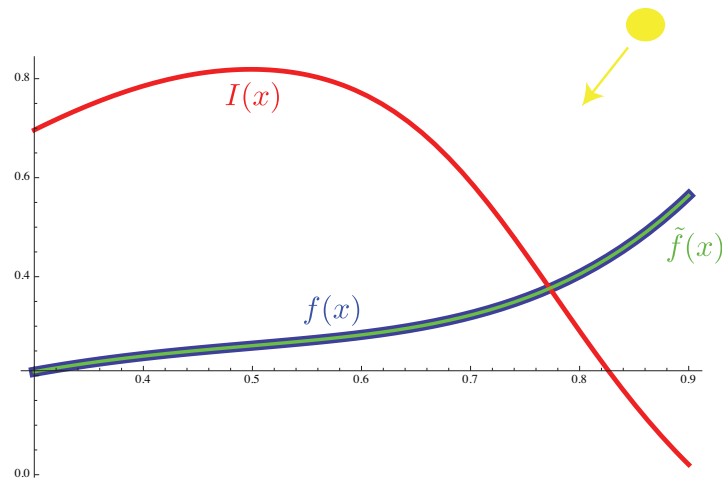


Figure 10. The SFS flow problem in one dimension. The blue curve is the 1D surface, which has the equation $\sin(x) + x^2 + x^5$. The red curve is the intensity function for the 1D setup, given a light source of norm 1 and direction as shown. The green curve (overlaid) is the recovered surface $f(x)$ using our P.D.E. formulation from Corollary 4.1 below, given three boundary conditions. As you can see, the reconstruction is indistinguishable from the correct surface.

These equations are novel; we call them *the second-order shading equations*. Note that the left-hand side of (3.3) has an I_{uu} rather than $\nabla I \cdot D_{\hat{u}} \hat{u}$, which must always be equal to 0. Note that there is no dependence on the light source; thus, these equations directly restrict the derivatives of our local surface patch without knowledge of the light source. We have included in the appendix the expanded versions of these equations in terms of those derivatives.

4. Applications of the shading equations. Below, we investigate local surface ambiguity via these second-order shading equations. We consider four cases, arranged from simple to complicated:

1. In the 1D case, these equations can be solved directly in a P.D.E. formulation to recover the curve exactly.
2. A second-order surface assumption, with a frontal-parallel tangent plane, leads to four explicit solutions.
3. A second-order surface assumption with arbitrary tangent planes can be solved implicitly.
4. “Critical” points and curves in the image have reduced shading ambiguity in the general case.

4.1. Case 1: 1D SFS. For simplification and to build intuition, we consider the problem of SFS in one dimension: Given a 1D intensity function $I(x)$, solve for the smooth curve $f(x)$ corresponding to $I(x)$ under Lambertian shading. Although this problem can be solved with other means, we use it to illustrate the P.D.E. approach to solving these shading equations. In this example, we build the intensity function using an unknown light source and recover the shape exactly using our second-order shading equations. We treat our equation as a P.D.E. and solve it numerically. See Figure 10 for the problem setup.

4.1.1. P.D.E. formulation. On the curve $f(x)$, the point sets of constant brightness are now single points rather than isophote curves. Thus, we cannot talk about $D_{\hat{v}}\hat{v}$ and I_{uu} . However, $D_{\hat{u}}\hat{u}$ still makes sense, as \vec{u} is now defined as the tangent f_x to the curve $f(x)$. Thus, we can apply (3.3).

We need to convert the surface geometric properties in (3.3) into their simpler 1D analogues. For example, the Hessian H becomes f_{xx} , the directional derivative $\vec{u}[H]$ becomes f_{xxx} , and $dN(\vec{u})$ becomes the change in the normal as we move along the tangent direction:

$$dN(\vec{u}) = \frac{f_{xx}}{1 + (f_x)^2}.$$

Putting it all together, we get the simplified shading equation in one dimension, as follows.

Corollary 4.1.

$$(4.1) \quad I_{xx} = -I \frac{f_{xx}^2}{(1 + a^2)^2} - 2I_x \frac{af_{xx}}{(1 + a^2)} + I_x \frac{f_{xxx}}{f_{xx}}.$$

Now, the functions $\{I, I_x, I_{xx}\}$ are all known from the intensity curve, so (4.1) is a third-order differential equation. Solving this with the appropriate boundary conditions will give us our curve $f(x)$. Note that these equations are ill-defined when $f_{xx} = 0$, so one must be careful and use approximations near these critical points.

4.1.2. Boundary conditions. As (4.1) is a third-order equation, it needs three boundary conditions. We assume known Dirichlet boundary conditions on both endpoints, with a single von Neumann condition at one of the endpoints. With these boundary conditions, we use Mathematica's NDSolve function to get the solution curve $\tilde{f}(x)$. (See Figure 10.) Although this is a toy example, it illustrates the precision that may be available in 2D SFS if we can solve the shading equations in a P.D.E. formulation.

4.1.3. Extension to 2D shading. If a surface satisfies (3.1), (3.2), and (3.3) at every point in the image, then that local surface will be a possible solution to SFS for that patch. That is, when imaged under some set of light source positions, it will result in an identical image. Thus, these equations implicitly define all smooth surfaces that can satisfy a shaded patch. However, in order to solve that P.D.E., one must have boundary conditions, and it is unclear in the 2D case exactly what these boundary conditions should be. In addition, we have no guarantee that there will be a unique solution, as these are nonlinear equations and thus don't satisfy the standard P.D.E. uniqueness theorem. In fact, we know that many different global surfaces can lead to the exact same image. Thus, we will consider some solutions under simplifications (equivalently, assumptions on our surface). For completeness, we write the shading equations in the standard P.D.E. fashion in Appendix B.

4.2. Case 2: Second-order assumption and frontal-parallel. The shading equations are quite complicated, nonlinear, and of third order. Although it may be possible to directly solve them as a P.D.E. system, we will initially look at surfaces where the equations reduce nicely. (See also [31].) Consider a second-order Monge patch: $S = (x, y, f(x, y))$ with $f(x, y) = ax + by + cx^2 + dxy + ey^2$. Since there are no third-order terms, the directional derivative of

the Hessian will be a 0-matrix, and the equations simplify as follows.

Corollary 4.2.

$$(4.2a) \quad \nabla I \cdot D_{\hat{v}} \hat{v} = (I) \|dN(\vec{v})\|^2,$$

$$(4.2b) \quad \nabla I \cdot D_{\hat{u}} \hat{v} = (I) \langle dN(\vec{v}), dN(\vec{u}) \rangle_G + \|\nabla I\| \frac{\nabla f^T I I(\vec{v})}{\sqrt{1 + \|\nabla f\|^2}},$$

$$(4.2c) \quad I_{uu} = -(I) \|dN(\vec{u})\|^2 - 2\|\nabla I\| \frac{\nabla f^T I I(\vec{u})}{\sqrt{1 + \|\nabla f\|^2}}.$$

Equation (4.2a) corresponds to the equation in the introductory figure, Figure 3. Specifying further, let the tangent plane be frontal-parallel, i.e., where the normal to the tangent plane is parallel to the view vector. In this case, at the origin, vectors on the image plane are only translations of vectors on the tangent plane (in contrast to the general case, where there may be rotations, dilations, etc.). The first fundamental form G is now the identity matrix. Thus, the shape operator reduces: $dN = -G^{-1}II = -II = -n_3H = -H$. In addition, the gradient $\nabla f = \vec{0}$. Thus, the second term on the right-hand side of each equation is equal to $\vec{0}$, as follows.

Corollary 4.3.

$$(4.3a) \quad \frac{\nabla I \cdot D_{\hat{v}} \hat{v}}{I} = \vec{v} H^2 \vec{v},$$

$$(4.3b) \quad \frac{\nabla I \cdot D_{\hat{u}} \hat{v}}{I} = \vec{v} H^2 \vec{u},$$

$$(4.3c) \quad -\frac{I_{uu}}{I} = \vec{u} H^2 \vec{u}.$$

In this important special case, we get the elegant result that the normalized second derivatives of intensity are proportional to the square of surface second derivatives. This has been hypothesized in other work [35]. The twofold ambiguity, defined by the transformation $H \rightarrow -H$, emerges naturally. This is the well-explored *concave/convex* ambiguity. But there is more.

Without loss of generality, we assume that the Hessian is expressed in the basis $\{\vec{v}, \vec{u}\}$. Then, the equations simplify once more:

$$(4.4a) \quad \frac{\nabla I \cdot D_{\hat{v}} \hat{v}}{I} = (f_{xx}^2 + f_{xy}^2),$$

$$(4.4b) \quad \frac{\nabla I \cdot D_{\hat{u}} \hat{v}}{I} = (f_{xx} + f_{yy}) f_{xy},$$

$$(4.4c) \quad -\frac{I_{uu}}{I} = (f_{yy}^2 + f_{xy}^2).$$

If we plot these three equations in 3-space defined by coordinate axes of $\{f_{xx}, f_{xy}, f_{yy}\}$, we see that the equations consist of two cylinders (situated perpendicular to each other) and a hyperbolic cylinder, which has a set of four intersection points. Thus, we have a second twofold ambiguity, which corresponds to a saddle/ellipsoid type of ambiguity. See Figure 11. This is essentially unstudied in the SFS literature (but see [15]).

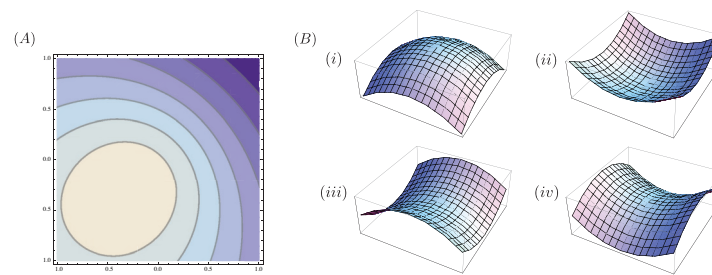


Figure 11. For a frontal-parallel second-order Monge patch, we have exactly four surfaces available for each shading flow (A). There is a concave/convex ambiguity in each row and a saddle/ellipsoid ambiguity in each column.

Note that we reduced the possible surfaces greatly by assuming that the third-order surface derivatives were 0 and that the tangent plane is frontal-parallel. That is, we assumed knowledge of four parameters (on the third-order derivatives) and two parameters (tangent plane). In general, without these assumptions, the shading ambiguity space (for second-order intensity) will be six-dimensional. In section 4.3, we will relax the condition of a frontal-parallel tangent plane.

4.3. Case 3: A Second-order assumption. We turn back to the question posed in the first example, but with only the second-order assumption on a surface patch. That is, we again write $S = (x, y, f(x, y))$ with $f(x, y) = ax + by + cx^2 + dxy + ey^2$. This is a more complicated example than the previous two. In this case, we work with the system defined by (4.2a), (4.2c), (4.2b). Recently, Xiong et al. [39] devised a shape reconstruction algorithm based on this quadratic assumption.

In the local second-order patch, there are five parameters that need to be calculated: the two tangent plane orientation parameters and the three surface curvatures. There are also an additional three parameters (two for the light source position and one for the albedo) that are involved in the image formation process. If we consider the previous discussion, then we have six conditions $\{I, I_x, I_y, I_{xx}, I_{xy}, I_{yy}\}$ with eight parameters. Our equations factor out both the albedo and light source directions—thus, we ignore $\{I, I_x, I_y\}$ and end up with three conditions $\{I_{xx}, I_{xy}, I_{yy}\}$ on five surface parameters. It is reasonable to expect that we will have a 2D family of surfaces corresponding to each patch. In addition, we expect that the family of surfaces may be parametrized by the light source position on the upper hemisphere or equivalently the two parameters of the tangent plane.

This agrees with what we saw in section 4.2. Since we now have no assumption on the two parameters of the tangent plane, we must have a 2D ambiguity multiplied by any ambiguity we found in that section.

We can apply the shading equations above to the Monge patch $f(x, y)$ to get three polynomial equations in $\{a, b, c, d, e\}$. We denote them as $g_i(c, d, e)$, $i = 1, 2, 3$. (For this analysis, we used Mathematica.) We will not display these polynomial equations here, as they are equivalent to those in Appendix C. By choosing either the tangent plane coefficients or the dominant light source direction, these fourth-order polynomial equations $\{g_i\}$ define the re-

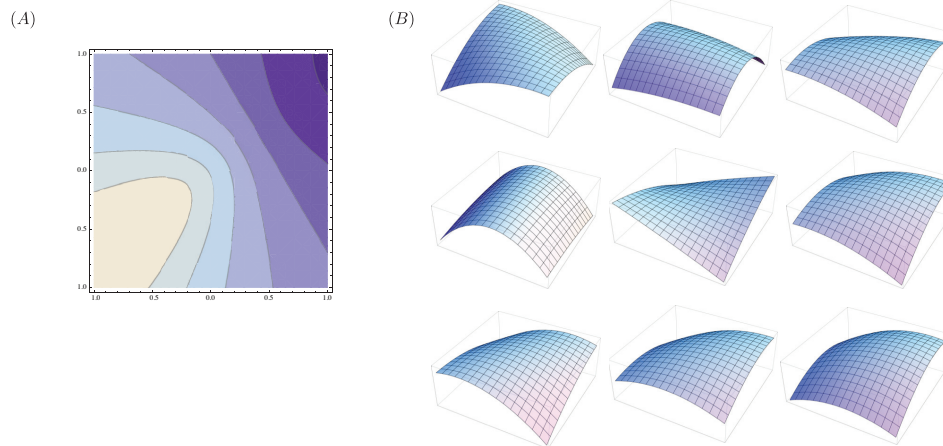


Figure 12. Panel (A) shows a shading flow, and (B) shows nine second-order Monge patches, all with different tangent planes that are solutions to the polynomial equations and thus can result in the shading flow in panel (A) when lit properly.

maining coefficients $\{c, d, e\}$.

For any chosen $\{a, b\}$ there may be either 0, 2, or 4 real roots of these polynomial equations. Each root represents a corresponding second-order Monge patch that would result in the same local shading flow. Since the analysis is difficult, we have observed experimentally that the polynomial system with a given shading flow and tangent plane choice will have real roots only outside a rectangle containing the origin. That is, the region of tangent planes where there is no solution is of the form $\{-x_0 \leq a \leq x_1 \cup -y_0 \leq b \leq y_1\}$. Of course, the rectangle's exact dimensions depend on the tangent plane choice. Unfortunately, due to the complexity of this polynomial system, we are unable to state the exact relationship between the rectangle, the shading flow, and the choice of tangent plane.

However, given the shading flow and any single choice of the tangent plane $\{a, b\}$, we can solve the $\{g_i\}$ to calculate all the possible roots of the equation for that tangent plane choice, which is the necessary part for our goal of shape reconstruction. In Figure 12, we show an example of a shading flow that corresponds to each of the following surfaces, all with slightly different tangent planes.

4.4. Case 4: Fourfold ambiguity. If there is to be a real solution to the polynomial system $\{g_i\}$ for a chosen $\{a_0, b_0\}$, we must have either two or four solutions. The reason for part of this ambiguity is due to the squared nature of the $\|dN(\vec{v})\|$ terms, just as we saw with the $H \rightarrow -H$ ambiguity in the frontal-parallel case. This leads to the standard concave/convex ambiguity pair. The second pair of solutions (if it exists) is due to the saddle/spherical ambiguity, as shown in Figures 11 and 13. We get one surface for each positive and negative pair of the principal curvatures: $\{k_1 > 0, k_2 > 0\}$, $\{k_1 > 0, k_2 < 0\}$, $\{k_1 < 0, k_2 > 0\}$, $\{k_1 < 0, k_2 < 0\}$. To see the complete ambiguity in movie form, please view the associated multimedia files (92647_01.gif [local/web 13.3MB], 92647_02.gif [local/web 13.6MB], 92647_03.gif [local/web 13.7MB], and 92647_04.gif [local/web 13.2MB]). There is one multimedia file for each branch, as explained further in Figure 13.

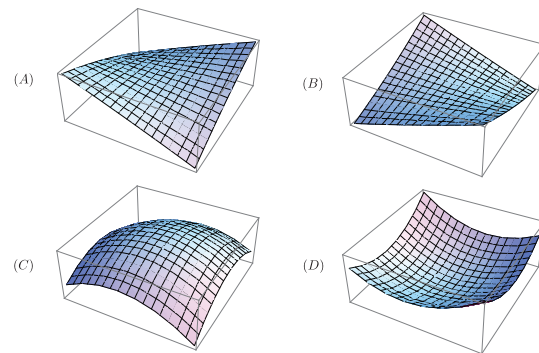


Figure 13. Panel (A) is the surface corresponding to the frontal-parallel tangent plane, as seen in the center of Figure 12(B). In addition to the 2D ambiguity, as illustrated in Figure 12 there can also be up to four different surfaces for a given tangent plane. Here, (B), (C), and (D) are the other surfaces corresponding to panel (A). Note that there are a pair of saddle surfaces and a pair of spherical surfaces. The transformation taking (A) to (B) and (C) to (D) is the concave-convex ambiguity. In addition, there is a novel ambiguity between saddles and spherical shapes. To see the complete ambiguity in movie form, please view the associated multimedia files (92647_01.gif [local/web 13.3MB] for panel (A), 92647_02.gif [local/web 13.6MB] for panel (B), 92647_03.gif [local/web 13.7MB] for panel (C), and 92647_04.gif [local/web 13.2MB] for panel (D)). In each animation, the left panel is a shaded third-order surface; the middle is a zoom-in of the outlined area from the left panel, corresponding to the region where second-order terms are dominant and thus second-order equations are accurate; and the right panel shows all second-order surfaces (of the appropriate family) and corresponding light source positions (yellow vector) that will create the image in the middle panel.

To summarize, we can use these equations to recover all the second-order surface information (up to the fourfold ambiguity), given any tangent plane orientation. However, we do not have the tangent plane information a priori. Thus, even with the local second-order assumption, we already must deal with at least a 2D shading ambiguity at every local patch due to the unknown light source positions.

For this reason, we believe the SFS problem can—and should—either be solved at certain points in the image (considered next) or be combined with other means for obtaining tangent plane information.

4.5. Ambiguity reduction at critical points. Much work has focused on the question: “Where should one draw lines on a surface in order to give the best impression of the surface shape?” Recently, Decarlo et al. [9] have considered “suggestive contours,” and Judd, Durand, and Adelson [24] have suggested apparent ridges. How do we decide which feature lines are “better”? [8] Why are certain curves so helpful in psychophysics? We believe these questions can be answered by looking at the shading ambiguity on these feature lines.

Consider the example of contours of critical points on a surface. We define them here as the points where the brightness gradient is zero. (In reality, we consider $\nabla I < \epsilon$.) Thus, these are connected points that are local maxima or minima. Koenderink has analyzed isolated points of this type in a simplified case [30]. For now, consider the generic case when the Gaussian curvature is not zero and so H^{-1} is well-defined. As the isophote direction \hat{v} is no longer well-defined at these points, we need to change the basis to $\{x, y\}$. Our equations (3.1),

(3.2), and (3.3) become

$$(4.5a) \quad I_{xx} = -(I)\|dN(\vec{e}_1)\|^2,$$

$$(4.5b) \quad I_{xy} = -(I)\langle dN(\vec{e}_1), dN(\vec{e}_2) \rangle_G,$$

$$(4.5c) \quad I_{yy} = -(I)\|dN(\vec{e}_2)\|^2.$$

This is quite similar to the second-order, frontal-parallel case, but we didn't need any assumptions! Rather, we can apply these equations only to highlight points in the image. This cursory analysis may explain why highlight lines are so effective at revealing surface shape psychophysically. We believe that understanding shading ambiguity can be a useful metric for deciding between different definitions of "shape representing contours." We can also go the other way; we can use the dimension of the shading ambiguity to define contours (sets of points) where the surface information is mathematically more restricted by the shading than at a generic point; key among these points are ridges [32]. Because of the complexity of this analysis, it will be treated in a companion paper.

5. Discussion. Given a local image patch I , assumed to be Lambertian, we can consider the continuous pixel intensity information in terms of derivatives of intensity. At every point in the image, we have the information contained in $\{I, I_x, I_y, I_{xx}, I_{xy}, I_{yy}\}$. We could consider more derivatives, but the second-order derivatives are the minimum order needed for removing the explicit light source variables. If we consider more brightness derivatives, the additional unknown parameters will grow faster than the additional constraints.

Let us consider how to use each informational element. The intensity I at a point alone is useless in considering shape information (or even tangent plane orientation), as we don't know the albedo. With unknown albedo, any angle between the unknown light sources and unknown normal is possible. (Of course, we have information from intensity at *several* nearby points, but we are considering that information to be contained in the derivatives of I .)

The amount of information in I_x and I_y is usually expressed as a brightness gradient. Unfortunately, given any surface with second fundamental form II of full rank, we can find a set of light source positions that will result in that brightness gradient. Thus, $\{I_x, I_y\}$ does not give us any information about the surface unless we assume a prior on light source positions: the set of surfaces before and after we consider the brightness gradient is the same. In terms of counting conditions, the brightness gradient gives us two conditions on the scene, but having to include the explicit light source positions in the equations adds at least two more parameters. However, once we have solved for our surface, we can use the brightness gradient to solve for the light source direction, which may vary for each local patch.

Finally, consider the second derivatives: $\{I_{xx}, I_{xy}, I_{yy}\}$. As we have shown, these are the lowest-order derivatives of intensity that can be factored into components of surface shape and image properties, with no explicit dependence on the light source. Thus, our SFS reconstruction efforts will focus on the use of these equations in order to create a light source-invariant algorithm.

Pentland [35] also attempted to calculate local surface shape via the constraints $\{I, I_x, I_y, I_{xx}, I_{xy}, I_{yy}\}$. In order to make it well-posed, he needed a local spherical assumption. With this assumption, he had six parameters to solve for: two for the light source direction, one for the albedo ρ , two for the tangent plane orientation, and one for the radius of curvature.

Our work differs in that, rather than putting a hard constraint on our shape, we allow for all possible local surfaces.

At particular points, i.e., at a boundary [25] or a cusp [33], we can get an unambiguous local estimate. However, to realize a global reconstruction, we will have to choose among generic local patches via compatibility conditions. There are two obvious compatibility metrics to use. First, we would like overlapping patches to have similar normals. Second, we would like the light sources used for nearby patches to be close in \mathbb{R}^3 . We plan to implement this via a relaxation labeling scheme [22] in an upcoming paper.

6. Conclusion. We have modeled the SFS problem as a light source-invariant map from a unit length vector field to a local surface and analyzed the mathematical relationship between the curvature of the isophotes and the curvatures of the surface in terms of differential geometry. To gain intuition, we considered these relationships on various simplified domains. We were able to investigate the ambiguity family when either the surface was second-order or the brightness gradient ∇I was zero. We showed a novel ambiguity between saddle shapes and ellipsoids.

We close with a neurobiological point. It is known that the higher visual areas are selective for surface properties, including their curvatures [40]. It is also known that many different forms of orientation images, such as oriented texture noise and glossy patterns (see references in [16]) are perceived as surfaces. To our knowledge the calculations here provide a suggestion of how this inference might proceed from the shading flow to surfaces. Our calculations also illustrate the ambiguity that remains. While the human visual system gives the impression of solving SFS, sometimes the solution is very physically inaccurate [29]. Michelangelo clearly “understood” the importance of drawing highlights, boundaries, and ridges to support the shading inference. Much remains to be done with the differential geometric approach to completely understand why.

Appendix A. Covariant differentiation of the Hessian H : Proof of Proposition 2.4.

A.1. Tensor decomposition of H . To covariantly differentiate H , we note that H is a $(0, 2)$ tensor, and so we can expand it as a sum of tensor products of 1-forms. We follow the notation in [12]. Write H^1, H^2 as the two rows of H , and E_1, E_2 as the standard basis 1-forms. In a tensor representation, H^1 and H^2 are also both 1-form fields (covariant tensors):

$$\begin{aligned} \text{(A.1a)} \quad H &= \begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix} \\ \text{(A.1b)} \quad &= [f_{xx} \ f_{xy}] \otimes [1 \ 0] + [f_{xy} \ f_{yy}] \otimes [0 \ 1] \\ \text{(A.1c)} \quad &= H^1 \otimes E_1 + H^2 \otimes E_2. \end{aligned}$$

To covariantly differentiate H , we apply a product rule for tensor products:

$$\begin{aligned} \text{(A.2)} \quad \nabla_{\bar{v}} H &= H^1 \otimes \nabla_{\bar{v}} E_1 + H^2 \otimes \nabla_{\bar{v}} E_2 \\ &\quad + \nabla_{\bar{v}} H^1 \otimes E_1 + \nabla_{\bar{v}} H^2 \otimes E_2. \end{aligned}$$

Note that each of these four terms is also a $(0, 2)$ tensor, and thus each term requires as input two vectors. Without loss of generality, we calculate one of the individual terms $\nabla_v H^1 \otimes E_1$;

the rest are analogous. The covariant derivative of a covariant tensor requires actions on the tensor inputs, so we introduce dummy vectors \vec{w}_1, \vec{w}_2 to use in our expression.

A.2. Parallel transport of H .

Remark 2. As mentioned in Remark 1, we recall the second, equivalent, definition of covariant differentiation here. We define it intrinsically, that is, independent of the ambient space \mathbb{R}^3 . We will not go into the derivations regarding *connections* or Christoffel symbols, which can be found in [12] and [11]. We just summarize by saying that *parallel transport* is a way to “equate” nearby vectors in nearby tangent planes along a curve $\beta(s)$. Using notation as in [12], we will write the parallel transport in the forward direction of the vector field $\vec{w}(\beta(s))$ as $\tau_s^\rightarrow(\vec{w}(\beta(s)))$. Conversely, the parallel transport backwards along the curve is written $\tau_s^\leftarrow(\vec{w}(\beta(s)))$. Then, the covariant derivative can be defined intrinsically as

$$\nabla_{\beta'(0)}\vec{w} = \lim_{s \rightarrow 0} \left(\frac{\tau_s^\leftarrow(\vec{w}(\beta(s))) - \vec{w}(\beta(0))}{s} \right).$$

Thus, covariant differentiation resolves the tangent plane orientation problem by first transporting the vector $\vec{w}(\beta(s)) \in T_{\beta(s)}(S)$ back to a “parallel” vector in $T_{\beta(0)}(S)$ before doing the standard derivative subtraction.

Now, due to the duality between 1-forms and vectors, when we apply covariant differentiation to a 1-form, we parallel transport the vector it acts on forwards. (This is opposite to a covariant derivative of a vector, which is parallel transported backwards in the derivative.) Define $\beta(s)$ as a curve passing through P with velocity \vec{v} . Note that the 1-form H^1 and its input vector \vec{w}_1 are defined along $\beta(s)$ and may be indexed at different positions. We will denote the position of H^1 using a subscript, such as $H_{\beta(s)}^1$. Using the definition of covariant derivative for covariant tensors in [12], we obtain

$$(A.3a) \quad (\nabla_{\vec{v}}H^1 \otimes E_1)(\vec{w}_1, \vec{w}_2) = E_1(\vec{w}_2) \cdot \left(\lim_{s \rightarrow 0} \frac{H_{\beta(s)}^1(\tau_s^\rightarrow(\vec{w}_1(\beta(0)))) - H_{\beta(0)}^1(\vec{w}_1(\beta(0)))}{s} \right)$$

$$(A.3b) \quad = E_1(\vec{w}_2) \cdot \left(H_{\beta(0)}^1 \left(\lim_{s \rightarrow 0} \frac{(\tau_s^\rightarrow(\vec{w}_1(\beta(0)))) - (\vec{w}_1(\beta(0)))}{s} \right) \right)$$

$$(A.3c) \quad - E_1(\vec{w}_2) \cdot (\vec{v}[H^1]\vec{w}_1)$$

$$(A.3d) \quad = E_1(\vec{w}_2) \cdot H^1(T_{\vec{w}_1}^\rightarrow) - E_1(\vec{w}_2) \cdot (\vec{v}[H^1]\vec{w}_1).$$

The dot represents multiplication here. To go from (A.3a) to (A.3b), we used the substitution

$$-\vec{v}[H^1](\vec{w}_1) = \lim_{s \rightarrow 0} \left(\frac{-H_{\beta(s)}^1(\tau_s^\rightarrow(\vec{w}_1(\beta(0)))) + H_{\beta(0)}^1(\tau_s^\rightarrow(\vec{w}_1(\beta(0))))}{s} \right).$$

In English, this just states, “the directional derivative of 1-form H^1 is defined as the difference in H^1 as we parallel transport its input.” For notational simplicity, in (A.3d) we assigned the change from parallel transport of an arbitrary vector w_1 to be $T_{w_1}^\rightarrow$:

$$(A.4) \quad \lim_{s \rightarrow 0} \frac{\tau_s^\rightarrow(\vec{w}_1(\beta(0))) - \vec{w}_1(\beta(0))}{s} = T_{w_1}^\rightarrow.$$

$T_{w_1}^{\vec{}}$ can be written using the *Christoffel symbols*, but this will lead to unnecessary notation as the terms involving T_{w_1} will eventually cancel.

We now repeat the process for the other three terms in (A.2) and compile the four terms into the original matrix representation to get

$$(A.5) \quad (\nabla_v H)(\vec{w}_1, \vec{w}_2) = T_{w_2}^{\vec{}} H \vec{w}_1 + \vec{w}_2^T H T_{w_1}^{\vec{}} - \vec{w}_2^T \vec{v} [H] \vec{w}_1.$$

Appendix B. Proofs for shading equations $D_{\hat{u}} \hat{u}$ and $D_{\hat{u}} \hat{v}$. The proofs for the shading equations (3.2) and (3.3) are analogous to the proof for the first shading equation, (3.1). Rather than repeat the analysis almost verbatim, we instead describe where substitutions need to be made.

B.1. For $D_{\hat{u}} \hat{v}$. Here, we take the directional derivative of the constraint $\langle \vec{l}_t, dN(\vec{v}) \rangle_G$ in the \vec{u} direction. Thus, we need to modify (2.6a) so that we take the directional derivative with respect to \vec{u} rather than \vec{v} . From then on, every time we see a \vec{v} as a direction in which to take a derivative, we will instead have \vec{u} . The analysis for $D_{\hat{u}} \hat{v}$ then follows the $D_{\hat{v}} \hat{v}$ case exactly except in the following one place.

The first term in (2.14b) (which was $T_{l_t}^{\vec{}} II \vec{v}$) is now $T_{l_t}^{\vec{}} II \vec{u}$ and thus contributes a term rather than 0. However this simplifies (using the Christoffel symbols Γ_{ij}^k) to

$$(B.1a) \quad T_{l_t}^{\vec{}} = \left[\sum_{i,j} \Gamma_{ij}^1 \vec{l}_t^i u^j \quad \sum_{i,j} \Gamma_{ij}^2 \vec{l}_t^i u^j \right]$$

$$(B.1b) \quad = \left[\frac{f_x (\vec{l}_t^T II \vec{u})}{\sqrt{1+f_x^2+f_y^2}} \quad \frac{f_y (\vec{l}_t^T II \vec{u})}{\sqrt{1+f_x^2+f_y^2}} \right]$$

$$(B.1c) \quad = \vec{l}_t^T II \vec{u} \frac{\begin{bmatrix} f_x & f_y \end{bmatrix}}{\sqrt{1+f_x^2+f_y^2}}$$

$$(B.1d) \quad = \|\nabla I\| \frac{\nabla f}{\sqrt{1+\|\nabla f\|^2}}.$$

Thus,

$$(B.2) \quad T_{l_t}^{\vec{}} II \vec{u} = \|\nabla I\| \frac{(\nabla f)^T II \vec{u}}{\sqrt{1+\|\nabla f\|^2}}.$$

And this is precisely the extra term included in (3.2).

B.2. For $D_{\hat{u}} \hat{u}$. Here, we take the directional derivative of the constraint $\langle \vec{l}_t, dN(\vec{u}) \rangle_G = I_{\vec{u}}$ in the \vec{u} direction, giving us I_{uu} . Thus, we will get I_{uu} on the left-hand side of (2.6a). However, the analogous second term for (2.15e) will now be 0, as $D_{\vec{u}} \vec{u}$ is always perpendicular to ∇I . Thus, we will get an I_{uu} on the left-hand side of (3.3) rather than a term analogous to $D_{\vec{v}} \vec{v}$ on the right-hand side of (2.15e). This leads to each term on the right-hand side of (3.3) being the opposite sign of the previous equations (3.1) and (3.2).

The other differences between this proof and the previous one for $D_{\hat{u}} \hat{v}$ is that the terms containing $dN(\vec{v})$ in the previous proof are now $dN(\vec{u})$. In (2.10b), the second term is now

$-\vec{u}[n_3]G^{-1}H(\vec{u})$. In the previous proofs, the corresponding term enters an inner product with \vec{l}_t (which is equal to 0) and thus contributes nothing. Here, we must calculate it separately.

Straightforward calculation yields

$$(B.3) \quad \vec{u}[n_3] = \frac{\nabla f^T I I \vec{u}}{\sqrt{1 + \|\nabla f\|^2}}.$$

Thus,

$$(B.4a) \quad \langle \vec{l}_t, -\vec{u}[n_3]G^{-1}H(\vec{u}) \rangle_G = -\frac{\nabla f^T I I \vec{u}}{\sqrt{1 + \|\nabla f\|^2}} \langle \vec{l}_t, dN(\vec{u}) \rangle_G$$

$$(B.4b) \quad = -\|\nabla I\| \frac{\nabla f^T I I \vec{u}}{\sqrt{1 + \|\nabla f\|^2}}.$$

Therefore, the sum of the two extra terms we get when applying the method for $D_{\hat{u}}\hat{u}$ is solely

$$-2\|\nabla I\| \frac{\nabla f^T I I \vec{u}}{\sqrt{1 + \|\nabla f\|^2}}.$$

Adding these respective terms into the formulas for $D_{\hat{u}}\hat{v}$ and $D_{\hat{u}}\hat{u}$ and changing the appropriate \vec{v} to \vec{u} give the second-order shading equation stated in (3.3).

Appendix C. P.D.E.s. For completeness, we add the shading equations in P.D.E. fashion in the $\{x, y\}$ basis, without the differential geometric notation:

$$(C.1) \quad \begin{aligned} 0 = & I_{xx} \\ & + \left(\frac{(1 + f_x^2)f_{xy}^2 - 2f_x f_y f_{xy} f_{xx} + (1 + f_y^2)f_{xx}^2}{(1 + f_x^2 + f_y^2)^2} \right) I \\ & + 2I_x \frac{f_x f_{xx} + f_y f_{xy}}{1 + f_x^2 + f_y^2} \\ & - \left(\frac{I_x(f_{yy}f_{xxx} - f_{xy}f_{xxy}) + I_y(-f_{xy}f_{xxx} + f_{xx}f_{xxy})}{f_{xx}f_{yy} - f_{xy}^2} \right), \end{aligned}$$

$$(C.2) \quad \begin{aligned} 0 = & I_{yy} \\ & + \left(\frac{(1 + f_x^2)f_{yy}^2 - 2f_x f_y f_{xy} f_{yy} + (1 + f_y^2)f_{xy}^2}{(1 + f_x^2 + f_y^2)^2} \right) I \\ & + 2I_y \frac{f_x f_{xy} + f_y f_{yy}}{1 + f_x^2 + f_y^2} \\ & - \left(\frac{I_x(f_{yy}f_{xyy} - f_{xy}f_{yyy}) + I_y(-f_{xy}f_{xyy} + f_{xx}f_{yyy})}{f_{xx}f_{yy} - f_{xy}^2} \right), \end{aligned}$$

$$\begin{aligned}
\text{(C.3)} \quad 0 &= I_{xy} \\
&+ \left(\frac{f_{xy}(f_{xx} + f_{yy} + f_y^2 f_{xx} + f_x^2 f_{yy}) - f_x f_y (f_{xy}^2 + f_{xx} f_{yy})}{(1 + f_x^2 + f_y^2)^2} \right) I \\
&+ \frac{f_x I_y f_{xx} + (f_x I_x + f_y I_y) f_{xy} + f_y I_x f_{yy}}{1 + f_x^2 + f_y^2} \\
&- \left(\frac{I_x (f_{yy} f_{xxy} - f_{xy} f_{xyy}) + I_y (-f_{xy} f_{xxy} + f_{xx} f_{xyy})}{f_{xx} f_{yy} - f_{xy}^2} \right).
\end{aligned}$$

Acknowledgments. We thank Dan Holtmann-Rice, Matt Lawlor, Steve Cholewiak, Roland Fleming, and the reviewers for helpful discussions and comments on the manuscript.

REFERENCES

- [1] J. BARRON AND J. MALIK, *Shape, Illumination, and Reflectance from Shading*, Technical Report, 2013.
- [2] P. BELHUMEUR AND D. KRIEGMAN, *What is the set of images of an object under all possible illumination conditions?*, *Internat. J. Computer Vision*, 28 (1998), pp. 1–16.
- [3] P. BELHUMEUR, D. KRIEGMAN, AND A. YUILLE, *The bas-relief ambiguity*, *Internat. J. Computer Vision*, 35 (1999), pp. 33–44.
- [4] O. BEN-SHAHAR AND S. ZUCKER, *Geometrical computations explain projection patterns of long-range horizontal connections in visual cortex*, *Neural Comput.*, 16 (2004), pp. 445–476.
- [5] P. BRETON AND S.W. ZUCKER, *Shadows and shading flow fields*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '96)*, 1996, pp. 782–789.
- [6] M. BREUSS, E. CRISTIANI, J.-D. DUROU, M. FALCONE, AND O. VOGEL, *Perspective shape from shading: Ambiguity analysis and numerical approximations*, *SIAM J. Imaging Sci.*, 5 (2012), pp. 311–342.
- [7] M.J. BROOKS AND B.K.P. HORN, *Shape and source from shading*, in *Proceedings of the International Joint Conference on Artificial Intelligence*, 1985, pp. 932–936.
- [8] F. COLE, K. SANIK, D. DECARLO, A. FINKELSTEIN, T. FUNKHOUSER, S. RUSINKIEWICZ, AND M. SINGH, *How well do line drawings depict shape?*, *ACM Trans. Graph.*, 28 (2009).
- [9] D. DECARLO, A. FINKELSTEIN, S. RUSINKIEWICZ, AND A. SANTELLA, *Suggestive contours for conveying shape*, in *Proceedings of SIGGRAPH*, 2003, pp. 848–855.
- [10] P. DIEFT AND J. SYLVESTER, *Some remarks on the shape-from-shading problem in computer vision*, *J. Math. Anal. Appl.*, 84 (1981), pp. 235–248.
- [11] M.P. DOCARMO, *Differential Geometry of Curves and Surfaces*, Prentice–Hall, Upper Saddle River, NJ, 1976.
- [12] C.T.J. DODSON AND T. POTSON, *Tensor Geometry*, Springer, Berlin, Heidelberg, 1991.
- [13] P. DUPUIS AND J. OLIENSIS, *An optimal control formulation and related numerical methods for a problem in shape reconstruction*, *Ann. Appl. Probab.*, 4 (1994), pp. 287–346.
- [14] A. ECKER AND A.D. JEPSON, *Polynomial shape from shading*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 25–27.
- [15] R. ERENS, A. KAPPERS, AND J.J. KOENDERINK, *Perception of local shape from shading*, *Perception & Psychophys.*, 54 (1993), pp. 145–156.
- [16] R. FLEMING, D. HOLTSMANN-RICE, AND H. BULTHOFF, *Estimation of 3D shape from image orientations*, *Proc. Natl. Acad. Sci. USA*, 108 (2011), pp. 20438–20443.
- [17] R. FLEMING, A. TORRALBA, AND E.H. ADELSON, *Specular reflections and the perception of shape*, *J. Vision*, 4 (2004), 10.
- [18] D.A. FORSYTH, *Variable-source shading analysis*, *Internat. J. Computer Vision*, 91 (2011), pp. 280–302.
- [19] J. GARDING, *Surface orientation and curvature from differential texture distortion*, in *Proceedings of the 5th International Conference on Computer Vision*, 1995, pp. 733–739.
- [20] B.K.P. HORN AND M. BROOKS, *The variational approach to shape from shading*, *Computer Vision Graphics Image Process.*, 33 (1986), pp. 174–208.

- [21] D. HUBEL, *Eye, Brain, and Vision*, Scientific American Library, 1988.
- [22] R.A. HUMMEL AND S. ZUCKER, *On the foundations of relaxation labeling processes*, IEEE Trans. Pattern Anal. Machine Intell., 5 (1983), pp. 267–287.
- [23] K. IKEUCHI AND B.K.P. HORN, *Numerical shape from shading and occluding boundaries*, Artificial Intell., 1981, pp. 141–183.
- [24] T. JUDD, F. DURAND, AND E.H. ADELSON, *Apparent ridges for line drawing*, ACM Trans. Graph., 26 (2007), 19.
- [25] J.J. KOENDERINK, *What does the occluding contour tell us about solid shape?*, Perception, 13 (1984), pp. 321–330.
- [26] J.J. KOENDERINK, *Solid Shape*, The MIT Press, Cambridge, MA, 1990.
- [27] J.J. KOENDERINK AND A.J. VAN DOORN, *Photometric invariants related to solid shape*, Optica Acta, 27 (1980), pp. 981–996.
- [28] J.J. KOENDERINK AND A.J. VAN DOORN, *Two-plus-one-dimensional differential geometry*, Pattern Recognition Lett., 15 (1994), pp. 439–443.
- [29] J.J. KOENDERINK, A.J. VAN DOORN, A.M. KAPPERS, AND J.T. TODD, *Ambiguity and the 'mental eye' in pictorial relief*, Perception, 30 (2001), pp. 431–438.
- [30] J.J. KOENDERINK AND A.J. VAN DOORN, *Illuminance critical points on generic smooth surfaces*, J. Opt. Soc. Amer. A, 5 (1993), pp. 844–854.
- [31] B. KUNSBERG AND S.W. ZUCKER, *The differential geometry of shape from shading: Biology reveals curvature structure*, in Proceedings of the 8th IEEE Workshop on Perceptual Organization in Computer Vision, 2012, pp. 39–46.
- [32] B. KUNSBERG AND S.W. ZUCKER, *Shape-from-shading and cortical computation: A new formulation*, J. Vision, 12 (2012), 233.
- [33] M. LAWLOR, D. HOLTSMANN-RICE, P. HUGGINS, O. BEN-SHAHAR, AND S.W. ZUCKER, *Boundaries, shading, and border ownership: A cusp at their interaction*, J. Physiol. - Paris, 103 (2009), pp. 18–36.
- [34] J. OLIENSIS, *Uniqueness in shape from shading*, Internat. J. Computer Vision, 2 (1991), pp. 75–104.
- [35] A. PENTLAND, *Local shading analysis*, IEEE Trans. Pattern Anal. Machine Intell., PAMI-6 (1984), pp. 170–187.
- [36] E. PRADOS AND O. FAUGERAS, *Shape from shading: A well-posed problem?*, in Proceedings of the 2nd IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 870–877.
- [37] E. PRADOS AND O. FAUGERAS, *Shape from shading*, in Handbook of Mathematical Models in Computer Vision, N. Paragios, Y. Chen, and O. Faugeras, eds., Springer Science, New York, 2006, pp. 375–388.
- [38] J. WAGEMANS, A.J. VAN DOORN, AND J.J. KOENDERINK, *The shading cue in context*, i-Perception, 1 (2010), pp. 159–178.
- [39] Y. XIONG, A. CHAKRABARTI, R. BASRI, S.J. GORTLER, D. JACOBS, AND T. ZICKLER, *From Shading to Local Shape*, preprint, arXiv:1310.2916, 2013.
- [40] Y. YAMANE, E.T. CARLSON, K.C. BOWMAN, Z. WANG, AND C.E. CONNOR, *A neural code for three-dimensional object shape in macaque inferotemporal cortex*, Nature Neurosci., 11 (2008), pp. 1352–1360.
- [41] R. ZHANG, T. PING-SONG, J. CRYER, AND M. SHAH, *Shape from shading: A survey*, IEEE Trans. Pattern Anal. Machine Intell., 21 (1999), pp. 690–706.