COMPUTER VISION, GRAPHICS, AND IMAGE PROCESSING **22**, 70–94 (1983)

Rotationally Symmetric Operators for Surface Interpolation[1]

MICHAEL BRADY AND BERTHOLD K. P. HORN

Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Cambridge, Massachusetts 02139

Received November 23, 1981

The use of rotationally symmetric operators in vision is reviewed and conditions for rotational symmetry are derived for linear and quadratic forms in the first and second partial directional derivatives of a function $f(x, y)$. Surface interpolation is considered to be the process of computing the most conservative solution consistent with boundary conditions. The "most conservative" solution is modeled using the calculus of variations to find the minimum function that satisfies a given performance index. To guarantee the existence of a minimum function, Grimson (W. E. L. Grimson, *From Images to Surfaces*: *A Computational Study of the Human Early Visual System*, MIT Press: Cambridge, Mass., 1981.) has recently suggested that the performance index should be a seminorm. It is shown that all quadratic forms in the second partial derivatives of the surface satisfy this criterion. The seminorms that are, in addition, rotationally symmetric form a vector space whose basis is the square Laplacian and the quadratic variation. Whereas both seminorms give rise to the same Euler condition in the interior, the quadratic variation offers the tighter constraint at the boundary and is to be preferred for surface interpolation.

1. INTRODUCTION

Two separate themes from the computer vision literature come together in this paper: the use of rotationally symmetric operators, and the idea that several modules of visual perception require that the "most conservative" solution that meets a given set of boundary conditions be computed. The two themes are combined in an investigation of which operator to use in the interpolation of smooth surfaces from one-dimensional boundary constraints. Such constraints arise naturally in a variety of visual problems.

In the next section we review the role of rotationally symmetric operators in computer vision, and we derive conditions which linear and quadratic forms in the first and second directional derivatives must satisfy to be rotationally symmetric. We then discuss the idea that vision is a conservative process, citing examples from both figure perception and scene analysis. The "most conservative" solution is modeled using the calculus of variations to find the minimum function that satisfies a given performance index. A major problem associated with the use of the calculus of variations is guaranteeing the existence of a minimum function (see, for example [9, p. 173]). A theorem of Grimson [12, Theorem 2] proves that a sufficient condition for the existence of a minimum is that the performance index should be a seminorm on the space of functions. The condition is not necessary. For example, Horn [16]

[1]Support for the Laboratory's Artificial Intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research Contract N00014-80-C-0505.

0734-189X/83 $3.00

has determined the curve that minimizes the integral square curvature subject to tangency conditions at the end points; the performance index is not a seminorm.

Grimson [12] notes that many intuitively plausible performance indices based on mean and Gaussian curvature are not seminorms, but that the square Laplacian $f_{xx}^2 + 2f_{xx}f_{yy} + f_{yy}^2$ and the quadratic variation $f_{xx}^2 + 2f_{xy}^2 + f_{yy}^2$ are. We show here that *any quadratic form* in f_{xx}, f_{xy}, and f_{yy} is a seminorm.

To further constrain the choice of performance index in the infinite set of quadratic forms, we require, in addition, that the quadratic form should be rotationally symmetric. We prove that there are essentially two choices: the square Laplacian and the quadratic variation. All the remaining possibilities are linear combinations, that is, they form a vector space with these two as a basis.

To choose between the square Laplacian and the quadratic variation, we consider their respective Euler conditions and natural boundary conditions [9]. The Euler conditions are identical, but the natural boundary conditions, which are derived from the statics of a deformed thin plate, favor the quadratic variation since they offer a tighter constraint in this case.

2. ROTATIONALLY SYMMETRIC OPERATORS IN VISION

A major concern of computer vision is the isolation of constraints that combine with the information provided in the image to yield an interpretation. Early work on polyhedra [8, 18, 23, 40, 35, 36, 20] focused on the discovery of constraints deriving from the image forming process, constraints that relate image fragments, like junctions and lines, to their scene counterparts, vertices and edges. As computer vision turned its attention away from plane-faced objects to the natural world, other constraints were required. Often the constraints expressed some facet of the intuitive notion of "smoothness" and did so in a way that supported useful computations [34, 7, 19, 43, 17]. Recently, smoothness and image forming have been combined using differential geometry [12, 42, 5].

One constraint that is usually implicit, but is occasionally made explicit, expresses the idea that perceptual processes are often approximately *isotropic*. It seems that humans usually do not show strong directional preferences when detecting edges, motion, or reflectance boundaries. We seem to be equally adept at perceiving the layout and orientation of a visible surface regardless of its orientation relative to the view vector. Ullman [37] argues for an explicit isotropy constraint in his work on subjective contours (see also Knuth [21]).

Processes that are isotropic are naturally computed by rotationally symmetric operators, since the values they return are unaffected by the coordinate system chosen for the image. Conversely, rotationally symmetric operators compute isotropic information. As we shall see, many operators that have been proposed for vision are not rotationally symmetric but directionally selective. Some authors have, however, proposed rotationally symmetric operators, particularly for early visual processing.

Precise definitions of rotational symmetry for functions, operators (or functionals), and, by specialization, matrices are given in the following section. In the rest of this section we assume that the definitions are already understood.

Some kinds of blurring in an image forming system can be approximated by convolution with a Gaussian. The rotationally symmetric Gaussian can be defined by

$$G(r) = \frac{1}{2}\pi\sigma^2 \exp\left(\frac{-r^2}{2\sigma^2}\right).$$

Pratt [29] presents several techniques, such as convolution with the generalized inverse of the blur function, for restoring the image (see, for example, his Figs. 14.2.1, 14.3.2).

The Laplacian $\Delta = f_{xx} + f_{yy}$ is well known to be rotationally symmetric[2] and its use has been proposed several times in computer vision and image processing. If an image is blurred in a way that can be approximately modeled by passing the image through a system with a Gaussian point spread function, then it can be sharpened by subtracting a multiple of its Laplacian [32, p. 184; 30, p. 107]. Pratt [29, Fig. 17.4.5] illustrates the use of the Laplacian for enhancing the edges in an image.

Weska *et al.* [41] note that convolving a step edge with a Laplacian operator gives rise to a pulse pair: a negative pulse at the transition from the lower plateau to the edge, and a positive pulse at the transition from the edge to the upper plateau (see also [15, 26]). They suggested that the image intensities at the locations of the positive and negative pulses could be used to set thresholds to use in segmenting the image into regions.

Several authors have noted the relative insensitivity of human perception to small intensity gradients [13, 25–27]. They have noted that the effect can be explained by assuming that the vision system uses operators approximating second derivatives. This so-called lateral inhibition effect seems to be performed by center–surround operators in the retina (see, for example [31]). The Laplacian is a rotationally symmetric second differential operator, and an attractive candidate to perform lateral inhibition.

The use of the Laplacian for edge detection was proposed by Horn [15] in a study of the determination of lightness. Following Land and McCann [22], Horn restricted attention to images of planes colored with patches of uniform reflectance or color. Within a patch, grey level variations are due to small variations in illumination, and they are smooth compared to the abrupt changes between patches. The conventional approach to detecting significant changes in intensity had been to note that the gradient of the image is small within a region, but is infinite across a reflectance boundary between regions. For a particular image tesselation and quantization of grey levels, the gradient is always finite. It is usually much larger, however, at a reflectance boundary than it is within a region. Horn rejected using the gradient since "the first partial derivatives are directional and thus unsuitable since they will for example completely eliminate evidence of edges running in a direction parallel to their direction of differentiation." The Laplacian is the lowest order linear combination of derivatives that is rotationally symmetric. A reflectance boundary can be *detected* by the paired positive and negative peaks on either side of the boundary, and *localized* by noting the position where the Laplacian crosses zero between the peaks.[3]

[2]A proof of this is given in Section 3 below.

[3]See Binford [5] for more on the distinction between detection and localization of an intensity change.

Marr and Hildreth [26] have proposed that edges are detected in the human visual system by an operator that approximates ΔG, where Δ is the Laplacian and G is a rotationally symmetric Gaussian. We shall show in the next section that the application of a rotationally symmetric operator, such as the Laplacian, to a rotationally symmetric function, such as the Gaussian, is itself rotationally symmetric. If follows that the Marr–Hildreth operator is rotationally symmetric. Marr and Hildreth note that intensity changes occur at a number of scales and are often superimposed. They suggest that an image should be smoothed by a number of bandpass filters to isolate the changes at a particular range of scales. The Gaussian is chosen as the filter to optimize localization of changes in both the spatial and frequency domains.

We noted above that the Gaussian and the Laplacian have figured prominently in early visual processing. The Gaussian has mostly been used to approximate the point spread function corresponding to the blurring of a point source. Marr and Hildreth *deliberately* introduce Gaussian blurring. They further note that ΔG can be approximated by a difference of Gaussians $G_1 - G_2$. Nishihara and Larson [28] note that the difference of Gaussians is to be preferred on grounds of efficiency. Macleod [24] proposes an edge detection operator that is the difference of two Gaussians. However, no analysis of its performance is given, and no indication is given that the operator approximates a low-pass filtered second derivative.

Regarding the use of the Laplacian, Marr and Hildreth do not seem to make isotropy an explicit constraint on edge detection. Instead, Hildreth [14, p. 13] notes that "a number of practical considerations, which will be illuminated in the discussion of the implementation, suggested that the... operators not be directional." Suppose instead that directional operators are used. The simplest algorithm for edge detection has two stages. First, the image is convolved with the directional operators in "sufficiently many" directions. Second, the outputs are combined to determine the orientation and extent of intensity changes. Regarding the first stage, both Marr and Hildreth [26, p. 193] and Hildreth [14, p. 40] claim that the cost of convolving the image with a "sufficient" number of operators is excessive. They show that a single rotationally symmetric operator (the Laplacian) gives precisely the same results if a condition called "linear variation" holds. Regarding the second stage, Hildreth [14, p. 36] observes that edges in a direction close to that of the mask are elongated in the direction of the mask. She also notes that operators at several orientations give significant responses to any given edge, and that combining the responses is nontrivial.

There are two essentially different issues here that need to be clearly separated. Intensity changes first have to be detected and then localized as a set of "feature points" marking the position of the change in the image, and the characteristics of the corresponding edge. The detection of feature points is inherently isotropic, as Horn [15] noted. The feature points have then to be combined to produce *descriptions of edge segments*. Edge segments are clearly directional; indeed, a central problem concerns the determination of the direction of an edge in an image. The computation of rich descriptions of edge segments is, as Hildreth notes, not at all easy. Marr's [25] original "primal sketch" work was almost entirely concerned with it. Binford [5] discusses the application of directional operators to compute the directionality of an edge.

The Gaussian and Laplacian are not the only rotationally symmetric operators that have been proposed in computer vision. Prewitt [30, p. 10] observes that "derivatives of all orders can be used to form isotropic nonlinear differential operators, provided that derivatives of odd order appear only in even functions. The simplest of these... is the squared gradient," namely $\nabla \cdot \nabla$, where ∇ is the column vector

$$\begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{bmatrix}.$$

Earlier in the same article, Prewitt [30, p. 85] suggests that "the Hankel transformation enters naturally in the analysis of systems with isotropic point spread functions and greatly facilitates restoration." This suggestion does not appear to have been investigated in computer vision.

We noted earlier that an important aspect of modeling perception is the isolation of constraints which capture some facet of smoothness. Horn and Schunck [17] consider the determination of optical flow fields and note that "if every point of the brightness pattern can move independently, there is little hope of recovering the velocities." One way to express the additional constraint of smoothness is to minimize the integral of the performance index

$$S(u, v) = \left(u_x^2 + u_y^2\right) + \left(v_x^2 + v_y^2\right)$$

where u and v are the x and y components of the optical flow, and subscripts denote partial differentiation. We show in the next section that this operator is rotationally symmetric. In many simple situations the smoothness constraint is significantly wrong only at occluding boundaries.

We conclude this review of the use of rotationally symmetric operators in vision with Grimson's [12] work on surface interpolation. As it will be the focus of Section 5, our remarks will be brief. The Marr–Poggio theory of human stereo vision yields the disparity (scaled depth) at matched edge points that are computed by the Marr–Hildreth approach described above. The disparity map is as sparse as the set of matched edge points, whereas human perception is of smooth surfaces passing through the given disparity points. Grimson [12] interpolates a smooth surface from the given set of edge points by a local parallel algorithm that applies a rotationally symmetric operator to minimize the quadratic variation introduced above.

3. CONDITIONS FOR ROTATIONAL SYMMETRY

A function $f\colon \Re^2 \mapsto \Re$, is *rotationally symmetric* if its polar form is only dependent on radial distance $r = (x^2 + y^2)^{1/2}$ and not on direction $\phi = \tan^{-1} y/x$. Clearly, a function is rotationally symmetric if and only if it can be represented as a function of $(x^2 + y^2)^{1/2}$. An alternative definition can be given that is often more convenient for functions, and that can be generalized to operators. A function is rotationally symmetric if and only if it yields the same value under an arbitrary rotation of coordinates.

An anticlockwise rotation from one set of image coordinates (x, y) to another (X, Y) is effected by a rotation matrix

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = R \begin{bmatrix} x \\ y \end{bmatrix}. \tag{0}$$

For convenience, we shall denote $\cos\phi$ by c and $\sin\phi$ by s. To simplify notation, we shall not make explicit the dependence of the rotation matrix R on the angle ϕ. A function f is rotationally symmetric if and only if the untransformed version $f(x, y)$ is equal to the transformed version $f(X, Y)$. We shall occasionally find it useful to borrow the mathematical shorthand that equates a function $f(X, Y)$ with a function of a single vector argument $f(R[x, y]^T)$.

Example 1. The function $f_1(x, y) = (x^2 + y^2)$ is rotationally symmetric:

$$\begin{aligned} f_1(X, Y) &= \left((xc + ys)^2 + (yc - xs)^2\right) \\ &= (x^2 + y^2) \\ &= f_1(x, y). \end{aligned}$$

Example 2. The function $f_2(x, y) = xy$ is *not* rotationally symmetric:

$$\begin{aligned} f_2(X, Y) &= (xc + ys)(yc - xs) \\ &= xy\cos 2\phi + \frac{y^2 - x^2}{2}\sin 2\phi \end{aligned}$$

and so $f_2(X, Y) = f_2(x, y)$ only when $\phi = 0$ or $\phi = \pi$.

We can extend the definition of rotational symmetry to operators

$$\mathfrak{O} : (\Re^2 \mapsto \Re) \mapsto (\Re^2 \mapsto \Re).$$

An operator $\mathfrak{O}$ is rotationally symmetric if $\mathfrak{O}(f)$ is a rotationally symmetric function, for all functions $f: \Re^2 \mapsto \Re$.

Example 3. The function produced by the operator $\mathfrak{O}_1$, defined by

$$\mathfrak{O}_1(f)(x, y) = e^{f(x, y)}$$

is rotationally symmetric if and only if f is. In general then, the operator $\mathfrak{O}_1$ is not rotationally symmetric. However, the Gaussian is a rotationally symmetric operator as it combines Examples 1 and 3.

Most of the operators of interest in computer vision are combinations of the first and second directional derivatives $\partial/\partial x$, $\partial/\partial y$, $\partial^2/\partial x^2$, $\partial^2/\partial x \partial y$, $\partial^2/\partial y \partial x$, and $\partial^2/\partial y^2$. We need to determine the effect of a coordinate rotation on these directional derivatives. By the chain rule

$$\frac{\partial}{\partial x} = \frac{\partial X}{\partial x}\frac{\partial}{\partial X} + \frac{\partial Y}{\partial x}\frac{\partial}{\partial Y}$$
$$= c\frac{\partial}{\partial X} - s\frac{\partial}{\partial Y}.$$

Similarly,

$$\frac{\partial}{\partial y} = s\frac{\partial}{\partial X} + c\frac{\partial}{\partial Y}.$$

It follows that

$$\begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{bmatrix} = R^T \begin{bmatrix} \dfrac{\partial}{\partial X} \\ \dfrac{\partial}{\partial Y} \end{bmatrix}$$

where T denotes matrix transpose. Since R is a rotation matrix, its transpose equals its inverse, so

$$\begin{bmatrix} \dfrac{\partial}{\partial X} \\ \dfrac{\partial}{\partial Y} \end{bmatrix} = R \begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{bmatrix}. \tag{1}$$

Operators in general, and differential operators in particular, depend upon the choice of coordinate frame. To make the dependence of the differential operator on the choice of coordinate frame explicit, we introduce the notation

$$\mathcal{O}_{(x,y)}.$$

With this notation, (1) becomes

$$\nabla_{(X,Y)} = R\nabla_{(x,y)} \tag{2}$$

where $\nabla_{(x,y)}$ is the column vector

$$\begin{bmatrix} \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{bmatrix}.$$

PROPOSITION 1. *Linear combinations of $\partial/\partial x$ and $\partial/\partial y$ are not rotationally symmetric.*

Proof. Any linear form in the first directional derivatives has the form

$$[\lambda \quad \mu]\nabla_{(x,y)}.$$

The condition for rotational symmetry is

$$[\lambda \quad \mu]\nabla_{(X,Y)} = [\lambda \quad \mu]\nabla_{(x,y)}.$$

By (2)

$$[\lambda \quad \mu]\nabla_{(X,Y)} = [\lambda \quad \mu]R\nabla_{(x,y)}$$

and so the linear differential operator is rotationally symmetric if and only if

$$[\lambda \quad \mu] = [\lambda \quad \mu]R$$

so that $[\lambda \quad \mu]$ is an eigenvector of R. The eigenvalues of R are $c + is$ and $c - is$. So there are no real eigenvectors unless ϕ is a multiple of π. Since the condition is not satisfied for all ϕ, no linear combination is rotationally symmetric. □

The same style of analysis can be applied to other combinations of first derivatives such as the operator

$$\Theta_2(f) = \frac{f_x - f_y}{f_x + f_y}.$$

It is easy to show that $\Theta_{2(X,Y)}$ is not equal to $\Theta_{2(x,y)}$, for example when $\phi = \pi/2$.

In Section 2, we referred to an operator proposed by Prewitt [30], namely

$$\left(\frac{\partial}{\partial x}\right)^2 + \left(\frac{\partial}{\partial y}\right)^2$$

that is, the vector dot product

$$\nabla^T_{(x,y)}\nabla_{(x,y)}.$$

More generally, we often consider quadratic differential expressions such as

$$\nabla^T_{(x,y)}\begin{bmatrix}\lambda & \mu \\ \nu & \xi\end{bmatrix}\nabla_{(x,y)}.$$

Such an expression is called a *quadratic form* if the matrix is symmetric, that is, $\mu = \nu$. By (1),

$$\nabla_{(X,Y)} = R\nabla_{(x,y)}$$

so that

$$\nabla^T_{(x,y)}M\nabla_{(x,y)} = \nabla^T_{(X,Y)}M\nabla_{(X,Y)}$$

if and only if

$$R^TMR = M$$

where R is an arbitrary rotation matrix, and

$$M = \begin{bmatrix} \lambda & \mu \\ \nu & \xi \end{bmatrix}.$$

Since the transpose R^T of a rotation matrix R is the inverse of R, a quadratic form is rotationally symmetric if and only if the corresponding matrix M commutes with all rotation matrices. We will refer to matrices M having this property as being rotationally symmetric.

LEMMA 1. *A* 2×2 *matrix is rotationally symmetric if and only if it has the form*

$$M = \begin{bmatrix} \lambda & \mu \\ -\mu & \lambda \end{bmatrix}.$$

Proof. We require $RM = MR$ for all rotation matrices R, that is,

$$\begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} \lambda & \mu \\ \nu & \xi \end{bmatrix} = \begin{bmatrix} \lambda & \mu \\ \nu & \xi \end{bmatrix} \begin{bmatrix} c & -s \\ s & c \end{bmatrix}.$$

Expanding, and equating terms, this holds if and only if

$$\mu + \nu = 0$$
$$\lambda = \xi.$$

Alternatively, only the operations of scaling by a constant k and multiplication by a rotation matrix R' commute with all rotation matrices in two dimensions. So $M = kR'$ for some scale factor k and some rotation matrix R'. □

PROPOSITION 2. *Up to scaling, the only rotationally symmetric quadratic form in* $\partial/\partial x$ *and* $\partial/\partial y$ *is* $\nabla_{(x,y)} \cdot \nabla_{(x,y)}$.

Proof. A quadratic form in $\partial/\partial x$ and $\partial/\partial y$ has the form

$$\nabla^T_{(x,y)} \begin{bmatrix} \lambda & \mu \\ \mu & \xi \end{bmatrix} \nabla_{(x,y)}. \tag{3}$$

To be rotationally symmetric, as well as symmetric (so that it is a quadratic form), Lemma 1 implies that

$$\lambda = \xi$$
$$\mu = 0.$$

It follows that the matrix in (3) is λI_2. □

The operator $f_x^2 + f_y^2$ is commonly used as a measure of the contrast across an intensity change. Notice that other popular measures of the contrast, such as $(f_x + f_y)^2$, $(f_x - f_y)^2$, or $\|f_x\| + \|f_y\|$, are not rotationally symmetric, and therefore respond differently to edges in different directions [32, p. 279].

We now consider linear and quadratic forms in $\partial^2/\partial x^2$, $\partial^2/\partial x\partial y$, $\partial^2/\partial y\partial x$, and $\partial^2/\partial y^2$. It is convenient to *not* assume $\partial^2/\partial x\partial y = \partial^2/\partial y\partial x$ for the developments that follow.

The first task is to find a matrix R^* so that

$$\begin{bmatrix} \dfrac{\partial^2}{\partial x^2} \\ \dfrac{\partial^2}{\partial x\partial y} \\ \dfrac{\partial^2}{\partial y\partial x} \\ \dfrac{\partial^2}{\partial y^2} \end{bmatrix} = R^* \begin{bmatrix} \dfrac{\partial^2}{\partial X^2} \\ \dfrac{\partial^2}{\partial X\partial Y} \\ \dfrac{\partial^2}{\partial Y\partial X} \\ \dfrac{\partial^2}{\partial Y^2} \end{bmatrix}. \tag{4}$$

The (i, j) element of the matrix R [4] will be denoted by r_{ij}. Applying the chain rule as before, but this time to relate the second derivatives in (X, Y) to those in (x, y), we find that the 4×4 matrix R^* can be written

$$R^* = \begin{bmatrix} r_{11}R^T & r_{21}R^T \\ r_{12}R^T & r_{22}R^T \end{bmatrix}. \tag{5}$$

DEFINITION 1. [4, p. 41]. Let $A = [a_{ij}]$ and $B = [b_{ij}]$ be $m \times m$ and $n \times n$ matrices respectively. The $mn \times mn$ matrix $A \otimes B$, called the *Kronecker product* of A and B, is defined by multiplying each element $a(i, j)$ of A by the matrix B, to form the block matrix

$$\begin{bmatrix} a_{11}B & a_{12}B & \dots & a_{1m}B \\ a_{21}B & a_{22}B & \dots & a_{2m}B \\ \vdots & \vdots & & \vdots \\ a_{m1}B & a_{m2}B & \dots & a_{mm}B \end{bmatrix}. \tag{6}$$

With this notation

$$R^* = R^T \otimes R^T$$

so that

$$R^* = \begin{bmatrix} c^2 & -sc & -sc & s^2 \\ sc & c^2 & -s^2 & -sc \\ sc & -s^2 & c^2 & -sc \\ s^2 & sc & sc & c^2 \end{bmatrix}. \tag{7}$$

[4] Recall the definition of the matrix R from (0).

Note that the elements of $A \otimes B$ are naturally indexed by 4-tuples

$$\{A \otimes B\}_{ijkl} = a_{ij} b_{kl}.$$

We state a number of simple properties of the $\otimes$ operation. They are essentially straightforward consequences of the properties of ordinary multiplication, and are stated without proof.

PROPOSITION 3.

$$(i) \qquad (A \otimes B)^T = A^T \otimes B^T$$

$$(ii) \qquad (A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$$

$$(iii) \qquad (A \otimes B) \otimes C = A \otimes (B \otimes C).$$

For the remainder of the paper, we restrict attention to the application of $\otimes$ to R and its transpose.

PROPOSITION 4. *The rotationally symmetric linear combinations of* $\partial^2/\partial x^2$, $\partial^2/\partial x \partial y$, $\partial^2/\partial y \partial x$, *and* $\partial^2/\partial y^2$ *are linear combinations of the Laplacian* $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2$, *and the smoothness measure* $\partial^2/\partial x \partial y - \partial^2/\partial y \partial x$.

Proof. Let the linear combination be

$$[\lambda \quad \mu \quad \nu \quad \xi] \begin{bmatrix} \dfrac{\partial^2}{\partial x^2} \\ \dfrac{\partial^2}{\partial x \partial y} \\ \dfrac{\partial^2}{\partial y \partial x} \\ \dfrac{\partial^2}{\partial y^2} \end{bmatrix}.$$

Following the proof of Proposition 1, the condition for rotational symmetry is

$$[\lambda \quad \mu \quad \nu \quad \xi] R^T \otimes R^T = [\lambda \quad \mu \quad \nu \quad \xi]$$

for all rotation matrices R and the corresponding rotation angle ϕ. Expanding $R^T \otimes R^T$ by (7), we find

$$[\lambda \quad \mu \quad \nu \quad \xi] \begin{bmatrix} c^2 & -sc & -sc & s^2 \\ sc & c^2 & -s^2 & -sc \\ sc & -s^2 & c^2 & -sc \\ s^2 & sc & sc & c^2 \end{bmatrix} = [\lambda \quad \mu \quad \nu \quad \xi]$$

so that

$$[\lambda \quad \mu \quad \nu \quad \xi]\begin{bmatrix} -s^2 & -sc & -sc & s^2 \\ sc & -s^2 & -s^2 & -sc \\ sc & -s^2 & -s^2 & -sc \\ s^2 & sc & sc & -s^2 \end{bmatrix} = [0 \quad 0 \quad 0 \quad 0].$$

It follows that

$$[\lambda - \xi \quad \mu + \nu \quad 0 \quad 0]\begin{bmatrix} -2s^2 & -2sc & -2sc & 2s^2 \\ 2sc & -2s^2 & -2s^2 & -2sc \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = [0 \quad 0 \quad 0 \quad 0].$$

The determinant of the upper left 2×2 submatrix is

$$(4s^4 + 4s^2c^2) = 4s^2.$$

Since this is not zero for all angles ϕ, $\lambda - \xi$ and $\mu + \nu$ are both zero. A basis for the infinite set of linear combinations satisfying these conditions is provided by setting λ and μ equal to one, which proves the Proposition. □

Before turning to quadratic forms, analogous to Proposition 2, we define a projection operator on $R^T \otimes R^T$ that makes explicit the assumption $f_{xy} = f_{yx}$.

DEFINITION 2. Let $D = [d_{ij}]$ be a 4×4 matrix. The *projection* of D is the 3×3 matrix D^*

$$\begin{bmatrix} d_{11} & (d_{12} + d_{13}) & d_{14} \\ (d_{21} + d_{31}) & (d_{22} + d_{32} + d_{23} + d_{33}) & (d_{24} + d_{34}) \\ d_{41} & (d_{42} + d_{43}) & d_{44} \end{bmatrix}.$$

That is, the second and third columns as well as the second and third rows are combined by addition.

PROPOSITION 5.

$$[a \quad b \quad b \quad c]D[a \quad b \quad b \quad c]^T$$

is equivalent to

$$[a \quad b \quad c]D^*[a \quad b \quad c]^T$$

where D^* is the projection of D.

The proof is completed by equating terms, and is omitted. We now give the main result of this section.

PROPOSITION 6. *The rotationally symmetric quadratic forms in $\partial^2/\partial x^2$, $\partial^2/\partial x \partial y$, $\partial^2/\partial y \partial x$, and $\partial^2/\partial y^2$ form a vector space. If $\partial^2/\partial x \partial y = \partial^2/\partial y \partial x$, the matrices* *associated with the rotationally symmetric quadratic forms project to* 3×3 *matrices of the form*

$$\begin{bmatrix} \alpha+\beta & 0 & \beta \\ 0 & 2\alpha & 0 \\ \beta & 0 & \alpha+\beta \end{bmatrix}.$$

It follows that the rotationally symmetric quadratic forms that satisfy $\partial^2/\partial x \partial y = \partial^2/\partial y \partial x$ *form a vector space that has the quadratic variation* $(\partial^2/\partial x^2)^2 + 2(\partial^2/\partial x \partial y)^2 + (\partial^2/\partial y^2)^2$ *and the square Laplacian* $(\partial^2/\partial x^2 + \partial^2/\partial y^2)^2$ *as a basis.*

Proof. Since the matrix in a quadratic form is defined to be symmetric, a quadratic form in $\partial^2/\partial x^2$, $\partial^2/\partial x \partial y$, $\partial^2/\partial y \partial x$, and $\partial^2/\partial y^2$ can be written

$$\begin{bmatrix} \dfrac{\partial^2}{\partial x^2} & \dfrac{\partial^2}{\partial x \partial y} & \dfrac{\partial^2}{\partial y \partial x} & \dfrac{\partial^2}{\partial y^2} \end{bmatrix} \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \begin{bmatrix} \dfrac{\partial^2}{\partial x^2} \\ \dfrac{\partial^2}{\partial x \partial y} \\ \dfrac{\partial^2}{\partial y \partial x} \\ \dfrac{\partial^2}{\partial y^2} \end{bmatrix}$$

where A and C are symmetric 2×2 matrices, and B is 2×2. As usual, the quadratic form is rotationally symmetric if and only if

$$R^T \otimes R^T \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} R^T \otimes R^T$$

where R is an arbitrary rotation matrix. It follows that

$$\begin{bmatrix} cR^T & sR^T \\ -sR^T & cR^T \end{bmatrix} \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \begin{bmatrix} cR^T & sR^T \\ -sR^T & cR^T \end{bmatrix}$$

and hence that

$$\begin{bmatrix} cR^TA + sR^TB^T & cR^TB + sR^TC \\ -sR^TA + cR^TB^T & -sR^TB + cR^TC \end{bmatrix} = \begin{bmatrix} cAR^T - sBR^T & sAR^T + cBR^T \\ cB^TR^T - sCR^T & sB^TR^T + cCR^T \end{bmatrix}.$$

Equating submatrices, we find that for all rotation angles ϕ

$$c(R^TA - AR^T) + s(R^TB^T + BR^T) = 0 \tag{8}$$

$$c(R^TC - CR^T) - s(B^TR^T + R^TB) = 0 \tag{9}$$

$$s(CR^T - R^TA) + c(R^TB^T - B^TR^T) = 0 \tag{10}$$

$$s(R^TC - AR^T) + c(R^TB - BR^T) = 0. \tag{11}$$

Consider (10) or (11) when $\phi = \pi/2$. Equating terms, we find that

$$\begin{aligned} a_{11} &= c_{22} \\ a_{22} &= c_{11} \\ a_{12} &= -c_{21} \\ a_{21} &= -c_{12}. \end{aligned} \tag{12}$$

Similarly, (8) or (9), when $\phi = \pi/2$, yields

$$b_{11} + b_{22} = 0. \tag{13}$$

Expanding (8) for general ϕ yields

$$b_{11} + a_{12} = 0 \tag{14}$$

$$b_{22} - a_{21} = 0 \tag{15}$$

$$b_{21} + b_{12} + a_{22} - a_{11} = 0. \tag{16}$$

Combining (12) through (16) we find that, to be rotationally symmetric, the matrix

$$\begin{bmatrix} A & B \\ B^T & C \end{bmatrix}$$

has the form

$$\begin{bmatrix} \alpha+\beta & \gamma & -\gamma & \beta \\ \gamma & \alpha-\delta & \delta & \gamma \\ -\gamma & \delta & \alpha-\delta & -\gamma \\ \beta & \gamma & -\gamma & \alpha+\beta \end{bmatrix}.$$

A matrix of this form projects to

$$\begin{bmatrix} \alpha+\beta & 0 & \beta \\ 0 & 2\alpha & 0 \\ \beta & 0 & \alpha+\beta \end{bmatrix}$$

where $\alpha = b_{12} - a_{11}$ and $\beta = b_{12}$. It is easy to show that linear combinations of matrices of this form are of the same form, so that the rotationally symmetric quadratic forms constitute a vector space. Clearly, the square Laplacian and the quadratic variation, corresponding to the cases $\alpha = 1$, $\beta = 0$, and $\alpha = 0$, $\beta = 1$, respectively, form a basis.

We show that the measure of smoothness of optical flow proposed by Horn and Schunck [17] is rotationally symmetric. Recall from Section 2 that the measure is defined by the operator

$$S(u, v) = \left(u_x^2 + u_y^2\right) + \left(v_x^2 + v_y^2\right).$$

We extend the Kronecker product operator $\otimes$ to vectors, and then show how to define $S(u, v)$ in terms of vector Kronecker products.

DEFINITION 3. (a) Let $\mathbf{a} = [a_1 \cdots a_m]$ and $\mathbf{b} = [b_1 \cdots b_n]$ be vectors. The *Kronecker product* of $\mathbf{a}$ and $\mathbf{b}$ is the mn dimensional vector $[a_1 b_1 \cdots a_1 b_n a_2 b_1 \cdots a_m b_n]$;

(b) By extension, if $\mathcal{O} = [\mathcal{O}_1 \ldots \mathcal{O}_m]$ is a vector of operators and $\mathbf{f} = [f_1 \ldots f_n]$ is a vector of functions, the Kronecker product of $\mathcal{O}$ and $\mathbf{f}$ is the mn dimensional vector of functions

$$[\mathcal{O}_1(f_1) \ldots \mathcal{O}_1(f_n) \ldots \mathcal{O}_m(f_n)].$$

The components u and v of optical flow are functions of x, y, and t. Recall that $\nabla_{(x,y)} = [\partial/\partial x \quad \partial/\partial y]^T$. According to Definition 3

$$\nabla_{(x,y)} \otimes [u \quad v]^T = \left[\frac{\partial u}{\partial x} \quad \frac{\partial u}{\partial y} \quad \frac{\partial v}{\partial x} \quad \frac{\partial v}{\partial y} \right]$$

so that

$$S(u, v) = \left(\nabla_{(x,y)} \otimes [u \quad v]^T\right) \cdot \left(\nabla_{(x,y)} \otimes [u \quad v]^T\right).$$

If the coordinate frame is rotated through ϕ by the matrix R, the optical flow components become $R[u \quad v]^T$. The Horn–Schunck measure is rotationally symmetric if and only if

$$(R \otimes R)^T (R \otimes R) = I_4$$

where I_4 is the 4×4 identity matrix. The rotational symmetry is a simple consequence of Proposition 3.

A rotationally symmetric operator has the general form

$$\mathcal{O}_{(x,y)}(\nabla, \nabla \otimes \nabla, \nabla \otimes \nabla \otimes \nabla, \ldots)$$

and its application to a rotationally symmetric function $f(x, y)$ has the form

$$\mathcal{O}_{(x,y)}(f(x, y)).$$

To see that this is rotationally symmetric, we rotate the coordinate frame to (X, Y) by a matrix R as before. Since $\mathcal{O}$ and f are rotationally symmetric, all the occurrences of R (including its Kronecker square, cube, and so on) introduced by the frame change can be deleted. It follows that the application of a rotationally symmetric operator to a rotationally symmetric function is itself rotationally symmetric. In particular, the $\Delta(G)$ filters of the Marr–Hildreth theory of edge detection are rotationally symmetric.

4. VISION AS A CONSERVATIVE PROCESS

The second theme of this paper is that a number of vision modules construct the *most conservative* interpretation that is consistent with the given data, and that is subject to an appropriate set of suitably formulated constraints. A major concern of computer vision has always been the isolation of constraints that enable the interpretation of an image. Constraints embody observations about the way the world is—at least, most of the time. Although such observations can be as specific as cataloging familiar figures and shapes, it has proved more fruitful to first uncover constraints that correspond to general observations that are widely applicable. Constraints are used together with the data computed from the image to construct an interpretation. The *representations* of the information from the image and the constraints determine, and are determined by, the interpretation process. For example, early blocks world programs represented constraints as catalogs of labelings, an approach that led naturally to search processes for interpretation [8, 20].

As computer vision has turned its attention to images of the natural world, constraints have concerned the smoothness of surfaces and movement. The relationship to boundary value problems of physics and mathematics suggests itself. The information computed from the image sets the boundary conditions, and the constraints determine which (and whether a) solution to the boundary value problem is found. Horn [15] solved an instance of Poisson's problem using Green's functions to determine the lightness of an image.

Following a different approach, Ullman [38] studied the perception of apparent motion generated by two successive frames consisting of isolated dots of equal intensity moving independently of each other. Without constraint, all possible pairings, or "correspondences," of dots in the first frame with dots in the second are equally likely. Ullman defined the "most likely" correspondence to be the one that minimized the sum

$$\sum_{\substack{1 \le i \le n \\ 1 \le j \le m}} x_{ij} q_{ij}$$

where n is the number of dots in the first frame, m is the number of dots in the second frame, and x_{ij} is one if the ith dot of the first frame P_i is paired with the jth dot of the second frame Q_j, else zero. The weight q_{ij} is the "cost" of pairing P_i with Q_j, and might, for example, be related to the image distance between the paired points. The problem of finding the minimal correspondence is considered in terms of integer programming. If correspondences are assumed to be covering mappings, the following linear constraints apply to the x_{ij}:

$$\forall i, \qquad 1 \le i \le n \sum_{1 \le j \le m} x_{ij} \ge 1$$

and

$$\forall j, \qquad 1 \le j \le m \sum_{1 \le i \le n} x_{ij} \ge 1.$$

Ullman restricted the set of Q_j that can be paired with P_i to those whose positions were close to P_i. Following Arrow, Hurwicz, and Uzawa [2], he set up the iterative scheme

$$\begin{aligned} x_{ij}^{t+1} &= u_i^t + v_j^t - q_{ij}^t \\ u_i^{t+1} &= \sum_{1 \le i \le n} x_{ij}^t - 1 \\ v_j^{t+1} &= \sum_{1 \le j \le m} x_{ij}^t - 1. \end{aligned}$$

The approach can be extended to mappings that are not one–one, as well as to continuous motion. A major problem with the approach is guaranteeing the convergence of the algorithm. This is determined largely by the properties of the costs q_{ij}, but this was not investigated, aside from a comment on the empirical determination of the q_{ij} (see also [39]).

One limitation of Ullman's approach is that it is restricted to minimizing a *known* linear objective function that is subject to linear constraints. The method can be extended to constrained nonlinear programming in which the goal is to minimize a known function $f(\mathbf{x})$ subject to a set of equality and inequality constraints of the form $g_i(\mathbf{x}) \leq 0$. In general, however, criteria based on other than intuition need to be found for selecting the function f to be minimized. To do this, one can apply the calculus of variations (see, for example, [9, Chap. IV]). The familiar differential calculus shows how to find a real valued parameter that minimizes some function. The calculus of variations extends the differential calculus by showing how one can determine a *function* f^*, which is subject to a given set of boundary conditions, and minimizes the integral

$$\mathfrak{F}(f) = \iint_G F(x, y, f, f_x, f_y, f_{xx}, f_{xy}, f_{yy})\, dx\, dy \tag{17}$$

over a given region of integration G.[5] The function F is called a "performance index" and generalizes the notion of cost function associated with linear and nonlinear programming. In the next section we consider the choice of a performance index for interpolating smooth surfaces from one-dimensional boundary conditions.

Associated with a variation problem of the form (17) is the *Euler equation*, which provides a necessary, though by no means sufficient, condition which a function f must satisfy if it is to minimize the variational integral $\mathfrak{F}(f)$. For the particular variational problem given in (17), the Euler equation is

$$F_f - \frac{\partial}{\partial x}F_{f_x} - \frac{\partial}{\partial y}F_{f_y} + \frac{\partial^2}{\partial x^2}F_{f_{xx}} + \frac{\partial^2}{\partial x \partial y}F_{f_{xy}} + \frac{\partial^2}{\partial y^2}F_{f_{yy}} = 0. \tag{18}$$

In the case where there is only a single dependent variable x, the partial derivatives are total and the Euler equation becomes

$$F_f - \frac{d}{dx}F_{f_x} + \frac{d^2}{dx^2}F_{f_{xx}} = 0. \tag{19}$$

There are two important considerations associated with the use of the calculus of variations. First, unlike the differential calculus, the existence of an extremum f^* of the integral given in (17) cannot be taken for granted. Courant and Hilbert [9, p. 173] note that "a characteristic difficulty of the calculus of variations is that problems which can be meaningfully formulated may not have solutions." Conditions for the existence of a minimum have recently been proposed by Grimson [12] and will be discussed in the next section.

Second, associated with any variational problem is a set of *natural boundary conditions* which imposes a necessary condition on any feasible solution to the Euler

[5]For simplicity of presentation, we restrict attention to functions f of one or two variables x, y.

equation at the boundary. Courant and Hilbert [9, p. 211] note that "in general, we can, by adding boundary terms or boundary integrals, essentially modify the natural boundary conditions without altering the Euler equations." Determining the "most conservative" solution means finding a performance index that guarantees the existence of an extremum function f^* and provides the tightest set of natural boundary conditions that are consistent with the given data.

The calculus of variations has recently been applied by a number of authors to interpolate plane and space curves and surfaces. We review the applications in that order. First, Horn [16] has recently determined the curve which passes through two specified points with specified orientation while minimizing

$$\int \kappa^2 \, ds \tag{20}$$

where κ is the curvature and s is the arc length. This is the true shape of a spline used in "lofting" [10, p. 228]. In a thin beam, curvature is proportional to the bending moment. The total elastic energy stored in a thin beam is therefore proportional to the integral of the square of the curvature. Since the shape taken on by a thin beam is the one which minimizes the internal strain energy, the curve that solves (20) is called the "curve of least energy." The variational problem is to minimize

$$\int \frac{f_{xx}^2}{\left(1+f_x^2\right)^{5/2}} \, dx.$$

This has the form of (17). Horn [16, p. 19] shows that the Euler equation is

$$-c\kappa = \sqrt{\cos \psi}$$

where ψ is the angle between the tangent to the curve and the axis of symmetry. The solution to this differential equation is an incomplete elliptic integral of the first kind. Brady, Grimson, and Langridge [6] consider a "small angle" approximation to the curve of least energy, in which first derivatives can be ignored. The performance index that they use is f_{xx}^2, for reasons that will become evident in the next section. They find that in that case the solution is a cubic. Horn [16, p. 2] notes that the fact that a curve has near minimum energy does not mean that it lies close to the curve of minimum energy. Note that the existence of the curve of least energy is guaranteed as Horn has derived an analytical formula for it. Approximations to it, such as the one by Brady *et al.*, are similarly guaranteed to exist.

Barrow and Tenenbaum [3] investigate the problem of interpreting a line as the image of a space curve that is an occluding boundary. They observe that the problem has two parts: (i) determining the tangent vector $\mathbf{t}$ at each point on the space curve, and (ii) determining the surface normal at each point, given that it is constrained to be orthogonal to the tangent.

They suggest minimizing a performance index F that is a function of the curvature κ and the torsion τ (possibly together with their derivatives) and expresses a suitable notion of "smoothness." They first consider uniformity of curvature as a measure of smoothness, that is, $F = d\kappa/ds = \kappa_1$, where s measures distance along the space curve. They reject this measure on the grounds that κ_1 can be made arbitrarily small by "stretching out the space curve so that it approaches a twisting straight line." To overcome this difficulty, they propose that the space curve should also be "as planar as possible or, more precisely, that the integral of its torsion should be minimized."

Barrow and Tenenbaum finally suggest finding the space curve that projects to the given image line and minimizes the performance index $[d(\kappa\mathbf{b})/ds]^2$, where $\mathbf{b}$ is the binormal. They report that an algorithm based on their analysis produced the "correct 3-D interpretations for simple and closed curves, such as an ellipse, which was interpreted as a circle." However, they note that the rate of convergence was slow and dependent on the initial data. No consideration is given to the Euler equations, to the existence of an extremum given a line drawing $\{x(s), y(s)\}$, or to the natural boundary conditions associated with the performance index $[d(\kappa\mathbf{b})/ds]^2$. Empirical evidence that the method works on a number of simple test cases is encouraging; but there is no analysis of the scope of the method.

In the same paper, Barrow and Tenenbaum [3] consider the interpolation of a smooth surface from depth and local surface orientation values at all points along the surface boundary. Their approach is to "seek a technique that yields exact reconstructions for the special symmetric cases of spherical and cylindrical surfaces, as well as intuitively reasonable reconstructions for other smooth surfaces" [3]. They observe that if $\mathbf{n}$ is the surface normal of a cylinder, then the x and y components of the normal $\mathbf{n}_x$ and $\mathbf{n}_y$ are linear functions of x and y, so long as the axis of the cylinder lies in the x–y plane. This observation forms the basis of an algorithm to estimate the surface normal by least squares fitting of the parameters of the partial derivatives of the normal. As before, no analysis is given of the Euler equation, the natural boundary conditions, or the convergence of their algorithm for different types of surface.

5. A PERFORMANCE INDEX FOR SURFACE INTERPOLATION

In the review of the application of the calculus of variations to visual perception in the previous section we drew attention to three important considerations. First, the Euler equations provide a necessary condition on possible extremal functions. Second, the existence of an extremum cannot be taken for granted, even when the minimization problem seems plausible on some grounds. Third, the natural boundary conditions impose a necessary condition on any feasible solution to the Euler equation at the boundary. The most thorough analysis of the second of these problems in computer vision, framed in the context of surface interpolation, is due to Grimson [12], who proves the following theorem.

THEOREM (Grimson, see Rudin [33]). *Suppose there exists a complete seminorm F on a space of functions $\mathfrak{F}$, and that F satisfies the parallelogram law. Then, every nonempty closed convex set $\mathfrak{S} \subseteq \mathfrak{F}$ contains a unique element f^* of minimal norm $F(f^*)$, up to possibly an element of the null space of F.*

A seminorm F is a function $V \mapsto \mathfrak{R}^+$ from a vector space V to the positive real numbers that satisfies

$$F(v + w) \le F(v) + F(w)$$

$$F(av) = |a|F(v).$$

Informally, a seminorm is a generalization of the Euclidean metric, and provides a measure of a vector. The second condition generalizes the triangle inequality, for example. The null space of the seminorm F consists of all those vectors v_0 that map to zero. Since

$$F(v + v_0) = F(v)$$

any element of the null space can be added to a vector of minimal norm to yield another vector of minimal norm. Hence the qualifying phrase "unique... up to possibly an element of the null space of F." The parallelogram law states that

$$[F(v + w)]^2 + [F(v - w)]^2 = 2[F(v)]^2 + 2[F(w)]^2$$

for all vectors v, w. Finally, the seminorm is complete if all Cauchy sequences converge. As is well known, the elements of vector spaces can be functions. This enables Grimson to prove the following corollary, that guarantees the existence of an extremum function in calculus of variations "most conservative" interpolation problems.

COROLLARY (Grimson [12]). *Let the set of known points be* $\{(x_i, y_i, z_i) | 1 \leq i \leq n\}$. *Let* $\mathfrak{F}$ *be a vector space of possible functions on* $\Re^2$ *and let* $\mathfrak{S}$ *be the subset of* $\mathfrak{F}$ *that interpolates the known data. That is, for all functions* $f \in \mathfrak{S}$, $f(x_i, y_i) = z_i$. *Let F be a complete seminorm on* $\mathfrak{S}$ *that satisfies the parallelogram law. Then there exists a unique (up to the null space of F) function* f^* *that interpolates the data and has minimal norm. In particular, if F is a performance index then there is a function* f^* *that minimizes the integral*

$$\mathfrak{F}(f) = \int F.$$

In short, if the conditions of the corollary are fulfilled, the existence of a "most conservative" surface that meets the boundary conditions is guaranteed. As we shall see, the condition of being a seminorm is the most restrictive required of the performance index. The conditions are sufficient to guarantee the existence of a minimum, but they are not necessary. For example, κ^2 is not a seminorm,[6] nevertheless Horn's [16] analysis shows that there is a unique minimum. It is far from clear whether Barrow and Tenenbaum's [3] analyses of curve and surface interpolation have a guaranteed minimum in all cases.

Grimson notes that several intuitively plausible performance indices are not seminorms. For example, the two most popular measures of curvature are not. Suppose that κ_1 and κ_2 are the principal curvatures of a surface [10, p. 111]; then the Gaussian curvature κ_g is the product $\kappa_1\kappa_2$ and the mean curvature κ_m is the sum $\kappa_1 + \kappa_2$. For a surface $f(x, y)$

$$\kappa_g(f) = \frac{f_{xx}f_{yy} - f_{xy}^2}{\left(1 + f_x^2 + f_y^2\right)^2}.$$

[6]Which is why Brady *et al.* [6], used the small angle approximation f_{xx}^2.

Since the curvatures can be negative, while a seminorm is required to be positive, it is necessary to investigate

$$\left\{ \int \kappa_g^2 \, dx \, dy \right\}^{1/2}.$$

Grimson [12] observes that $\kappa_g^2(af) \neq |a|\kappa_g^2(f)$ because of the denominator. If f_x and f_y are small, the denominator is approximately equal to one, and the numerator is a seminorm. Note that it is

$$f_{xx} f_{yy} - f_{xy}^2. \tag{21}$$

Grimson shows that the mean curvature κ_m is also not a seminorm for exactly the same reason. The analogous small angle approximation is

$$(f_{xx} + f_{yy})^2 = (\Delta f)^2$$

the square Laplacian, which is a seminorm. We find it convenient to denote the square Laplacian by F_l. Grimson [12] chooses the quadratic variation

$$f_{xx}^2 + 2f_{xy}^2 + f_{yy}^2$$

on the grounds that its null space, consisting of all linear functions, is smaller than the null space of the square Laplacian. If we denote the quadratic variation by F_q, we see that the approximation to the Gaussian curvature given in (21) is $(F_l - F_q)/2$.

How shall we choose a performance index for surface interpolation, given that it has to satisfy the conditions of the corollary? We have exhibited three candidates; are there more? Notice first that each of the seminorms given above are quadratic forms in f_{xx}, f_{xy}, and f_{yy}. It is easy to show that *any* quadratic form satisfies the seminorm and parallelogram conditions, so there is an infinite set of plausible seminorms to use to find the "most conservative" interpolated surface. We need an extra condition, and the one we choose is rotational symmetry, since we suppose that surface interpolation is an isotropic process. Proposition 6 of Section 3 shows that the rotationally symmetric quadratic forms in f_{xx}, f_{xy}, and f_{yy} form a vector space that has the square Laplacian and the quadratic variation as a basis. The choice of which performance index to use is thus effectively reduced to the square Laplacian, the quadratic variation, and linear combinations of them. How shall we choose between those two? In the light of our earlier discussion, two criteria suggest themselves: the Euler equations and the natural boundary conditions.

PROPOSITION 7. *All rotationally symmetric quadratic forms lead to an identical Euler equation*

$$\Delta^2(f) = 0.$$

Proof. We exploit the fact that the square Laplacian and the quadratic variation are a basis of the rotationally symmetric quadratic forms.

(a) *Square Laplacian.* The performance index is

$$F_l = \left(f_{xx} + f_{yy}\right)^2.$$

By (18) the Euler equation is

$$\frac{\partial^2}{\partial x^2}\{2(f_{xx} + f_{yy})\} + \frac{\partial^2}{\partial y^2}\{2(f_{xx} + f_{yy})\} = 0$$

that is,

$$(\Delta f)^2 = 0$$

as required.

(b) *Quadratic variation.* The Euler equation is

$$2f_{xxxx} + 4f_{xyxy} + 2f_{yyyy} = 0$$

that is,

$$(\Delta f)^2 = 0$$

provided that f is continuous of fourth order.

(c) *Linear combinations of F_l and F_q.* Linear combinations clearly give rise to the identical Euler equation.

The gist of Proposition 7 is that there is no difference between F_q and F_l in the interior away from the boundary conditions. We can see that result of Proposition 7 in an interesting alternative way. Recall that

$$F_l - F_q = 2\left(f_{xx}f_{yy} - f_{xy}^2\right)$$

is the seminorm approximation to the Gaussian curvature (Eq. 21). The latter expression is an instance of a *divergence expression*, and Courant and Hilbert [9, p. 196] note, "If the difference between the integrands of two variational problems is a divergence expression, then the Euler equations and therefore the families of extremals are identical for the two variational problems."

Since F_q and F_l have identical Euler equations, we analyze their natural boundary conditions to choose between them. We could approach this problem directly, but a more revealing route is available. Courant and Hilbert [9, p. 250] consider the statics of a thin plate. In particular, they determine the shape it assumes for a given force $p(s)$ along its boundary Γ and bending moments $m(s)$ normal to its boundary.

Courant and Hilbert note that the energy stored in the plate is the integral of a quadratic form in the principal curvatures κ_1 and κ_2 of the surface, a result which can be derived from noting that the elastic energy stored in a thin strip (corresponding to any normal section) is proportional to the square curvature. It follows that the stored energy is locally

$$\begin{aligned}
\mathcal{E}_1 &= \alpha\left(\kappa_1^2 + \kappa_2^2\right) + 2\beta\kappa_1\kappa_2 \\
&= \alpha\left(\kappa_1 + \kappa_2\right)^2 + 2(\beta - \alpha)\kappa_1\kappa_2 \\
&= \alpha\kappa_m + 2(\beta - \alpha)\kappa_g \\
&\sim \alpha F_l + 2(\beta - \alpha)\frac{(F_l - F_q)}{2} \\
&= \beta F_l + (\alpha - \beta)F_q \\
&= \alpha\left(\mu F_l + (1 - \mu)F_q\right)
\end{aligned}$$

where $\mu = \beta/\alpha$. It follows that the energy stored in the thin plate is a convex linear combination of the square Laplacian and the quadratic variation, which formally establishes its connection to the visual perceptual problem studied here. Observe that setting the weight $\mu = 1$ gives the square Laplacian, while setting it equal to zero gives the quadratic variation. Note also that this expression for the stored energy makes use of the small angle approximation to the curvature used in (21).

A second source of stored energy derives from the boundary conditions that are represented as a function $p(s)$ along the boundary Γ of the plate and a bending moment $m(s)$ applied normal to the plate. Courant and Hilbert [9, p. 251] show that the natural boundary conditions associated with the plate are

$$-\Delta f + (1 - \mu)\left(f_{xx}x_s^2 + 2f_{xy}x_s y_s + f_{yy}y_s^2\right) = p(s)$$

$$\frac{\partial}{\partial n}\Delta f + (1 - \mu)\frac{\partial}{\partial s}\left(f_{xx}x_s x_n + f_{xy}(x_s y_n + x_n y_s) + f_{yy}y_s y_n\right) = m(s)$$

that is,

$$-\Delta f + (1 - \mu)\left([x_s y_s]H[x_s y_s]^T\right) = p(s)$$

$$\frac{\partial}{\partial n}\Delta f + (1 - \mu)\frac{\partial}{\partial s}\left([x_n y_n]H[x_s y_s]^T\right) = m(s)$$

where H is the Hessian matrix

$$\begin{bmatrix} f_{xx} & f_{xy} \\ f_{xy} & f_{yy} \end{bmatrix}.$$

Gladwell and Wait [11] quote a version of this result due to Agmon [1], such that the biharmonic operator, which we showed was the natural boundary condition for the surface interpolation problem, has Dirichlet forms that are linear combinations of the square Laplacian and the quadratic variation. As an example of the constraint, consider a straight line contour aligned with the x axis. Then $[x_s y_s] = [1\ 0]$ and $[x_n y_n] = [0\ 1]$. The natural boundary conditions reduce to

$$f_{yy} + \mu f_{xx} = -p(s)$$

$$f_{yyy} + (2 - \mu)f_{yxx} = m(s).$$

The constraint is tightest when μ is not equal to one. A similar result can be obtained for a straight line contour inclined at an angle α to the x axis. The first of the natural boundary conditions is

$$f_{xx}(\sin^2\alpha + \mu\cos^2\alpha) + f_{yy}(\cos^2\alpha + \mu\sin^2\alpha) + (1 - \mu)\sin 2\alpha f_{xy}.$$

If $\mu = 1$, there is no constraint from the cross derivative. If μ is not equal to 1, at most one of the terms can be zero. We conclude that, for interpolation problems in which the small angle approximations used throughout our analysis hold, it is preferable to choose μ not equal to one, that is, to not use the square Laplacian as a performance index. The quadratic variation is an obvious choice, but so are linear combinations of the square Laplacian and the quadratic variation for which μ is not equal to one. Grimson [12] chooses the quadratic variation since its null space is smaller than that of the square Laplacian. This is a precise way of saying that it imposes a tighter constraint. For example, the function $f(x, y) = xy$ is the null space of the square Laplacian but not in the null space of the quadratic variation. Since the quadratic variation has the smallest null space among the linear combinations of the square Laplacian and quadratic variation, this is an additional reason for choosing it. We would further expect that any differences between the quadratic variation and the square Laplacian would show up near the given boundary data but not in the interior, far removed from the boundary. This is what Grimson finds in a set of examples that compare surfaces interpolated using the quadratic variation and the square Laplacian.

ACKNOWLEDGMENTS

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. The authors thank Eric Grimson for comments on drafts of this article.

REFERENCES

1. S. Agmon, *Lectures on Elliptic Boundary Value Problems*, Van Nostrand, Princeton, New Jersey, 1965.
2. K. J. Arrow, L. Hurwicz, and H. Uzawa, *Studies in Linear and Nonlinear Programming*, Stanford Univ. Press, Palo Alto, Calif., 1958.
3. H. G. Barrow and J. M. Tenenbaum, Interpreting line drawings as three dimensional surfaces, *Artif. Intell.* **17**, 1981, 75–116.
4. A. ben Israel and T. N. Greville, *Generalized Inverses: Theory and Applications*, Wiley, New York, 1974.
5. T. O. Binford, Inferring surfaces from images, *Artificial Intelligence* **17**, 1981.
6. Michael Brady, W. E. L. Grimson, and D. Langridge, The shape of subjective contours, *Proc. AAAI* **1**, 1980, 15–17.
7. M. J. Brooks, Surface normals from closed paths, *Proc. Int. Jt. Conf. Artif. Intell.* **6**, 1979, 98–101.
8. M. B. Clowes, On seeing things, *Artificial Intelligence* **2**, 1971, 79–112.
9. R. Courant and D. Hilbert, *Methods of Mathematical Physics*, Wiley–Interscience, New York, 1953.
10. I. D. Faux and M. J. Pratt, *Computational Geometry for Design and Manufacture*, Ellis Horwood, Chichester, U.K., 1979.
11. I. Gladwell and R. Wait (Eds.), *Survey of Numerical Methods for Partial Differential Equations*, Oxford Univ. Press (Clarendon), London/New York, 1979.
12. W. E. L. Grimson, *From Images to Surfaces: A Computational Study of the Human Early Visual System*, MIT Press, Cambridge, Mass. 1981.