Digital Step Edges from Zero Crossing of Second Directional Derivatives

ROBERT M. HARALICK, FELLOW, IEEE

Abstract—We use the facet model to accomplish step edge detection. The essence of the facet model is that any analysis made on the basis of the pixel values in some neighborhood has its final authoritative interpretation relative to the underlying gray tone intensity surface of which the neighborhood pixel values are observed noisy samples. With regard to edge detection, we define an edge to occur in a pixel if and only if there is some point in the pixel's area having a negatively sloped zero crossing of the second directional derivative taken in the direction of a nonzero gradient at the pixel's center. Thus, to determine whether or not a pixel should be marked as a step edge pixel, its underlying gray tone intensity surface must be estimated on the basis of the pixels in its neighborhood. For this, we use a functional form consisting of a linear combination of the tensor products of discrete orthogonal polynomials of up to degree three. The appropriate directional derivatives are easily computed from this kind of a function.

Upon comparing the performance of this zero crossing of second directional derivative operator with the Prewitt gradient operator and the Marr-Hildreth zero crossing of the Laplacian operator, we find that it is the best performer; next is the Prewitt gradient operator. The Marr-Hildreth zero crossing of the Laplacian operator performs the worst.

Index Terms—Edge operator, facet model, image processing, image segmentation, zero crossings of second directional derivative.

I. Introduction

WHAT IS an edge in a digital image? The first intuitive notion is that a digital edge occurs on the boundary between two pixels when the respective brightness values of the two pixels are significantly different. Here "significantly different" may depend upon the distribution of brightness values around each of the pixels.

We often point to a region on an image and say this region is brighter than its surrounding area. Having noticed this we would then say that an edge exists between each pair of neighboring pixels where one pixel is inside the brighter region and the other is outside the region. *Such edges are referred to as step edges.*

Step edges are not the only kind of edge. If we scan through a region in a left-right manner observing the brightness values steadily increasing and then after a certain point observe that the brightness values are steadily decreasing, we are likely to say that there is an edge at the point of change from increasing to decreasing brightness values. *Such edges are called roof edges.*

Therefore, it is clear from our use of the word "edge" that edge refers to places in the image where there appears to be a jump in brightness value or a local extremum in brightness value derivative. Jumps in brightness values are the kinds of edges originally detected by Roberts [18]. Relative extrema of first derivative in a one-dimensional form is used by Ehrich and Schroeder [6] and in an isotropic two-dimensional suboptimal form by Marr and Hildreth [13].

In some sense this summary statement about edges is quite revealing since in a discrete array of brightness values there are jumps, between neighboring brightness values if the brightness values are different, even if only slightly different. Perhaps more to the heart of the matter, there exists no calculus definition of derivative for a discrete array of brightness values.

One clear way to interpret jumps in value or local extrema of derivatives when referring to a discrete array of values is to assume that the discrete array of values comes about by sampling a real-valued function f defined on the domain of the image which is a bounded and connected subset of the real plane R^2. The finite difference typically used in the numerical approximation of first-order derivatives are usually based on assuming the function f to be linear. From this point of view, the jumps in value or extrema in derivative really must refer to points of high first derivative of f or to points of relative extrema in the second derivatives of f. Edge detection must then involve fitting a function to the sample values. Prewitt [17] was the first to suggest the fitting idea. Hueckel [11], [12], Brooks [3], Haralick [8], Haralick and Watson [10], Morgenthaler and Rosenfeld [15], Zucker and Hummel [20], and Morgenthaler [14] all use the surface fit concept in determining edges.

Edge finders should then regard the digital picture function as a sampling of the underlying function f, where some kind of random noise has been added to the true function values. To do this, the edge finder must assume some parametric form for the underlying function f, use the sampled brightness values of the digital picture function to estimate the parameters, and finally make decisions regarding the locations of discontinuities and the locations of relative extrema of partial derivatives based on the estimated values of the parameters.

Of course, it is impossible to determine the true locations of discontinuities in value or relative extrema in derivatives directly from a sampling of the function values. The locations are estimated analytically after doing function approximation. Sharp discontinuities can reveal themselves in high values for estimates of first partial derivatives. Relative extrema in first directional derivative can reveal themselves as zero crossings of the second directional derivative. Thus, if it were known that the first and second partial derivatives of any possible underly-

Manuscript received February 9, 1982; revised December 20, 1982.

The author is with the Departments of Electrical Engineering and Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061.

Reprinted from *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume PAMI-6, Number 1, January 1984, pages 58-68. Copyright © 1984 by The Institute of Electrical and Electronics Engineers, Inc.

ing image function had known bounds, then any estimated first- or second-order partials which exceed these known bounds must be due to discontinuities in value or in derivative of the underlying function. This is the basis for the gradient magnitude and Laplacian magnitude edge detectors which have appeared in the literature: detect an edge if the gradient is high enough.

However, edges can be weak but well localized. Such edges, as well as the strong edges just discussed, manifest themselves as local extrema of the directional derivative of the estimated gray tone intensity function taken across the edge. This idea for edges is the basis of the edge detector discussed here. In this paper, we assume that in each neighborhood of the image the underlying gray tone intensity function f takes the parametric form of a polynomial in the row and column coordinates and that the sampling producing the digital picture function is a regular equal interval grid sampling of the square plane which is the domain of f. Thus, in each neighborhood f takes the form

$$f(r, c) = k_1 + k_2 r + k_3 c + k_4 r^2 + k_5 rc + k_6 c^2 + k_7 r^3 + k_8 r^2 c + k_9 rc^2 + k_{10} c^3 .$$

As just mentioned, we place edges not at locations of high gradient, but at locations of spatial gradient maxima. More precisely, a pixel is marked as an edge pixel if in the pixel's immediate area there is a zero crossing of the second directional derivative taken in the direction of the gradient [9] and the slope of the zero crossing is negative. Thus, this kind of edge detector will respond to weak but spatially peaked gradients.

The underlying functions from which the directional derivatives are computed are easy to represent as linear combinations of the polynomials in any polynomial basis set. A polynomial basis set which permits the independent estimation of each coefficient would be the easiest to use. Such a basis is discussed in the appendix.

Section II discusses how the discretely sampled data values are used to estimate the coefficients of the linear combinations: coefficient estimates for exactly fitting or estimates for least square fitting are calculated as linear combinations of the sampled data values.

Having used the pixel values in a neighborhood to estimate the underlying polynomial function we can now determine the value of the partial derivatives at any location in the neighborhood and use those values in edge finding. Having to deal with partials in both the row and column directions makes using these derivatives a little more complicated than using the simple derivatives of one-dimensional functions. Section III discusses the directional derivative, how it is related to the row and column partial derivatives, and how the coefficients of the fitted polynomial get used in the edge detector. In Section IV we discuss the statistical confidence of the estimate of edge existence and the edge angle. In Section V we show results comparing the directional derivative zero crossing edge operator with the generalized Prewitt gradient operator and the related Marr-Hildreth zero crossing of the Laplacian operator.

II. Fitting Data with Discrete Orthogonal Polynomials

Let an index set R with the symmetry property $r \in R$ implies $-r \in R$ be given. Let the number of elements in R be N. Using the construction technique in the Appendix, we may construct the set $\{P_0(r), \cdots, P_{N-1}(r)\}$ of discrete orthogonal polynomials over R. Using the tensor product technique also discussed in the Appendix we can construct discrete orthogonal polynomials over a two-dimensional neighborhood. Some one- and two-dimensional discrete orthogonal polynomials are shown as follows:

Index Set	Discrete Orthogonal Polynomial Set
$\{-1/2, 1/2\}$	$\{1, r\}$
$\{-1, 0, 1\}$	$\{1, r, r^2 - 2/3\}$
$\{-3/2, -1/2, 1/2, 3/2\}$	$\{1, r, r^2 - 5/4, r^3 - 41/20r\}$
$\{-2, -1, 0, 1, 2\}$	$\{1, r, r^2 - 2, r^3 - 17/5,$ $r^4 + 3r^2 + 72/35\}$
$\{-1, 0, 1\} \times \{-1, 0, 1\}$	$\{1, r, c, r^2 - 2/3, rc, c^2 - 2/3,$ $r(c^2 - 2/3), c(r^2 - 2/3),$ $(r^2 - 2/3)(c^2 - 2/3)\}.$

For each $r \in R$, let a data value $d(r)$ be observed. The exact fitting problem is to determine coefficients $a_0, \cdots, a_{N-1}$ such that

$$d(r) = \sum_{n=0}^{N-1} a_n P_n(r). \tag{1}$$

The approximate fitting problem is to determine coefficients $a_0, \cdots, a_K$, $K \leqslant N - 1$ such that

$$e^2 = \sum_{r \in R} \left[d(r) - \sum_{n=0}^{K} a_n P_n(r) \right]^2$$

is minimized. In either case the result is

$$a_m = \sum_{r \in R} P_m(r) d(r) / \sum_{s \in R} P_m^2(s). \tag{2}$$

The exact fitting coefficients and the least squares coefficients are identical for $m = 0, \cdots, K$. Similar equations hold for the two-dimensional case.

Equation (2) means that each fitting coefficient a_m can be computed as linear combination of the data values. For each index r in the index set, the data value $d(r)$ is multiplied by the weight

$$P_m(r) / \sum_{s \in R} P_m^2(s)$$

which is just an appropriate normalization of an evaluation of the polynomial P_m at the index r. Figs. 1 and 2 show these weights for the 3 × 3 and 4 × 4 neighborhoods.

Once the fitting coefficients a_k, $k = 1, \cdots, K$, have been computed, the estimated polynomial $Q(r)$ is given by

$$Q(r) = \sum_{n=0}^{K} a_n P_n(r).$$

1/9

1	1	1
1	1	1
1	1	1

1

1/6

-1	-1	-1
0	0	0
1	1	1

r

1/6

-1	0	1
-1	0	1
-1	0	1

c

1/6

-1	-1	-1
-2	-2	-2
1	1	1

$r^2-2/3$

1/4

1	0	-1
0	0	0
-1	0	1

rc

1/6

1	-2	1
1	-2	1
1	-2	1

c

$c^2-2/3$

1/4

-1	0	1
2	0	-2
-1	0	1

$(r^2-2/3)c$

1/4

-1	2	-1
0	0	0
1	-2	1

$(c^2-2/3)r$

1/4

1	-2	1
-2	4	-2
1	-2	1

$(r^2-2/3)\ (c^2-2/3)$

Fig. 1. Illustrates the 9 masks for the 3 × 3 window.

1/16

1	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

1

1/40

-3	-3	-3	-3
-1	-1	-1	-1
1	1	1	1
3	3	3	3

r

1/40

-3	-1	1	3
-3	-1	1	3
-3	-1	1	3
-3	-1	1	3

c

1/16

1	1	1	1
-1	-1	-1	-1
-1	-1	-1	-1
1	1	1	1

$r^2- 5/4$

1/100

9	3	-3	-9
3	1	-1	-9
-3	-1	1	3
-9	-3	3	9

rc

1/16

1	-1	-1	1
1	-1	-1	1
1	-1	-1	1
1	-1	-1	1

$c^2- 5/4$

Fig. 2. Illustrates the masks used to obtain the coefficients of all polynomials up to the quadratic ones for a 4 × 4 window.

This equation permits us to interpret $Q(r)$ as a well-behaved real-valued function defined on the real line. To determine

$$\frac{dQ}{dr}\,(r_0)$$

we need only to evaluate

$$\sum_{n=0}^{N} a_n \frac{dP_n}{dr}\,(r_0). \tag{3}$$

In this manner, the estimate for any derivative at any point may be obtained. Similarly for any definite integrals. Section IV gives a statistical analysis of the errors of the estimate.

$$g(r,c)=a_{00}+a_{10}r+a_{01}c$$

	-1	-1	-1
1/6	0	0	0
	1	1	1

$$g(r,c)= a_{00}+a_{10}r+a_{01}c$$
$$a + a_{20}(r^2-2/3)+a_{11}rc +a_{02}(c^2-2/3)$$
$$+ a_{21}(r^2-2/3)c + a_{12}(c^2- 2/3)r$$

	0	-1	0
1/2	0	0	0
	0	1	0

Fig. 3. Illustrates that the kernel mask used to estimate a quantity such as row derivative can depend on the order of the assumed model.

Beaudet [2] uses this technique for estimating derivatives employed in rotationally invariant image operators.

It should be noted that the kernel used to estimate a derivative depends on the neighborhood size, the order of the fit, and the basis functions used for the fit. These parameters constitute the assumed model. The importance of the model is illustrated by the example of Fig. 3. This difference means that the model used must be justified, the justification being that it is a good fit to the data. From this we learn that the use of first-order models to estimate first-order partials as is often done may not produce correct results if the first-order fit is not good enough. In particular, it may require a third-order model to get good estimates of first-order partial derivatives.

III. The Directional Derivative Edge Finder

We stated in the introduction that a digital step edge occurs at pixels having a negatively sloped zero crossing of the second directional derivative taken in the direction of the gradient. In this section we discuss the relationship between the directional derivatives and the coefficients from the polynomial fit.

We denote the directional derivative of f at the point (r, c) in the direction α by $f'_\alpha(r, c)$. It is defined as

$$f'_\alpha(r, c) = \lim_{h\to 0} \frac{f(r + h \sin \alpha, c + h \cos \alpha) - f(r, c)}{h}. \tag{4}$$

The direction angle α is the clockwise angle from the column axis. It follows directly from this definition that

$$f'_\alpha(r, c) = \frac{\partial f}{\partial r}(r, c) \sin \alpha + \frac{\partial f}{\partial c}(r, c) \cos \alpha. \tag{5}$$

We denote the second directional derivative of f at the point (r, c) in the direction α by $f''_\alpha(r, c)$ and it quickly follows by substituting f'_α for f in (5) that

$$f''_\alpha = \frac{\partial^2 f}{\partial r^2} \sin^2 \alpha + \frac{2\partial^2 f}{\partial rc} \sin \alpha \cos \alpha + \frac{\partial^2 f}{\partial c^2} \cos^2 \alpha. \tag{6}$$

Taking f to be a cubic polynomial in r and c which can be estimated by the discrete orthogonal polynomial fitting procedure, we can compute the gradient of f and the gradient direction angle α at the center of the neighborhood used to estimate f. In order for our notation to be invariant to the different discrete orthogonal polynomials which result from different neighborhood sizes, we rewrite this cubic in canonical form as

$$\begin{aligned} f(r, c) = k_1 &+ k_2 r + k_3 c \\ &+ k_4 r^2 + k_5 rc + k_6 c^2 \\ &+ k_7 r^3 + k_8 r^2 c + k_9 rc^2 + k_{10} c^3. \end{aligned} \tag{7}$$

We obtain the angle by α by

$$\begin{aligned} \sin \alpha &= k_2/\sqrt{(k_2^2 + k_3^2)} \\ \cos \alpha &= k_3/\sqrt{(k_2^2 + k_3^2)}. \end{aligned} \tag{8}$$

At any point (r, c), the second directional derivative in the direction α is given by

$$\begin{aligned} f''_\alpha(r, c) = {} & (6k_7 \sin^2 \alpha + 4k_8 \sin \alpha \cos \alpha + 2k_9 \cos^2 \alpha) r \\ & + (6k_{10} \cos^2 \alpha + 4k_9 \sin \alpha \cos \alpha + 2k_8 \sin^2 \alpha) c \\ & + (2k_4 \sin^2 \alpha + 2k_5 \sin \alpha \cos \alpha + 2k_6 \cos^2 \alpha). \end{aligned} \tag{9}$$

We wish to only consider points (r, c) on the line in direction α. Hence, $r = \rho \sin \alpha$ and $c = \rho \cos \alpha$. Then

$$\begin{aligned} f''_\alpha(\rho) = {} & 6\,[k_7 \sin^3 \alpha + k_8 \sin^2 \alpha \cos \alpha \\ & + k_9 \sin \alpha \cos^2 \alpha + k_{10} \cos^3 \alpha]\, \rho \\ & + 2\,[k_4 \sin^2 \alpha + k_5 \sin \alpha \cos \alpha + k_6 \cos^2 \alpha] \\ = {} & A\rho + B. \end{aligned} \tag{10}$$

If for some ρ, $|\rho| < \rho_0$, where ρ_0 is slightly smaller than the length of the side of a pixel, $f''_\alpha(\rho) < 0, f''_\alpha(\rho) = 0$ and $f'_\alpha(\rho) \neq 0$ we have discovered a negatively sloped zero crossing of the estimated second directional derivative taken in the estimated direction of the gradient and we mark the center pixel of the neighborhood as an edge pixel.

IV. Statistical Analysis

Noise induces a randomness in the least squares coefficients which then induces a randomness in the estimated gradient value, the estimated angle of the gradient, and the estimated location of the zero crossing. In this section we show how to handle this randomness by appropriate hypothesis tests or confidence interval estimation.

A. General Model

We let P_n, $n = 1, \cdots, N$, denote the discrete orthonormal basis functions, η denote the independent and identically distributed noise, and g denote the gray tone intensity function. Under this model, each neighborhood of the observed image can be written as

$$g(r, c) = \sum_{n=1}^{N} a_n P_n(r, c) + \eta(r, c) \tag{11}$$

where

$$\sum_{r, c} P_n(r, c)\, P_m(r, c) = \begin{matrix} 0, \\ 1 \end{matrix} \begin{cases} n \neq m \\ n = m \end{cases}$$

and the least squares estimates $a'_1, \cdots, a'_N$ for the unknown coefficients $a_1, \cdots, a_n$ are given by

$$a'_n = \sum_{r,c} g(r, c)\, P_n(r, c). \tag{12}$$

Substituting the formula for $g(r, c)$ into the equation for a'_n and simplifying results in

$$a'_n = a_n + \sum_{r, c} P_n(r, c)\, \eta(r, c) \qquad (13)$$

clearly showing that a'_n has a deterministic part and a random part, the randomness due to the noise. We assume that the noise is independent normal having mean 0 and variance σ^2. Therefore, the estimated coefficient a'_n has mean a_n, variance σ^2 and is uncorrelated with every other coefficient:

$$E[a'_n] = a_n$$

$$E[a'_m a'_n] = a_m a_n, m \neq n$$

$$E[a'^2_n] = a^2_n + \sigma^2$$

$$V[a'_n] = \sigma^2.$$

The residual error e is defined as the difference between the observed values and fitted values. It too is a random variable:

$$e(r, c) = g(r, c) - \sum_{n=1}^{N} a'_n P_n(r, c)$$

$$= \sum_{n=1}^{N} (a_n - a'_n) P_n(r, c) + \eta(r, c). \qquad (14)$$

It is not difficult to see that at each (r, c), the residual error has mean zero and is uncorrelated with each estimated coefficient a'_n since

$$E[a'_n\, e(r, c)] = 0.$$

After some algebraic substitutions and manipulation, the total residual error S^2 can be written as

$$S^2 = \sum_{r, c} e^2(r, c) = \sum_{r, c} \eta^2(r, c) - \sum_{n=1}^{N} (a_n - a'_n)^2. \qquad (15)$$

Thus, if the noise is assumed normal and there are K pixels in a window

$$\sum_{r, c} \eta^2(r, c)/\sigma^2 \text{ has } X^2_k,$$

a chi-squared variate with K degree of freedom,

$$\sum_{n=1}^{N} (a_n - a'_n)^2/\sigma^2 \text{ has } X^2_N$$

which makes

$$\sum_{r, c} e^2(r, c)/\sigma^2 \text{ have } X^2_{k-N}.$$

B. Estimating the First Partials

If the discrete orthogonal basis functions are polynomials then each first partial derivative at (0, 0) in the row and column directions is given as some linear combination of the estimated coefficients. Furthermore, the linear combination for the row partial will be orthogonal to the linear combination in the column partial. It is not difficult to derive the expected value and variance of the first row partial μ'_r and the first column partial μ'_c. They are

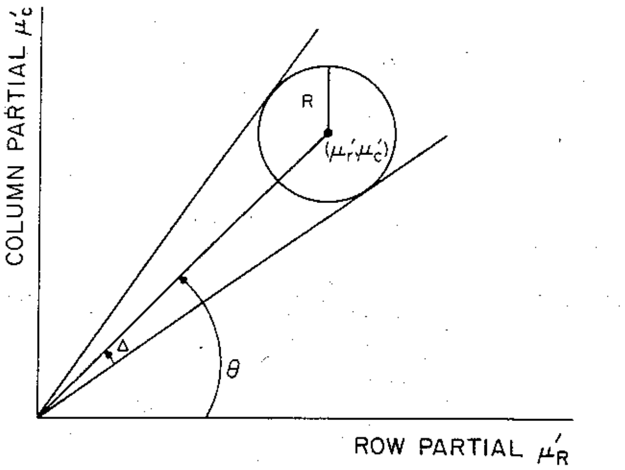


Fig. 4. Illustrates the geometry of the confidence interval estimation for the edge angle.

$$E[\mu'_r] = \mu_r$$

$$E[\mu'_c] = \mu_c$$

$$V[\mu'_r] = \sigma^2 k$$

$$V[\mu'_c] = \sigma^2 k$$

$$E[\mu'_r \mu'_c] = \mu_r \mu_c.$$

Hence, the estimates for the row and column partial derivatives are uncorrelated.

C. Hypothesis Testing for Zero Gradient

To see the effect of the randomness on the estimate of the gradient magnitude, consider testing the hypothesis that $\mu'_r = \mu'_c = 0$. This hypothesis must be rejected if there is to be a zero crossing of second directional derivative. Under this hypothesis,

$$\frac{\mu'^2_r + \mu'^2_c}{k\sigma^2}$$

has a X^2_2 distribution.

The total residual error normalized by the noise variance, S^2/σ^2, has a X^2_{K-N} distribution. Hence

$$\frac{(\mu'^2_r + \mu'^2_c)/2}{kS^2/(K - N)}$$

has an $F_{2, K-N}$ distribution and the hypothesis of $\mu'_r = \mu'_c = 0$ would be rejected for suitably large values.

D. Confidence Interval for Gradient Direction

To see the effect of the randomness on the estimate of the direction of the gradient, consider the relationships portrayed in Fig. 4. The axes are the row and column partials μ'_r and μ'_c. The direction angle Θ of the gradient is given by

$$\cos \Theta = \mu_r/(\mu^2_r + \mu^2_c)^{1/2}$$

$$\sin \Theta = \mu_c/(\mu^2_r + \mu^2_c)^{1/2}. \qquad (16)$$

The center of the circle is at the estimate (μ_r', μ_c'). Upon substituting the estimates μ_r' and μ_c' for μ_r and μ_c, we obtain the estimated direction angle Θ' by

$$\cos \Theta' = \mu_r'/(\mu_r'^2 + \mu_c'^2)^{1/2}$$
$$\sin \Theta' = \mu_c'/(\mu_r'^2 + \mu_c'^2)^{1/2}. \tag{17}$$

From a Bayesian point of view, the area of the circle represents the conditional probability that the unknown (μ_r, μ_c) lies within a distance R from the observed (μ_r', μ_c') given that the variance of μ_r' and μ_c' is known and equal to $k\sigma^2$. Assuming a normal distribution for the noise, this conditional probability is $q = 1 - e^{-R^2/2k\sigma^2}$. Hence, if probability q is given, the corresponding radius R is

$$R = k\sigma[-2 \log (1 - q)]^{1/2}. \tag{18}$$

To determine a confidence interval for Θ of the form $\Theta' - \Delta \leqslant \Theta \leqslant \Theta' + \Delta$, we have from Fig. 4 that

$$\sin^2 \Delta = \frac{k\sigma^2(-2 \log (1 - q))}{\mu_r'^2 + \mu_c'^2}. \tag{19}$$

Note that the 2Δ confidence interval length depends on the probability q of the circle confidence region for (μ_r, μ_c) and the unknown noise variance σ^2. Although σ^2 is not known, we do know S^2 which has a $\sigma^2\ X_{K-N}^2$ distribution. We can handle the problem of the unknown σ^2 by determining a joint confidence region for (μ_r, μ_c) and σ^2 [7]. Taking p to be the probability that a chi-squared random variable with $K - N$ degrees of freedom has an observed value greater than $X_{K-N,p}^2$ we have the confidence interval $[0, S^2/X_{K-N,p}^2]$ for σ^2 having at least probability p. Replacing σ^2 in (19) by $S^2/X_{N-K,p}^2$ we obtain

$$\sin^2 \Delta = \frac{kS^2(-2 \log (1 - q))}{X_{K-N,p}^2(\mu_r'^2 + \mu_c'^2)}. \tag{20}$$

A confidence interval for Θ having at least probability pq is then $(\Theta' - \Delta, \Theta' + \Delta)$.

E. Edge Hypothesis Testing

In this section we first take the edge direction α to be a fixed constant. We let μ_A and μ_B be the expected values of the random variables A and B appearing in (10):

$$f_\alpha''(\rho) = A\rho + B.$$

The null hypothesis is that an edge exists and it is satisfied if for some ρ, $0 \leqslant \rho \leqslant \rho_0$, where ρ_0 is just smaller than the length of a side of a pixel, we have $\mu_A \rho + \mu_B = 0$ and $\mu_A < 0$.

The observed random variables are A, B, and the residual fitting error S^2. The bivariate random variable

$$\begin{pmatrix} A \\ B \end{pmatrix}$$

is normal having mean

$$\begin{pmatrix} \mu_A \\ \mu_B \end{pmatrix}$$

and covariance

$$\sigma^2 \begin{bmatrix} k_A & 0 \\ 0 & k_B \end{bmatrix}$$

where k_A and k_B are known constants. For a window of K pixels and a cubic fit, S^2/σ^2 has a X_{K-10}^2.

From this it follows that

$$Z(\mu_A, \mu_B) = \frac{[(A - \mu_A)/k_A]^2 + [(B - \mu_B/k_B^2)]/2}{S^2/(K - 10)}$$

has an F_{2K-10} distribution.

We define $R = \{(x, y) |$ for some ρ, $0 \leqslant \rho \leqslant d$, $x\rho + y = 0\}$ Then the null hypothesis is rejected at the p significance level if

$$\min_{(\mu_A, \mu_B) \in R} Z(\mu_A, \mu_B)$$

is larger than $F_{2K-10\ 1-p}$.

An edge strength probability can be defined by q where q satisfies

$$\min_{(\mu_A, \mu_B) \in R} Z(\mu_A, \mu_B) = F_{2k-10\ q}.$$

Of course, the edge direction α is not fixed. But as derived at the end of Section IV-C, we do have a confidence interval for it. And for each value of α in the confidence interval, the random variable $A(\alpha)$ and $B(\alpha)$ can be computed and the null hypothesis tested. If for all α in the confidence interval the null hypothesis is rejected, then the existence of an edge is also rejected.

In practice, we can perform a nonexact hypothesis test selecting only the left end, middle, and right end values of α from its confidence interval. If for each of these three values of α the null hypothesis is rejected, then the existence of an edge is also rejected.

V. Experimental Results

To understand the performance of the second directional derivative zero crossing digital step edge operator we examine its behavior on a well structured simulated data set and on a real aerial image. For the simulated data set, we use a 100 X 100 pixel image of a checkerboard, the checks being 20 X 20 pixels. The dark checks have gray tone intensity 75 and the light checks have gray tone intensity 175. To this perfect checkerboard we add independent Gaussian noise having mean zero and standard deviation 50. Defining the signal to noise ratio as 10 times the logarithm of the range of signal divided by rms of the noise, the simulated image has a 3 dB signal to noise ratio. The perfect and noisy checkerboards are shown in Fig. 5.

Section V-A illustrates the performance of the classic 3 X 3 edge operators with and without preaveraging compared against the generalized Prewitt operator. Section V-B illustrates the performance of the 11 X 11 Marr-Hildreth zero crossing of the Laplacian operator, the 11 X 11 Prewitt operator, and the 11 X 11 zero crossing of second directional derivative operator. The zero crossing of second directional derivative surpasses the performance of the other two on the twofold basis of probability of correct assignment and error distance which is defined as the average distance to closest true edge pixel of pixels which are assigned nonedge but which are true edge pixels.

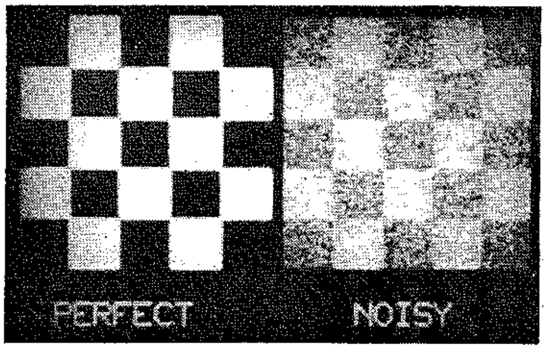


Fig. 5. Illustrates the controlled perfect and noisy checkerboard images.

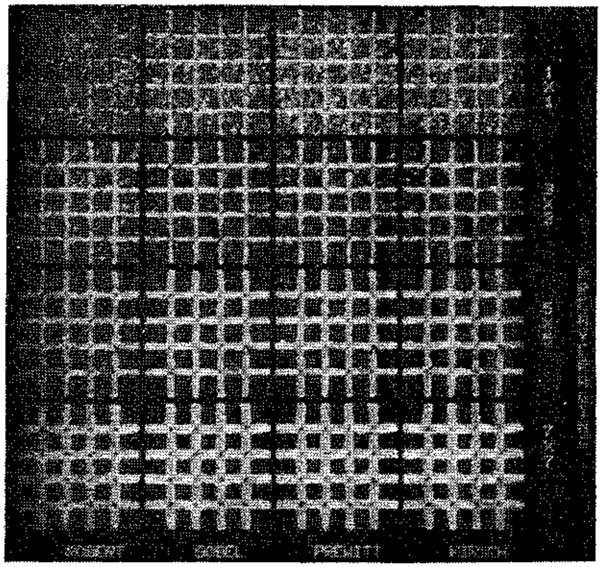


Fig. 6. Illustrates the 3 × 3 Roberts, Sobel, Prewitt, and Kirsch edge operators with a box filter preaveraging of 1 × 1, 3 × 3, 5 × 5, and 7 × 7.

A. The Classic Edge Operators

The classic 3 X 3 gradient operators all perform badly as shown in Fig. 6. Note that the usual definition of the Roberts operator has been modified in the natural way so that it uses a 3 X 3 mask.

Averaging before the application of the gradient operator is considered [19] to be one cure for such bad performance on noisy images. Fig. 6 also shows the same operators applied after a box filtering with a 3 X 3, 5 X 5, and 7 X 7 neighborhood sizes.

An alternative to the preaveraging is to define the gradient operator with a larger window. This is easily done with the Prewitt operator [17] which fits a quadratic surface in every window and uses the square root of the sum of the squares of the linear term coefficients to the estimate the gradient. (A linear fit would actually yield the same result. A cubic fit is the first higher order fit which would yield a different result.) This is illustrated in Fig. 7. A 3 X 3 preaverage followed by a 3 X 3 gradient operator yields a resulting neighborhood size of 5 X 5. Thus in Fig. 8 we also show the 3 X 3 preaverage followed by a 3 X 3 gradient under the 5 X 5 Prewitt and we show the 5 X 5 preaverage followed by the 3 X 3 gradient under the 7 X 7 Prewitt. The noise is higher in the preaverage edge-detector. For comparison purposes the 5 X 5 [16] compass operator is shown alongside the 5 X 5 Prewitt in Fig. 8. They give virtually the same result. The Prewitt operator has the advantage of requiring half the computation.

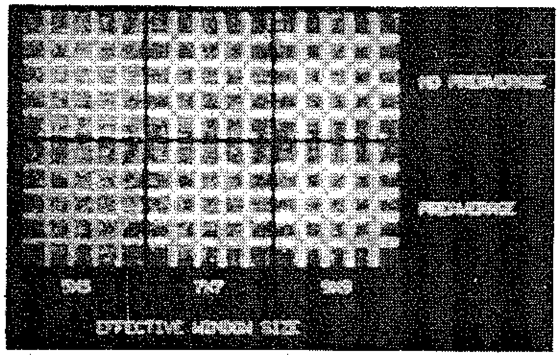


Fig. 7. Illustrates the Prewitt operator done by using a least squares quadratic fit in the neighborhood versus doing preaveraging and using a smaller fitting neighborhood size. The no preaveraging results show slightly higher contrast.

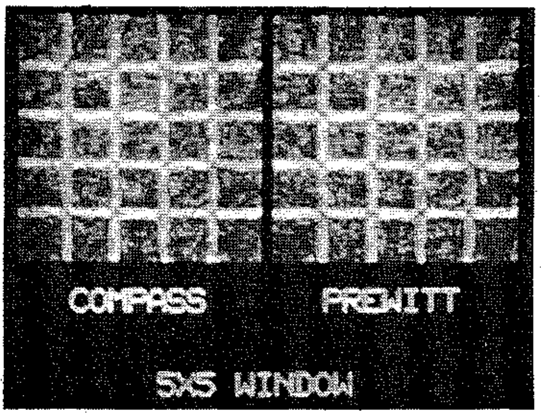


Fig. 8. Compares the Nevatia and Babu compass operator with the Prewitt operator in a 5 X 5 neighborhood.

It is obvious from these results that good gradient operators must have larger neighborhood sizes than 3 X 3. Unfortunately, the larger neighborhood sizes also yield thicker edges.

To detect edges, the gradient value must be thresholded. In each case, we chose a threshold value which makes the conditional probability of assigning an edge given that there is an edge equal to the conditional probability of there being a true edge given that an edge is assigned. True edges are defined to be the two pixel wide region in which each pixel neighbors some pixel having a value different from it on the perfect checkerboard. Fig. 9 shows the thresholded Prewitt operator (quadratic fit) for a variety of neighborhood sizes. Notice that because the gradient is zero at the saddle points (the corner where four checks meet) any operator depending on the gradient to detect an edge will have trouble there.

B. The Second Derivative Zero Crossing Edge Operators

Marr and Hildreth [13] suggest an edge operator based on the zero crossing of a generalized Laplacian. In effect, this is a nondirectional or isotropic second derivative zero crossing

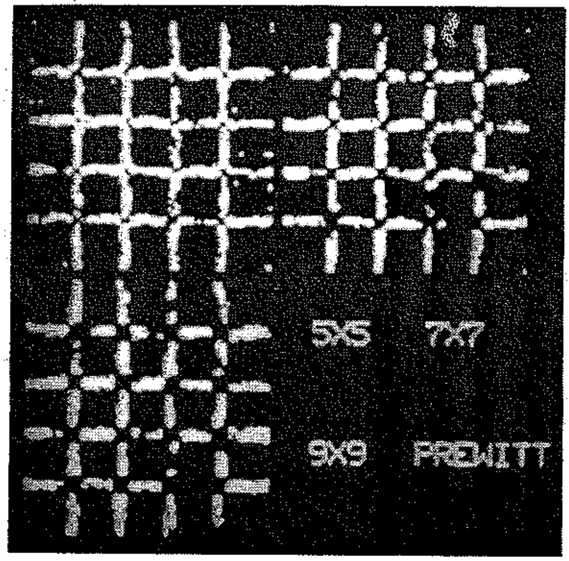


Fig. 9. Illustrates the edges obtained by thresholding the results of the Prewitt operator.

0	0	0	−1	−1	−2	−1	−1	0	0	0
0	0	−2	−4	−8	−9	−8	−4	−2	0	0
0	−2	−7	−15	−22	−23	−22	−15	−7	−2	0
−1	−4	−15	−24	−14	−1	−14	−24	−15	−4	−1
−1	−8	−22	−14	52	103	52	−14	−22	−8	−1
−2	−9	−23	−1	103	178	103	−1	−23	−9	−2
−1	−8	−22	−14	52	103	52	−14	−22	−8	−1
−1	−4	−15	−24	−14	−1	−14	−24	−15	−4	−1
0	−2	−7	−15	−22	−23	−22	−15	−7	−2	0
0	0	−2	−4	−8	−9	−8	−4	−2	0	0
0	0	0	−1	−1	−2	−1	−1	0	0	0

σ=1.4

−24	−21	−17	−13	−10	−10	−10	−13	−17	−21	−24
−21	−16	−10	−4	0	2	0	−4	−10	−16	−21
−17	−10	−1	6	11	13	11	6	−1	−10	−17
−13	−4	6	15	22	24	22	15	6	−4	−13
−10	0	11	22	29	31	29	22	11	0	−10
−10	2	13	24	31	34	31	24	13	2	−10
−10	0	11	22	29	31	29	22	11	0	−10
−13	−4	6	15	22	24	22	15	6	−4	−13
−17	−10	−1	6	11	13	11	6	−1	−10	−17
−21	−16	−10	−4	0	2	0	−4	−10	−16	−21
−24	−21	−17	−13	−10	−10	−10	−13	−17	−21	−24

σ=5.0

−34	−25	−18	−12	−8	−7	−8	−12	−18	−25	−34
−25	−15	−7	−1	3	4	3	−1	−7	−15	−25
−18	−7	2	8	12	14	12	8	2	−7	−18
−12	−1	8	15	20	21	20	15	8	−1	−12
−8	3	12	20	24	26	24	20	12	3	−8
−7	4	14	21	26	27	26	21	14	4	−7
−8	3	12	20	24	26	24	20	12	3	−8
−12	−1	8	15	20	21	20	15	8	−1	−12
−18	−7	2	8	12	14	12	8	2	−7	−18
−25	−15	−7	−1	3	4	3	−1	−7	−15	−25
−34	−25	−18	−12	−8	−7	−8	−12	−18	−25	−34

σ=10.00

Fig. 10. Shows that 11 × 11 kernels used for Mexican hat or generlized Laplacian operator defined by sampling

$$A\left(1-k\,\frac{r^2+c^2}{\sigma^2}\,e^{-1/2\,\frac{r^2+c^2}{\sigma^2}}\right)$$

at integer coordinates (r, c) $r, c = -5, \cdots, +5$. The constant k is chosen to make the sum of the values in the kernel to be zero, within the quantization error. The constant A just scales the values so that integer arithmetic can be used.

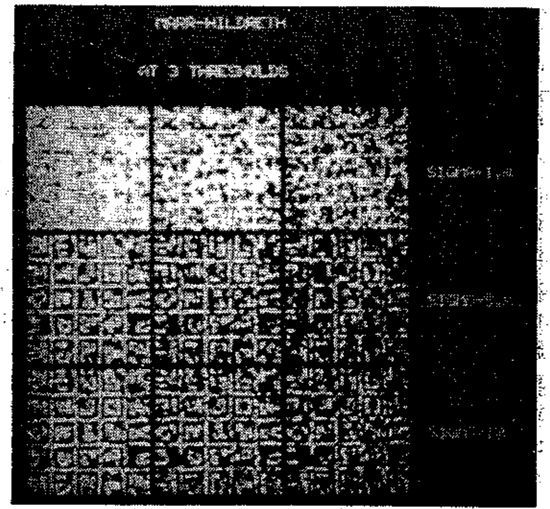


Fig. 11. Illustrates the edges obtained by the 11 × 11 Marr-Hildreth zero crossing of Laplacian operator set for three different zero crossing thresholds and three different standard deviations for the associated Mexican hat filter.

operator. The mask for this generalized Laplacian operator is given by sampling the kernel

$$A\left(1-k\,\frac{r^2+c^2}{\sigma^2}\,e^{-1/2\,\frac{r^2+c^2}{\sigma^2}}\right)$$

at the integer row column coordinates (r, c) designating the center of each pixel position in the neighborhood and then setting the value k so that the sum of the resulting weights is zero. Fig. 10 shows the resulting kernels for $\sigma = 1.4$, 5, and 10. The value $\sigma = 1.4$ is near the value usually used.

Edges are detected at all pixels whose generalized Laplacian value is of one sign and one of whose neighbor's generalized Laplacian value is of the opposite sign. A zero crossing threshold strength can be introduced here by insisting that the difference between the positive value and the negative value must exceed the threshold value before the pixel is declared to be an edge pixel. Fig. 11 illustrates the edge images produced by this technique for a variety of threshold values and a variety of values for σ for an 11 × 11 window. It is apparent that if all edge pixels are to be detected, there will be many pixels declared to be edge pixels which are really not edge pixels. And if there are to be no pixels which are to be declared edge pixels which are not edge pixels, then there will be many edge pixels which are not detected. Its performance appears to be poorer than the Prewitt operator.

The directional second derivative zero crossing edge operator introduced in this paper is shown in Fig. 12 for a variety of gradient threshold values. If the gradient exceeds the threshold value and a zero crossing occurs in a direction of $\pm 14.9°$ of the gradient direction within a circle of one pixel length centered in the pixel, then the pixel is declared to be an edge pixel. This technique performs worst at the saddle points, the corners where four checks meet, because the gradient in zero there. The second derivative zero crossing edge operator compares favorably with the more computationally expensive maximum likelihood boundary estimation technique of Cooper *et al.* [5] who show results using a higher signal to noise ratio (5.05 dB) synthetic image.

Table I shows the comparison among the Prewitt operator and the directional and the Marr-Hildreth nondirectional second derivative zero crossing edge operators. As before, the

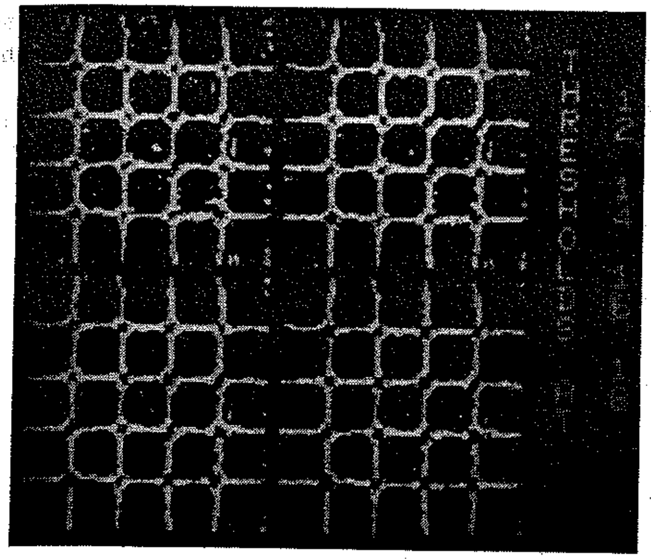


Fig. 12. Illustrates the directional derivative edge operator for a window size of 11 X 11 and deciding that the true gradient is nonzero when the estimated gradient is higher than the thresholds of 12, 14, 16, or 18.

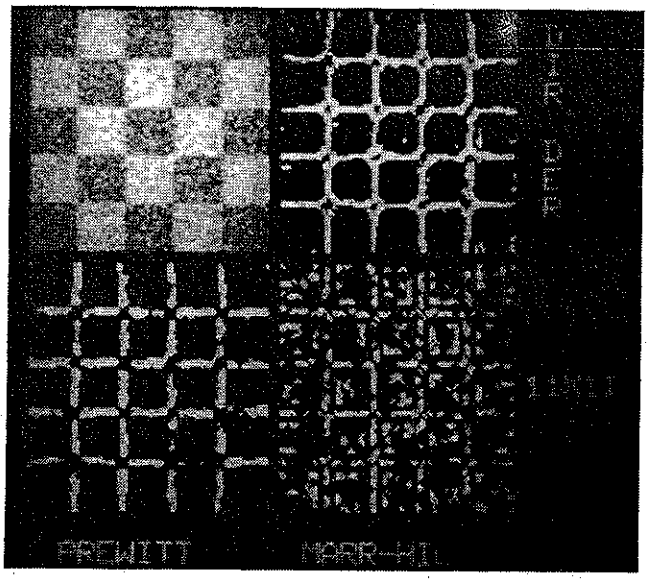


Fig. 13. Compares the directional derivative edge operator with the Marr–Hildreth edge operator and the Prewitt edge operator. The thresholds chosen were the best possible ones.

TABLE I

Compares the Performance of Three Edge Operators Using an 11 X 11 Window on the Noisy Checkerboard Image. Thresholds are Chosen to Equalize, as Best as Possible, $P(AE/TE)$, the Conditional Probability of Assigned Edge Given True Edge and the Conditional Probability, $P(TE/AE)$ of True Edge Given Assigned Edge. The Error Distance is the Average Distance to Closest True Edge Pixels of Pixels which are Assigned Nonedge but which are True Edge. A Visual Evaluation also Leaves the Impression that the Directional Derivative Operator Produces Better Edge Continuity and has Less Noise than the Other Two.

	Prewitt	Marr–Hildreth	Directional Derivative
Parameters	Gradient Threshold = 18.5	Zero Crossing Strength = 4.0 $\sigma = 5.0$	Gradient Threshold = 14.0 $\rho = 0.5$
$P(AE\|TE)$	0.6738	0.3977	0.7207
$P(TE\|AE)$	0.6872	0.4159	0.7197
Error Distance	1.79	1.76	1.16

threshold used is the one equalizing the conditional probability of assigned edge given true edge and the conditional probability of true edge given assigned edge. It appears that the performance of the directional derivative operator is better than the Prewitt operator and the Marr–Hildreth operator, both on the basis of the correct assignment probability and the error distance which is the average distance to closest true edge pixels of pixels which are assigned nonedge labels but which are true edge pixels.

Fig. 13 shows the corresponding edge images of the 11 X 11 Prewitt operator using a cubic fit rather than a quadratic fit, the 11 X 11 Marr–Hildreth operator, and the 11 X 11 directional derivative zero crossing operator. The thresholds used are the ones to equalize the conditional probabilities as given in Table I.

For the case of constant variance additive noise, thresholding on the basis of the hypothesis test of Section IV-C yields essentially the same results as simply thresholding the gradient value.

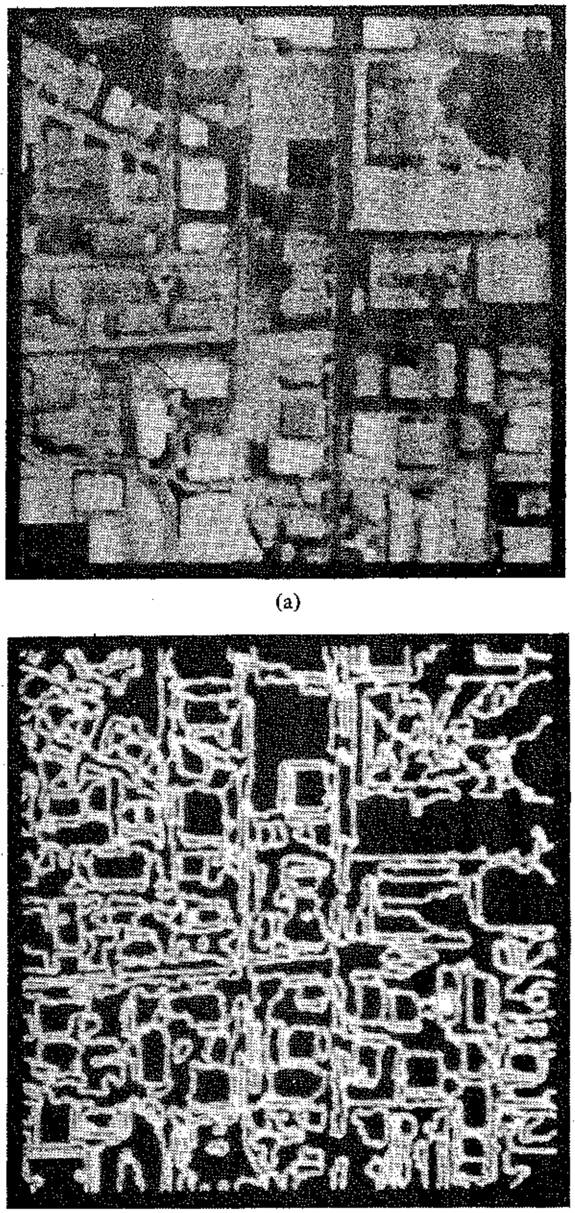
Fig. 14 illustrates the second directional derivative zero crossing operator on an aerial image which has been median filtered and then enhanced by replacing each pixel with the closer of its 3 X 3 neighborhood minimum or maximun. The technique is so good that it is possible to determine region boundaries essentially by doing a connected components on nonedge pixels. Fig. 14(b) shows the cleaned edge image which is obtained by doing a connected components on the non edge pixels, then removing all pixels whose region has fewer than 20 pixels. The resulting boundaries are given as pixels which have a neighbor with a different label than its own.

Initial raw edges which leave gaps in a region boundary will in effect make the regions merge in the connected components step. Thus the small number of missing boundaries is surprising. To be sure, we are not advocating connected components as an image segmentation technique. The fact that it works as well as it does is an indication of the strength of the edge detector.

VI. Conclusions

We have argued that numeric digital image operations should be explained in terms of their actions on the underlying gray tone intensity surface of which the digital image is an observed noisey sample. We called this model the local facet model for digital image processing and showed how the facet model can be used to estimate in each neighborhood the underlying gray tone intensity surface.

We described a digital step edge operator which detects edges at all pixels whose estimated second directional derivative taken in the direction of the gradient has a zero crossing within the pixel's area. We discussed the statistical analysis of this technique, illustrating how to determine confidence intervals for the direction of the gradient and how this interval de-



(a)

(b)

Fig. 14. (a) Illustrates an aerial photograph. (b) Illustrates the directional derivative edges obtained from the aerial photograph by first 3 X 3 median filtering, then replacing each pixel by the closer of its 3 X 3 neighborhood minimum or maximum, then taking the directional derivative edges using a 7 X 7 window, then doing a connected components on the nonedge pixels, and removing all regions having fewer than 20 pixels, and then displaying any pixel neighboring an pixel different than it as an edge pixel.

termines a confidence interval for the placement of the zero crossing.

We have compared the performance of the directional derivative zero crossing edge operator with that of some classic edge operators, the generalized Prewitt gradient operator, and the Marr-Hildreth zero crossing edge operator. We found that in both the simulated and real image data sets the directional derivative zero crossing edge operator had superior performance.

We have illustrated that for good performance it is important to use larger neighborhood sizes than 3 X 3 and have shown that better results are achieved by defining the edge operator naturally in the large neighborhood rather than preaveraging and then using a smaller neighborhood edge operator on the averaged image.

There is much work yet to be done. A comparison should be done between the image segmentation discussed in Chen and Pavlidis [4] and the segmentation achievable by the edge technique presented in this paper. We need to explore the relationship of basis function kind (polynomial, trigonometric polynomial, etc.), order of fit, and neighborhood size to the goodness of fit. Evaluation must be made of the confidence intervals produced by the technique. The technique needs to be generalized so that it works on saddle points created by two edges crossing. A suitable edge linking method needs to be developed which uses these confidence intervals. Ways of incorporating semantic information and ways of using variable resolution need to be developed. An analogous technique for roof edges needs to be developed. We hope to explore these issues in future papers.

Appendix

The Discrete Orthogonal Polynomials

The discretely orthogonal polynomial basis set of size N which we use has polynomials from degree zero through degree $N-1$. These polynomials are unique and sometimes called the discrete Chebyshev polynomials [1]. In this Appendix we show how to construct them for one or two variables.

A. Discrete Orthogonal Polynomial Construction Technique

Let the discrete integer index set R be symmetric in the sense that $r \in R$ implies $-r \in R$. Let $P_n(r)$ be the nth order polynomial. We define the construction technique for discrete orthogonal polynomials iteratively.

Define $P_0(r) = 1$. Suppose $P_0(r), \cdots, P_{n-1}(r)$ have been defined. In general, $P_n(r) = r^n + a_{n-1}r^{n-1} + \cdots + a_1 r + a_0$. $P_n(r)$ must be orthogonal to each polynomial $P_0(r), \cdots, P_{n-1}(r)$. Hence, we must have the n equations

$$\sum_{r \in R} P_k(r)(r^n + a_{n-1}r^{n-1} + \cdots + a_1 r + a_0) = 0,$$
$$k = 0, \cdots, n-1. \tag{A1}$$

These equations are linear equations in the unknown $a_0, \cdots, a_{n-1}$ and are easily solved by standard techniques.

The first five polynomial functions formulas are

$$P_0(r) = 1$$
$$P_1(r) = r$$
$$P_2(r) = r^2 - \mu_2/\mu_0$$
$$P_3(r) = r^3 - (\mu_4/\mu_2)r$$
$$P_4(r) = \frac{r^4 + (\mu_2\mu_4 - \mu_6)r^2 + (\mu_2\mu_6 - \mu_4^2)}{\mu_0\mu_4 - \mu_2}$$

where

$$\mu_k = \sum_{s \in R} s^k.$$

B. Two-Dimensional Discrete Orthogonal Polynomials

Two-dimensional discrete orthogonal polynomials can be created from two sets of one-dimensional discrete orthogonal polynomials by taking tensor products. Let R and C be index sets satisfying the symmetry condition $r \in R$ implies $-r \in R$ and $c \in C$ implies $-c \in C$. Let $\{P_0(r), \cdots, P_N(r)\}$ be a set of discrete polynomials on R. Let $\{Q_0(c), \cdots, Q_M(c)\}$ be a set of discrete polynomials on C. Then the set $\{P_0(r) Q_0(c), \cdots, P_n(r) Q_m(c), \cdots, P_N(r) Q_M(c)\}$ is a set of discrete polynomials on $R \times C$.

The proof of this fact is easy. Consider whether $P_i(r) Q_j(c)$ is orthogonal to $P_n(r) Q_m(c)$. When $n \neq i$ or $m \neq j$. Then

$$\sum_{r \in R} \sum_{c \in C} P_i(r) Q_j(c) P_n(r) Q_m(c)$$

$$= \sum_{r \in R} P_i(r) P_n(r) \sum_{c \in C} Q_j(c) Q_m(c).$$

Since by assumption $n \neq i$ or $m \neq j$ the first sum or second sum must be zero, thereby proving the orthogonality.

References

[1] P. Beckmann, *Orthogonal Polynomials for Engineers and Physicists.* Boulder, CO: Golem, 1973.

[2] P. Beaudet, "Rotationally invariant image operators," in *Proc. 4th Int. Joint Conf. Pattern Recognition*, Tokyo, Japan, Nov. 1978, pp. 579–583.

[3] M. J. Brooks, "Rationalizing edge detectors," *Comput. Graphics Image Processing*, vol. 8, pp. 277–285, 1978.

[4] P. C. Chen and T. Pavlidis, "Image segmentation as an estimation problem," in *Image Modeling*, A. Rosenfeld, Ed. New York: Academic, 1981.

[5] D. B. Cooper, H. Elliott, F. Cohen, L. Reiss, and P. Symosek, "Stochastic boundary estimation and object recognition," in *Image Modeling*, A. Rosenfeld, Ed. New York: Academic, 1981.

[6] R. Ehrich and F. Schroeder, "Contextual boundary formation by one-dimensional edge detection and scan line matching," *Comput. Graphics Image Processing*, vol. 16, pp. 116–149, 1981.

[7] R. Foutz, personal communication, 1981.

[8] R. Haralick, "Edge and region analysis for digital image data," *Comput. Graphics Image Processing*, vol. 12, pp. 60–73, 1980.

[9] —, "Zero-crossing of second directional derivative edge operator," presented at the SPIE Symp. on Robot Vision, Washington, DC, May 1982.

[10] R. Haralick and L. Watson, "A facet model for image data," *Comput. Graphics Image Processing*, vol. 15, pp. 113–129, 1981.

[11] M. Hueckel, "A local visual operator which recognizes edges and lines," *J. Ass. Comput. Mach.*, vol. 20, pp. 634–647, 1973.

[12] —, "An operator which locates edges in digitized pictures, *J. Ass. Comput. Mach.*, vol. 18, pp. 113–125, 1971.

[13] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Royal Soc. London*, ser. B, vol. 207, pp. 187–217, 1980.

[14] D. Morgenthaler, "A new hybrid edge detector," *Comput. Graphics Image Processing*, vol. 16, pp. 166–176, 1981.

[15] D. Morgenthaler and A. Rosenfeld, "Multidimensional edge section by hypersurface fitting," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-3, pp. 482–486, July 1981.

[16] R. Nevatia and R. Babu, "Linear feature extraction and description," *Comput. Graphics Image Processing*, vol. 13, pp. 257–269, 1980.

[17] J. Prewitt, "Object enhancement and extraction," in *Picture Processing and Psychopictorics*, B. Lipkin and A. Rosenfeld, Eds. New York: Academic, 1970, pp. 75–149.

[18] L. G. Roberts "Machine perception of three-dimensional solids," in *Optical and Electrooptical Imformation Processing*, J. T. Trippett *et al.*, Eds. Cambridge, MA: MIT Press, 1965, pp. 159–197.

[19] A. Rosenfeld and A. Kak, *Digital Picture Processing.* New York: Academic, 1976.

[20] S. Zucker and R. Hummel, "An optimal three-dimensional edge operator," in *Proc. Pattern Recognition and Image Processing Conf.*, Chicago, IL, Aug. 1979, pp. 162–168.

Robert M. Haralick (M'69–SM'76–F'84) was born in Brooklyn, NY, on September 30, 1943. He received the B.A., B.S., M.S., and Ph.D. degrees from the University of Kansas, Lawrence, in 1964, 1966, 1967, and 1969, respectively.

He has worked at Autonetics and IBM. In 1965 he worked for the Center for Research, University of Kansas, as a Research Engineer and in 1969 he joined the faculty of the Department of Electrical Engineering, where he served as a Professor from 1975 to 1978. In 1979 he joined the faculty of the Departments of Electrical Engineering and Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, where he is now a Professor and Director of the Spatial Data Analysis Laboratory. He has done research in pattern recognition, multiimage processing, remote sensing, texture analysis, data compression, clustering, artificial intelligence, and general systems theory. He is responsible for the development of GIPSY (general image processing system), a multiimage processing package which runs on a minicomputer system.

Dr. Haralick is a member of the Association for Computing Machinery, Sigma Xi, the Pattern Recognition Society, and the Society for General Systems Research.