

# A Computational Approach to Edge Detection

JOHN CANNY, MEMBER, IEEE

**Abstract**—This paper describes a computational approach to edge detection. The success of the approach depends on the definition of a *comprehensive* set of goals for the computation of edge points. These goals must be precise enough to delimit the desired behavior of the detector while making minimal assumptions about the form of the solution. We define detection and localization criteria for a class of edges, and present mathematical forms for these criteria as functionals on the operator impulse response. A third criterion is then added to ensure that the detector has only one response to a single edge. We use the criteria in numerical optimization to derive detectors for several common image features, including step edges. On specializing the analysis to step edges, we find that there is a natural uncertainty principle between detection and localization performance, which are the two main goals. With this principle we derive a single operator shape which is optimal at any scale. The optimal detector has a simple approximate implementation in which edges are marked at maxima in gradient magnitude of a Gaussian-smoothed image. We extend this simple detector using operators of several widths to cope with different signal-to-noise ratios in the image. We present a general method, called feature synthesis, for the fine-to-coarse integration of information from operators at different scales. Finally we show that step edge detector performance improves considerably as the operator point spread function is extended along the edge. This detection scheme uses several elongated operators at each point, and the directional operator outputs are integrated with the gradient maximum detector.

**Index Terms**—Edge detection, feature extraction, image processing, machine vision, multiscale image analysis.

## I. INTRODUCTION

EDGE detectors of some kind, particularly step edge detectors, have been an essential part of many computer vision systems. The edge detection process serves to simplify the analysis of images by drastically reducing the amount of data to be processed, while at the same time preserving useful structural information about object boundaries. There is certainly a great deal of diversity in the applications of edge detection, but it is felt that many applications share a common set of requirements. These requirements yield an abstract edge detection problem, the solution of which can be applied in any of the original problem domains.

We should mention some specific applications here. The Binford-Horn line finder [14] used the output of an edge

detector as input to a program which could isolate simple geometric solids. More recently the model-based vision system ACRONYM [3] used an edge detector as the front end to a sophisticated recognition program. Shape from motion [29], [13] can be used to infer the structure of three-dimensional objects from the motion of edge contours or edge points in the image plane. Several modern theories of stereopsis assume that images are preprocessed by an edge detector before matching is done [19], [20]. Beattie [1] describes an edge-based labeling scheme for low-level image understanding. Finally, some novel methods have been suggested for the extraction of three-dimensional information from image contours, namely shape from contour [27] and shape from texture [31].

In all of these examples there are common criteria relevant to edge detector performance. The first and most obvious is low error rate. It is important that edges that occur in the image should not be missed and that there be no spurious responses. In all the above cases, system performance will be hampered by edge detector errors. The second criterion is that the edge points be well localized. That is, the distance between the points marked by the detector and the "center" of the true edge should be minimized. This is particularly true of stereo and shape from motion, where small disparities are measured between left and right images or between images produced at slightly different times.

In this paper we will develop a mathematical form for these two criteria which can be used to design detectors for arbitrary edges. We will also discover that the first two criteria are not "tight" enough, and that it is necessary to add a third criterion to circumvent the possibility of multiple responses to a single edge. Using numerical optimization, we derive optimal operators for ridge and roof edges. We will then specialize the criteria for step edges and give a parametric closed form for the solution. In the process we will discover that there is an uncertainty principle relating detection and localization of noisy step edges, and that there is a direct tradeoff between the two. One consequence of this relationship is that there is a single unique "shape" of impulse response for an optimal step edge detector, and that the tradeoff between detection and localization can be varied by changing the spatial width of the detector. Several examples of the detector performance on real images will be given.

## II. ONE-DIMENSIONAL FORMULATION

To facilitate the analysis we first consider one-dimensional edge profiles. That is, we will assume that two-

Manuscript received December 10, 1984; revised November 27, 1985. Recommended for acceptance by S. L. Tanimoto. This work was supported in part by the System Development Foundation, in part by the Office of Naval Research under Contract N00014-81-K-0494, and in part by the Advanced Research Projects Agency under Office of Naval Research Contracts N00014-80-C-0505 and N00014-82-K-0334.

The author is with the Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139.

IEEE Log Number 8610412.

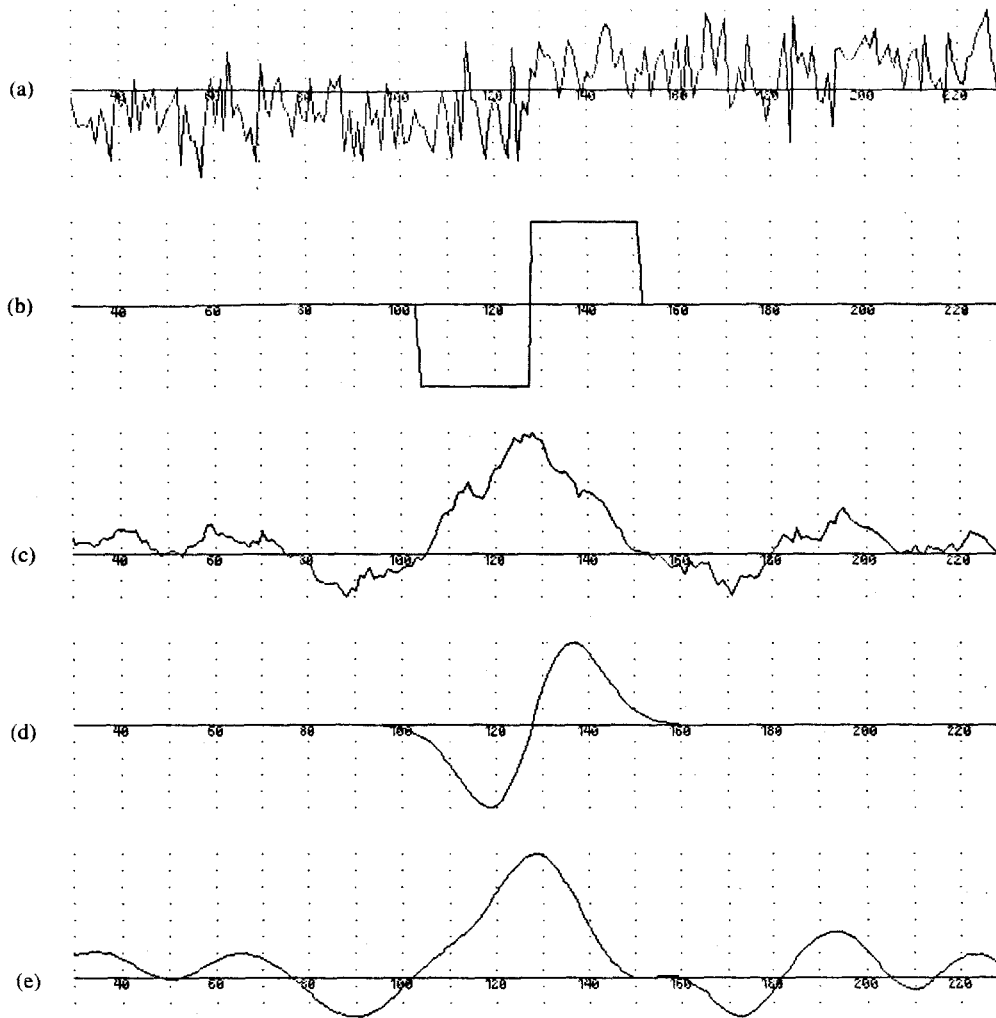


Fig. 1. (a) A noisy step edge. (b) Difference of boxes operator. (c) Difference of boxes operator applied to the edge. (d) First derivative of Gaussian operator. (e) First derivative of Gaussian applied to the edge.

dimensional edges locally have a constant cross-section in some direction. This would be true for example, of smooth edge contours or of ridges, but not true of corners. We will assume that the image consists of the edge and additive white Gaussian noise.

The detection problem is formulated as follows: We begin with an edge of known cross-section bathed in white Gaussian noise as in Fig. 1(a), which shows a step edge. We convolve this with a filter whose impulse response could be illustrated by either Fig. 1(b) or (d). The outputs of the convolutions are shown, respectively, in Fig. 1(c) and (e). We will mark the center of an edge at a local maximum in the output of the convolution. The design problem then becomes one of finding the filter which gives the best performance with respect to the criteria given below. For example, the filter in Fig. 1(d) performs much better than Fig. 1(b) on this example, because the response of the latter exhibits several local maxima in the region of the edge.

In summary, the three performance criteria are as follows:

1) Good detection. There should be a low probability

of failing to mark real edge points, and low probability of falsely marking nonedge points. Since both these probabilities are monotonically decreasing functions of the output signal-to-noise ratio, this criterion corresponds to maximizing signal-to-noise ratio.

2) Good localization. The points marked as edge points by the operator should be as close as possible to the center of the true edge.

3) Only one response to a single edge. This is implicitly captured in the first criterion since when there are two responses to the same edge, one of them must be considered false. However, the mathematical form of the first criterion did not capture the multiple response requirement and it had to be made explicit.

#### A. Detection and Localization Criteria

A crucial step in our method is to capture the intuitive criteria given above in a mathematical form which is readily solvable. We deal first with signal-to-noise ratio and localization. Let the impulse response of the filter be  $f(x)$ , and denote the edge itself by  $G(x)$ . We will assume that the edge is centered at  $x = 0$ . Then the response of the

filter to this edge at its center  $H_G$  is given by a convolution integral:

$$H_G = \int_{-W}^{+W} G(-x) f(x) dx \quad (1)$$

assuming the filter has a finite impulse response bounded by  $[-W, W]$ . The root-mean-squared response to the noise  $n(x)$  only, will be

$$H_n = n_0 \left[ \int_{-W}^{+W} f^2(x) dx \right]^{1/2} \quad (2)$$

where  $n_0^2$  is the mean-squared noise amplitude per unit length. We define our first criterion, the output signal-to-noise ratio, as the quotient of these two responses.

$$\text{SNR} = \frac{\left| \int_{-W}^{+W} G(-x) f(x) dx \right|}{n_0 \sqrt{\int_{-W}^{+W} f^2(x) dx}} \quad (3)$$

For the localization criterion, we want some measure which increases as localization improves, and we will use the reciprocal of the root-mean-squared distance of the marked edge from the center of the true edge. Since we have decided to mark edges at local maxima in the response of the operator  $f(x)$ , the first derivative of the response will be zero at these points. Note also that since edges are centered at  $x = 0$ , in the absence of noise there should be a local maximum in the response at  $x = 0$ .

Let  $H_n(x)$  be the response of the filter to noise only, and  $H_G(x)$  be its response to the edge, and suppose there is a local maximum in the total response at the point  $x = x_0$ . Then we have

$$H'_n(x_0) + H'_G(x_0) = 0. \quad (4)$$

The Taylor expansion of  $H'_G(x_0)$  about the origin gives

$$H'_G(x_0) = H'_G(0) + H''_G(0)x_0 + O(x_0^2). \quad (5)$$

By assumption  $H'_G(0) = 0$ , i.e., the response of the filter in the absence of noise has a local maximum at the origin, so the first term in the expansion can be ignored. The displacement  $x_0$  of the actual maximum is assumed to be small so we will ignore quadratic and higher terms. In fact by a simple argument we can show that if the edge  $G(x)$  is either symmetric or antisymmetric, all even terms in  $x_0$  vanish. Suppose  $G(x)$  is antisymmetric, and express  $f(x)$  as a sum of a symmetric component and an antisymmetric component. The convolution of the symmetric component with  $G(x)$  contributes nothing to the numerator of the SNR, but it does contribute to the noise component in the denominator. Therefore, if  $f(x)$  has any symmetric component, its SNR will be worse than a purely antisymmetric filter. A dual argument holds for symmetric edges, so that if the edge  $G(x)$  is symmetric or antisymmetric, the filter  $f(x)$  will follow suit. The net result of this is that the response  $H_G(x)$  is always symmet-

ric, and that its derivatives of odd orders [which appear in the coefficients of even order in (5)] are zero at the origin. Equations (4) and (5) give

$$H''_G(0)x_0 \approx -H'_n(x_0). \quad (6)$$

Now  $H'_n(x_0)$  is a Gaussian random quantity whose variance is the mean-squared value of  $H'_n(x_0)$ , and is given by

$$E[H'_n(x_0)^2] = n_0^2 \int_{-W}^{+W} f'^2(x) dx \quad (7)$$

where  $E[y]$  is the expectation value of  $y$ . Combining this result with (6) and substituting for  $H''_G(0)$  gives

$$E[x_0^2] \approx \frac{n_0^2 \int_{-W}^{+W} f'^2(x) dx}{\left[ \int_{-W}^{+W} G'(-x) f'(x) dx \right]^2} = \delta x_0^2 \quad (8)$$

where  $\delta x_0$  is an approximation to the standard deviation of  $x_0$ . The localization is defined as the reciprocal of  $\delta x_0$ .

$$\text{Localization} = \frac{\left| \int_{-W}^{+W} G'(-x) f'(x) dx \right|}{n_0 \sqrt{\int_{-W}^{+W} f'^2(x) dx}}. \quad (9)$$

Equations (3) and (9) are mathematical forms for the first two criteria, and the design problem reduces to the maximization of both of these simultaneously. In order to do this, we maximize the product of (3) and (9). We could conceivably have combined (3) and (9) using any function that is monotonic in two arguments, but the use of the product simplifies the analysis for step edges, as should become clear in Section III. For the present we will make use of the product of the criteria for arbitrary edges, i.e., we seek to maximize

$$\frac{\left| \int_{-W}^{+W} G(-x) f(x) dx \right|}{n_0 \sqrt{\int_{-W}^{+W} f^2(x) dx}} \frac{\left| \int_{-W}^{+W} G'(-x) f'(x) dx \right|}{n_0 \sqrt{\int_{-W}^{+W} f'^2(x) dx}}. \quad (10)$$

There may be some additional constraints on the solution, such as the multiple response constraint (12) described next.

### B. Eliminating Multiple Responses

In our specification of the edge detection problem, we decided that edges would be marked at local maxima in the response of a linear filter applied to the image. The detection criterion given in the last section measures the effectiveness of the filter in discriminating between signal and noise at the center of an edge. It does not take into account the behavior of the filter *nearby* the edge center. The first two criteria can be trivially maximized as fol-

lows. From the Schwarz inequality for integrals we can show that SNR (3) is bounded above by

$$n_0^{-1} \sqrt{\int_{-W}^{+W} G^2(x) dx}$$

and localization (9) by

$$n_0^{-1} \sqrt{\int_{-W}^{+W} G'^2(x) dx}.$$

Both bounds are attained, and the product of SNR and localization is maximized when  $f(x) = G(-x)$  in  $[-W, W]$ .

Thus, according to the first two criteria, the optimal detector for step edges is a truncated step, or difference of boxes operator. The difference of boxes was used by Rosenfeld and Thurston [25], and in conjunction with lateral inhibition by Herskovits and Binford [11]. However it has a very high bandwidth and tends to exhibit many maxima in its response to noisy step edges, which is a serious problem when the imaging system adds noise or when the image itself contains textured regions. These extra edges should be considered erroneous according to the first of our criteria. However, the analytic form of this criterion was derived from the response at a single point (the center of the edge) and did not consider the interaction of the responses at several nearby points. If we examine the output of a difference of boxes edge detector we find that the response to a noisy step is a roughly triangular peak with numerous sharp maxima in the vicinity of the edge (see Fig. 1).

These maxima are so close together that it is not possible to select one as the response to the step while identifying the others as noise. We need to add to our criteria the requirement that the function  $f$  will not have "too many" responses to a single step edge in the vicinity of the step. We need to limit the number of peaks in the response so that there will be a low probability of declaring more than one edge. Ideally, we would like to make the distance between peaks in the noise response approximate the width of the response of the operator to a single step. This width will be some fraction of the operator width  $W$ .

In order to express this as a functional constraint on  $f$ , we need to obtain an expression for the distance between adjacent noise peaks. We first note that the mean distance between adjacent maxima in the output is twice the distance between adjacent zero-crossings in the derivative of the operator output. Then we make use of a result due to Rice [24] that the average distance between zero-crossings of the response of a function  $g$  to Gaussian noise is

$$x_{ave} = \pi \left( \frac{-R(0)}{R''(0)} \right)^{1/2} \quad (11)$$

where  $R(\tau)$  is the autocorrelation function of  $g$ . In our case we are looking for the mean zero-crossing spacing for the function  $f'$ . Now since

$$R(0) = \int_{-\infty}^{+\infty} g^2(x) dx \quad \text{and} \quad R''(0) = - \int_{-\infty}^{+\infty} g'^2(x) dx$$

the mean distance between zero-crossings of  $f'$  will be

$$x_{zc}(f) = \pi \left( \frac{\int_{-\infty}^{+\infty} f'^2(x) dx}{\int_{-\infty}^{+\infty} f''^2(x) dx} \right)^{1/2} \quad (12)$$

The distance between adjacent maxima in the noise response of  $f$ , denoted  $x_{max}$ , will be twice  $x_{zc}$ . We set this distance to be some fraction  $k$  of the operator width.

$$x_{max}(f) = 2x_{zc}(f) = kW. \quad (13)$$

This is a natural form for the constraint because the response of the filter will be concentrated in a region of width  $2W$ , and the expected number of noise maxima in this region is  $N_n$  where

$$N_n = \frac{2W}{x_{max}} = \frac{2}{k}. \quad (14)$$

Fixing  $k$  fixes the number of noise maxima that could lead to a false response.

We remark here that the intermaximum spacing (12) scales with the operator width. That is, we first define an operator  $f_w$  which is the result of stretching  $f$  by a factor of  $w$ ,  $f_w(x) = f(x/w)$ . Then after substituting into (12) we find that the intermaximum spacing for  $f_w$  is  $x_{zc}(f_w) = wx_{zc}(f)$ . Therefore, if a function  $f$  satisfies the multiple response constraint (13) for fixed  $k$ , then the function  $f_w$  will also satisfy it, assuming  $W$  scales with  $w$ . For any fixed  $k$ , the multiple response criterion is invariant with respect to spatial scaling of  $f$ .

### III. FINDING OPTIMAL DETECTORS BY NUMERICAL OPTIMIZATION

In general it will be difficult (or impossible) to find a closed form for the function  $f$  which maximizes (10) subject to the multiple response constraint. Even when  $G$  has a particularly simple form (e.g., it is a step edge), the form of  $f$  may be complicated. However, if we are given a candidate function  $f$ , evaluation of (10) and (12) is straightforward. In particular, if the function  $f$  is represented by a discrete time sequence, evaluation of (10) requires only the computation of four inner products between sequences. This suggests that numerical optimization can be done directly on the sampled operator impulse response.

The output will not be an analytic form for the operator, but an implementation of a detector for the edge of interest will require discrete point-spread functions anyway. It is also possible to include additional constraints by using a *penalty method* [15]. In this scheme, the constrained optimization is reduced to one, or possibly several, unconstrained optimizations. For each constraint we define a penalty function which has a nonzero value when one

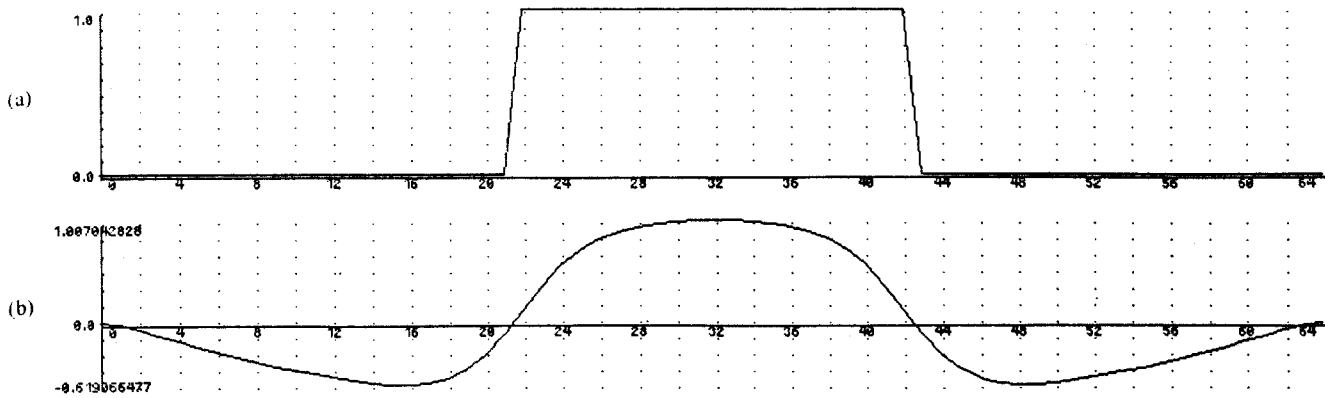


Fig. 2. A ridge profile and the optimal operator for it.

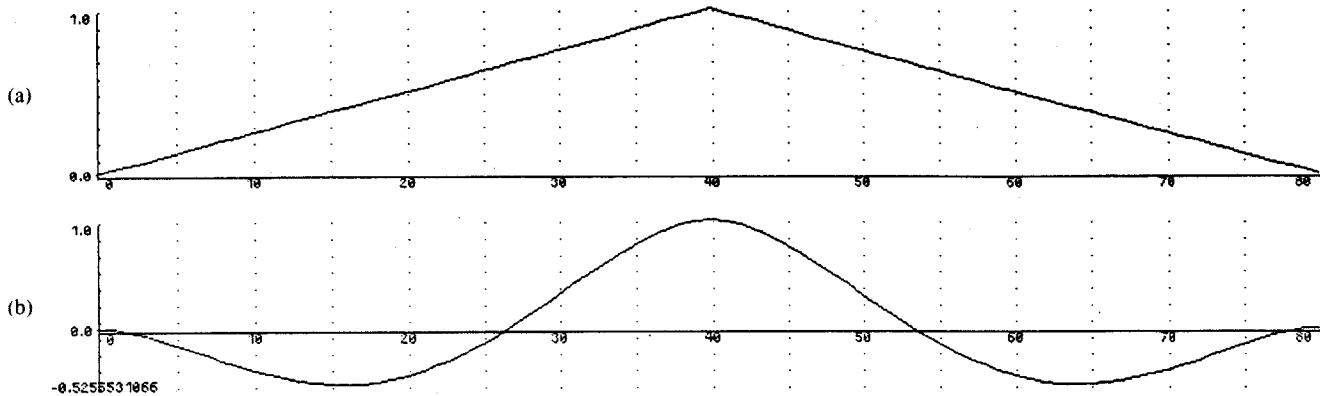


Fig. 3. A roof profile and an optimal operator for roofs.

of the constraints is violated. We then find the  $f$  which maximizes

$$\text{SNR}(f) * \text{Localization}(f) - \sum \mu_i P_i(f) \quad (15)$$

where  $P_i$  is a function which has a positive value only when a constraint is violated. The larger the value of  $\mu_i$  the more nearly the constraints will be satisfied, but at the same time the greater the likelihood that the problem will be ill-conditioned. A sequence of values of  $\mu_i$  may need to be used, with the final form of  $f$  from each optimization used as the starting form for the next. The  $\mu_i$  are increased at each iteration so that the value of  $P_i(f)$  will be reduced, until the constraints are "almost" satisfied.

An example of the method applied to the problem of detecting "ridge" profiles is shown in Fig. 2. For a ridge, the function  $G$  is defined to be a flat plateau of width  $w$ , with step transitions to zero at the ends. The auxiliary constraints are

- The multiple response constraint. This constraint is taken directly from (12), and does not depend on the form of the edge.
- The operator should have zero dc component. That is it should have zero output to constant input.

Since the width of the operator is dependent on the width of the ridge, there is a suggestion that several widths of operators should be used. This has not been done in the present implementation however. A wide ridge can be considered to be two closely spaced edges, and the im-

plementation already includes detectors for these. The only reason for using a ridge detector is that there are ridges in images that are too small to be dealt with effectively by the narrowest edge operator. These occur frequently because there are many edges (e.g., scratches and cracks or printed matter) which lie at or beyond the resolution of the camera and result in contours only one or two pixels wide.

A similar procedure was used to find an optimal operator for roof edges. These edges typically occur at the concave junctions of two planar faces in polyhedral objects. The results are shown in Fig. 3. Again there are two subsidiary constraints, one for multiple responses and one for zero response to constant input.

A roof edge detector has not been incorporated into the implementation of the edge detector because it was found that ideal roof edges were relatively rare. In any case the ridge detector is an approximation to the ideal roof detector, and is adequate to cope with roofs. The situation may be different in the case of an edge detector designed explicitly to deal with images of polyhedra, like the Binford-Horn line-finder [14].

The method just described has been used to find optimal operators for both ridge and roof profiles and in addition it successfully finds the optimal step edge operator derived in Section IV. It should be possible to use it to find operators for arbitrary one-dimensional edges, and it should be possible to apply the method in two dimensions to find optimal detectors for various types of corner.

#### IV. A DETECTOR FOR STEP EDGES

We now specialize the results of the last section to the case where the input  $G(x)$  is step edge. Specifically we set  $G(x) = Au_{-1}(x)$  where  $u_n(x)$  is the  $n$ th derivative of a delta function, and  $A$  is the amplitude of the step. That is,

$$u_{-1}(x) = \begin{cases} 0, & \text{for } x < 0; \\ 1, & \text{for } x \geq 0; \end{cases} \quad (16)$$

and substituting for  $G(x)$  in (3) and (9) gives

$$\text{SNR} = \frac{A \left| \int_{-w}^0 f(x) dx \right|}{n_0 \sqrt{\int_{-w}^{+w} f^2(x) dx}} \quad (17)$$

$$\text{Localization} = \frac{A |f'(0)|}{n_0 \sqrt{\int_{-w}^{+w} f'^2(x) dx}} \quad (18)$$

Both of these criteria improve directly with the ratio  $A/n_0$  which might be termed the signal-to-noise ratio of the image. We now remove this dependence on the image and define two performance measures  $\Sigma$  and  $\Lambda$  which depend on the filter only:

$$\text{SNR} = \frac{A}{n_0} \Sigma(f) \quad \Sigma(f) = \frac{\left| \int_{-w}^0 f(x) dx \right|}{\sqrt{\int_{-w}^{+w} f^2(x) dx}} \quad (19)$$

$$\text{Localization} = \frac{A}{n_0} \Lambda(f') \quad \Lambda(f') = \frac{|f'(0)|}{\sqrt{\int_{-w}^{+w} f'^2(x) dx}} \quad (20)$$

Suppose now that we form a spatially scaled filter  $f_w$  from  $f$ , where  $f_w(x) = f(x/w)$ . Recall from the end of Section II that the multiple response criterion is unaffected by spatial scaling. When we substitute  $f_w$  into (19) and (20) we obtain for the performance of the scaled filter:

$$\Sigma(f_w) = \sqrt{w} \Sigma(f) \quad \text{and} \quad \Lambda(f'_w) = \frac{1}{\sqrt{w}} \Lambda(f'). \quad (21)$$

The first of these equations is quite intuitive, and implies that a filter with a broad impulse response will have better signal-to-noise ratio than a narrow filter when applied to a step edge. The second is less obvious, and it implies that a narrow filter will give better localization than a broad one. What is surprising is that the changes are inversely related, that is, both criteria either increase or decrease by  $\sqrt{w}$ . There is an uncertainty principle relating the detection and localization performance of the

step edge detector. Through spatial scaling of  $f$  we can trade off detection performance against localization, but we cannot improve both simultaneously. This suggests that a natural choice for the composite criterion would be the product of (19) and (20), since this product would be invariant under changes in scale.

$$\Sigma(f) \Lambda(f') = \frac{\left| \int_{-w}^0 f(x) dx \right|}{\sqrt{\int_{-w}^{+w} f^2(x) dx}} \frac{|f'(0)|}{\sqrt{\int_{-w}^{+w} f'^2(x) dx}} \quad (22)$$

The solutions to the maximization of this expression will be a *class of functions* all related by spatial scaling. In fact this result is independent of the method of combination of the criteria. To see this we assume that there is a function  $f$  which gives the best localization  $\Lambda$  for a particular  $\Sigma$ . That is, we find  $f$  such that

$$\Sigma(f) = c_1 \quad \text{and} \quad \Lambda(f') \text{ is maximized.} \quad (23)$$

Now suppose we seek a second function  $f_w$  which gives the best possible localization while its signal-to-noise ratio is fixed to a different value, i.e.,

$$\Sigma(f_w) = c_2 \quad \text{while} \quad \Lambda(f'_w) \text{ is maximized.} \quad (24)$$

If we now define  $f_1(x)$  in terms of  $f_w(x)$  as  $f_1(x) = f_w(xw)$  where

$$w = c_2^2/c_1^2$$

then the constraint on  $f_w$  in (24) translates to a constraint on  $f_1$  which is identical to (23), and (24) can be rewritten as

$$\Sigma(f_1) = c_1 \quad \text{and} \quad \frac{1}{\sqrt{w}} \Lambda(f'_1) \text{ is maximized} \quad (25)$$

which has the solution  $f_1 = f$ . So if we find a single such function  $f$ , we can obtain maximal localization for any fixed signal-to-noise ratio by scaling  $f$ . The design problem for step edge detection has a *single unique* (up to spatial scaling) solution regardless of the absolute values of signal to noise ratio or localization.

The optimal filter is implicitly defined by (22), but we must transform the problem slightly before we can apply the calculus of variations. Specifically, we transform the maximization of (22) into a constrained minimization that involves only integral functionals. All but one of the integrals in (22) are set to undetermined constant values. We then find the extreme value of the remaining integral (since it will correspond to an extreme in the total expression) as a function of the undetermined constants. The values of the constants are then chosen so as to maximize the original expression, which is now a function only of these constants. Given the constants, we can uniquely specify the function  $f(x)$  which gives a maximum of the composite criterion.

A second modification involves the limits of the integrals. The two integrals in the denominator of (22) have

limits at  $+W$  and  $-W$ , while the integral in the numerator has one limit at 0 and the other at  $-W$ . Since the function  $f$  should be antisymmetric, we can use the latter limits for all integrals. The denominator integrals will have half the value over this subrange that they would have over the full range. Also, this enables the value of  $f'(0)$  to be set as a boundary condition, rather than expressed as an integral of  $f''$ . If the integral to be minimized shares the same limits as the constraint integrals, it is possible to exploit the *isoperimetric constraint* condition (see [6, p. 216]). When this condition is fulfilled, the constrained optimization can be reduced to an unconstrained optimization using Lagrange multipliers for the constraint functionals. The problem of finding the maximum of (22) reduces to the minimization of the integral in the denominator of the SNR term, subject to the constraint that the other integrals remain constant. By the principle of reciprocity, we could have chosen to extremize any of the integrals while keeping the others constant, and the solution should be the same.

We seek some function  $f$  chosen from a space of *admissible* functions that minimizes the integral

$$\int_{-W}^0 f^2(x) dx \quad (26)$$

subject to

$$\begin{aligned} \int_{-W}^0 f(x) dx &= c_1 & \int_{-W}^0 f'^2(x) dx &= c_2 \\ \int_{-W}^0 f''^2(x) dx &= c_3 & f'(0) &= c_4. \end{aligned} \quad (27)$$

The space of admissible functions in this case will be the space of all continuous functions that satisfy certain boundary conditions, namely that  $f(0) = 0$  and  $f(-W) = 0$ . These boundary conditions are necessary to ensure that the integrals evaluated over finite limits accurately represent the infinite convolution integrals. That is, if the  $n$ th derivative of  $f$  appears in some integral, the function must be continuous in its  $(n-1)$ st derivative over the range  $(-\infty, +\infty)$ . This implies that the values of  $f$  and its first  $(n-1)$  derivatives must be zero at the limits of integration, since they are zero outside this range.

The functional to be minimized is of the form  $\int_a^b F(x, f, f', f'')$  and we have a series of constraints that can be written in the form  $\int_a^b G_i(x, f, f', f'') = c_i$ . Since the constraints are isoperimetric, i.e., they share the same limits of integration as the integral being minimized, we can form a composite functional using Lagrange multipliers [6]. The functional is a linear combination of the functionals that appear in the expression to be minimized and in the constraints. Finding a solution to the unconstrained maximization of  $\Psi(x, f, f', f'')$  is equivalent to finding the solution to the constrained problem. The composite functional is

$$\begin{aligned} \Psi(x, f, f', f'') &= F(x, f, f', f'') + \lambda_1 G_1(x, f, f', f'') \\ &+ \lambda_2 G_2(x, f, f', f'') + \dots \end{aligned}$$

Substituting,

$$\Psi(x, f, f', f'') = f^2 + \lambda_1 f'^2 + \lambda_2 f''^2 + \lambda_3 f. \quad (28)$$

It may be seen from the form of this equation that the choice of which integral is extremized and which are constraints is arbitrary, the solution will be the same. This is an example of the *reciprocity* that was mentioned earlier. The choice of an integral from the denominator is simply convenient since the standard form of variational problem is a minimization problem. The Euler equation that corresponds to the functional  $\Psi$  is

$$\Psi_f - \frac{d}{dx} \Psi_{f'} + \frac{d^2}{dx^2} \Psi_{f''} = 0 \quad (29)$$

where  $\Psi_f$  denotes the partial derivative of  $\Psi$  with respect to  $f$ , etc. We substitute for  $\Psi$  from (28) in the Euler equation giving:

$$2f(x) - 2\lambda_1 f''(x) + 2\lambda_2 f''''(x) + \lambda_3 = 0. \quad (30)$$

The solution of this differential equation is the sum of a constant and a set of four exponentials of the form  $e^{\gamma x}$  where  $\gamma$  derives from the solution of the corresponding homogeneous differential equation. Now  $\gamma$  must satisfy

$$2 - 2\lambda_1 \gamma^2 + 2\lambda_2 \gamma^4 = 0$$

so

$$\gamma^2 = \frac{\lambda_1}{2\lambda_2} \pm \frac{\sqrt{\lambda_1^2 - 4\lambda_2}}{2\lambda_2}. \quad (31)$$

This equation may have roots that are purely imaginary, purely real, or complex depending on the values of  $\lambda_1$  and  $\lambda_2$ . From the composite functional  $\Psi$  we can infer that  $\lambda_2$  is positive (since the integral of  $f''^2$  is to be minimized) but it is not clear what the sign or magnitude of  $\lambda_1$  should be. The Euler equation supplies a necessary condition for the existence of a minimum, but it is not a sufficient condition. By formulating such a condition we can resolve the ambiguity in the value of  $\lambda_1$ . To do this we must consider the second variation of the functional. Let

$$J[f] = \int_{x_0}^{x_1} \Psi(x, f, f', f'') dx.$$

Then by Taylor's theorem (see also [6, p. 214]),

$$J[f + \epsilon g] = J[f] + \epsilon J_1[f, g] + \frac{1}{2} \epsilon^2 J_2[f + \rho g, g]$$

where  $\rho$  is some number between 0 and  $\epsilon$ , and  $g$  is chosen from the space of admissible functions, and where

$$\begin{aligned} J_1[f, g] &= \int_{x_0}^{x_1} \Psi_f g + \Psi_{f'} g' + \Psi_{f''} g'' dx \\ J_2[f, g] &= \int_{x_0}^{x_1} \Psi_{ff} g^2 + \Psi_{ff'} g' g' + \Psi_{ff''} g'' g'' \\ &+ 2\Psi_{ff'} g g' + 2\Psi_{f'f''} g' g'' + 2\Psi_{f''f''} g'' g'' dx. \end{aligned} \quad (32)$$

Note that  $J_1$  is nothing more than the integral of  $g$  times the Euler equation for  $f$  (transformed using integration by parts) and will be zero if  $f$  satisfies the Euler equation. We can now define the second variation  $\delta^2 J$  as

$$\delta^2 J = \frac{\epsilon^2}{2} J_2[f, g].$$

The necessary condition for a minimum is  $\delta^2 J \geq 0$ . We compute the second partial derivatives of  $\Psi$  from (28) and we get

$$J_1[f + g] = \int_{x_0}^{x_1} 2g^2 + 2\lambda_1 g'^2 + 2\lambda_2 g''^2 dx \geq 0. \quad (33)$$

Using the fact that  $g$  is an admissible function and therefore vanishes at the integration limits, we transform the above using integration by parts to

$$2 \int_{x_0}^{x_1} g^2 - \lambda_1 g g'' + \lambda_2 g''^2 dx \geq 0$$

which can be written as

$$2 \int_{x_0}^{x_1} \left( g^2 - \frac{\lambda_1}{2} g''^2 \right) + \left( \lambda_2 - \frac{\lambda_1^2}{4} \right) g''^2 dx \geq 0.$$

The integral is guaranteed to be positive if the expression being integrated is positive for all  $x$ , so if

$$4\lambda_2 > \lambda_1^2$$

then the integral will be positive for all  $x$  and for arbitrary  $g$ , and the extremum will certainly be a minimum. If we refer back to (31) we find that this condition is precisely that which gives complex roots for  $\gamma$ , so we have both guaranteed the existence of a minimum and resolved a possible ambiguity in the form of the solution. We can now proceed with the derivation and assume four complex roots of the form  $\gamma = \pm\alpha \pm i\omega$  with  $\alpha, \omega$  real. Now  $\gamma^2 = \alpha^2 - \omega^2 \pm 2i\alpha\omega$  and equating real and imaginary parts with (31) we obtain

$$\alpha^2 - \omega^2 = \frac{\lambda_1}{2\lambda_2} \quad \text{and} \quad 4\alpha^2\omega^2 = \frac{4\lambda_2 - \lambda_1^2}{4\lambda_2^2}. \quad (34)$$

The general solution in the range  $[-W, 0]$  may now be written

$$f(x) = a_1 e^{\alpha x} \sin \omega x + a_2 e^{\alpha x} \cos \omega x + a_3 e^{-\alpha x} \sin \omega x + a_4 e^{-\alpha x} \cos \omega x + c. \quad (35)$$

This function is subject to the boundary conditions

$$f(0) = 0 \quad f(-W) = 0 \quad f'(0) = s \quad f'(-W) = 0$$

where  $s$  is an unknown constant equal to the slope of the function  $f$  at the origin. Since  $f(x)$  is asymmetric, we can extend the above definition to the range  $[-W, W]$  using  $f(-x) = -f(x)$ . The four boundary conditions enable us to solve for the quantities  $a_1$  through  $a_4$  in terms of the unknown constants  $\alpha, \omega, c$ , and  $s$ . The boundary conditions may be rewritten

$$\begin{aligned} a_2 + a_4 + c &= 0 \\ a_1 e^{\alpha} \sin \omega + a_2 e^{\alpha} \cos \omega + a_3 e^{-\alpha} \sin \omega \\ &+ a_4 e^{-\alpha} \cos \omega + c = 0 \\ a_1 \omega + a_2 \alpha + a_3 \omega - a_4 \alpha &= s \\ a_1 e^{\alpha} (\alpha \sin \omega + \omega \cos \omega) + a_2 e^{\alpha} (\alpha \cos \omega \\ &- \omega \sin \omega) + a_3 e^{-\alpha} (-\alpha \sin \omega + \omega \cos \omega) \\ &+ a_4 e^{-\alpha} (-\alpha \cos \omega - \omega \sin \omega) = 0. \end{aligned} \quad (36)$$

These equations are linear in the four unknowns  $a_1, a_2, a_3, a_4$  and when solved they yield

$$\begin{aligned} a_1 &= c(\alpha(\beta - \alpha) \sin 2\omega - \alpha\omega \cos 2\omega + (-2\omega^2 \sinh \alpha \\ &+ 2\alpha^2 e^{-\alpha}) \sin \omega + 2\alpha\omega \sinh \alpha \cos \omega \\ &+ \omega e^{-2\alpha}(\alpha + \beta) - \beta\omega)/4(\omega^2 \sinh^2 \alpha - \alpha^2 \sin^2 \omega) \\ a_2 &= c(\alpha(\beta - \alpha) \cos 2\omega + \alpha\omega \sin 2\omega - 2\alpha\omega \cosh \alpha \\ &\cdot \sin \omega - 2\omega^2 \sinh \alpha \cos \omega + 2\omega^2 e^{-\alpha} \sinh \alpha \\ &+ \alpha(\alpha - \beta))/4(\omega^2 \sinh^2 \alpha - \alpha^2 \sin^2 \omega) \\ a_3 &= c(-\alpha(\beta + \alpha) \sin 2\omega + \alpha\omega \cos 2\omega + (2\omega^2 \sinh \alpha \\ &+ 2\alpha^2 e^{\alpha}) \sin \omega + 2\alpha\omega \sinh \alpha \cos \omega \\ &+ \omega e^{2\alpha}(\beta - \alpha) - \beta\omega)/4(\omega^2 \sinh^2 \alpha - \alpha^2 \sin^2 \omega) \\ a_4 &= c(-\alpha(\beta + \alpha) \cos 2\omega - \alpha\omega \sin 2\omega + 2\alpha\omega \cosh \alpha \\ &\cdot \sin \omega + 2\omega^2 \sinh \alpha \cos \omega - 2\omega^2 e^{\alpha} \sinh \alpha \\ &+ \alpha(\alpha - \beta))/4(\omega^2 \sinh^2 \alpha - \alpha^2 \sin^2 \omega) \end{aligned} \quad (37)$$

where  $\beta$  is the slope  $s$  at the origin divided by the constant  $c$ . On inspection of these expressions we can see that  $a_3$  can be obtained from  $a_1$  by replacing  $\alpha$  by  $-\alpha$ , and similarly for  $a_4$  from  $a_2$ .

The function  $f$  is now parametrized in terms of the constants  $\alpha, \omega, \beta$ , and  $c$ . We have still to find the values of these parameters which maximize the quotient of integrals that forms our composite criterion. To do this we first express each of the integrals in terms of the constants. Since these integrals are very long and uninteresting, they are not given here but may be found in [4]. We have reduced the problem of optimizing over an infinite-dimensional space of functions to a nonlinear optimization in three variables  $\alpha, \omega$ , and  $\beta$  (not surprisingly, the combined criterion does not depend on  $c$ ). Unfortunately the resulting criterion, which must still satisfy the multiple response constraint, is probably too complex to be solved analytically, and numerical methods must be used to provide the final solution.

The shape of  $f$  will depend on the multiple response constraint, i.e., it will depend on how far apart we force the adjacent responses. Fig. 5 shows the operators that result from particular choices of this distance. Recall that there was no single best function for arbitrary  $\omega$ , but a class of functions which were obtained by scaling a pro-



prototype function by  $\omega$ . We will want to force the responses further apart as the signal-to-noise ratio in the image is lowered, but it is not clear what the value of signal-to-noise ratio will be for a single operator. In the context in which this operator is used, several operator widths are available, and a decision procedure is applied to select the smallest operator that has an output signal-to-noise ratio above a fixed threshold. With this arrangement the operators will spend much of the time operating close to their output  $\Sigma$  thresholds. We try to choose a spacing for which the probability of a multiple response error is comparable to the probability of an error due to thresholding.

A rough estimate for the probability of a spurious maximum in the neighborhood of the true maximum can be formed as follows. If we look at the response of  $f$  to an ideal step we find that its second derivative has magnitude  $|Af'(0)|$  at  $x = 0$ . There will be only one maximum near the center of the edge if  $|Af'(0)|$  is greater than the second derivative of the response to noise only. This latter quantity, denoted  $s_n$ , is a Gaussian random variable with standard deviation

$$n_0 \sigma_s = n_0 \left( \int_{-W}^{+W} f''^2(x) dx \right)^{1/2}.$$

The probability  $p_m$  that the noise slope  $s_n$  exceeds  $Af'(0)$  is given in terms of the normal distribution function  $\Phi$

$$p_m = 1 - \Phi \left( \frac{A|f'(0)|}{n_0 \sigma_s} \right). \quad (38)$$

We can choose a value for this probability as an acceptable error rate and this will determine the ratio of  $f'(0)$  to  $\sigma_s$ . We can relate the probability of a multiple response  $p_m$  to the probability of falsely marking an edge  $p_f$  which is

$$p_f = 1 - \Phi \left( \frac{A}{n_0} \Sigma \right) \quad (39)$$

by setting  $p_m = p_f$ . This is a natural choice since it makes a detection error or a multiple response error equally likely. Then from (38) and (39) we have

$$\frac{|f'(0)|}{\sigma_s} = \Sigma. \quad (40)$$

In practice it was impossible to find filters which satisfied this constraint, so instead we search for a filter satisfying

$$\frac{|f'(0)|}{\sigma_s} = r \Sigma \quad (41)$$

where  $r$  is as close as possible to 1. The performance indexes and parameter values for several filters are given in Fig. 4. The  $a_i$  coefficients for all these filters can be found from (37), by fixing  $c$  to, say,  $c = 1$ . Unfortunately, the largest value of  $r$  that could be obtained using the constrained numerical optimization was about 0.576 for filter number 6 in the table. In our implementation, we have

Filter Parameters						
n	$x_{max}$	$\Sigma A$	$r$	$\alpha$	$\omega$	$\beta$
1	0.15	4.21	0.215	21.59550	0.12250	63.97566
2	0.3	2.87	0.313	12.47120	0.38284	31.26860
3	0.5	2.13	0.417	7.85869	2.62856	18.28800
4	0.8	1.57	0.515	5.06500	2.56770	11.06100
5	1.0	1.33	0.561	3.45580	0.07161	4.80684
6	1.2	1.12	0.576	2.05220	1.56939	2.91540
7	1.4	0.75	0.484	0.00297	3.50350	7.47700

Fig. 4. Filter parameters and performance measures for the filters illustrated in Fig. 5.

approximated this filter using the first derivative of a Gaussian as described in the next section.

The first derivative of Gaussian operator, or even filter 6 itself, should not be taken as the final word in edge detection filters, even with respect to the criteria we have used. If we are willing to tolerate a slight reduction in multiple response performance  $r$ , we can obtain significant improvements in the other two criteria. For example, filters 4 and 5 both have significantly better  $\Sigma A$  product than filter 6, and only slightly lower  $r$ . From Fig. 5 we can see that these filters have steeper slope at the origin, suggesting that the performance gain is mostly in localization, although this has not been verified experimentally. A thorough empirical comparison of these other operators remains to be done, and the theory in this case is unclear on how best to make the tradeoff.

## V. AN EFFICIENT APPROXIMATION

The operator derived in the last section as filter number 6, and illustrated in Fig. 6, can be approximated by the first derivative of a Gaussian  $G'(x)$ , where

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

The reason for doing this is that there are very efficient ways to compute the two-dimensional extension of the filter if it can be represented as some derivative of a Gaussian. This is described in detail elsewhere [4], but for the present we will compare the theoretical performance of a first derivative of a Gaussian filter to the optimal operator. The impulse response of the first derivative filter is

$$f(x) = -\frac{x}{\sigma^2} \exp \left( -\frac{x^2}{2\sigma^2} \right) \quad (42)$$

and the terms in the performance criteria have the values

$$\begin{aligned} |f'(0)| &= \frac{1}{\sigma_s} \\ \int_{-\infty}^{+\infty} f(x) dx &= 1 & \int_{-\infty}^{+\infty} f^2(x) dx &= \frac{\sqrt{\pi}}{2\sigma} \\ \int_{-\infty}^{+\infty} f'^2(x) dx &= \frac{3\sqrt{\pi}}{4\sigma^3} & \int_{-\infty}^{+\infty} f''^2(x) dx &= \frac{15\sqrt{\pi}}{8\sigma^5}. \end{aligned} \quad (43)$$

The overall performance index for this operator is

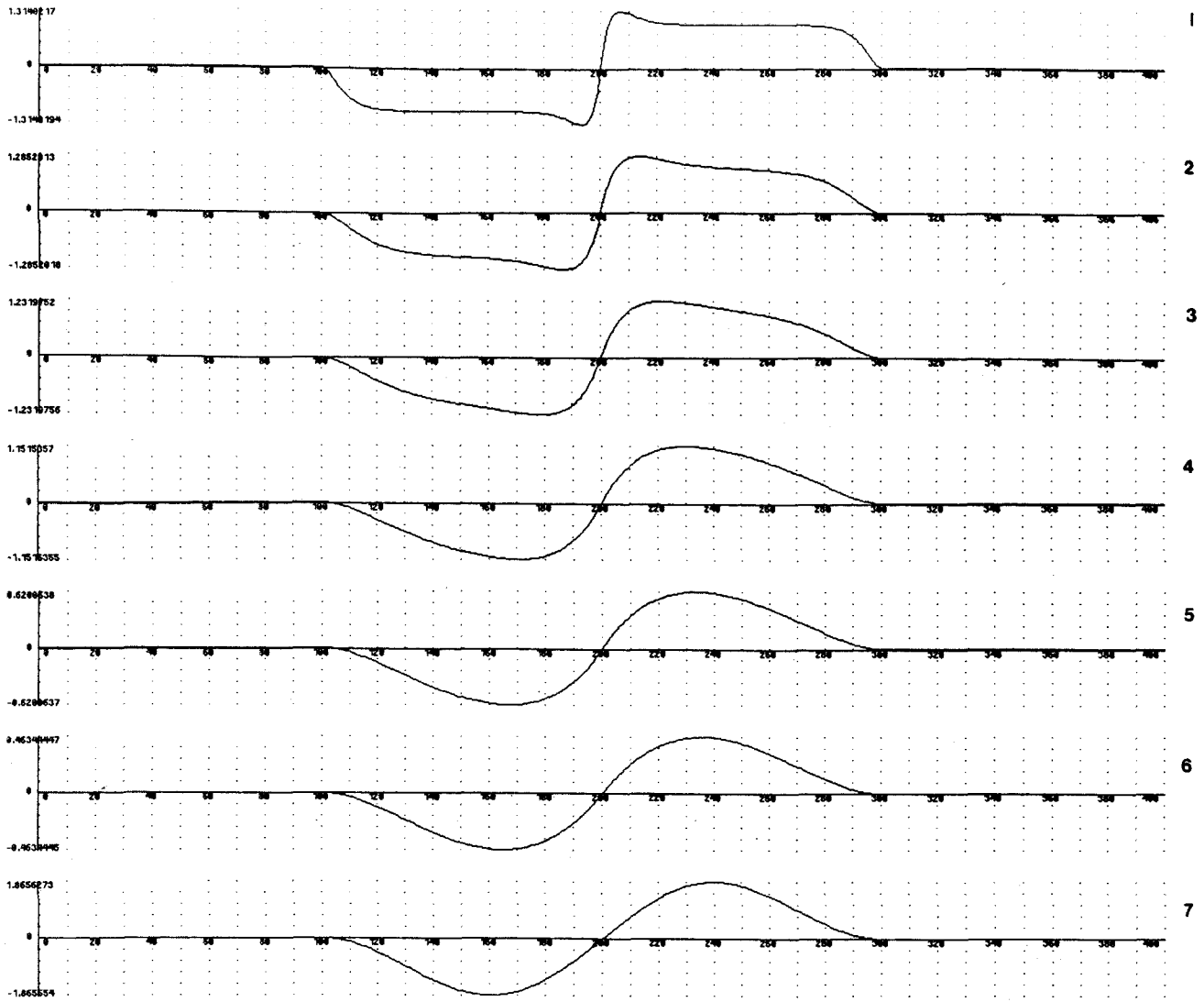


Fig. 5. Optimal step edge operators for various values of  $x_{max}$ . From top to bottom, they are  $x_{max} = 0.15, 0.3, 0.5, 0.8, 1.0, 1.2, 1.4$ .

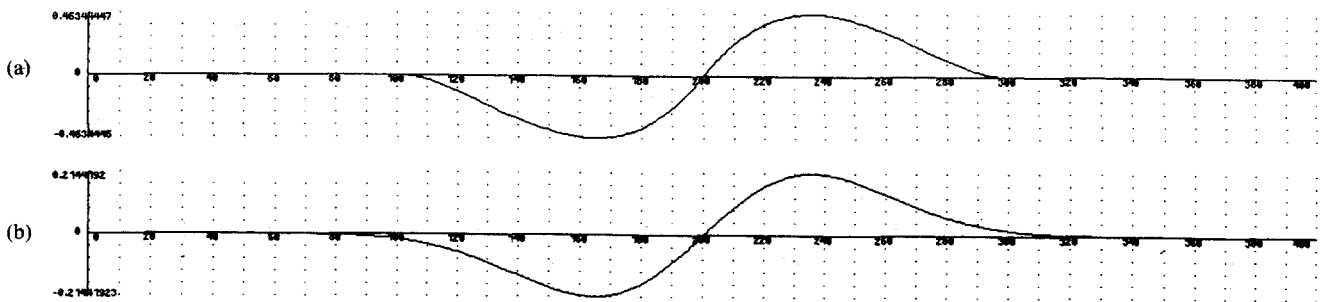


Fig. 6. (a) The optimal step edge operator. (b) The first derivative of a Gaussian.

$$\Sigma\Lambda = \sqrt{\frac{8}{3\pi}} \approx 0.92 \quad (44)$$

while the  $r$  value is, from (41),

$$r = \sqrt{\frac{4}{15}} \approx 0.51.$$

The performance of the first derivative of Gaussian operator above is worse than the optimal operator by about 20 percent and its multiple response measure  $r$ , is worse by about 10 percent. It would probably be difficult to detect a difference of this magnitude by looking at the performance of the two operators on real images, and because the first derivative of Gaussian operator can be computed with much less effort in two dimensions, it has

been used exclusively in experiments. The impulse responses of the two operators can be compared in Fig. 6.

A close approximation of the first derivative of Gaussian operator was suggested by Macleod [16] for step edge detection. Macleod's operator is a difference of two displaced two-dimensional Gaussians. It was evaluated in Fram and Deutsch [7] and compared very favorably with several other schemes considered in that paper. There are also strong links with the Laplacian of Gaussian operator suggested by Marr and Hildreth [18]. In fact, a one-dimensional Marr-Hildreth edge detector is almost identical with the operator we have derived because maxima in the output of a first derivative operator will correspond to zero-crossings in the Laplacian operator as used by Marr and Hildreth. In two dimensions however, the directional properties of our detector enhance its detection and localization performance compared to the Laplacian. Another important difference is that the amplitude of the response at a maximum provides a good estimate of edge strength, because the SNR criterion is the ratio of this response to the noise response. The Marr-Hildreth operator does not use any form of thresholding, but an adaptive thresholding scheme can be used to advantage with our first derivative operator. In the next section we describe such a scheme, which includes noise estimation and a novel method for thresholding edge points along contours.

We have derived our optimal operator to deal with known image features in Gaussian noise. Edge detection between textured regions is another important problem. This is straightforward if the texture can be modelled as the response of some filter  $t(x)$  to Gaussian noise. We can then treat the texture as a noise signal, and the response of the filter  $f(x)$  to the texture is the same as the response of the filter  $(f * t)(x)$  to Gaussian noise. Making this replacement in each integral in the performance criteria that computes a noise response gives us the texture edge design problem. The generalization to other types of texture is not as easy, and for good discrimination between known texture types, a better approach would involve a Markov image model as in [5].

## VI. NOISE ESTIMATION AND THRESHOLDING

To estimate noise from an operator output, we need to be able to separate its response to noise from the response due to step edges. Since the performance of the system will be critically dependent on the accuracy of this estimate, it should also be formulated as an optimization. Wiener filtering is a method for optimally estimating one component of a two-component signal, and can be used to advantage in this application. It requires knowledge of the autocorrelation functions of the two components and of the combined signal. Once the noise component has been optimally separated, we form a global histogram of noise amplitude, and estimate the noise strength from some fixed percentile of the noise signal.

Let  $g_1(x)$  be the signal we are trying to detect (in this case the noise output), and  $g_2(x)$  be some disturbance (paradoxically this will be the edge response of our filter),

then denote the autocorrelation function of  $g_1$  as  $R_{11}(\tau)$  and that of  $g_2$  as  $R_{22}(\tau)$ , and their cross-correlation as  $R_{12}(\tau)$ , where the correlation of two real functions is defined as follows:

$$R_{ij}(\tau) = \int_{-\infty}^{+\infty} g_i(x) g_j(x + \tau) dx.$$

We assume in this case that the signal and disturbance are uncorrelated, so  $R_{12}(\tau) = 0$ . The optimal filter is  $K(x)$ , which is implicitly defined as follows [30]:

$$R_{11}(\tau) = \int_{-\infty}^{+\infty} (R_{11}(\tau - x) + R_{22}(\tau - x)) K(x) dx.$$

Since the autocorrelation of the output of a filter in response to white noise is equal to the autocorrelation of its impulse response, we have

$$R_{11}(x) = k_3 \left( \frac{x^2}{2\sigma^2} - 1 \right) \exp \left( -\frac{x^2}{4\sigma^2} \right)$$

If  $g_2$  is the response of the operator derived in (42) to a step edge then we will have  $g_2(x) = k \exp(-x/2\sigma^2)$  and

$$R_{22}(x) = k_2 \exp \left( -\frac{x^2}{4\sigma^2} \right).$$

In the case where the amplitude of the edge is large compared to the noise,  $R_{22} + R_{11}$  is approximately a Gaussian and  $R_{11}$  is the second derivative of a Gaussian of the same  $\sigma$ . Then the optimal form of  $K$  is the second derivative of an impulse function.

The filter  $K$  above is convolved with the output of the edge detection operator and the result is squared. The next step is the estimation of the mean-squared noise from the local values. Here there are several possibilities. The simplest is to average the squared values over some neighborhood, either using a moving average filter or by taking an average over the entire image. Unfortunately, experience has shown that the filter  $K$  is very sensitive to step edges, and that as a consequence the noise estimate from any form of averaging is heavily colored by the density and strength of edges.

In order to gain better separation between signal and noise we can make use of the fact that the amplitude distribution of the filter response tends to be different for edges and noise. By our model, the noise response should have a Gaussian distribution, while the step edge response will be composed of large values occurring very infrequently. If we take a histogram of the filter values, we should find that the positions of the low percentiles (say less than 80 percent) will be determined mainly the noise energy, and that they are therefore useful estimators for noise. A global histogram estimate is actually used in the current implementation of the algorithm.

Even with noise estimation, the edge detector will be susceptible to streaking if it uses only a single threshold. Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the

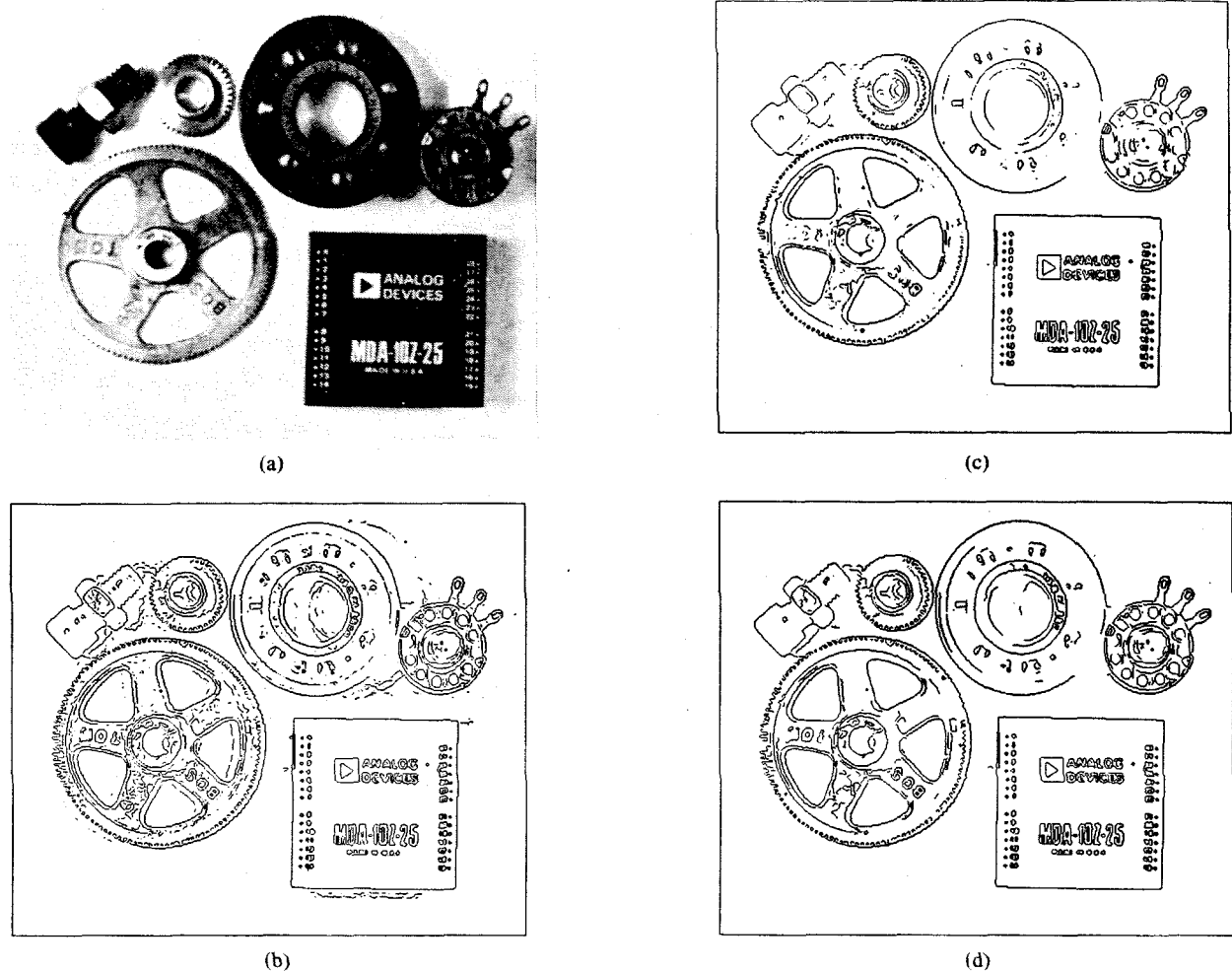


Fig. 7. (a) Parts image, 576 by 454 pixels. (b) Image thresholded at  $T_1$ . (c) Image thresholded at  $2 T_1$ . (d) Image thresholded with hysteresis using both the thresholds in (a) and (b).

threshold along the length of the contour. Suppose we have a single threshold set at  $T_1$ , and that there is an edge in the image such that the response of the operator has mean value  $T_1$ . There will be some fluctuation of the output amplitude due to noise, even if the noise is very slight. We expect the contour to be above threshold only about half the time. This leads to a broken edge contour. While this is a pathological case, streaking is a very common problem with edge detectors that employ thresholding. It is very difficult to set a threshold so that there is small probability of marking noise edges while retaining high sensitivity. An example of the effect of streaking is given in Fig. 7.

One possible solution to this problem, used by Pentland [22] with Marr-Hildreth zero-crossings, is to average the edge strength of a contour over part of its length. If the average is above the threshold, the entire segment is marked. If the average is below threshold, no part of the contour appears in the output. The contour is segmented by breaking it at maxima in curvature. This segmentation is necessary in the case of zero-crossings since the zero-crossings always form closed contours, which obviously do not always correspond to contours in the image.

In the current algorithm, no attempt is made to presegment contours. Instead the thresholding is done with hysteresis. If any part of a contour is above a high threshold, those points are immediately output, as is the entire connected segment of contour which contains the points and which lies above a low threshold. The probability of streaking is greatly reduced because for a contour to be broken it must now fluctuate above the high threshold and below the low threshold. Also the probability of isolated false edge points is reduced because the strength of such points must be above a higher threshold. The ratio of the high to low threshold in the implementation is in the range two or three to one.

## VII. TWO OR MORE DIMENSIONS

In one dimension we can characterize the position of a step edge in space with one position coordinate. In two dimensions an edge also has an orientation. In this section we will use the term "edge direction" to mean the direction of the tangent to the contour that the edge defines in two dimensions. Suppose we wish to detect edges of a particular orientation. We create a two-dimensional mask for this orientation by convolving a linear edge detection

function aligned normal to the edge direction with a projection function parallel to the edge direction. A substantial savings in computational effort is possible if the projection function is a Gaussian with the same  $\sigma$  as the (first derivative of the) Gaussian used as the detection function. It is possible to create such masks by convolving the image with a symmetric two-dimensional Gaussian and then differentiating normal to the edge direction. In fact we do not have to do this in every direction because the slope of a smooth surface in any direction can be determined exactly from its slope in two directions. This form of directional operator, while simple and inexpensive to compute, forms the heart of the more elaborate detector which will be described in the next few sections.

Suppose we wish to convolve the image with an operator  $G_n$  which is the first derivative of a two-dimensional Gaussian  $G$  in some direction  $n$ , i.e.,

$$G = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

and

$$G_n = \frac{\partial G}{\partial n} = n \cdot \nabla G. \quad (45)$$

Ideally,  $n$  should be oriented normal to the direction of an edge to be detected, and although this direction is not known *a priori*, we can form a good estimate of it from the smoothed gradient direction

$$n = \frac{\nabla(G * I)}{|\nabla(G * I)|} \quad (46)$$

where  $*$  denotes convolution. This turns out to be a very good estimator for edge normal direction for steps, since a smoothed step has strong gradient normal to the edge. It is exact for straight line edges in the absence of noise, and the Gaussian smoothing keeps it relatively insensitive to noise.

An edge point is defined to be a local maximum (in the direction  $n$ ) of the operator  $G_n$  applied to the image  $I$ . At a local maximum, we have

$$\frac{\partial}{\partial n} G_n * I = 0$$

and substituting for  $G_n$  from (45) and associating Gaussian convolution, the above becomes

$$\frac{\partial^2}{\partial n^2} G * I = 0. \quad (47)$$

At such an edge point, the edge strength will be the magnitude of

$$|G_n * I| = |\nabla(G * I)|. \quad (48)$$

Because of the associativity of convolution, we can first convolve with a symmetric Gaussian  $G$  and then compute directional second derivative zeros to locate edges (47), and use the magnitude of (48) to estimate edge strength. This is equivalent to detecting and locating the edge using

the directional operator  $G_n$ , but we need not know the direction  $n$  before convolution.

The form of nonlinear second derivative operator in (47) has also been used by Torre and Poggio [28] and by Haralick [10]. It also appears in Prewitt [23] in the context of edge enhancement. A rather different two-dimensional extension is proposed by Spacek [26] who uses one-dimensional filters aligned normal to the edge direction but without extending them along the edge direction. Spacek starts with a one-dimensional formulation which maximizes the product of the three performance criteria defined in Section II, and leads to a step edge operator which differs slightly from the one we derived in Section IV. Gennert [8] addresses the two-dimensional edge detector problem directly, and applies a set of directional first derivative operators at each point in the image. The operators have limited extent along the edge direction and produce good results at sharp changes in edge orientation and corners.

The operator (47) actually locates either maxima or minima by locating the zero-crossings in the second derivative in the edge direction. In principle it could be used to implement an edge detector in an arbitrary number of dimensions, by first convolving the image with a symmetric  $n$ -dimensional Gaussian. The convolution with an  $n$ -dimensional Gaussian is highly efficient because the Gaussian is separable into  $n$  one-dimensional filters.

But there are other more pressing reasons for using a smooth projection function such as a Gaussian. When we apply a linear operator to a two-dimensional image, we form at every point in the output a weighted sum of some of the input values. For the edge detector described here, this sum will be a difference between local averages on different sides of the edge. This output, before nonmaximum suppression, represents a kind of moving average of the image. Ideally we would like to use an infinite projection function, but real edges are of limited extent. It is therefore necessary to window the projection function [9]. If the window function is abruptly truncated, e.g., if it is rectangular, the filtered image will not be smooth because of the very high bandwidth of this window. This effect is related to the Gibbs phenomenon in Fourier theory which occurs when a signal is transformed over a finite window. When nonmaximum suppression is applied to this rough signal we find that edge contours tend to "wander" or that in severe cases they are not even continuous.

The solution is to use a smooth window function. In statistics, the Hamming and Hanning windows are typically used for moving averages. The Gaussian is a reasonable approximation to both of these, and it certainly has very low bandwidth for a given spatial width. (The Gaussian is the unique function with minimal product of bandwidth and frequency.) The effect of the window function becomes very marked for large operator sizes and it is probably the biggest single reason why operators with large support were not practical until the work of Marr and Hildreth on the Laplacian of Gaussian.

It is worthwhile here to compare the performance of

this kind of directional second derivative operator with the Laplacian. First we note that the two-dimensional Laplacian can be decomposed into components of second derivative in two arbitrary orthogonal directions. If we choose to take one of the derivatives in the direction of principal gradient, we find that the operator output will contain one contribution that is essentially the same as the operator described above, and also a contribution that is aligned along the edge direction. This second component contributes nothing to localization or detection (the surface is roughly constant in this direction), but increases the output noise.

In later sections we will describe an edge detector which incorporates operators of varying orientation and aspect ratio, but these are a superset of the operators used in the simple detector described above. In typical images, most of the edges are marked by the operators of the smallest width, and most of these by nonelongated operators. The simple detector performs well enough in these cases, and as detector complexity increases, performance gains tend to diminish. However, as we shall see in the following sections, there are cases when larger or more directional operators should be used, and that they do improve performance when they are applicable. The key to making such a complicated detector produce a coherent output is to design effective decision procedures for choosing between operator outputs at each point in the image.

### VIII. THE NEED FOR MULTIPLE WIDTHS

Having determined the optimal shape for the operator, we now face the problem of choosing the width of the operator so as to give the best detection/localization tradeoff in a particular application. In general the signal-to-noise ratio will be different for each edge within an image, and so it will be necessary to incorporate several widths of operator in the scheme. The decision as to which operator to use must be made dynamically by the algorithm and this requires a local estimate of the noise energy in the region surrounding the candidate edge. Once the noise energy is known, the signal-to-noise ratios of each of the operators will be known. If we then use a model of the probability distribution of the noise, we can effectively calculate the probability of a candidate edge being a false edge (for a given edge, this probability will be different for different operator widths).

If we assume that the *a priori* penalty associated with a falsely detected edge is independent of the edge strength, it is appropriate to threshold the detector outputs on probability of error rather than on magnitude of response. Once the probability threshold is set, the minimum acceptable signal-to-noise ratio is determined. However, there may be several operators with signal-to-noise ratios above the threshold, and in this case the smallest operator should be chosen, since it gives the best localization. We can afford to be conservative in the setting of the threshold since edges missed by the smallest operators may be picked up by the larger ones. Effectively the global tradeoff between error rate and localization remains, since choosing a high

signal-to-noise ratio threshold leads to a lower error rate, but will tend to give poorer localization since fewer edges will be recorded from the smaller operators.

In summary then, the first heuristic for choosing between operator outputs is that *small operator widths should be used whenever they have sufficient  $\Sigma$* . This is similar to the selection criterion proposed by Marr and Hildreth [18] for choosing between different Laplacian of Gaussian channels. In their case the argument was based on the observation that the smaller channels have higher resolution, i.e., there is less possibility of interference from neighboring edges. That argument is also very relevant in the present context, as to date there has been no consideration of the possibility of more than one edge in a given operator support. Interestingly, Rosenfeld and Thurston [25] proposed exactly the opposite criterion in the choice of operator for edge detection in texture. The argument given was that the larger operators give better averaging and therefore (presumably) better signal-to-noise ratios.

Taking the fine-to-coarse heuristic as a starting point, we need to form a local decision procedure that will enable us to decide whether to mark one or more edges when several operators in a neighborhood are responding. If the operator with the smallest width responds to an edge and if it has a signal-to-noise ratio above the threshold, we should immediately mark an edge at that point. We now face the problem that there will almost certainly be edges marked by the larger operators, but that these edges will probably not be exactly coincident with the first edge. A possible answer to this would be to suppress the outputs of all nearby operators. This has the undesirable effect of preventing the large channels for responding to "fuzzy" edges that are superimposed on the sharp edge.

Instead we use a "feature synthesis" approach. We begin by marking all the edges from the smallest operators. From these edges, we synthesize the large operator outputs that would have been produced if these were the only edges in the image. We then compare the actual operator outputs to the synthetic outputs. We mark additional edges only if the large operator has significantly greater response than what we would predict from the synthetic output. The simplest way to produce the synthetic outputs is to take the edges marked by a small operator in a particular direction, and convolve with a Gaussian normal to the edge direction for this operator. The  $\sigma$  of this Gaussian should be the same as the  $\sigma$  of the large channel detection filter.

This procedure can be applied repeatedly to first mark the edges from the second smallest scale that were not marked by at the first, and then to find the edges from the third scale that were not marked by either of the first two, etc. Thus we build up a cumulative edge map by adding those edges at each scale that were not marked by smaller scales. It turns out that in many cases the majority of edges are picked up by the smallest channel, and the later channels mark mostly shadow and shading edges, or edges between textured regions.

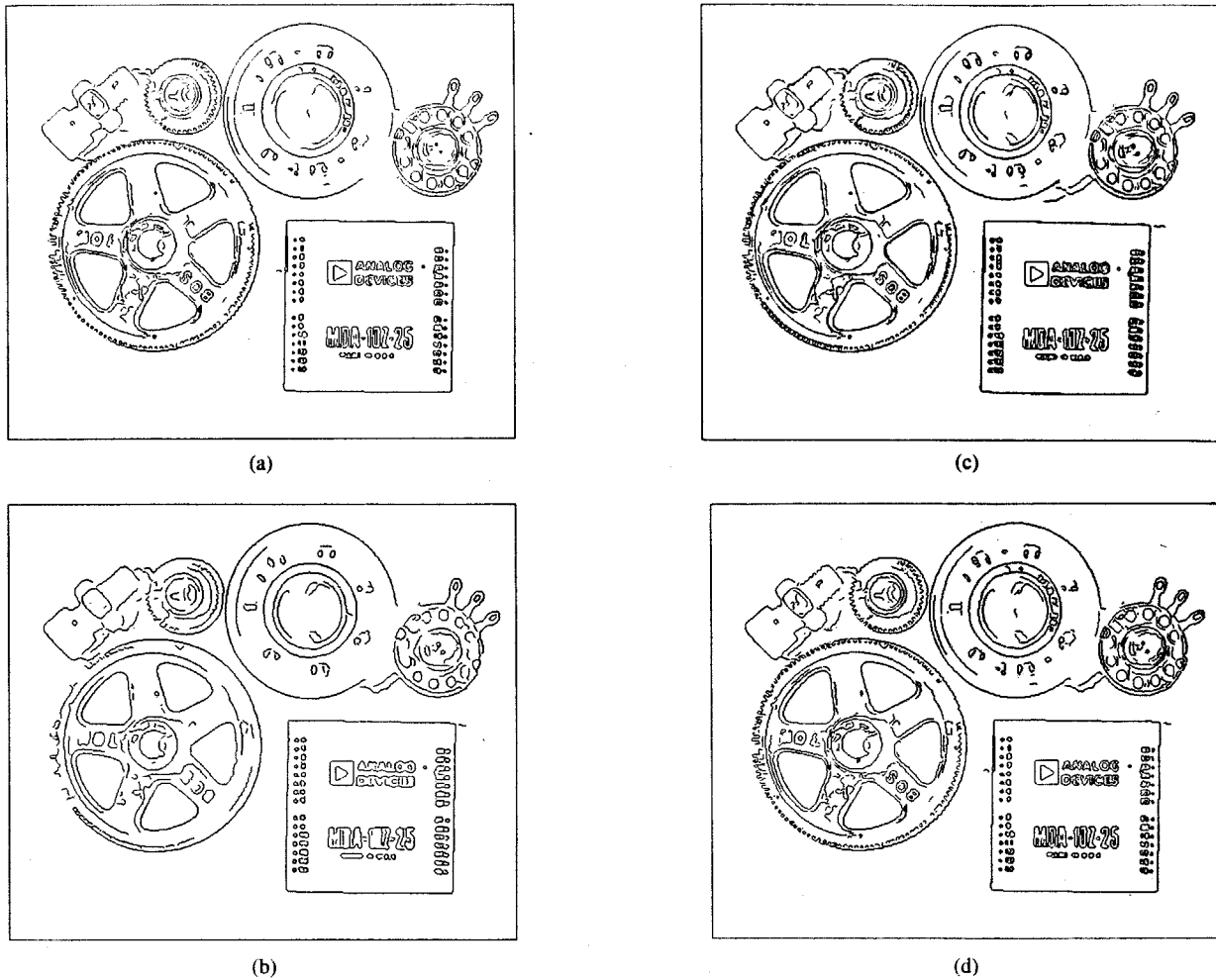


Fig. 8. (a) Edges from parts image at  $\sigma = 1.0$ . (b) Edges at  $\sigma = 2.0$ . (c) Superposition of the edges. (d) Edges combined using feature synthesis.

Some examples of feature synthesis applied to some sample images are shown in Figs. 8 and 9. Notice that most of the edges in Fig. 8 are marked by the smaller scale operator, and only a few additional edges, mostly shadows, are picked up by the coarser scale. However when the two sets of edges are superimposed, we notice that in many cases the responses of the two operators to the same edge are not spatially coincident. When feature synthesis is applied we find that redundant responses of the larger operator are eliminated leading to a sharp edge map.

By contrast, in Fig. 9 the edges marked by the two operators are essentially independent, and direct superposition of the edges gives a useful edge map. When we apply feature synthesis to these sets of edges we find that most of the edges at the coarser scale remain. Both Figs. 8 and 9 were produced by the edge detector with exactly the same set of parameters (other than operator size), and they were chosen to represent opposing extremes of image content across scale.

#### IX. THE NEED FOR DIRECTIONAL OPERATORS

So far we have assumed that the projection function is a Gaussian with the same  $\sigma$  as the Gaussian used for the

detection function. In fact both the detection and localization of the operator improve as the length of the projection function increases. We now prove this for the operator signal-to-noise ratio. The proof for localization is similar. We will consider a step edge in the  $x$  direction which passes through the origin. This edge can be represented by the equation

$$I(x, y) = Au_{-1}(y)$$

where  $u_{-1}$  is the unit step function, and  $A$  is the amplitude of the edge as before. Suppose that there is additive Gaussian noise of mean squared value  $n_{00}^2$  per unit area. If we convolve this signal with a filter whose impulse response is  $f(x, y)$ , then the response to the edge (at the origin) is

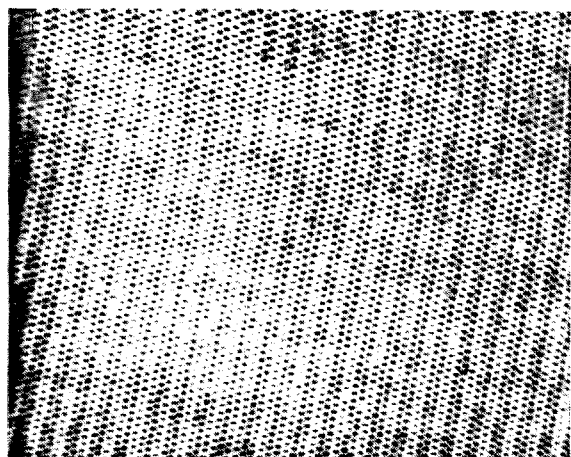
$$\int_{-\infty}^0 \int_{-\infty}^{+\infty} f(x, y) dx dy.$$

The root mean squared response to the noise only is

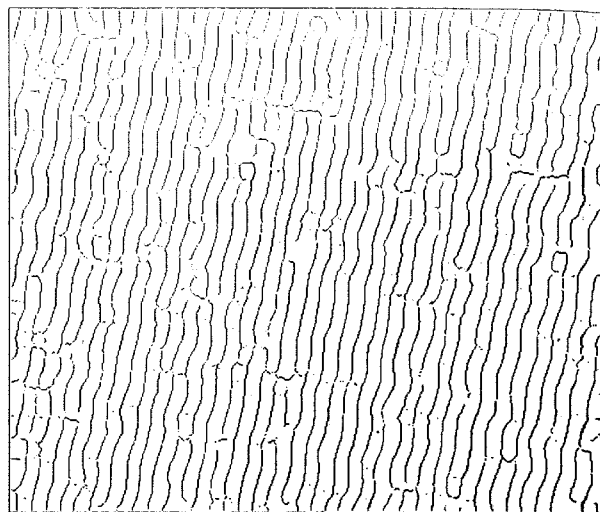
$$n_{00} \left( \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f^2(x, y) dx dy \right)^{1/2}$$

The signal-to-noise ratio is the quotient of these two

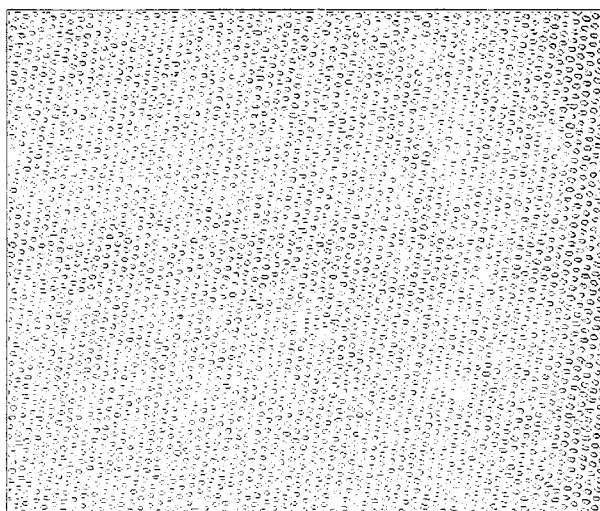




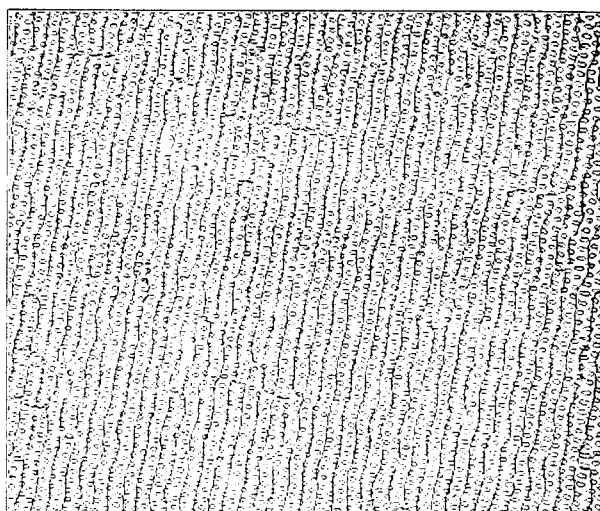
(a)



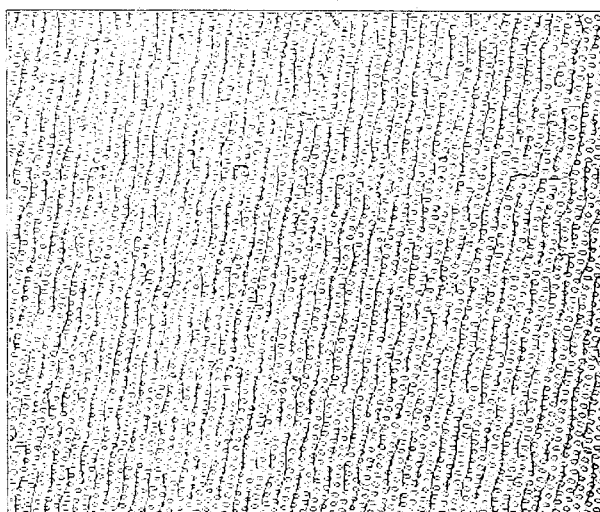
(c)



(b)



(d)



(e)

Fig. 9. (a) Handywipe image 576 by 454 pixels. (b) Edges from handywipe image at  $\sigma = 1.0$ . (c)  $\sigma = 5.0$ . (d) Superposition of the edges. (e) Edges combined using feature synthesis.



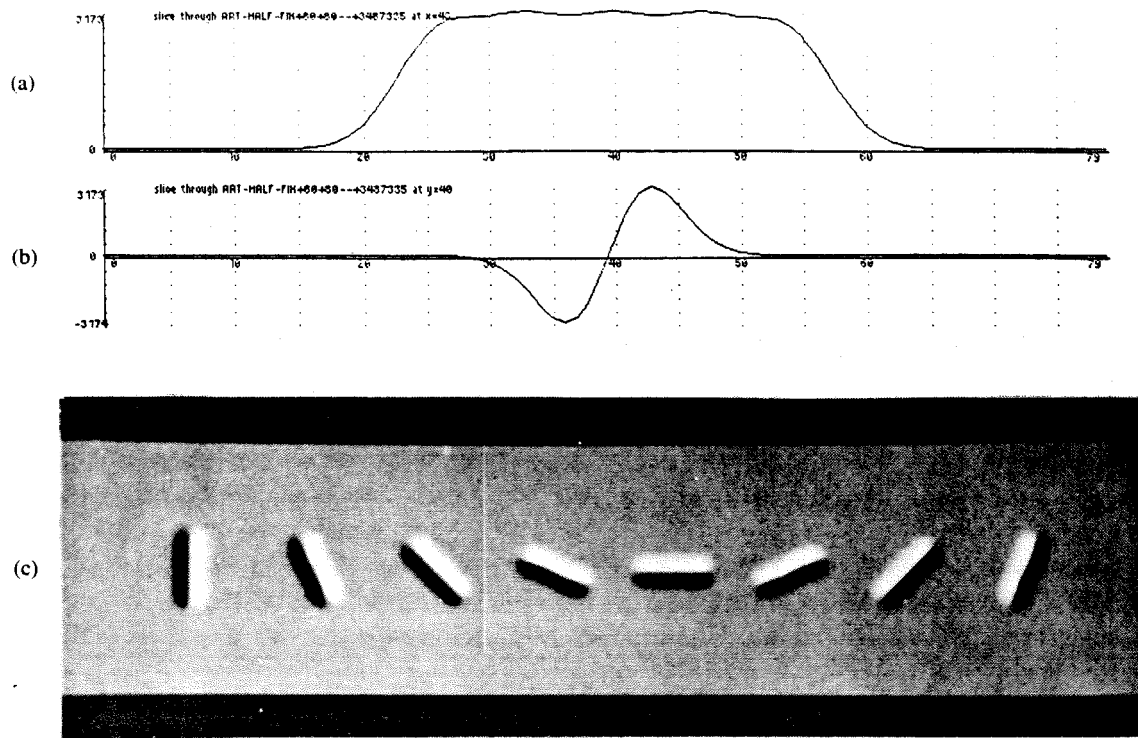


Fig. 10. Directional step edge mask. (a) Cross section parallel to the edge direction. (b) Cross section normal to edge direction. (c) Two-dimensional impulse responses of several masks.

integrals, and will be denoted by  $\Sigma$ . We have already seen what happens if we scale the function normal to the edge (21). We now do the same to the projection function by replacing  $f(x, y)$  by  $f_l(x, y) = f(x, (y/l))$ . The integrals become

$$\begin{aligned} & \int_{-\infty}^0 \int_{-\infty}^{+\infty} f\left(x, \frac{y}{l}\right) dx dy \\ &= \int_{-\infty}^0 \int_{-\infty}^{+\infty} f(x, y_1) l dx dy_1 \\ n_{00} & \left( \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f^2\left(x, \frac{y}{l}\right) dx dy \right)^{1/2} \\ &= n_{00} \left( \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f^2(x, y_1) l dx dy_1 \right)^{1/2} \quad (49) \end{aligned}$$

And the ratio of the two is now  $\sqrt{l\Sigma}$ . The localization  $\Lambda$  also improves as  $\sqrt{l}$ . It is clearly desirable that we use as large a projection function as possible. There are practical limitations on this however, in particular edges in an image are of limited extent, and few are perfectly linear. However, most edges continue for some distance, in fact much further than the 3 or 4 pixel supports of most edge operators. Even curved edges can be approximated by linear segments at a small enough scale. Considering the advantages, it is obviously preferable to use directional operators whenever they are applicable. The only proviso is that the detection scheme must ensure that they are used only when the image fits a linear edge model.

The present algorithm tests for applicability of each di-

rectional mask by forming a goodness-of-fit estimate. It does this at the same time as the mask itself is computed. An efficient way of forming long directional masks is to sample the output of nonelongated masks with the same direction. This output is sampled at regular intervals in a line parallel to the edge direction. If the samples are close together (less than  $2\sigma$  apart), the resulting mask is essentially flat over most of its range in the edge direction and falls smoothly off to zero at its ends. Two cross sections of such a mask are shown in Fig. 10. In this diagram (as in the present implementation) there are five samples over the operator support.

Simultaneously with the computation of the mask, it is possible to establish goodness of fit by a simple squared-error measure. The mask is computed by summing some number of circular mask outputs (say 5) in a line. If the mask lies over a step edge in its preferred direction, these 5 values will be roughly the same. If the edge is curved or not aligned with the mask direction, the values will vary. We use the variance of these values as an estimate of the goodness of fit of the actual edge to an ideal step model. We then suppress the output of a directional mask if its variance is greater than some fraction of the squared output. Where no directional operator has sufficient goodness of fit at a point, the algorithm will use the output of the nonelongated operator described in Section VII. This simple goodness-of-fit measure is sufficient to eliminate the problems that traditionally plague directional operators, such as false responses to highly curved edges and extension of edges beyond corners; see Hildreth [12].

This particular form of projection function, that is a

function with constant value over some range which decays to zero at each end with two roughly half-Gaussians, is very similar to a commonly used extension of the Hanning window. This latter function is flat for some distance and decays to zero at each end with two half-cosine bells [2]. We can therefore expect our function to have good properties as a moving average estimator, which as we saw in Section VII, is an important role fulfilled by the projection function.

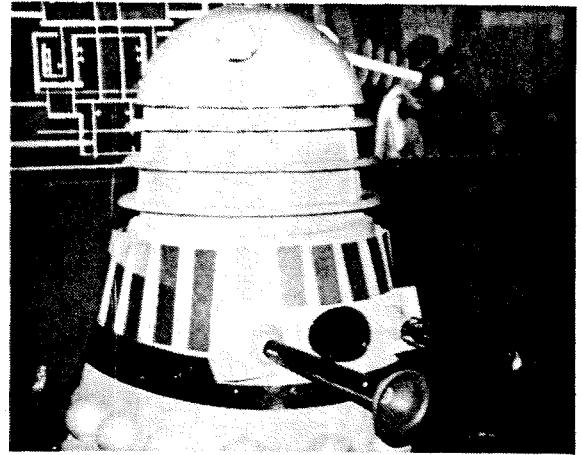
All that remains to be done in the design of directional operators is the specification of the number of directions, or equivalently the angle between two adjacent directions. To determine the latter, we need to determine the angular selectivity of a directional operator as a function of the angle  $\theta$  between the edge direction and the preferred direction of the operator. Assume that we form the operator by taking an odd number  $2N + 1$  of samples. Let the number of a sample be  $n$  where  $n$  is in the range  $-N \dots +N$ . Recall that the directional operator is formed by convolving with a symmetric Gaussian, differentiating normal to the preferred edge direction of the operator, and then sampling along the preferred direction. The differentiated surface will be a ridge which makes an angle  $\theta$  to the preferred edge direction. Its height will vary as  $\cos \theta$ , and the distance of the  $n$ th sample from the center of the ridge will be  $nd \sin \theta$  where  $d$  is the distance between samples. The normalized output will be

$$O_n(\theta) = \frac{\cos \theta}{2N + 1} \left[ \sum_{n=-N}^N \exp \left( -\frac{(nd \sin \theta)^2}{2\sigma^2} \right) \right]. \quad (50)$$

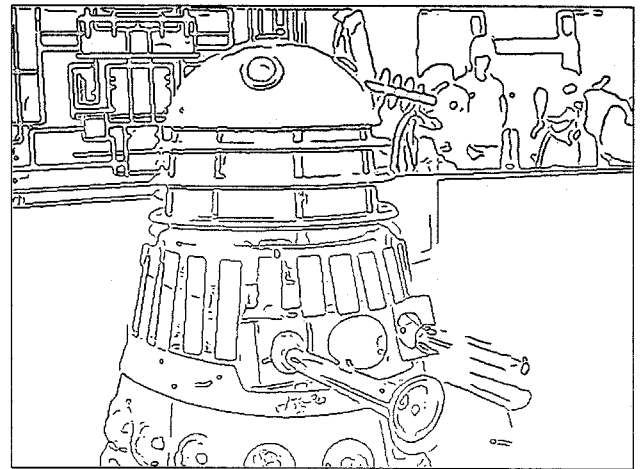
If there are  $m$  operator directions, then the angle between the preferred directions of two adjacent operators will be  $180/m$ . The worst case angle between the edge and the nearest preferred operator direction is therefore  $90/m$ . In the current implementation the value of  $d/\sigma$  is about 1.4 and there are 6 operator directions. The worst case for  $\theta$  is 15 degrees, and for this case the operator output will fall to about 85 percent of its maximum value. Directional operators very much like the ones we have derived were suggested by Marr [17], but were discarded in favor of the Laplacian of Gaussian [18]. In part this was because the computation of several directional operators at each point in the image was thought to require an excessive amount of computation. In fact the sampling scheme described above requires only five multiplications per operator. An example of edge detection using five-point directional operators is given in Fig. 11.

#### X. CONCLUSIONS

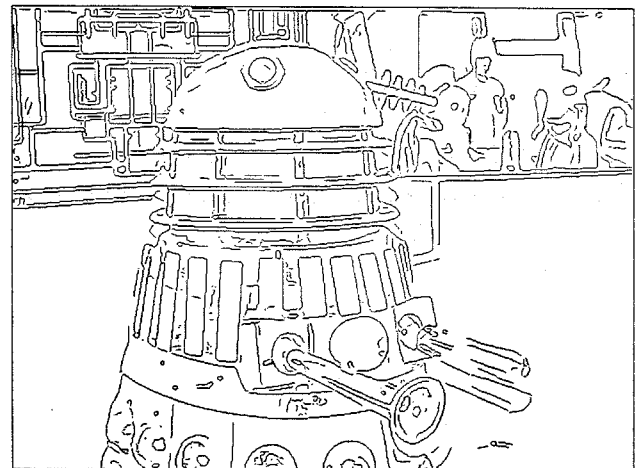
We have described a procedure for the design of edge detectors for arbitrary edge profiles. The design was based on the specification of detection and localization criteria in a mathematical form. It was necessary to augment the original two criteria with a multiple response measure in order to fully capture the intuition of good detection. A mathematical form for the criteria was presented, and nu-



(a)



(b)



(c)

Fig. 11. (a) Dalek image 576 by 454 pixels. (b) Edges found using circular operator. (c) Directional edges (6 mask orientations).

merical optimization was used to find optimal operators for roof and ridge edges. The analysis was then restricted to consideration of optimal operators for step edges. The result was a class of operators related by spatial scaling. There was a direct tradeoff in detection performance versus localization, and this was determined by the spatial

width. The impulse response of the optimal step edge operator was shown to approximate the first derivative of a Gaussian.

A detector was proposed which used adaptive thresholding with hysteresis to eliminate streaking of edge contours. The thresholds were set according to the amount of noise in the image, as determined by a noise estimation scheme. This detector made use of several operator widths to cope with varying image signal-to-noise ratios, and operator outputs were combined using a method called feature synthesis, where the responses of the smaller operators were used to predict the large operator responses. If the actual large operator outputs differ significantly from the predicted values, new edge points are marked. It is therefore possible to describe edges that occur at different scales, even if they are spatially coincident.

In two dimensions it was shown that marking edge points at maxima of gradient magnitude in the gradient direction is equivalent to finding zero-crossings of a certain nonlinear differential operator. It was shown that when edge contours are locally straight, highly directional operators will give better results than operators with a circular support. A method was proposed for the efficient generation of highly directional masks at several orientations, and their integration into a single description.

Among the possible extensions of the work, the most interesting unsolved problem is the integration of different edge detector outputs into a single description. A scheme which combined the edge and ridge detector outputs using feature synthesis was implemented, but the results were inconclusive. The problem is much more complicated here than for edge operators at different scales because there is no clear reason to prefer one edge type over another. Each edge set must be synthesized from the other, without a bias caused by overestimation in one direction.

The criteria we have presented can be used with slight modification for the design of other kinds of operator. For example, we may wish to design detectors for nonlinear two-dimensional features (such as corners). In this case the detection criterion would be a two-dimensional integral similar to (3), while a plausible localization criterion would need to take into account the variation of the edge position in both the  $x$  and  $y$  directions, and would not directly generalize from (9). There is a natural generalization to the detection of higher-dimensional edges, such as occur at material boundaries in tomographic scans. As was pointed out in Section VII, (47) can be used to find edges in images of arbitrary dimension, and the algorithm remains efficient in higher dimensions because  $n$ -dimensional Gaussian convolution can be broken down into  $n$  linear convolutions.

#### ACKNOWLEDGMENT

The author would like to thank Dr. J. M. Brady for his influence on the course of this work and for comments on early drafts of this paper. Thanks to the referees for their suggestions which have greatly improved the presentation

of the paper. In particular thanks to the referee who suggested the simple derivation based on the Schwarz inequality that appears on p. 682.

#### REFERENCES

- [1] R. J. Beattie, "Edge detection for semantically based early visual processing," Ph.D. dissertation, Univ. Edinburgh, 1984.
- [2] C. Bingham, M. D. Godfrey, and J. W. Tukey, "Modern techniques of power spectrum estimation," *IEEE Trans. Audio Electroacoust.*, vol. AU-15, no. 2, pp. 56-66, 1967.
- [3] R. A. Brooks, "Symbolic reasoning among 3-D models and 2-D images," Dep. Comput. Sci., Stanford Univ., Stanford, CA, Rep. AIM-343, 1981.
- [4] J. F. Canny, "Finding edges and lines in images," M.I.T. Artificial Intell. Lab., Cambridge, MA, Rep. AI-TR-720, 1983.
- [5] F. S. Cohen, D. B. Cooper, J. F. Silverman, and E. B. Hinkle, "Simple parallel hierarchical and relaxation algorithms for segmenting textured images based on noncasual Markovian random field models," in *Proc. 7th Int. Conf. Pattern Recognition and Image Processing*, Canada, 1984.
- [6] R. Courant and D. Hilbert, *Methods of Mathematical Physics*, vol. 1. New York: Wiley-Interscience, 1953.
- [7] J. R. Fram and E. S. Deutsch, "On the quantitative evaluation of edge detection schemes and their comparison with human performance," *IEEE Trans. Comput.*, vol. C-24, no. 6, pp. 616-628, 1975.
- [8] M. Gennert, "Detecting half-edges and vertices in images," in *IEEE Conf. Comput. Vision and Pattern Recognition*, Miami Beach, FL, June 24-26, 1986.
- [9] R. W. Hamming, *Digital Filters*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
- [10] R. M. Haralick, "Zero-crossings of second directional derivative edge operator," in *SPIE Proc. Robot Vision*, Arlington, VA, 1982.
- [11] A. Herskovits and T. O. Binford, "On boundary detection," M.I.T. Artificial Intell. Lab., Cambridge, MA, AI Memo 183, 1970.
- [12] E. C. Hildreth, "Implementation of a theory of edge detection," M.I.T. Artificial Intell. Lab., Cambridge, MA, Rep. AI-TR-579, 1980.
- [13] —, *The Measurement of Visual Motion*. Cambridge, MA: M.I.T. Press, 1983.
- [14] B. K. P. Horn, "The Binford-Horn line-finder," M.I.T. Artificial Intell. Lab., Cambridge, MA, AI Memo 285, 1971.
- [15] D. G. Luenberger, *Introduction to Linear and Non-Linear Programming*. Reading, MA: Addison-Wesley, 1973.
- [16] I. D. G. Macleod, "On finding structure in pictures," in *Picture Language Machines*, S. Kanef, Ed. New York: Academic, 1970, p. 231.
- [17] D. C. Marr, "Early processing of visual information," *Phil. Trans. Roy. Soc. London*, vol. B 275, pp. 483-524, 1976.
- [18] D. C. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London*, vol. B 207, pp. 187-217, 1980.
- [19] D. C. Marr and T. Poggio, "A theory of human stereo vision," *Proc. Roy. Soc. London*, vol. B 204, pp. 301-328, 1979.
- [20] J. E. W. Mayhew and J. P. Frisby, "Psychophysical and computational studies toward a theory of human stereopsis," *Artificial Intell. (Special Issue on Computer Vision)*, vol. 17, 1981.
- [21] T. Poggio, H. Voorhees, and A. Yuille, "A regularized solution to edge detection," M.I.T. Artificial Intell. Lab., Cambridge, MA, Rep. AIM-833, 1985.
- [22] A. P. Pentland, "Visual inference of shape: Computation from local features," Ph.D. dissertation, Dep. Psychol., Massachusetts Inst. Technol., Cambridge, MA, 1982.
- [23] J. M. S. Prewitt, "Object enhancement and extraction," in *Picture Processing and Psychopictorics*, B. Lipkin and A. Rosenfeld, Eds. New York: Academic, 1970, pp. 75-149.
- [24] S. O. Rice, "Mathematical analysis of random Noise," *Bell Syst. Tech. J.*, vol. 24, pp. 46-156, 1945.
- [25] A. Rosenfeld and M. Thurston, "Edge and curve detection for visual scene analysis," *IEEE Trans. Comput.*, vol. C-20, no. 5, pp. 562-569, 1971.
- [26] L. Spacek, "The computation of visual motion," Ph.D. dissertation, Univ. Essex at Colchester, 1984.
- [27] K. A. Stevens, "Surface perception from local analysis of texture and contour," M.I.T. Artificial Intell. Lab., Cambridge, MA, Rep. AI-TR-512, 1980.

- [28] V. Torre and T. Poggio, "On edge detection," M.I.T. Artificial Intell. Lab., Cambridge, MA, Rep. AIM-768, 1984.
- [29] S. Ullman, *The Interpretation of Visual Motion*. Cambridge, MA: M.I.T. Press, 1979.
- [30] N. Wiener, *Extrapolation, Interpolation and Smoothing of Stationary Time Series*. Cambridge, MA: M.I.T. Press, 1949.
- [31] A. P. Witkin, "Shape from contour," M.I.T. Artificial Intell. Lab., Cambridge, MA, Rep. AI-TR-589, 1980.



**John Canny** (S'81-M'82) was born in Adelaide, Australia, in 1958. He received the B.Sc. degree in computer science and the B.E. degree from Adelaide University in 1980 and 1981, respectively, and the S.M. degree from the Massachusetts Institute of Technology, Cambridge, in 1983.

He is with the Artificial Intelligence Laboratory, M.I.T. His research interests include low-level vision, model-based vision, motion planning for robots, and computer algebra.

Mr. Canny is a student member of the Association for Computing Machinery.