The Dissimilarity Corner Detector

J. Cooper
Dept. of Comp. Sci.
Uni. of Western Australia
Nedlands WA 6009
Australia.

S. Venkatesh
Dept. of Comp. Sci.
Curtin University
P.O. Box U1987, Perth WA 6001
Australia.

L. Kitchen
Dept. of Comp. Sci.
University of Melbourne
Parkville Vic 3052
Australia.

Abstract— **We present a corner detector that works by using dissimilarity along the contour direction to detect curves in the image contour. The operator is fast, robust to noise and almost self-thresholding. The standard deviation of the image noise must be specified, but this value is easily measured and the explicit modeling of image noise contributes to the robustness of the operator to noise. We also present a new interpretation of the Kitchen-Rosenfeld corner operator in which we show that this operator can also be viewed as the second derivative of the image function along the edge direction.**

I. INTRODUCTION

While primitive features such as edges and lines are vital visual clues, intersection of edges such as corners and junctions, commonly referred to as 2D features, provide rich information for examining frame-to-frame displacement characteristics of images. Thus, in applications involving disparity analysis such as motion detection and depth from stereo, we need to identify these 2D features. In general, edge detectors do not make good corner detectors because they give reduced output at corners, and because of this considerable research has been specifically directed towards isolating corners and junction points.

Kitchen and Rosenfeld [1] measure "cornerity" as the rate of change of gradient direction along an edge, multiplied by the gradient magnitude. Nagel [2] used a method based on minimizing the squared differences between a second order Taylor series expansion of grey level values from one frame to another. Noble [3] has shown how the Plessey corner detector estimates image curvature and has proposed an image representation that is based on the differential geometrical "topography" of the intensity surface. Spacek [4] relates cornerness to the difference in response between a directionally selective edge detector and a rotationally symmetric one.

In this paper we propose a two-stage approach to corner detection, in which first the contour direction is measured, and then image differences along the contour direction are computed. A knowledge of the noise characteristics is used to determine whether the image differences along the contour direction are sufficient to indicate a corner. The technique is less computationally expensive than existing corner detection techniques and robust to noise.

The corner detection algorithm is described in Section II. A comparative study of the results obtained with our corner detector and other corner detectors, specifically the Plessey [3], Kitchen-Rosenfeld (KR) [1] and Beaudet's DET operator [5] are carried out in Section III. We draw conclusions in Section IV.

Manuscript received April 7, 1991. This research was supported by the Department of Industry Technology and Commerce under the Artificial Intelligence Research Support Program, and by a CTEC Special Research Grant.

II. THE DISSIMILARITY CORNER DETECTOR

If two image patches along a straight edge segment are compared with each other, they will be very similar (Fig. 1a). If the edge segment is curved, as at a corner, the two image patches will be dissimilar (Fig. 1b). Thus corners can be detected where the similarity between image patches along the edge direction is low. This suggests a corner detection algorithm like that shown in Fig. 2.

A. *Image Self-Similarity and Early-Jump-Out*

The method makes extensive use of a similarity test that therefore must be fast. In this section, we outline a fast sequential analysis [6] method proposed by Barnea and Silverman [7] for determining the similarity between two $M \times M$ pixel image windows. We then present our interpretation and enhancements of the method in its application to corner detection.

Barnea and Silverman [7] suggested that at a strategically chosen sequence of locations, the absolute value of the difference between the greylevels of corresponding pixels in the two windows be added to an accumulating sum.

7803-0078/91/0600-1377$01.00 ©1991 IEEE

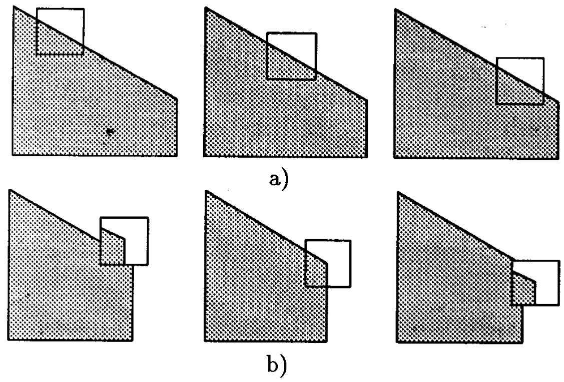


Fig. 1. If we take an image patch belonging to a straight edge and compare it with others along the edge direction we find that it is similar to them (a). If we do a similar comparison along the local edge direction at a corner we find that the patch is not similar to its neighbors (b).

```
for each pixel (x, y) in the image
    Obtain the components of the image gradient.
    If the gradient magnitude is low then
      The point (x, y) is not a corner
    else
        Calculate positions (x_l, y_l) and (x_r, y_r) to left
        and right along the local contour direction.
        Use a similarity test to decide whether the
        image patches centered at (x_l, y_l) or at
        (x_r, y_r) differ from that at (x, y).
        if so then
          (x, y) is a corner
        else
          (x, y) is not a corner.
        endif
    endif
end
```

Fig. 2. A simple corner detection algorithm.

At each step of the similarity test, we compare the accumulating value with the appropriate member of a threshold sequence. At any step, if the accumulating value is larger than the threshold, the process is terminated with the conclusion that the patches are dissimilar.

A speed-up factor of 50 is claimed for similarity tests when $M = 32$ [7] for the above technique in comparison with traditional correlation because the similarity test is in general terminated long before all M^2 operations are performed. It is note-worthy that Wald [6] only claims a speed-up factor of about 2 over non-sequential decision making procedures for the Sequential Probability Ratio Test. This means that the significant improvement over thresholded correlation afforded by the image similarity test is mainly due to its effect of choosing an efficient sample size for the statistical similarity test. The correlation test could use an efficient sample size through a suitable choice of M.

We interpret the comparison with the threshold sequence member as a test of the null hypothesis H_0:

> "The two windows are similar so that the differences in pixel values can be attributed entirely to noise."

We term the method the Early Jump-Out (EJO) technique and we have extended it to use multiple threshold sequences. We will call the threshold sequence mentioned above the Early Jump-Out High (EJH) sequence.

1) Early Jump-out High: Barnea and Silverman use "a computer solution of an analytic system of recursive equations" to generate a threshold sequence such that if H_0 was true then at each step, the probability that the accumulated value will be greater than the current threshold is some small value q. In this section, we show a simple way in which the threshold sequence can be derived. We first describe the method to derive the first member of the threshold sequence, and then show how subsequent members can be computed.

We model the image data at each pixel as an image signal with added Gaussian noise distributed as $N(0,\sigma)$. If the difference between the value of two pixels is due entirely to noise, then the difference is a random variable X distributed as $N(0,\sigma\sqrt{2})$ and whose probability distribution function (pdf) is given by

$$f_X(x) = \frac{1}{2\sigma\sqrt{2}} e^{-(1/4)(x/\sigma)^2}. \quad (1)$$

The absolute value of X is a random variable D_1 with pdf given by

$$f_{D_1}(d) = \begin{cases} \frac{1}{\sigma\sqrt{2}} e^{-(1/4)(x/\sigma)^2} & \text{if } d > 0 \\ \frac{1}{2\sigma\sqrt{2}} & \text{if } d = 0 \\ 0 & \text{else} \end{cases}. \quad (2)$$

We numerically integrate (2) to find the first member T_1 of the threshold sequence such that the probability that the first difference will be greater than this threshold is q. That is

$$\int_{T_1}^{\infty} f_{D_1}(x)dx = q. \quad (3)$$

Let R_N be the value of the accumulating result after N operations in the EJO algorithm. At step N, only if $R_N < T_N$ is the absolute value of the difference between a new pair of pixels added to the accumulating result. The pdf of the survivors of each thresholding operation is therefore of interest. The value of R_N subject to $R_N < T_N$ is a random variable S_N whose pdf is given by

$$f_{S_N}(s) = \begin{cases} \frac{1}{(1-q)} f_{D_N}(s) & \text{if } s < T_N \\ 0 & \text{else} \end{cases} . \tag{4}$$

Since R_{N+1} is obtained by adding R_N to the absolute value of the difference between a new pair of pixels, we have $D_{N+1} = S_N + D_1$. Therefore the pdf of D_{N+1} can be computed using

$$f_{D_{N+1}}(d) = f_{D_1}(d) * f_{S_N}(s), \tag{5}$$

where * is the convolution operator. For each N we can generate f_{D_N} and then use numerical integration to find a threshold T_N such that

$$\int_{T_N}^{\infty} f_{D_N}(x)dx = q. \tag{6}$$

In the preceding discussion we have assumed that each pixel difference has the same pdf as the first. If this assumption was not valid, we could have obtained the pdf of each difference independently and used the new pdf in the convolution step. Alternatively, the EJH sequence could have been computed from distributions obtained directly from image data.

2) Early Jump-out Low (EJL): The DISSimilarity corner detector (DISS) compares image patches along an edge. If the edge is straight, a high degree of similarity can be expected, the accumulating value will never exceed the EJH sequence, and the cost of EJH similarity test is then $O(M^2)$. We use another threshold sequence - the Early Jump-out Low (EJL) sequence [8], to allow the process of accumulating differences to be terminated early if the image patches are similar.

At each step of the Early Jump-out technique, the accumulated value is compared with the appropriate member of a threshold sequence in order to test a new null hypothesis H_0:

"The two windows are dissimilar to each other."

This time the sequence is pre-computed so that if H_0 is true then at each step the probability that the accumulating difference will be *less* than the current threshold is some small value p. As before, we first derive the first member of the EJL sequence, and then show how subsequent members of the sequence can be derived.

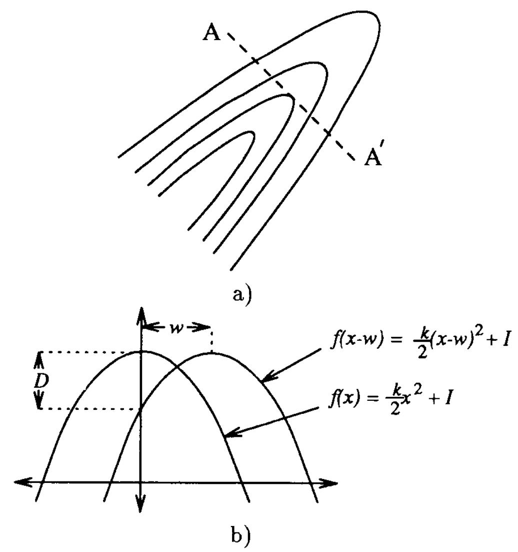


Fig. 3. If we take a cross section along the line AA′ through the small image region whose contour map is shown in a), we obtain a parabola $f(x)$ as in b). We can displace this parabola a distance w and calculate $D(x,w) = f(x) - f(x-w)$, which is a function of the second derivative k.

Consider contours of equal greylevel at a noise-free corner as shown in Fig. 3a. If line segment AA′ is parallel to the edge direction, then a profile taken along AA′ can be approximated by a second order Taylor series expansion as a parabola

$$f(x) = f(0) + f'(0)x + \frac{f''(0)x^2}{2}, \tag{7}$$

where f'' is the second derivative along the edge direction, f' is the first derivative, and $f(0)$ is the image intensity. As AA′ is parallel to the edge direction, f' is zero. If noise is taken into account,

$$f(x) = f(0) + \frac{f''(0)x^2}{2} + e(x), \tag{8}$$

where $e(x)$ is the error due to image noise (with standard deviation σ) at x.

If the parabola is displaced by a small distance w as shown in Fig. 3, then we can compute the difference function D as

$$D(x,w) = |f(x) - f(x-w)|$$

$$= \left| kxw - \frac{kw^2}{2} + e(x) - e(x+w) \right| \tag{9}$$

where $k = f''(0)$.

If k is large and $x \neq w/2$ then D has a normal distribution with mean $\left|kxw - \frac{kw^2}{2}\right|$ and standard deviation equal to $\sigma\sqrt{2}$. Further, the threshold T_1 is the first value of EJL sequence, and if the first difference D_1 is less than T_1, we jump out of the similarity test with the conclusion that k is small and the pixel is not a corner pixel. For particular values of x and w, we can choose values k_{min} and T_1 such that the probability of D_1 being less than T_1 is some small value p. Then k_{min} becomes the smallest curvature that can be reliably detected for the particular values of x, w and T_1.

For $x = 0$, and $w = 2$, D_1 distributed as $N(2k, \sigma\sqrt{2})$. The pdf of D_1 is given by

$$f_{D_1}(d) = \frac{1}{2\sigma\sqrt{\pi}} e^{(1/4)[(d-2k)/\sigma]^2}. \quad (10)$$

We numerically integrate (10) to find the first member T_1 of the threshold sequence such that

$$\int_{\infty}^{T_1} f_{D_1}(x)dx = p. \quad (11)$$

As with the Early Jump-out High threshold sequence, let R_N be the value of the accumulating result after N operations. At step N, only if $R_N > T_N$ is the absolute value of the difference between a new pair of pixels added to the accumulating result. The value of R_N subject to $R_N > T_N$ is a random variable S_N whose pdf is given by

$$f_{S_N}(s) = \begin{cases} \frac{1}{(1-p)} f_{D_N}(s) & \text{if } s > T_N \\ 0 & \text{else} \end{cases}. \quad (12)$$

As before, for $N > 1$, the pdf of D_{N+1} can be computed recursively from the pdf of S_N and D_1 using

$$f_{D_{N+1}}(d) = f_{D_1}(d) * f_{S_N}(s). \quad (13)$$

For each N we can generate $f_{D_{N+1}}(d)$ using the above distribution and then use numerical integration to find T_N such that

$$\int_{0}^{T_N} f_{D_N} N(x)dx) = p. \quad (14)$$

By specifying x, w and k_{min}, we thus determine the entire Early Jump-out Low threshold sequence T_N.

The value of k could be measured via a surface fitting approach. Let g be the second order Taylor series expansion of the local image function with $g' = (g_x, g_y)^T$ the image surface gradient and $g'' = \begin{pmatrix} g_{xx} & g_{xy} \\ g_{yx} & g_{yy} \end{pmatrix}$ the second derivative at $(0,0)^T$. Now k is defined as the second derivative along the edge direction. We can obtain the component of g'' along the edge direction by pre- and post- multiplying g'' by a unit vector that is orthogonal to g'. Thus we have

$$k = \frac{1}{\sqrt{g_x^2 + g_y^2}} \begin{pmatrix} -g_y \\ g_x \end{pmatrix}^T \begin{pmatrix} g_{xx} & g_{xy} \\ g_{yx} & g_{yy} \end{pmatrix} \frac{1}{\sqrt{g_x^2 + g_y^2}} \begin{pmatrix} -g_y \\ g_x \end{pmatrix}$$

$$= \frac{g_{xx}g_y^2 + g_{yy}g_x^2 + 2g_{xy}g_xg_y}{g_x^2 + g_y^2}. \quad (15)$$

It is interesting to note that the value of k is actually the quantity measured by the KR corner operator [1]. While Kitchen and Rosenfeld looked upon the k as the curvature of the contour multiplied by the edge magnitude, we have shown that this value is fact the curvature of the image surface along the edge direction. This straightforward interpretation allows a simple derivation and error analysis.

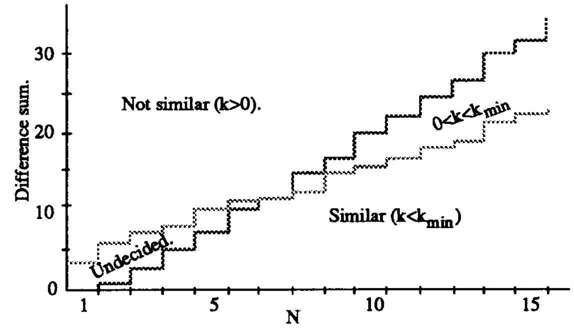


Fig. 4. Division of Early Jump-out space by EJH (light grey) and EJL (dark grey) threshold sequences calculated with $\sigma = 1.15$ and $T_1 = 0$. An accumulating difference sum starts in the undecided region between the two threshold sequences.

3) The Meaning of the Early Jump-out Sequences: Fig. 4 shows typical EJH and EJL threshold sequences, which divide "Early Jump-out space". A difference sum starts in the undecided region. If the difference sum outgrows the EJH sequence, we conclude that the two samples are not similar ($k > 0$). If the difference sum gets below the EJL sequence, we conclude that either the samples are similar or the value of k for the pixel in question is too low for a corner to be detected ($k < k_{min}$). If the difference sequence gets into the fish-tail region shown in Fig. 4, it indicates that both of the above conclusions could be accepted, which means that the image surfaces are not similar, but the strength of the feature is not sufficient ($0 < k < k_{min}$) - there is a corner there, but the value of k is so low that EJL would reject it as a corner.

If we let $T_1 = 0$ then we can detect the reliably detectable corner with the smallest value of k. In practice we use a higher value for T_1 than zero. This raises both the starting point and the slope of the EJL threshold sequence curve and reduces the time spent in the undecided region of Early Jump-out space by difference sums associated with weak corners. The fish-tail region in Fig. 4 then becomes important as difference sums can now get into this region when there is a detectable corner.

4) Compensating for errors: In our implementation, DISS uses the output of the Sobel operator to calculate the positions to the right and left along the contour of the current pixel. However, even on a straight edge there are two sources of error that cause the computed equi-contour pixel not to coincide with the position of the true contour.

The first error is due to spatial digitization. Consider an edge pixel at (x, y) with a gradient direction of θ on a straight edge. The predicted positions of the contour to left and right along the edge direction are generally displaced by a sub-pixel distance from the centers of the nearest pixels. We call this error the spatial digitization error. The second error arises from the error in the edge orientation as obtained from the Sobel operator [9] and from the fact that the Sobel operator has a region-of-support of radius 1 and we use its output to predict the contour position 2 pixels from its center.

On a straight edge, DISS rejects a straight edge as being a corner by finding that the predicted position of the current contour to left and right along the edge is in fact the true position of the contour. However, the errors mentioned above cause misalignment in the gradient direction and this results in apparent dissimilarity. We call the sum of both the errors outlined above the ***contour position error***. Fig. 5 shows that the apparent image difference error is proportional to the contour position error and the image gradient magnitude. The product of the contour position error and the image gradient magnitude is used to modify the EJH threshold sequence for the purpose of corner detection. At each step N in the similarity test, we jump out if $S(N) > T_N + N \times E$ where E is the error due to contour position error and $S(N)$ is the sum after N image differences.

III. EXPERIMENTAL RESULTS

The results obtained by our corner detector are compared with other methods, both in terms of the quality of output, and the speed of performance. The values $x = 0$, and $w = 2$ from (9) are used and if the accumulating difference sum uses more than 5 terms on any test, the test is terminated with the conclusion that the sum is growing too slowly. This effectively means that the largest region of support for any corner is 9×9.

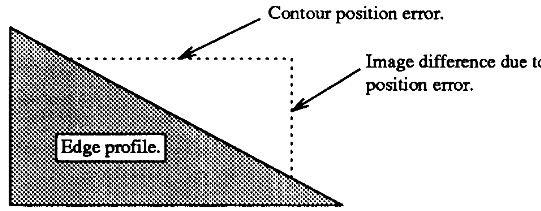


Fig. 5. The apparent image difference error due to contour position error.

We demonstrate that the Early Jump-out corner detector is computationally less expensive than the KR detector and Plessey corner detectors.

A. *The Corner Detectors*

Two synthetic corner images are created for the purpose of comparison. First a synthetic corner image composed of L, T, Y and + junctions is created. The contrast of the junctions is reduced progressively from top to bottom in the image, and this gives rise to a decreasing signal strength from top to bottom in the synthetic image. The image is then smoothed by convolution with a Gaussian ($\sigma = 0.6$), to simulate the blurring introduced by the imaging system. Finally, Gaussian noise of appropriate standard deviation is added, ($\sigma = 0.6$ for one image and $\sigma = 15$ for the other). The result is two synthetic images with corners of decreasing signal-to-noise ratio from top to bottom and is shown in Fig. 6.

To compare the four detectors in a non-qualitative manner we identify two errors as follows: We associate a 5×5 ***detection area*** centered at a true corner. Should a corner be detected in this area by a corner detector, the corner is correctly detected. Any corner detected outside this region is termed a *false positive* (FP), and failure to detect a corner in the detection area is termed *false negative* (FN). Thresholds are chosen for the competing corner detectors so that they give numbers of false negatives similar to those of our contender.

Table I shows the false negatives and false positives obtained when the four different corner detectors are applied to the synthetic images ($\sigma = 6$) and ($\sigma = 15$). At medium and high noise the KR, DET and Plessey operators all gave higher numbers of false positives than DISS.

TABLE I
ERROR PERFORMANCE OF CORNER OPERATORS

image noise	DISS FN	DISS FP	Plessey FN	Plessey FP	KR FN	KR FP	DET FN	DET FP
$\sigma = 6$	17	18	18	44	17	33	18	140
$\sigma = 15$	43	15	45	56	43	52	41	42
op's	$10 + 10$		$6n^2 + m^2 + 7$		$10n^2 + 11$		$6n^2 + 3$	
timing	1.48 sec		24.2 sec		2.17 sec		0.83 sec	

B. Number of Operations

One of our contributions to the technique of image patch similarity testing is the addition of the EJL sequence. It is the use of both the EJH and EJL sequences that leads to the speed of the corner detector. Table I compares the operators in terms of the number of operations performed. In the computation of the number of operations, the cost of one derivative operation is taken to be $2n^2$, where n is the size of the neighborhood of the difference operation used. For the dissimilarity-based operator, 10 operations are performed to compute the position and error, and 10 to perform the similarity test. The Plessey operator computes the $n \times n$ derivatives which it smoothes with an $m \times m$ Gaussian. Thus the cost of this operator is $6n^2 + m^2 + 7$. The costs for the DET and the KR operators are as computed in [1]. It should be noted that these operation counts are implementation specific. For instance, the convolution we used in the Plessey operator is separable and its convolution could therefore be reduced to $2m$.

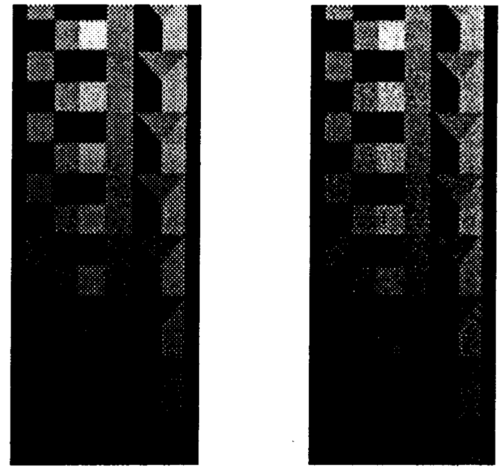
Table I also gives the timing figures comparing the four operators when applied to the high noise level test image. The speed of DISS is significantly better than that of the KR operator and nearly 16 times that of the Plessey operator. DET is indeed faster, but as shown in Table I is susceptible to noise.

IV. CONCLUSION

We present a fast new corner detector that works by using similarity tests along the contour direction to detect curves in the image contour. The speed of the operator can be mainly attributed to the fast Early Jump-out similarity tests. The use of the EJL sequence allows these tests to be terminated early when the test pixels are destined for rejection as corners. Explicit modeling of image noise contributes to the robustness of the operator to noise.

The corner detector presented here is almost self-thresholding. The standard deviation of the image noise must be specified, but this value can be easily measured.

We also present a new interpretation of the Kitchen-Rosenfeld corner operator in which we show that this operator can also be viewed as the second derivative of the image function along the edge direction.



a) b)

Fig. 6. Synthetic images with added gaussian noise ($\sigma = 15$) a), and ($\sigma = 15$) b).

REFERENCES

[1] L. Kitchen and A. Rosenfeld, "Gray Level Corner Detection," *Pattern Recognition Letters*, vol. 1, pp. 95–102, Dec. 1982.

[2] H. H. Nagel, "Overview on Image Sequence Analysis," in *Image Sequence Processing and Dynamic Scene Analysis* (T. S. Huang, ed.), pp. 2–39, New York: Springer-Verlag, 1983.

[3] A. J. Nobel, "Finding Corners," *Image and Vision Computing*, vol. 6, no. 2, pp. 121–128, May 1988.

[4] L. Spacek, *The Detection of Contours and their Visual Motion*. PhD thesis, University of Essex, 1985.

[5] P. R. Beaudet, "Rotationally Invarient Image Operators," in *International Joint Conference on Pattern Recognition*, pp. 579–583, April 1978.

[6] A. Wald, *Sequential Analysis*. London: Chapman & Hall, 1947.

[7] D. I. Barnea and H. F. Silverman, "A Class of Algorithms for Fast Digital Image Registration," *IEEE Transactions on Computers*, vol. C-21, no. 2, pp. 179–186, 1972.

[8] J. Cooper, S. Venkatesh, and L. Kitchen, "Early Jump-Out Corner Detectors," Tech. Rep. 90/14, University of Western Australia, jc@cs.mu.oz.au, 1990.

[9] L. J. Kitchen and J. A. Malin, "The Effect of Spatial Discretization on the Magnitude and Direction Response of Simple Differential Edge Operators on a Step Edge," *Computer Vision, Graphics and Image Processing*, vol. 47, pp. 243–258, 1989.