

A COMBINED CORNER AND EDGE DETECTOR

Chris Harris & Mike Stephens

Plessey Research Roke Manor, United Kingdom
© The Plessey Company plc. 1988

Consistency of image edge filtering is of prime importance for 3D interpretation of image sequences using feature tracking algorithms. To cater for image regions containing texture and isolated features, a combined corner and edge detector based on the local auto-correlation function is utilised, and it is shown to perform with good consistency on natural imagery.

INTRODUCTION

The problem we are addressing in Alvey Project MMI149 is that of using computer vision to understand the unconstrained 3D world, in which the viewed scenes will in general contain too wide a diversity of objects for top-down recognition techniques to work. For example, we desire to obtain an understanding of natural scenes, containing roads, buildings, trees, bushes, etc., as typified by the two frames from a sequence illustrated in Figure 1. The solution to this problem that we are pursuing is to use a computer vision system based upon motion analysis of a monocular image sequence from a mobile camera. By extraction and tracking of image features, representations of the 3D analogues of these features can be constructed.

To enable explicit tracking of image features to be performed, the image features must be discrete, and not form a continuum like texture, or edge pixels (edgels). For this reason, our earlier work¹ has concentrated on the extraction and tracking of feature-points or corners, since

they are discrete, reliable and meaningful². However, the lack of connectivity of feature-points is a major limitation in our obtaining higher level descriptions, such as surfaces and objects. We need the richer information that is available from edges³.

THE EDGE TRACKING PROBLEM

Matching between edge images on a pixel-by-pixel basis works for stereo, because of the known epi-polar camera geometry. However for the motion problem, where the camera motion is unknown, the aperture problem prevents us from undertaking explicit edgel matching. This could be overcome by solving for the motion beforehand, but we are still faced with the task of tracking each individual edge pixel and estimating its 3D location from, for example, Kalman Filtering. This approach is unattractive in comparison with assembling the edgels into edge segments, and tracking these segments as the features.

Now, the unconstrained imagery we shall be considering will contain both curved edges and texture of various scales. Representing edges as a set of straight line fragments⁴, and using these as our discrete features will be inappropriate, since curved lines and texture edges can be expected to fragment differently on each image of the sequence, and so be untrackable. Because of ill-conditioning, the use of parametrised curves (eg. circular arcs) cannot be expected to provide the solution, especially with real imagery.



a



b

Figure 1. Pair of images from an outdoor sequence.

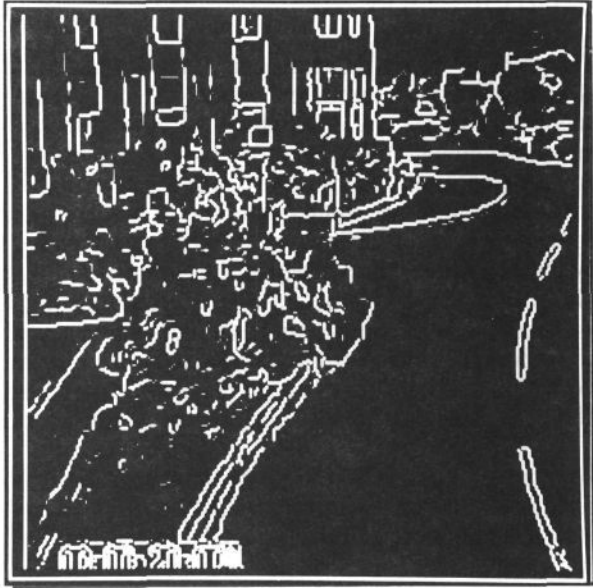
Having found fault with the above solutions to the problem of 3D edge interpretation, we question the necessity of trying to solve the problem at all! Psycho-visual experiments (the ambiguity of interpretation in viewing a rotating bent coat-hanger in silhouette), show that the problem of 3D interpretation of curved edges may indeed be effectively insoluble. This problem seldom occurs in reality because of the existence of small imperfections and markings on the edge which act as trackable feature-points.

Although an accurate, explicit 3D representation of a curving edge may be unobtainable, the connectivity it provides may be sufficient for many purposes - indeed the edge connectivity may be of more importance than explicit 3D measurements. Tracked edge connectivity, supplemented by 3D locations of corners and junctions, can provide both a wire-frame structural representation, and delimited image regions which can act as putative 3D surfaces.

This leaves us with the problem of performing reliable (ie. consistent) edge filtering. The state-of-the-art edge filters, such as⁵, are not designed to cope with junctions and corners, and are reluctant to provide any edge connectivity. This is illustrated in Figure 2 for the Canny edge operator, where the above- and below-threshold edgels are represented respectively in black and grey. Note that in the bushes, some, but not all, of the edges are readily matchable by eye. After hysteresis has been undertaken, followed by the deletion of spurs and short edges, the application of a junction completion algorithm results in the edges and junctions shown in Figure 3, edges being shown in grey, and junctions in black. In the bushes, very few of the edges are now readily matched. The problem here is that of edges with responses close to the detection threshold: a small change in edge strength or in the pixellation causes a large change in the edge topology. The use of edges to describe the bush is suspect, and it is perhaps better to describe it in terms of feature-points alone.

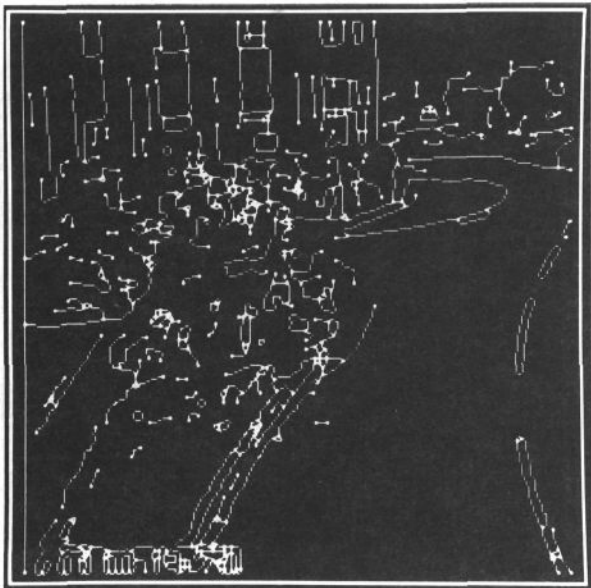


a

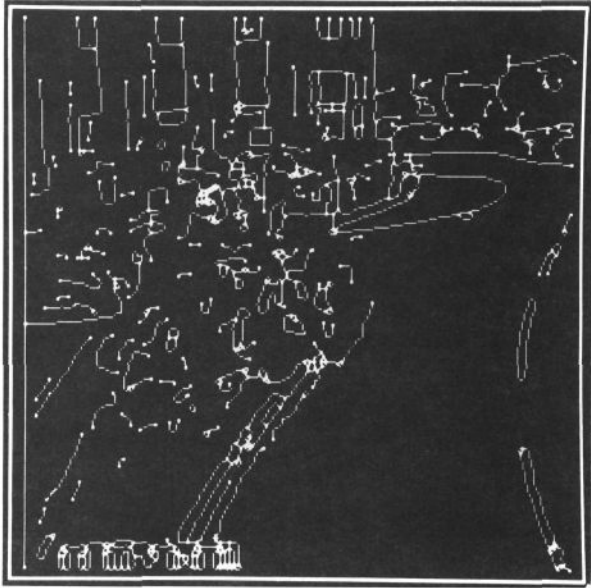


b

Figure 2. Unlinked Canny edges for the outdoor images



a



b

Figure 3. Linked Canny edges for the outdoor images

The solution to this problem is to attempt to detect both edges and corners in the image: junctions would then consist of edges meeting at corners. To pursue this approach, we shall start from Moravec's corner detector⁶.

MORAVEC REVISITED

Moravec's corner detector functions by considering a local window in the image, and determining the average changes of image intensity that result from shifting the window by a small amount in various directions. Three cases need to be considered:

- A. If the windowed image patch is flat (ie. approximately constant in intensity), then all shifts will result in only a small change;
- B. If the window straddles an edge, then a shift along the edge will result in a small change, but a shift perpendicular to the edge will result in a large change;
- C. If the windowed patch is a corner or isolated point, then all shifts will result in a large change. A corner can thus be detected by finding when the minimum change produced by any of the shifts is large.

We now give a mathematical specification of the above. Denoting the image intensities by I , the change E produced by a shift (x,y) is given by:

$$E_{x,y} = \sum_{u,v} w_{u,v} |I_{x+u,y+v} - I_{u,v}|^2$$

where w specifies the image window: it is unity within a specified rectangular region, and zero elsewhere. The shifts, (x,y) , that are considered comprise $\{(1,0), (1,1), (0,1), (-1,1)\}$. Thus Moravec's corner detector is simply this: look for local maxima in $\min(E)$ above some threshold value.

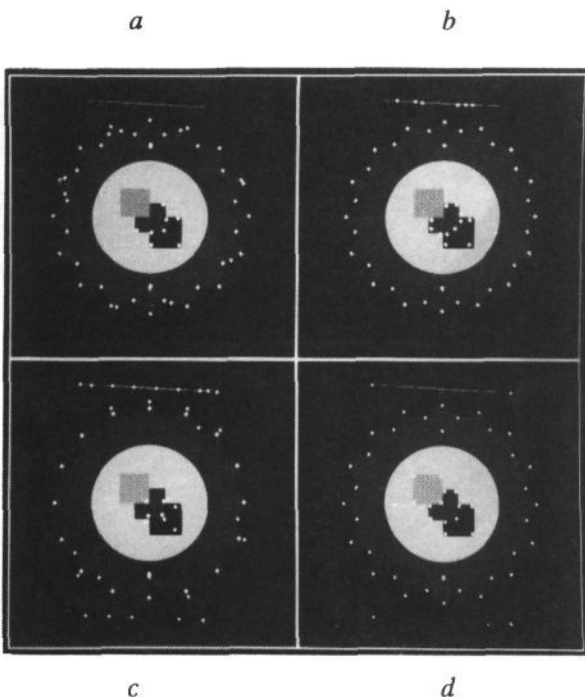


Figure 4. Corner detection on a test image

AUTO-CORRELATION DETECTOR

The performance of Moravec's corner detector on a test image is shown in Figure 4a; for comparison are shown the results of the Beaudet⁷ and Kitchen & Rosenfeld⁸ operators (Figures 4b and 4c respectively). The Moravec operator suffers from a number of problems; these are listed below, together with appropriate corrective measures:

1. The response is anisotropic because only a discrete set of shifts at every 45 degrees is considered - all possible small shifts can be covered by performing an analytic expansion about the shift origin:

$$\begin{aligned} E_{x,y} &= \sum_{u,v} w_{u,v} [I_{x+u,y+v} - I_{u,v}]^2 \\ &= \sum_{u,v} w_{u,v} [xX + yY + O(x^2, y^2)]^2 \end{aligned}$$

where the first gradients are approximated by

$$X = I \otimes (-1, 0, 1) \approx \partial I / \partial x$$

$$Y = I \otimes (-1, 0, 1)^T \approx \partial I / \partial y$$

Hence, for small shifts, E can be written

$$E(x,y) = Ax^2 + 2Cxy + By^2$$

where

$$A = X^2 \otimes w$$

$$B = Y^2 \otimes w$$

$$C = (XY) \otimes w$$

2. The response is noisy because the window is binary and rectangular - use a smooth circular window, for example a Gaussian:

$$w_{u,v} = \exp -(u^2 + v^2) / 2\sigma^2$$

3. The operator responds too readily to edges because only the minimum of E is taken into account - reformulate the corner measure to make use of the variation of E with the direction of shift.

The change, E , for the small shift (x,y) can be concisely written as

$$E(x,y) = (x,y) M (x,y)^T$$

where the 2×2 symmetric matrix M is

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}$$

Note that E is closely related to the local autocorrelation function, with M describing its shape at the origin (explicitly, the quadratic terms in the Taylor expansion). Let α, β be the eigenvalues of M . α and β will be proportional to the principal curvatures of the local auto-

correlation function, and form a rotationally invariant description of M . As before, there are three cases to be considered:

- A. If both curvatures are small, so that the local auto-correlation function is flat, then the windowed image region is of approximately constant intensity (ie. arbitrary shifts of the image patch cause little change in E);
- B. If one curvature is high and the other low, so that the local auto-correlation function is ridge shaped, then only shifts along the ridge (ie. along the edge) cause little change in E : this indicates an edge;
- C. If both curvatures are high, so that the local auto-correlation function is sharply peaked, then shifts in any direction will increase E : this indicates a corner.

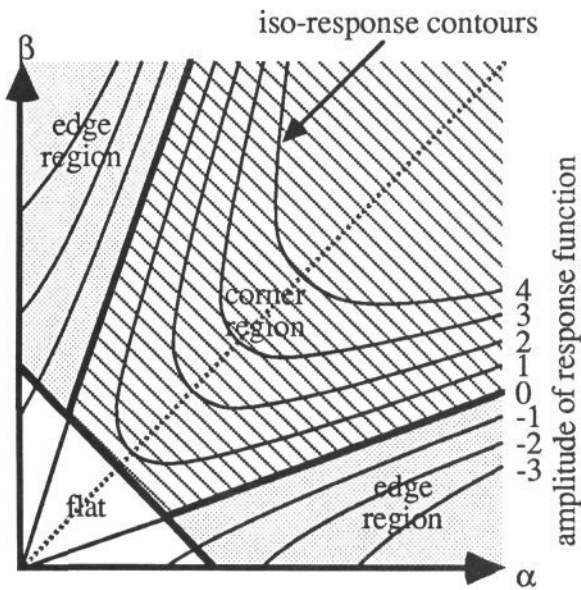


Figure 5. Auto-correlation principal curvature space- heavy lines give corner/edge/flat classification, fine lines are equi-response contours.

Consider the graph of (α, β) space. An ideal edge will have α large and β zero (this will be a surface of translation), but in reality β will merely be small in comparison to α , due to noise, pixellation and intensity quantisation. A corner will be indicated by both α and β being large, and a flat image region by both α and β being small. Since an increase of image contrast by a factor of p will increase α and β proportionately by p^2 , then if (α, β) is deemed to belong in an edge region, then so should $(\alpha p^2, \beta p^2)$, for positive values of p . Similar considerations apply to corners. Thus (α, β) space needs to be divided as shown by the heavy lines in Figure 5.

CORNER/EDGE RESPONSE FUNCTION

Not only do we need corner and edge classification regions, but also a measure of corner and edge quality or response. The size of the response will be used to select isolated corner pixels and to thin the edge pixels.

Let us first consider the measure of corner response, R , which we require to be a function of α and β alone, on grounds of rotational invariance. It is attractive to use $\text{Tr}(M)$ and $\text{Det}(M)$ in the formulation, as this avoids the explicit eigenvalue decomposition of M , thus

$$\text{Tr}(M) = \alpha + \beta = A + B$$

$$\text{Det}(M) = \alpha\beta = AB - C^2$$

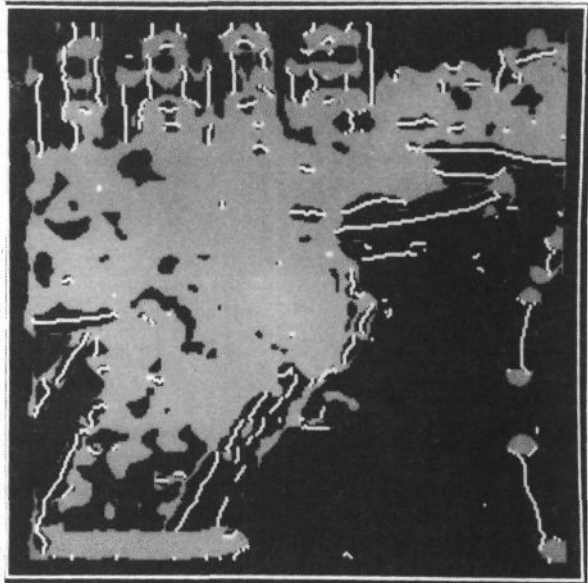
Consider the following inspired formulation for the corner response

$$R = \text{Det} - k \text{Tr}^2$$

Contours of constant R are shown by the fine lines in Figure 5. R is positive in the *corner region*, negative in the *edge regions*, and small in the *flat region*. Note that increasing the contrast (ie. moving radially away from the

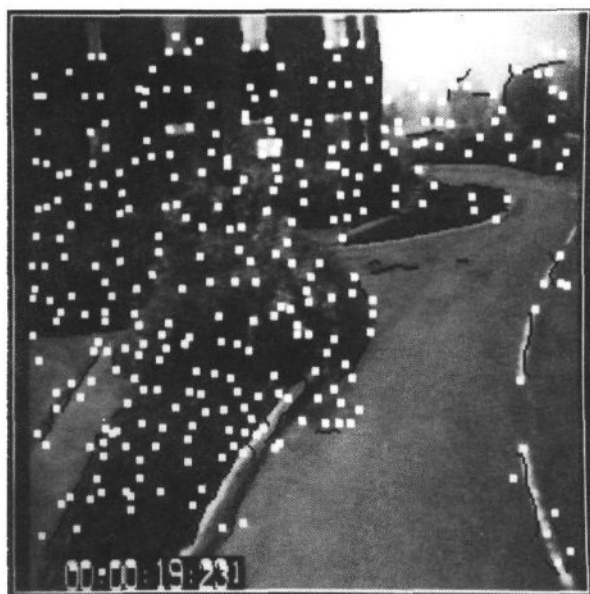


a

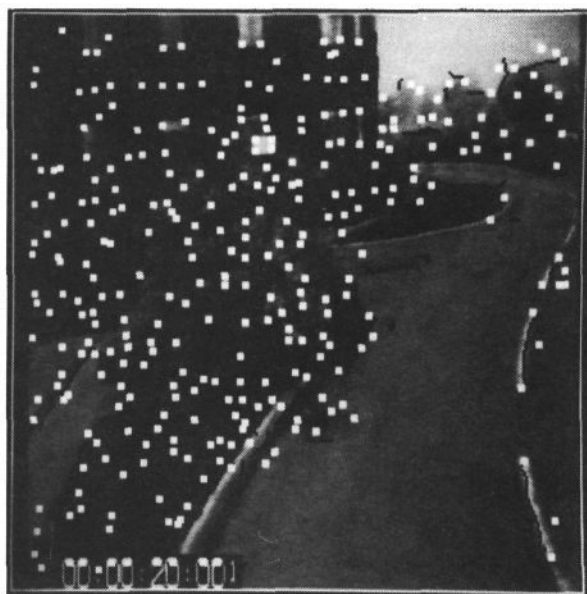


b

Figure 6. Edge/corner classification for the outdoor images (grey = corner regions, white = thinned edges).



a



b

Figure 7. Completed edges for the outdoor images
(white = corners, black = edges).

origin) in all cases increases the magnitude of the response. The flat region is specified by T_r falling below some selected threshold.

A corner region pixel (ie. one with a positive response) is selected as a nominated corner pixel if its response is an 8-way local maximum: corners so detected in the test image are shown in Figure 4d. Similarly, edge region pixels are deemed to be edgels if their responses are both negative and local minima in either the x or y directions, according to whether the magnitude of the first gradient in the x or y direction respectively is the larger. This results in thin edges. The raw edge/corner classification is shown in Figure 6, with black indicating corner regions, and grey, the thinned edges.

By applying low and high thresholds, edge hysteresis can be carried out, and this can enhance the continuity of edges. These classifications thus result in a 5-level image comprising: background, two corner classes and two edge classes. Further processing (similar to junction completion) will delete edge spurs and short isolated edges, and bridge short breaks in edges. This results in continuous thin edges that generally terminate in the corner regions. The edge terminators are then linked to the corner pixels residing within the corner regions, to form a connected edge-vertex graph, as shown in Figure 7. Note that many of the corners in the bush are unconnected to edges, as they reside in essentially textural regions. Although not readily apparent from the Figure, many of the corners and edges are directly matchable. Further work remains to be undertaken concerning the junction completion algorithm, which is currently quite rudimentary, and in the area of adaptive thresholding.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the use of imagery supplied by Mr J Sherlock of RSRE, and of the results

(the comparison of corner operators, Figure 4) obtained under MOD contract. The grey-level images used in this paper are subject to the following copyright: Copyright © Controller HMSO London 1988.

REFERENCES

1. Harris, C G & J M Pike, *3D Positional Integration from Image Sequences*, Proceedings third Alvey Vision Conference (AVC87), pp. 233-236, 1987; reproduced in *Image and Vision Computing*, vol 6, no 2, pp. 87-90, May 1988.
2. Charnley, D & R J Blissett, *Surface Reconstruction from Outdoor Image Sequences*, Proceedings fourth Alvey Vision Club (AVC88), 1988.
3. Stephens, M J & C G Harris, *3D Wire-Frame Integration from Image Sequences*, Proceedings fourth Alvey Vision Club (AVC88), 1988.
4. Ayache, N & F Lustman, *Fast and Reliable Passive Trinocular Stereovision*, Proceedings first ICCV, 1987.
5. Canny, J F, *Finding Edges and Lines in Images*, MIT technical report AI-TR-720, 1983.
6. Moravec, H, *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*, Tech Report CMU-RI-TR-3, Carnegie-Mellon University, Robotics Institute, September 1980.
7. Beaudet, P R, *Rotationally Invariant Image Operators*, International Joint Conference on Pattern Recognition, pp. 579-583 (1987).
8. Kitchen, L, and A. Rosenfeld, *Grey-level Corner Detection*, Pattern Recognition Letters, 1, pp. 95-102 (1982).

