[11] Segmentation and Description of Binocularly Viewed Contours

Tony T Pridmore, John Porrill and John E W Mayhew

AI Vision Research Unit
University of Sheffield, Sheffield S10 2TN, UK

Reprinted, with permission of Butterworth Scientific Ltd, from *Image and Vision Computing*, 1987, **5**, 132-138.

Abstract

Edge-based binocular correspondence produces a sparse disparity map, available information being distributed along space curves which project to matched image edges. To become useful these contours must be parsed into describable sections. We present a novel view of the segmentation/description process and describe an effective algorithm based on our model.

1. Introduction

The segmentation of arbitrary contours into meaningful sections is a longstanding problem receiving much attention. The goal of the present work is a description of the fragmented 3D contours to be found in edge-based binocular disparity data (figure 1). This representation provides a stepping stone to the construction of a complete wire frame model of the viewed scene.

Contour segmentation is typically seen as the identification either of discontinuities[1–7] or of describable subsections[8–10] operations which are usually treated as dual. As we are interested in providing primitive descriptions to higher processes we tend toward the latter view, though the algorithm reported here combines both approaches. Local segmentation operators are used to hypothesise discontinuities which are only retained if they delimit one or more describable segments. A contour is considered describable if the mean-squares residual associated with the most likely approximating primitive falls below some threshold. No attempt is made to accurately locate discontinuities, nor is a complete description required. In our view the primary goal of a bottom-up segmentation process should be to locate only those data sets which may be reliably described. This conservative approach provides an alternative to algorithms that obtain a fuller description at the cost of imposing interpretations which may not be appropriate.

Present (2D) approximation systems typically assume that a single primitive, usually a straight line, is sufficient to describe viewed curves. Most are based on computationally expensive split/merge algorithms guided by some measure of the accuracy of the approximation and terminating when an adequate description has been achieved (cf.[11]). For real scenes it is not clear that a single primitive will suffice. Furthermore, the generalisation of these techniques to multiple primitives is non-trivial and may involve optimisation theory (e.g. [12]).

Segmentation operators are more flexible; if descriptions are derived after discontinuity detection a wider set of primitives may be considered. It has, however, been argued[13] that reliable segmentation cannot be achieved without reference to the local structure of the data. A further problem is that a given discontinuity may not give rise to a unique, identifiable data item. It is quite probable, given the noise inherent in the disparity map, that the exact location of a particular discontinuity will not be covered by the data at all.

In the algorithm discussed below, simple segmentation operators provide heuristic guidance to a contour description process. This reduces the number of approximations attempted while allowing multiple primitives to be considered and relieving the segmentation operators of the responsibility of accurate discontinuity localisation.

In the following we describe segmentation and description techniques and present a recursive segmentation/description algorithm based upon the above considerations. Input to this algorithm are ordered strings of disparity measurements obtained via the Canny edge detector[14], PMF[15] and CONNECT[16].

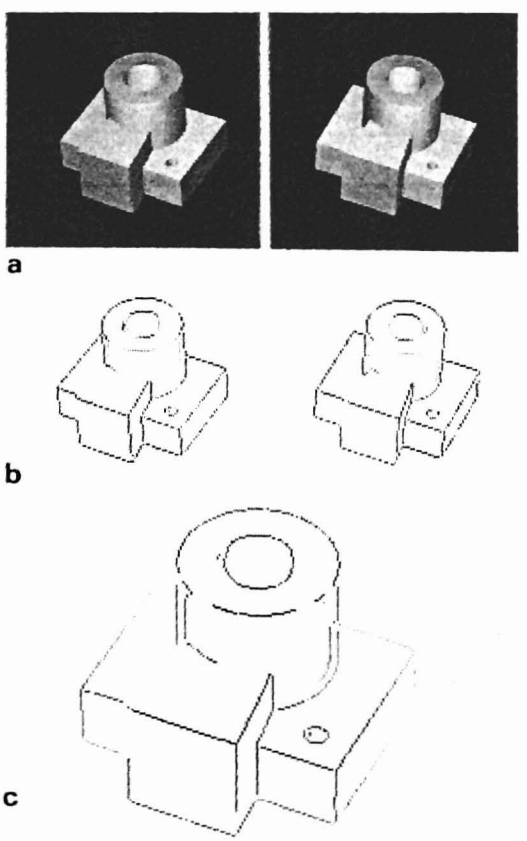


*Figure 1. Typical input data obtained from a pair of IBM Winsom images via PMF. The images (a), 256*256 pixels with 256 grey levels presented for cross-eyed fusion, give rise to edge assertions (b). The edge detector is a Canny operator with σ 1.0. Correspondence produces the sparse disparity map (c). Disparities are coded dark to light with increasing depth.*

2. Segmentation

Two operators are employed, recording curvature (κ) and its derivative ($\dot{\kappa}$) as functions of arc length. Both κ and $\dot{\kappa}$ estimates are obtained by differentiation of a locally approximating quadratic. Peaks in these measurements are assumed to mark discontinuities in orientation and curvature respectively, a supposition that is common in the literature[2–5,7].

It has long been appreciated that the performance of a given differential operator depends heavily upon the relative spatial extent, or scale, of the device and the features to which it is applied[17]. This observation has led to recent explorations of Scale Space[18,4]. Although the construction of a multiple scale representation is beyond the scope of the current project, the algorithm presented here does use smoothing to alleviate quantisation noise before computing curvature properties. The technique applied is the diffusion method of Porrill et. al.[19]. Only a small amount of smoothing is necessary; diffusion roughly equivalent to a gaussian of σ 2.5 is usually sufficient. When significantly larger σ are used peaks tend to migrate, making even approximate localisation difficult. After smoothing quadratics are fitted through triples of adjacent points.

Peaks (and troughs) in κ and $\dot{\kappa}$ are detected by thresholding absolute values. When examining $\dot{\kappa}$ it is important not to tag the side lobes of zero-crossings associated with peaks in curvature. For this reason supra-threshold $\dot{\kappa}$ estimates are only marked if no significant peaks in κ are found within a given neighbourhood. Asada and Brady[4] give the following expression for the arc length between the side lobes of a zero crossing caused by an angular discontinuity θ between contour segments with curvatures κ_1, κ_2 where σ is the standard deviation of an initial gaussian smoothing function:

$$d_{corner}=\frac{\sigma^2}{\phi}\left[\left[\kappa_1-\kappa_2\right]^2+\frac{4\phi^2}{\sigma^2}\right]^{\frac{1}{2}}$$

For a pure corner in which

$$\kappa=\kappa_1=\kappa_2$$

this simplifies to

$$d_{pure}=2\sigma$$

Assuming all angular discontinuities to be pure and the peak to lie approximately mid-way between the side lobes, we impose the condition that no significant curvature peak may lie within σ units of arc length of a marked peak in curvature difference. A curvature peak is considered significant if its absolute value is greater than the threshold that would be applied if such features were being sought. The use of diffusion means that our σ, the diffusion scale, is only an approximation to the gaussian parameter. It appears, however, to be a sufficiently close approximation for current purposes.

Note that κ and $\dot{\kappa}$ are measured in world, as opposed to disparity, coordinates. The transformation from disparity to world scales the depth component with respect to the other dimensions. Hence if curvature properties were estimated in disparity space the depth component would be compressed, playing a reduced role in segmentation. Segmenting in real coordinates allows depth information to contribute fully to all measurements. The increased error brought about by the disparity->world scaling does not appear to affect the segmentation process unduly, though the computation of mean squares residuals is greatly complicated. As quantisation error in disparity is isotropic in all three directions, the approximation techniques described below are applied to disparity values[20].

3. Description

In the current scheme, extrema in κ and $\dot{\kappa}$ serve to suggest likely areas of discontinuity; before any final decision can be made a geometrical description of the surrounding contour is required. An input string may be classified as a straight line, circular arc, planar or undescribable space curve. Mathematical details of the techniques involved are presented elsewhere[20], here we discuss the algorithm by which they are applied.

Given a set of points, orthogonal regression[20–22] supplies mean-squares residuals and metrical descriptions of the best fit plane, straight line and point. The residuals, related by the expression $res_{point}>res_{line}>res_{plane}$, are examined in decreasing order of magnitude, the first to fall below threshold* being taken as representative of the true description. Should all the residuals be above threshold, a default space curve tag is assigned. Short strings often match each primitive with a high degree of accuracy, in which case curves are assumed to be locally straight.

After regression, plane curves are passed to a three point circle fitting routine. If the circle residual is below threshold it is accepted, otherwise the plane descriptor becomes the primary representation. Although the choice of points may affect the residual, constraints imposed by CONNECT[16] force input strings to be at least Lipschitz continuous[23]. It is therefore unlikely that any significant discontinuities will be due to noise, making point selection less critical. As a further safeguard points are chosen from diffused data. This could introduce error if the data were heavily diffused, though it seems unlikely given the limited smoothing used here. As the extension of regression techniques to circle fitting is computationally expensive[22], the reduced cost of a three point fit easily outweighs its potential disadvantages.

Difficulties arise when describing curves containing horizontal segments. In such cases the binocular correspondence problem is effectively insoluble and any disparity values obtained must be considered unreliable. To avoid erroneous data, residual components arising from horizontal sections are computed in the image plane[20]. Geometrical descriptions, however, are always computed in 3-D, using non-horizontal data. Horizontal disparity elements are detected by thresholding the orientation of matched edge assertions. Although this method cannot solve the problem in its entirety, it does represent a first attempt to improve initial disparity measurements on the basis of later processing. If an input string is entirely horizontal its disparity values are used and the resulting segment(s) labelled.

4. An Algorithm

The segmentation/description scheme presented here admits many possible algorithms. We concern ourselves

with a single example, known as GDF (Geometrical Descriptive Filter). Processing begins with a call to the description algorithm, thereafter focusing attention on strings not immediately represented by a single straight or circular segment. Planar and undescribed space curves are passed to a recursive segmentation algorithm which may be summarised thus:-

(1) $\dot{\kappa}$ estimates arising from non-horizontal data are thresholded at 90% of their maximum value, supra-threshold data being tagged as possible segmentation points. Should the string be entirely horizontal all of the data is used.

(2) Tags are removed from any points which fail the side-lobe test

(3) If no $\dot{\kappa}$ tags remain or all substrings are below the required length, extrema in κ are sought. A threshold is again set at 90% of the maximum (non-horizontal) value and (non-horizontal) supra-threshold points tagged.

(4) If all hypothesised substrings fall below the length threshold horizontal data is removed by segmenting at the ends of horizontal sections.

(5) If no acceptably long segments result the description reverts to the previous plane or space curve representation, otherwise long substrings are passed to the descriptive processes. Any classified as space or plane curves are further subdivided by recursive application of the segmentation procedure.

GDF seeks the longest acceptable primitives while segmenting at the largest κ and/or $\dot{\kappa}$ values. Peaks in $\dot{\kappa}$ are sought first. If all hypothesised segmentation points are rejected κ values are examined. Should κ fail to provide an acceptable segmentation horizontal data is removed by placing segmentation points at the ends of horizontal sections. If this also fails the representation reverts to the previous space or planar curve description. Note that the length threshold measures the number of data points available, rather than the absolute length of the curve: reliable classification of short strings is problematic. Hypothesised segments of above threshold length are then passed to the description algorithm. After classification, any remaining plane or space curves are recursively segmented. In this way strings are subdivided until either a satisfactory representation is obtained or the segments remaining fall below a length threshold. On termination, a GDB (Geometrical Descriptive Base) format[25] file is produced.

In peak detection the choice of threshold is critical, being subject to a trade-off. If thresholds are set too high, extrema will be missed and the data only partially segmented, typically leading to extra plane and/or space curve descriptions. Underestimating may, on the other hand, lead to oversegmentation, breaking long strings into arbitrarily small segments. Both effects reduce the information content of the final representation, though the former is more serious. Later processes should be able to recover from fragmented data, though the computational cost incurred may be considerable. GDF sets thresholds dynamically at 90% of the maximum absolute value of the appropriate operator. An alternative strategy[8,11] would be to segment at the maximum. This, however, would restrict the system to marking a single discontinuity on each recursion. Under the current scheme it is common for several peaks to be marked, speeding segmentation and reducing the number of approximations required. As thresholds are set just below maximum the danger of marking noise is small. On successive recursions operator maxima, and therefore the thresholds applied, become smaller. GDF is, therefore, (trivially) guaranteed to terminate.

Note that a simple threshold is applied; no suppression of non-maximal estimates is assumed. As a result, most peaks/troughs generate a pair of segmentation points, one either side of the local extremum. An interesting feature of this technique is the way in which the distance between segmentation point and local extremum increases with the spatial extent of the discontinuity. Hence GDF tends to locate fine scale discontinuities with a greater degree of accuracy. This has the secondary effect that strings having rapidly changing properties are rarely passed to the descriptive processes.

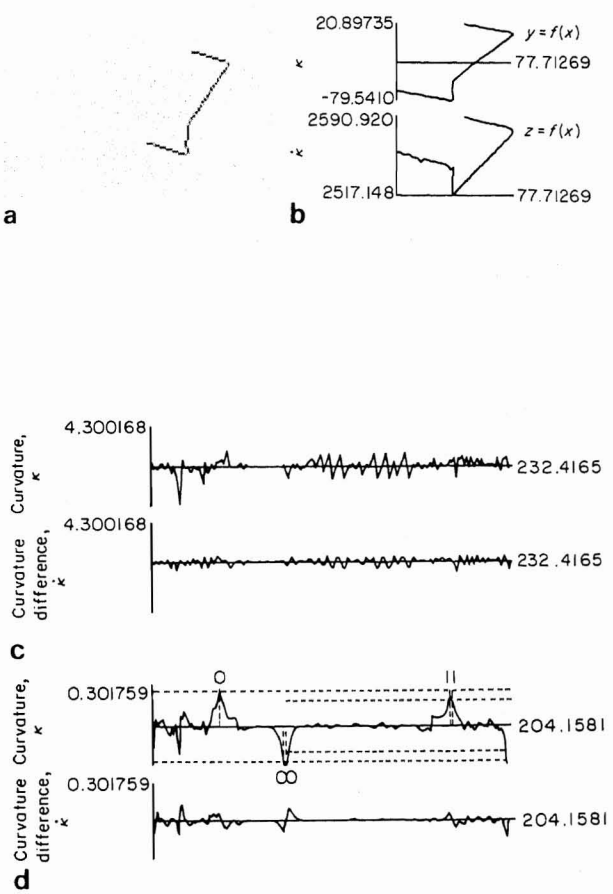
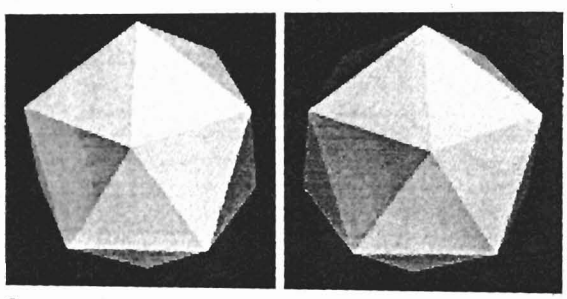


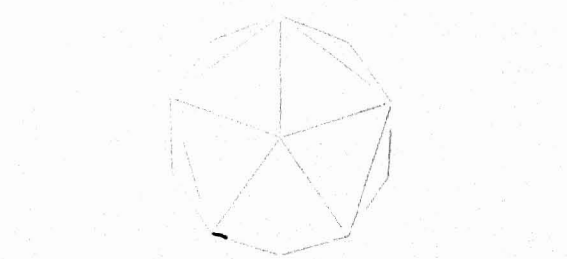
Figure 2. Recursive segmentation/description. b) data arising from the line highlighted in a). c) κ and $\dot{\kappa}$ values before diffusion. d) smoothed κ, $\dot{\kappa}$ plots (σ = 2.5). Segmentation points are marked by vertical lines tagged with recursion depths.

The approximation algorithm, where possible, extrapolates descriptions of non-horizontal edges into unreliable, horizontal data. If this is to be effective, horizontal strings must only contribute segmentation points as a last resort; the descriptive processes should first be given every chance to correct erroneous data. Threshold selection and application is, therefore, normally limited to non-horizontal depth estimates. Some strings are, however, entirely horizontal, in which case unreliable measurements must be used. Horizontal data introduces problems the techniques employed here can only begin to solve. The removal of horizontal segments when both differential operators fail to produce an acceptable segmentation is an explicit recognition of this limitation.

Figure 2 illustrates the application of GDF to the data of figure 1. The plots shown in fig. 2b display 3D position estimates, y and z coordinates as functions of x, arising from the line highlighted in figure 2a. Raw κ and $\dot{\kappa}$ values, obtained before diffusion and plotted as functions of arc length, are presented in figure 2c. Diffusion (σ=2.5) produces the smoothed plots shown in figure 2d. Note that three clearly distinguishable peaks have emerged. Dashed vertical lines represent segmentation points found by GDF, each tagged with an integer specifying the depth of recursion at which the discontinuity was identified. Threshold values, and the strings to which they were applied, are marked by horizontal dotted lines. The final representation in this case comprised three straight segments.



a



b

Figure 3: IBM Winsom generated test data. a) stereo images of a unit icosahedron viewed at 6 interocular distances. b) GDB representation. Circles are represented by broad lines, straight segments by fine.

Although the differential operators are said to hypothesise segmentation points, no hypotheses are ever fully rejected. All intermediate representations are recorded, building for each string a tree comprising non-terminal planar and/or space curve nodes and leaves representing straight or circular arcs. Note, however, that no attempt is made to produce a fuller description by combining segments. As Blake and Mayhew[26] point out, unrestricted computation of geometrical information is expensive and may lead to representations that are redundant given the task at hand. Even so, it is still sensible to exploit information obtained as a side effect of the normal description procedure.

5. Examples and Evaluation

The most important evaluation criterion for any 3D vision system is the extent to which it allows subsequent tasks to be performed. In the present case this clearly depends upon the geometrical accuracy of the contour descriptions supplied. As prior processing stages exert considerable influence on the final representation, any examination of GDF output incorporates some evaluation of the lower level components of the system.

Figures 3 and 4 show test data used to examine the accuracy of the GDB representation. To avoid camera calibration error, IBM Winsom generated images were employed. In these and all subsequent examples a Canny[14] edge operator (σ = 1.0), PMF[15] , CONNECT[16] and GDF (diffusion scale 2.5) were applied to the original 256*256 pixel, 256 grey level images. Figure 3 shows the images and GDB description arising from a unit icosahedron viewed at 6 interocular distances (approximately human reading distance). In figure 4 the icosahedron is intersected with a sphere, generating circular faces of known size, position and orientation.

Comparison of line equations derived from the Winsom model with straight line descriptors contained in GDF output shows a mean absolute error of 0.58 degrees in the internal angles of the icosahedron. The mean absolute error in line orientation is 0.24 degrees. Note that PMF was unable to match the horizontal edge at the bottom of the figure, this is an extreme example. Correspondence is usually achieved, though the resulting disparity data is always unreliable. Positional error was estimated by measuring the perpendicular distance from the mid-point of the GDB description to the true line, the mean error being 6.24 pixels. Winsom measures distances in image units, although no conversion to external units is possible, this is clearly a small error.

A similar examination of circle descriptions (figure 4b) shows a mean absolute error in circle radii and centre position of 0.71 and 1.20 pixels respectively. Each circle is associated with a plane descriptor. The mean absolute error in the internal angles of these planes is 2.05 degrees, while the mean absolute error in plane orientation is 2.49 degrees. Most circles give rise to some horizontal edges. Horizontal sections in the data of figure 4 are marked in figure 5a. Figure 5c shows the position estimates attributed to the arc highlighted in figure 5b. Note the flattened depth plot in the horizontal region, despite this distortion the circle description is recovered satisfactorily. In this case the error in plane orientation is 1.72 degrees, the circle radius and centre having errors of 1.38 and 2.01 pixels

respectively. It will be noted that full circles apparent in the original images (figure 4a) are represented in GDF output by pairs of semi-circles. This is a result of missing data in the edge detection phase. Although image noise is limited by the use of Winsom, such problems may still occur.

It is clear from the above that useful 3D descriptions can be derived, via GDF, from edge based disparity data. An important question, however, concerns the stability of the GDB representation over changes in viewpoint. Although noise in edge detection and the number and position of horizontal edges are obviously view-dependent, some measure of stability is to be expected. Figure 6 shows natural image pairs and GDB representations of two views (separated by a rotation of approximately 180 degrees) of a wire, seen from approximately four interocular distances. Under this geometry 250 pixels is approximately equal to 1 cm. Note that the location of segmentation points and the geometry of the final representation are similar. The most noticeable error arises in the radii of the small circular arcs close to the free end of the wire. A considerable amount of data is required if circles are to be recovered accurately, examination of the longer arc parallel to the short segment of figure 6d shows a reduced error. As a further test, GDF output has been successfully exploited in the model matching work of Pollard et. al[27]. A more detailed experimental evaluation of GDF may be found in Pridmore[24].

6. Conclusion

GDF has been implemented and appears both effective and robust. Several features should be stressed:

Segmentation operators provide heuristic control to contour description processes.

Descriptions obtained from non-horizontal data are, where possible, extrapolated into horizontal regions.

No attempt is made to label the exact positions of discontinuities. Rather we report the end points and geometrical properties of the largest acceptable approximating segments.

GDF makes no strong assumptions about viewed curves, preferring instead to capture only those sections which may be closely approximated by a set of simple primitives.

The construction of a complete wire frame from GDF output is currently being investigated.

7. Discussion Note

The use of thresholding is often justifiably criticised in the image processing literature on the grounds that it reduces generality: thresholds for edge detection for example often need to be tuned to particular world domains and/or imaging conditions.

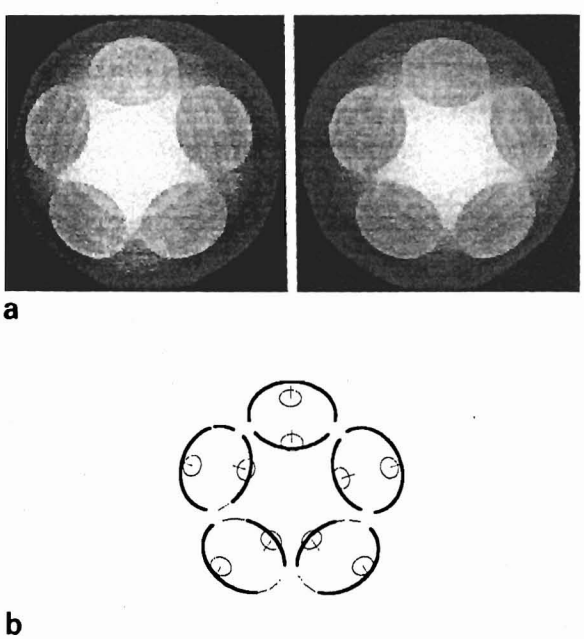


Figure 4: IBM Winsom generated test data. a) stereo images of a unit icosahedron intersected with a sphere. b) GDB representation. Circles are represented by broad lines, straight segments by fine.

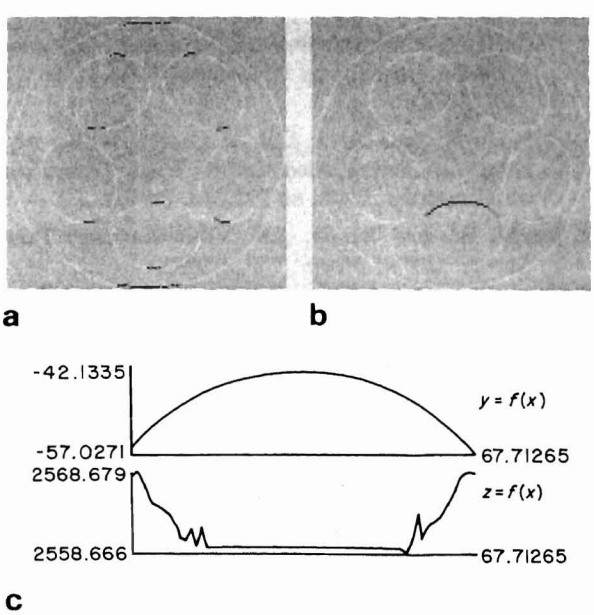


Figure 5. The effect of horizontal edges. a) disparity map obtained via PMF from the object of figure 4. Horizontal segments are drawn in black, other data in white. Position plots, y and z as functions of x, associated with the circular arc marked in black in (b) are presented in (c). Note the distorted z values arising from horizontal data.

This criticism is not applicable to our use of thresholds during curve fitting because those thresholds are to do with how any given set of edge locations can best be described geometrically. This is a different issue from how those edge tokens are obtained in the first place. Different edge operators will of course yield different edge tokens, and hence they would lead our system to produce different geometric descriptions. But that is not in itself a valid criticism of the use of thresholds within the processes we propose for obtaining geometric descriptions. We think it may be necessary to emphasise this point in view of the comments made by a referee. The residual values we threshold during orthogonal regression can be interpreted as measuring the standard deviation of error in edge location given a particular geometric fit (if errors in edge locations are assumed to be distributed normally which seems reasonable). Hence our use of thresholding is approximately equivalent to applying a χ^2 test of goodness of fit. The current threshold of 0.5 pixels amounts to the requirement that any accepted geometric description is unlikely to deviate from the data at any point by more that about one pixel.

Acknowledgements

We would like to thank Stephen Pollard and John Frisby for comments and advice and Chris Brown for his valuable technical assistance. This research was supported by SERC project grant no. GR/D 16796.

References

1 **Rosenberg, B.,** The Analysis of Convex Blobs, *Computer Graphics and Image Processing*, 1, pp 183-192, (1972).

2 **Davis, L.S.,** Understanding Shape: Angles and Sides, *IEEE Transactions on Computers*, C-26, 3, pp 236-242, (1977).

3 **Freeman H., and Davis, L.S.** A Corner Finding Algorithm for Chain Coded Curves, *IEEE Transactions on Computers*, C-26, pp 297-303, (1977).

4 **Asada, H., and Brady, J.M.,** The Curvature Primal Sketch, *MIT AI Memo*, 758, (1984).

5 **Sakai, T., Nagao, N, and Matsushima, H.,** Extraction of Invariant Picture sub-structures by Computer, *Computer Graphics and Image Processing*, 1, pp 81-96, (1972).

6 **Feng, H.Y., and Pavlidis, T.,** Finding Vertices in a Picture, *Computer Graphics and Image Processing*, 2, pp 103-117, (1973).

7 **Shirai, Y.,** Analysing Intensity Arrays Using Knowledge About Scenes, in *The Psychology of Computer Vision*, ed. P.H. Winston, McGraw-Hill, (1975).

8 **Ramer, U.,** An Iterative Procedure for the Polygonal Approximation of Plane Curves, *Computer Graphics and Image Processing*, 1, pp 244-56, (1972).

9 **Pavlidis, T.,** Waveform segmentation through Functional Approximation, *IEEE Transactions on Computers*, C-22, 7, pp 689-97, (1973).

10 **Pavlidis, T., and Horowitz, S.L.,** Segmentation of Plane Curves, *IEEE Transactions on Computers*, C-23, 8, pp 860-70, (1974).

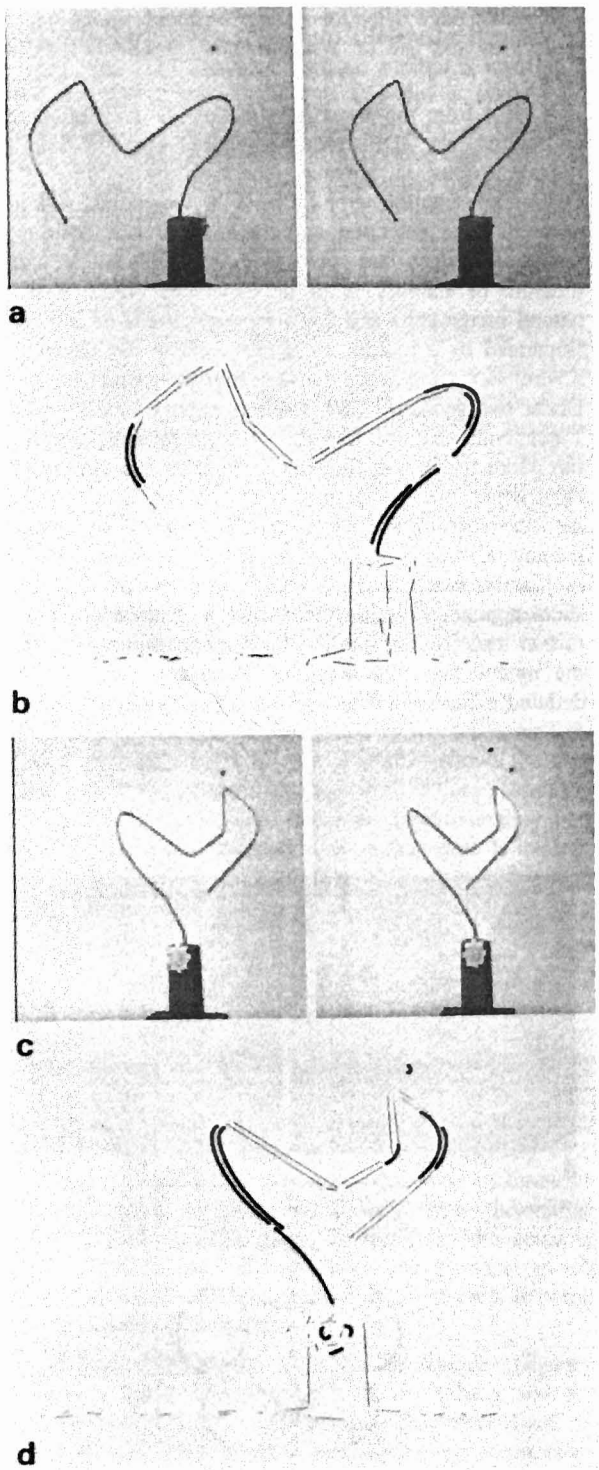


Figure 6: Stability over changes in viewpoint. a), c) natural stereo images of a bent wire taken from viewpoints separated by a rotation of approximately 180 degrees. b), d) GDB representations. The descriptions obtained are qualitatively and quantitatively similar.

11 **Ballard, D.H.,** Strip Trees: A Hierarchical Representation for Curves, *Communications of the Association for Computing Machinery*, 24, 5, pp 310-21, (1981).

12 **Plass, M., and Stone, M.,** Curve-Fitting with Piecewise Parametric Cubics, *Computer Graphics*, 17, 3, pp 229-239, (1983).

13 **Leclerc, Y., and Zucker, S.W.,** The Local Structure of Image Discontinuities in One Dimension, *McGill University Computer Vision and Robotics Laboratory Technical Report*, TR-83-19R, (1983).

14 **Canny, J.F., Finding Edges and Lines in Images,** *MIT AI memo*, 720, (1983).

15 **Pollard, S.B., Mayhew, J.E.W. and Frisby, J.P.,** PMF: A Stereo Correspondence Algorithm Using a Disparity Gradient Limit, *Perception*, 14, pp 449-470, (1985).

16 **Pridmore, T.P., Mayhew J.E.W., and Frisby, J.P., Production Rules for Grouping Edge-based Disparity Data,** Paper presented at the Alvey Image Interpretation Conference, Sussex University, and *AIVRU Memo*, 015, (1985).

17 **Marr, D., and Hildreth, E.,** Theory of Edge Detection, *Proc. Roy. Soc. Lond.*, B207, pp 187-217, (1980).

18 **Witkin, A.P., Scale Space Filtering,** *Proc. 8th IJCAI*, Karlsruhe, W. Germany, pp 1019-1023, (1983).

19 **Porrill, J., Mayhew, J.E.W., and Frisby, J.P.,** Scale Space Diffusion: Planes and space Curves, *AIVRU memo*, 018, (1986).

20 **Porrill, J., Pridmore, T.P., Mayhew, J.E.W., and Frisby, J.P.,** Fitting Planes, Lines and Circles to Stereo Disparity Data, *AIVRU memo*, 017, (1986).

21 **Pearson, K.,** On Lines and Planes of Closest Fit to Systems of Points in Space, *Phil. Mag. VI*, 2, pp 559, (1901).

22 **Ballard, D.H., and Brown, C.M.,** *Computer Vision,* Prentice Hall, (1982).

23 **Pollard, S.P., Porrill, J., Mayhew, J.E.W., and Frisby, J.P.,** Disparity Gradient, Lipschitz Continuity and Computing Binocular Correspondences, *Proc 3rd ISRR*, Gouviex, France, (1986).

24 **Pridmore, T.P.,** PhD Thesis, University of Sheffield, forthcoming.

25 **Pridmore, T.P., Bowen, J.B., and Mayhew, J.E.W.,** Geometrical Description of the CONNECT Graph #2, The Geometrical Base Description: A Specification, *AIVRU Memo*, 012, (1985).

26 **Blake, A., and Mayhew, J.E.W.,** Alvey 2&1/2-D Sketch Project: Proposed Structure For a Development System, *AIVRU Memo* , 009, (1985).

27 **Pollard, S.B., Porrill, J., Mayhew, J.E.W., and Frisby, J.P.,** Matching Geometrical Descriptions in Three Space, *Alvey Computer Vision and Image Interpretation Meeting and Image and Vision Computing (submitted)*, (1986).