Open access • Proceedings Article • DOI:10.5244/C.2.9

Object cues for model-based image interpretation — Source link

Ann Thornham, Christopher J. Taylor, D. H. Cooper

Published on: 01 Jan 1988 - Alvey Vision Conference

Topics: Real image, Local symmetry and Symmetry (geometry)

OBJECT CUES FOR MODEL BASED IMAGE INTERPRETATION

Ann Thornham, Christopher J. Taylor and David H. Cooper.

Wolfson Image Analysis Unit
Department of Medical Biophysics
University of Manchester
Oxford Road
Manchester M13 9PT

This paper is concerned with generating object cues from grey-level images for use in model-based image interpretation. We describe the idea of local grey-level symmetry and illustrate how points in the grey-level image with this property form local axes of symmetry. These axes together with appropriate scale information form the object cues. The degree of local symmetry in the grey-level image is made explicit by introducing a Centre of Gravity filter. The local axes of symmetry are shown to appear in the centre of gravity image as troughs and a method for labelling the troughs is described. We give results for real images and make an objective comparison with other published methods.

This work forms part of a project called 'Techniques for User Programmable Image Processing' (TUPIP) which is directed towards a computer vision system which can accept a description of a visual task from a user, who is not an image processing expert, and generate a solution, ie. be capable of performing the task. Within the proposed system the scene involved in the visual task is described in terms of an hierarchical model. In order to attempt to instantiate parts of this model the system must extract initial evidence from the image. The requirement is for simple symbolic entities, or cues, generated by applying low-level processing to the raw image data. These cues will be used to locate objects in the scene for model matching.

Much work has been devoted to developing optimal methods for detecting edges and uniform region primitives. For complex scenes it is, however, difficult to generate object hypotheses from such cues. Edge cues are too low-level in that they describe only local properties whilst in complex scenes objects do not necessarily correspond to uniform regions. With edge cues there also exists a combinatorial problem of associating multiple cues with multiple objects.

We believe that axes of symmetry would form useful object cues especially if the concept of scale could be introduced so that a course-to-fine description was available. We have developed a robust, scale sensitive method of generating axes of local symmetry directly from grey-level images, based on the idea of grey-level symmetry over a neighbourhood.

Axes of local symmetry have been used by other authors to describe shape. Many of the techniques reported, such as the medial axis transformation [1] (MAT) and the method of smoothed local symmetries [2] (SLS), involve detecting object boundaries first in order to find the axes of symmetry. They are inappropriate to the problem we consider here where we wish to use axial symmetries as cues to seed the process of boundary location. Other shape descriptors, such as the min-max MAT [3] (MMMAT), can be calculated directly from the grey-level image but have disadvantages which are discussed later.

OVERVIEW

To illustrate the concept of local grey-level symmetry we introduce a grey-level image which consists of a light, bar-shaped object on a darker background as shown in Figure 1. If we consider a disk-shaped neighbourhood for each point in the image, then for a point at the centre of the bar (Figure 1(a)) the grey-levels in the neighbourhood are evenly distributed in all directions about the point and the point has local grey-level symmetry. However, for a point away from the centre of the bar (Figure 1(b)) the grey-levels are not evenly distributed in all directions within the neighbourhood and the point does not have local grey-level symmetry.

We have introduced the idea that the position of the Centre of Gravity (COG) of the grey-levels within a neighbourhood can be used to compute the degree of local grey-level symmetry. We have developed a COG filter which operates over a grey-level image finding the coordinates of the centre of gravity for a neighbourhood centred at each point in the image. The distance of the centre of gravity from the centre of the neighbourhood is recorded in a filtered or COG image. If the grey-levels in the region are symmetric then the centre of gravity is at the centre of the region and the distance value is zero. If however the grey-levels are

This work is supported by a SERC grant and is part of ALVEY project MMI-093: 'Techniques for User Programmable Image Processing' (TUPIP).

AVC 1988 doi:10.5244/C.2.9

not symmetric the centre of gravity is away from the centre of the region and the distance value is non-zero. The magnitude of the distance value is a measure of the asymmetry of the grey-levels and hence the COG filter makes explicit the degree of local symmetry.

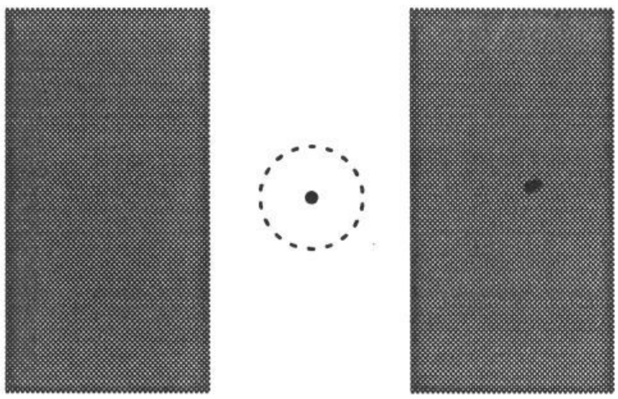

Figure 1(a). A disk shaped neighbourhood with local grey level symmetry.

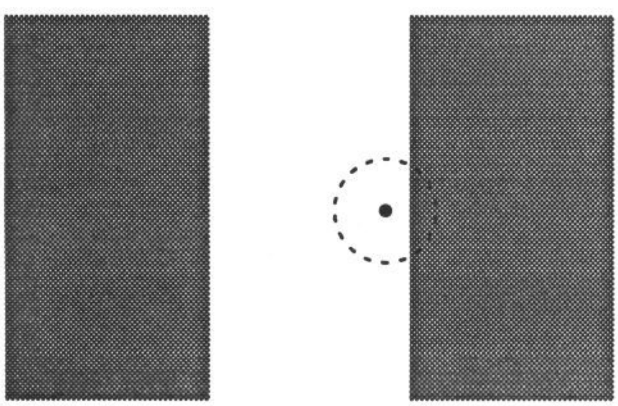

Figure 1(b). A disk shaped neighbourhood with locally asymmetric grey-levels.

Given a COG image it remains to extract the axes of local symmetry. If we consider the example illustrated in Figure 1, many points in the grey-level image satisfy the local grey-level symmetry criterion. In fact the number of points which have local symmetry depends upon the size of the neighbourhood around each point. This introduces the idea of scale: whether a point has local symmetry or not depends upon the size of the neighbourhood, or scale, at which the centre of gravity filter is operated. If the diameter of the neighbourhood is much less than the width of the bar then there are many points within the bar which have local symmetry. The number of points reduces as the diameter of the neighbourhood increases. In particular, when the diameter equals the width of the bar there is a line of points at the centre of the bar which have local symmetry. The line persists when the neighbourhood size exceeds the width of the bar. For the bar shaped object this line represents the local axis of symmetry or object cue that we seek and the points which make up this line appear in the COG image at the base of a valley or trough.

In general terms, when the COG filter operates over a grey-level image at a particular scale, if that scale equals or exceeds the scale of a 'feature' in the source image then a corresponding trough appears in the filtered image. At any one scale, these troughs represent local axes of symmetry. We have developed a principled and efficient method of trough detection which, when applied to the COG image allows axes of local grey-level symmetry to be extracted.

THE COG FILTER

Using the conventional definition, the coordinates of the centre of gravity $(\mathbf{I},\mathbf{J})$ of a neighbourhood centred at (x,y) are given by

$$\mathbf{I}(x,y) = \frac{\sum_{i,j} i\,.f(x+i,y+j)}{\sum_{i,j} f(x+i,y+j)}$$

$$\mathbf{J}(x,y) = \frac{\sum_{i,j} j\,.f(x+i,y+j)}{\sum_{i,j} f(x+i,y+j)}$$

where f(x,y) is the image value at (x,y) and i,j are pixel relative offsets. The summation is over the neighbourhood.

In practice we make two modifications to this definition. First we argue that the normalising denominator can be ignored, simplifying the calculation. This has the effect that off-symmetry response is proportional to local, rather than global, feature contrast: an arguably desirable consequence. Second, rather than using a simple disk-shaped neighbourhood we use a gaussian window which leads to better behaviour in the frequency domain. Our final definitions for the components of the COG are therefore

$$\mathbf{I}(x,y) = \sum_{i,j} i\,.f(x+i,y+j)\exp\{-(i^2+j^2)/\sigma^2\}$$

$$\mathbf{J}(x,y) = \sum_{i,j} j\,.f(x+i,y+j)\exp\{-(i^2+j^2)/\sigma^2\}$$

One advantage of this form for $(\mathbf{I},\mathbf{J})$ is that the filter can be decomposed into four one-dimensional convolutions and the computational complexity reduced from $O(n^2)$ to $O(n)$ for a neighbourhood of halfwidth n.

Finally the COG image is given by

$$R(x,y) = \sqrt{(I^2+J^2)} \ .$$

and represents the mass weighted distance of the centre of gravity from the centre of the neighbourhood at each point in the image.

Figure 2 is an example of a COG image generated for a dark pair of pliers on a light background. If the source image is inverted (to produce a light pair of pliers on a dark background) the same COG image is obtained.

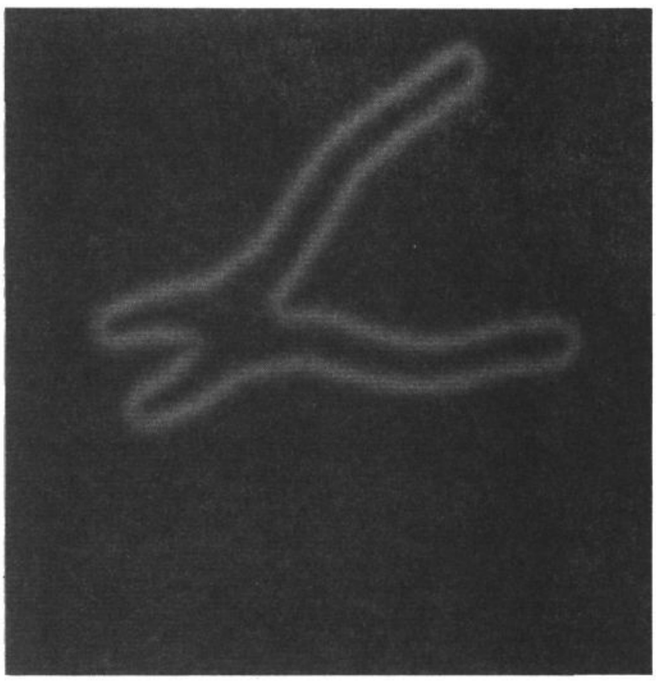

Figure 2. COG image of a pair of pliers.

THE TROUGH DETECTOR

We have established that the points of interest appear in the COG image as troughs. A number of ad hoc algorithms for detecting troughs have been reported previously.[4,5] We have attempted to find a more principled method. It is also important to note that the troughs in the COG image have certain characteristics which the trough detector must be able to exploit, for example, the filter halfwidth at which a trough point first appears provides important scale information about the corresponding 'feature' in the source image thus the trough detector must exclude 'flat bottomed' troughs. The trough detector should also be relatively insensitive to noise so that only genuine troughs are labelled.

Every point in the COG image must be examined to see whether or not it is a trough point. Without some a priori information about the orientation of potential troughs, every possible orientation must be considered at each point. Clearly the volume of computation is substantially reduced if a direction cue can be generated at each point so that the trough detector need be operated in one direction only. By finding a direction in this way the problem of trough detection can be reduced to one dimension.

As the trough points appear in the filtered image as dark line-like structures we have used the directional output from a line/edge detector [6] to direct the trough detector at each point. The line/edge detector has a 5*5 neighbourhood and gives an output quantised to eight directions. The trough detector reads the direction for each point in the filtered image and uses it to calculate the angle through which to rotate its 5*5 neighbourhood in order to align the axes of the neighbourhood with the direction of the potential trough. New rotated coordinates are calculated for each of the 25 points in the neighbourhood and bilinear interpolation is used to calculate the new grey-level at each point.

Given the rotated neighbourhood (Figure 3) the grey-level values in each row are summed to give 5 points on a one dimensional function.

row 2	a	b	c	d	e
row 1	f	g	h	i	j
row 0	k	l	m	n	o
row −1	p	q	r	s	t
row −2	u	v	w	x	y

→ trough direction

Figure 3. The rotated neighbourhood at a point.

We fit a least square parabola which has the equation

$$f(t) = a + bt + ct^2$$

to the 5 points (2, a+b+c+d+e), (1, f+g+h+i+j), (0, k+l+m+n+o), (−1, p+q+r+s+t) and (−2, u+v+w+x+y). Having evaluated the coefficients a,b and c we need criteria to establish whether row 0 lies on a trough. We use the constraint that the curve must have a minimum (ie. c is positive) and the minimum must lie in the interval [− 0.55, 0.55], to suppress non-minimal responses.

The rotation of the trough detector's 5*5 neighbourhood to find the new rotated coordinates, the interpolation to find the new grey-levels at the rotated coordinates and the calculation of the coefficients a,b and c in the least square parabola are all linear processes and can be combined together into one 7*7 convolution mask for each coefficient in each direction (8 directions * 3 coefficients = 24 masks). Combining the masks in this way makes generating the coefficients very efficient. Since only one direction is used at each point at most 3 whole-image convolutions are involved. Moreover, deciding whether the curve has a minimum

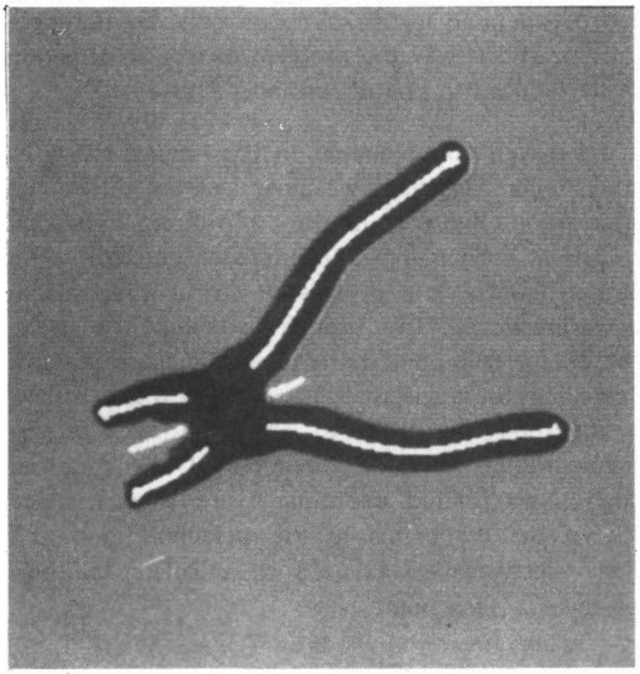

Figure 4. Trough points in the COG image of the pliers.

Figure 5. Chromosomes.

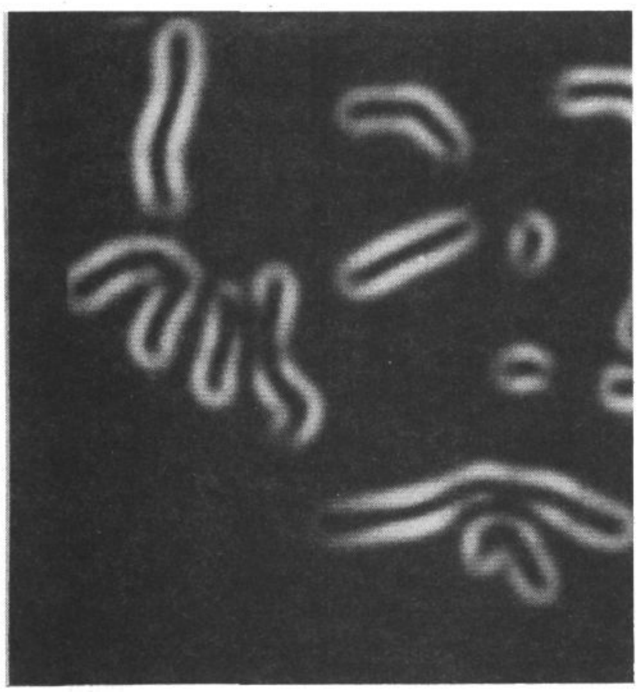

Figure 6. COG image of the chromosomes.

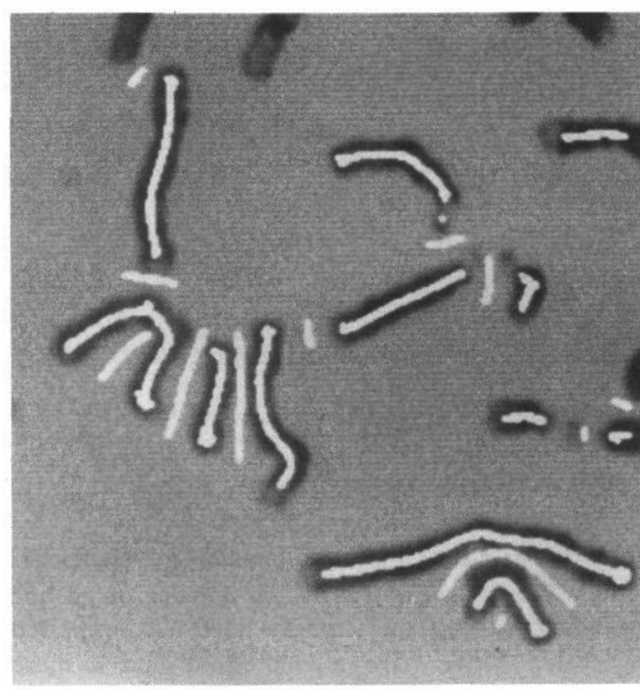

Figure 7. Trough points for the chromosomes.

involves evaluating only coefficient c at each point in the image. If the curve is found to have a minimum then calculating its location requires coefficient b as well.

The trough points in the COG image shown in Figure 2 have been identified and labelled and are shown (overlayed on the original image) in Figure 4. To obtain this result, the COG filter was run at a scale appropriate to the width of the four 'arms' of the pliers. The troughs which lie external to the object are clearly visible in this example as well as those internal to the object.

RESULTS

Real Images

Figures 5–9 show results obtained using real images. Figure 5 shows a microscope image of G–banded chromosomes. Figure 6 shows the corresponding COG image. The troughs extracted from the COG image and overlayed on the original chromosome image are shown in Figure 7. The COG filter was run at a scale appropriate to the width of the chromosomes. For a fairly straight isolated chromosome the trough points form a single axis of symmetry. If a chromosome is sufficiently bent, then an external axis of symmetry is

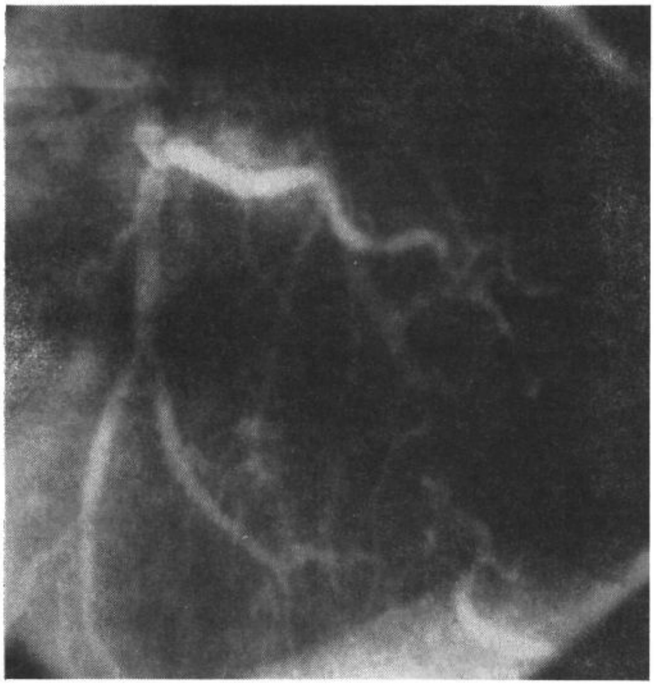

Figure 8. Coronary Angiogram.

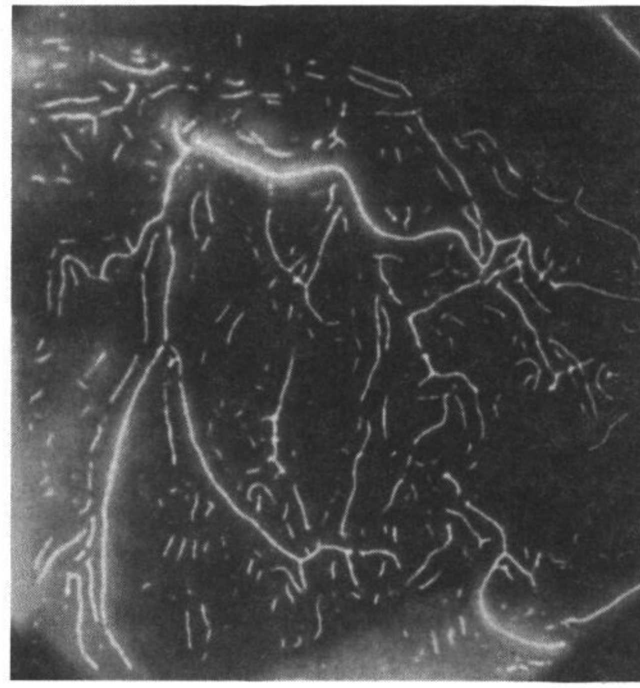

Figure 9. Coronary Angiogram with local axes of symmetry.

generated. If two chromosomes are separated by a distance equal to or less than the scale at which the COG filter was operated then an axis of symmetry appears between them. The trough points persist as the neighbourhood size of the COG filter increases but their position may be altered by the inclusion of other objects or features within the neighbourhood.

Figure 8 shows the blood vessels of the heart as visualised in a coronary angiogram. This represents a more significant challenge because of the very low signal-to-noise ratio. Figure 9 shows the extracted cues superimposed on the original image.

Reliability

We have tested the performance of our method and compared it with two others. Experiments were performed using a test image consisting of a dark bar on a light background with added gaussian noise. The position of the true symmetry axis was known by definition.

The tests compared the COG method with the MMMAT and with locating ridges and troughs directly in a smoothed source image. The method used to calculate **I** in the COG filter is mathematically equivalent to taking the first derivative with respect to x of the source image convolved with a two dimensional Gaussian (and similarly for **J**). The points in the COG image with local grey-level symmetry therefore correspond to local maxima and minima in the smoothed source image and an alternative approach would be to search directly for ridges and troughs in a smoothed image. The COG filter was run at an appropriate scale for the test object. For the direct trough/ridge finding method, gaussian smoothing with the same standard deviation as that used in the COG filter was applied.

All three methods involve a trough/ridge detector and for consistency the operator described in this paper was used in each case. The value of local curvature (the quadratic coefficient c) obtained by the trough/ridge detector for each image point was recorded so that the degree of separation of the signal and background distributions could be assessed.

Results were obtained using a number of different test images. We quote those for a 200*200 pixel image containing a vertical bar with a contrast of 20 grey-levels from the background, width 21 pixels and added noise of standard deviation 3. The accumulated results from 5 such images are shown in Table 1. For each method the distribution of grey-level curvature value obtained by the trough detector is tabulated for both the true axis points and background.

The results given are typical and show that, at this signal to noise ratio, the COG method is effectively 'perfect' with a clear separation between the local curvature distributions for axis and background. The direct method leads to very considerable overlap of the two distributions. The MMMAT method is better than the direct method but still leads to significant overlap. It is important to note that even where there is a very small probability that a background point will have a curvature value which overlaps the distribution of values for axis points (and thus may lead to false positive errors) the errors introduced may be significant, since it is normally the case that there are very many more background points than true axis points.

Table 1. Distribution of local grey-level curvature responses (at an arbitrary scale) obtained on test images for three methods of locating local axes of symmetry.

Curvature	COG Method		Direct Method		MMMAT Method	
	Axis	Background	Axis	Background	Axis	Background
0	0	193008	578	175669	0	136637
1	0	1452	116	2302	0	32483
2	0	437	68	682	0	25306
3	0	103	100	521	0	9806
4	0	21	47	259	0	2296
5	0	9	41	98	4	423
6	0	0	0	19	6	59
7	0	0	0	0	17	14
8	0	0	0	0	30	0
9	0	0	0	0	66	0
10	3	0	0	0	122	0
11	32	0	0	0	150	0
12	33	0	0	0	164	0
13	123	0	0	0	131	0
14	635	0	0	0	113	0
15	124	0	0	0	109	0
16	38	0	0	0	0	0
17	2	0	0	0	0	0
Total	990	195030	950	179550	912	207024

Speed

The COG filter has been implemented in microcode on both a Magiscan 2 (Joyce Loebl Ltd) and an IPB 3000 (Wolfson Image Analysis Unit) attached to a Sun 3/160 (Sun Microsystems Inc) workstation and so runs fairly efficiently. The actual speed with which the filtered image is produced depends upon the filter halfwidth. With a halfwidth of 10 pixels the COG filter takes 20 seconds to operate over an image 256*256 pixels and 6 bits deep.

The results for the trough detector have been obtained using an implementation in PASCAL which takes 3.5 minutes to operate over an image 256*256 pixels. We have recently implemented a microcoded version which takes approximately 10 seconds to operate over the same image. Both the COG filter and the trough detection algorithm could easily be performed in real-time on very simple special-purpose hardware.

DISCUSSION

The COG method generates local axes of symmetry which form good object cues for model instantiation. The method is fast and is robust in the presence of noise. The local axes of symmetry have been used as object cues for model instantiation as described elsewhere in these proceedings [7]. It is a straightforward matter to extend the COG method to label the axes of symmetry as ridges or troughs.

The trough detector described here works well and it is computationally less complex than methods involving fitting a two dimensional surface at each point in the image such as that described by Haralick [5] which also involves the use of arbitrary parameters.

One advantage of the COG method which we have not yet exploited is its scale dependence. Further work is necessary to look at combining cues generated at different scales. We also want to look at linking axis fragments. We intend to apply an improvement of the good continuation concept described by Dixon [8] for asbestos fibre counting.

REFERENCES

1. **Blum, H.** 'Biological Shape and Visual Science (Part 1)' *J. Theor. Biol.* Vol 38 (1973).
2. **Brady, M. and Asada, H.** 'Smoothed Local Symmetries and their Implementation' *MIT AI Memo AIM–757* (1984).
3. **Peleg, S. and Rosenfeld, A.** 'A Min–Max Medial Axis Transform' *IEEE PAMI* Vol 3 No 2 (1981).
4. **Pearson, D.E. and Robinson, J.A.** 'Visual Communication at Very Low Data Rates' *Proc IEEE* Vol 73 No 4 (1985).
5. **Haralick, R.M.** 'Ridges and Valleys on Digital Images' *Comp Vision, Graphics and Image Processing* Vol 22 (1983).
6. **Dixon, R.N. and Taylor, C.J.** 'Automated Asbestos Fibre Counting' in *Machine–Aided Image Analysis* IOP (1978).
7. **Cooper, D. Bryson, N. and Taylor C.J.** 'An Object Location Strategy using Shape and Grey Level Models' *Proceedings of the Alvey Vision Conference* (1988).
8. **Dixon, R.N.** 'Aspects of Picture Processing: Hardware and Software for General–Purpose Image Analysis' *Phd Thesis*, University of Manchester (1980).