**DICTA-91 Digital Image Computing: Techniques and Applications**
Royal Exhibition Building Conference Centre, Melbourne 4-6 December 1991
ed. A.J. Maeder and B.M. Jenkins *Australian Pattern Recognition Society* ISBN 0 646 07338 9

# A MULTISTAGE ELLIPSE AND LINE DETECTION ALGORITHM

Geoff West and Paul Rosin

School of Computing,
Curtin University of Technology,
GPO U1987, Perth 6000, WA

## ABSTRACT

This paper describes an algorithm for the detection of ellipses and lines in image edge data. Connected edge pixels are transformed into polygonal approximations by a two stage algorithm. Then a second two stage algorithm replaces combinations of lines by ellipses if the ellipse fit is better.

## 1. INTRODUCTION

In computer vision the extraction of meaningful features from images is an important technique. The most popular approach is based on edge detection. For model based object recognition, edges must be represented in a more manageable form than simply pixels. The type of representation required is highly application dependent but is typically based on a combination of straight line approximations and higher order curves such as arcs, conic sections, spline and curvature primitives.

A number of techniques have been proposed for determining polygonal approximations [2],[4]. However it is only recently that attention has been concentrated on the extraction of higher order representations because of the increased number of parameters or degrees of freedom and the ill-conditioned nature of the problem. However a number of important advances in this area have occurred [6],[3].

In this paper a technique is described for generating a higher order description by segmenting the edge data into combinations of ellipses and lines. There are four stages used: (1) lines are fitted to connected lists of edge pixels using Lowe's technique [1], (2) lines are grown by combining adjacent lines, (3) ellipses replace lines and finally (4) ellipses are grown by combining with adjacent lines and/or ellipses. In all stages replacement occurs only if the resultant fit is better. The concept of better fit is that suggested by Lowe which is termed a measure of significance. Significance is the maximum error between the fitted representation and the data (a line fitted to pixels or an ellipse fitted to lines) normalised by the length of the representation. The significance is a scale invariant measure which allows the replacing of (i) pixels by a line, (ii) combinations of lines by a line, (iii) combinations of lines by ellipses and (iv) combinations of lines and ellipses by ellipses. The same measure of significance is used in all four stages removing the requirement for any thresholds.

Previously published results [6] have demonstrated the utility of ellipse and line detection based on stages (1) and (3) of the technique described above. In this paper an improved version of the line and ellipse fitting algorithms is described which overcomes the disadvantages of using a binary search tree for stages (1) and (3) by adding new stages (2) and (4). It is shown that the addition of these stages improves the performance of the algorithm. In addition the improvements have been added to the algorithm for detecting arcs and lines (the LAD algorithm [9]).

## 2. LINE FITTING

### 2.1 Stage (1): binary search tree

The line fitting technique used is the familiar recursive binary tree search. Each curve is hypothesised to be a straight line, figure 1, and segmented at the point of maximum deviation from the curve to the straight line.

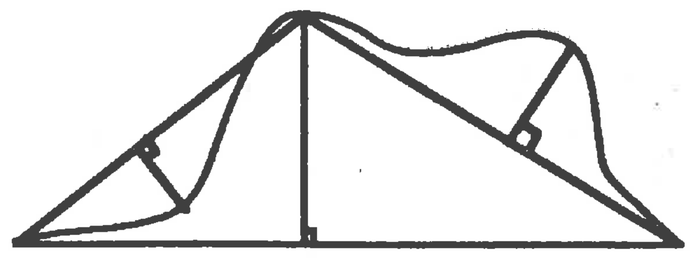
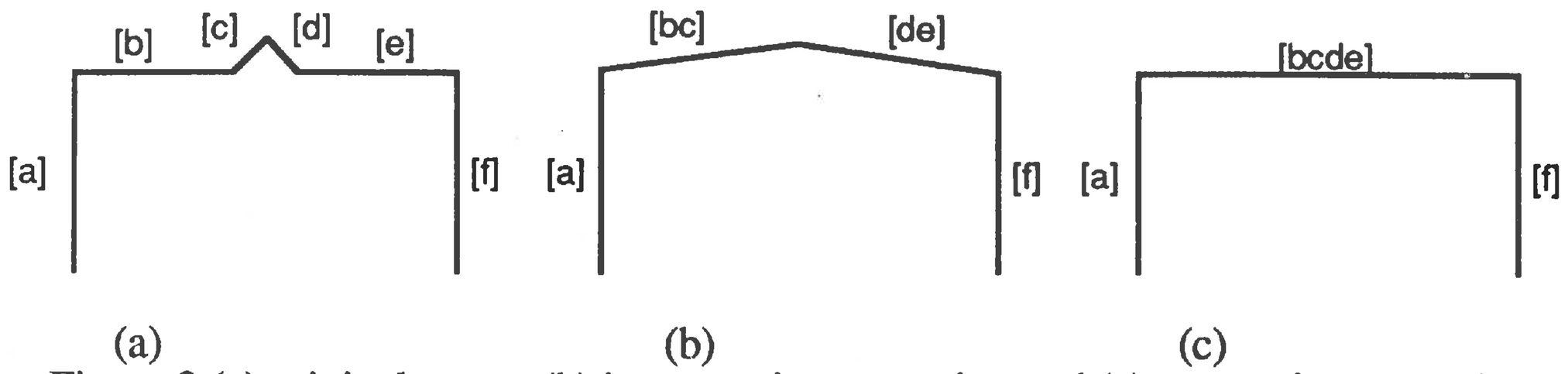


Figure 1 Segmentation of curves.

The process is then repeated for each of the two curves recursively. When the bottom of the tree is reached and the curve cannot be subdivided anymore, tail recursion is used to combine together those lines for which the combined line is a better representation than the other lines. This algorithm has been described in detail elsewhere [9]. The important points to note are (1) the use of points of maximum deviation for breakpoints and (2) the binary search tree preventing all adjacent combinations of lines from being compared. Consider the curve of figure 2a, the result of the algorithm is shown in 2b whereas the intuitively correct result is that in 2c. Figure 3 shows the interpretation tree for this curve.




(a) (b) (c)

Figure 2 (a) original curve, (b) incorrect interpretation and (c) correct interpretation.

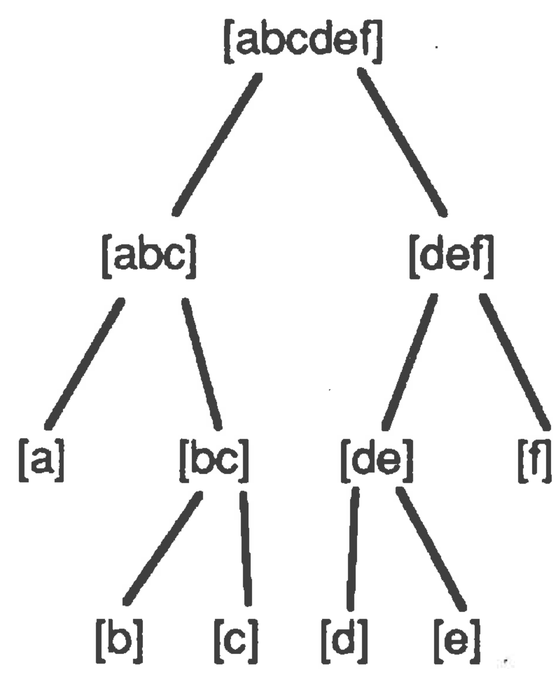



Figure 3 The interpretation tree for figure 2(b)

The tree shows that the curve has been segmented into the lines a,b,c,d,e and f, some of which have been combined under tail recursion to give the result of figure 2b.

## 2.2 Stage (2): combining adjacent lines

To overcome deficiencies such as shown in figure 2 a second stage is used that compares all adjacent lines and combines them if possible. Consider the example of figure 1 again. Figure 3 shows the tree that results from processing this curve.

Note the two lines that replace parts of the original curve [bc] and [de] are not compared at all in the tree, hence the incorrect result. The second stage takes the representations [a], [bc], [de] and [f] and computes the goodness of fit for combinations of a selected representation and its neighbours. Possible representations centred on [bc] when compared and possibly combined with its neighbours are: [abcde], [abc]&[de], [a]&[bcde], and [a]&[bc]&[de] (unchanged) where [AB] indicates one representation made up of previous representations [A] and [B]. Note that [abc] has already been tested as [a] and [bc] are adjacent in the tree. The new representation is the one that gives the lowest significance, the same measure used for the first stage. By iterating over the whole curve until no

further improvements can be made, all adjacent combinations are tested. Using the second stage on the example of figure 2b should result in representations [bc] and [de] being combined resulting in the result of figure 2c - the intuitively correct result. This is because the combination of [bc] and [de] should result in a lower significance than [bc] and [de] individually.

For part of the curve of figure 2b, segment [bc] and [de] each have a significance of approximately 1/6 and segment [bcde] has a significance of 1/12. Hence [bcde] is a better representation that [bc] or [de]. For a number of real images, table 1 shows results for the total number of lines that result from the original and improved line fitting algorithm. Each of these images contain a number of generalised cylinders. ICCV32 is the image for which results have previously been presented for line and ellipse fitting [6].

| | Input data | After 1st stage | After 2nd stage |
|---|---|---|---|
| Image name | Number of pixels | Number of lines | Number of lines |
| mugs6 | 21823 | 1267 | 1172 |
| iccv32 | 14764 | 890 | 815 |
| can3 | 14061 | 1103 | 1019 |
| cup3 | 8993 | 542 | 496 |

Table 1 Results for line fitting.

Stage (2) has improved the results in all three images reducing the total number of lines. The use of significance means that the algorithm favours lines that are more significant than others. Note a weighting factor can be used to force the algorithm to favour new combinations i.e. longer lines. Weighting the new combinations by a factor less than 1.0 will give the new line an artificially lower significance forcing the replacement by a longer line.

Figure 4 shows results for the image iccv32. For the original image of figure 4a, figure 4b and 4c show the output of the old and new algorithms respectively. The differences are shown in figures 4d and 4e. Figure 4e shows the lines that replace the lines shown in figure 4d. Note that many old lines have been replaced by fewer longer new lines.

## 3. ELLIPSE FITTING

### 3.1 Stage(3): binary search tree

The basic ellipse fitting algorithm has been presented elsewhere [6]. However a short description is required for completeness. The result of the line fitting algorithm is used as the input of the ellipse fitting algorithm which is, like the line fitting algorithm, based on a binary search. Figure 5a shows the output of line fitting for a curve consisting of two straight lines and one ellipse. In the original algorithm, an iterative Kalman filter was used to fit a conic section to the endpoints of the curve. This guarantied that an ellipse would be fitted whatever the shape of the curve. A standard least mean square conic fitting algorithm is now used. Although this generates hyperbola and parabola, it generates ellipses if the curve is an ellipse. Non-ellipse fits are ignored by giving the result a high significance value so it is always replaced. Replacing the Kalman filter has not altered the results significantly but the algorithm is now faster. Then the list of lines is split into two lists at the vertex of maximum deviation from the ellipse. The algorithm is then repeated for the two lists. Figure 5b shows the resulting interpretation after ellipse fitting which is not correct. The single elliptic arc has been segmented into two elliptic arcs. The intuitively correct result is shown in figure 5c. Figure 6 shows the interpretation tree for figure 5b. Note that the two elliptic arcs [bcd] and [efg] do not get compared in the tree.

Figure 4a Original image ICCV32.

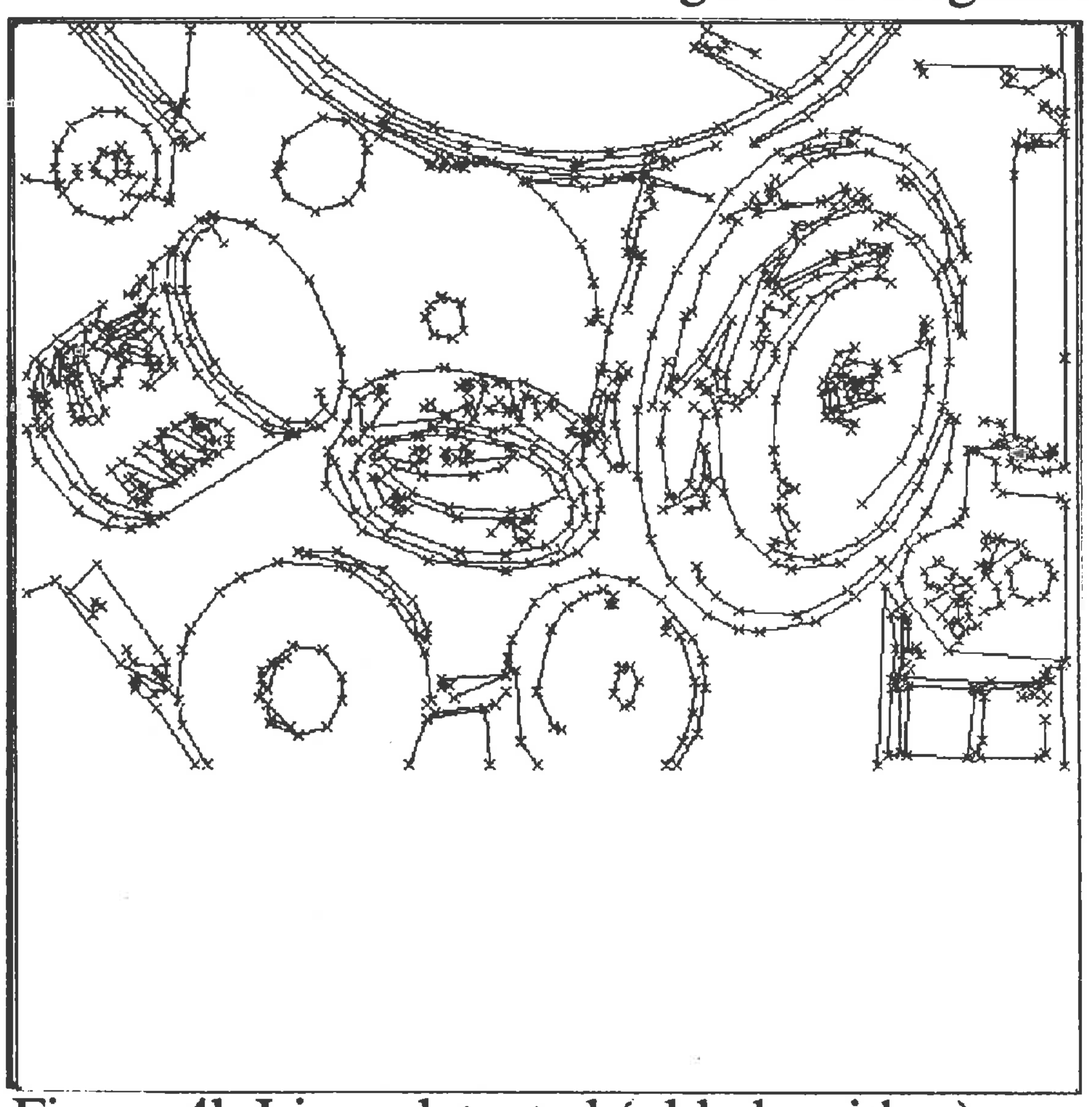


Figure 4b Lines detected (old algorithm).

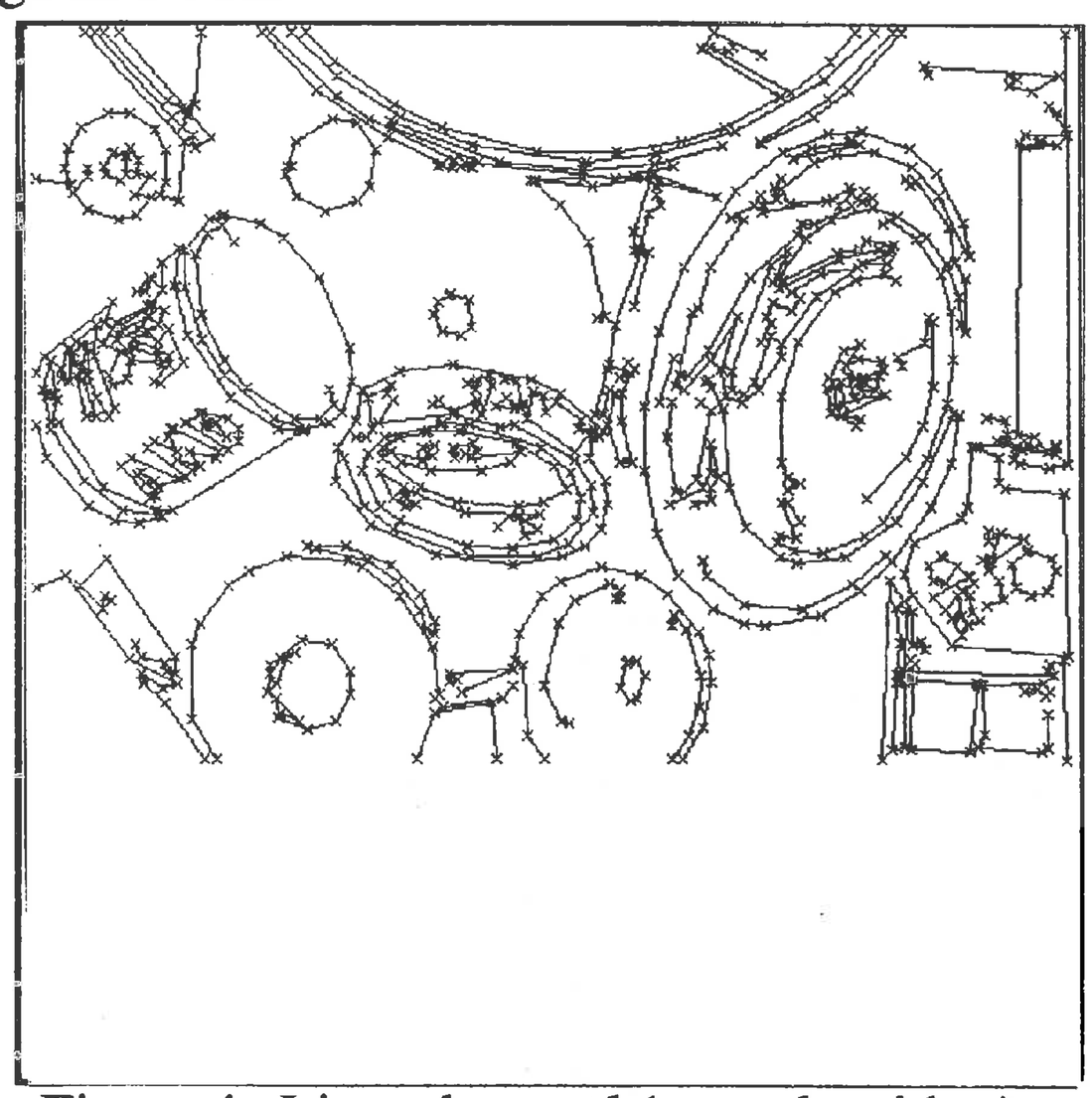


Figure 4c Lines detected (new algorithm).

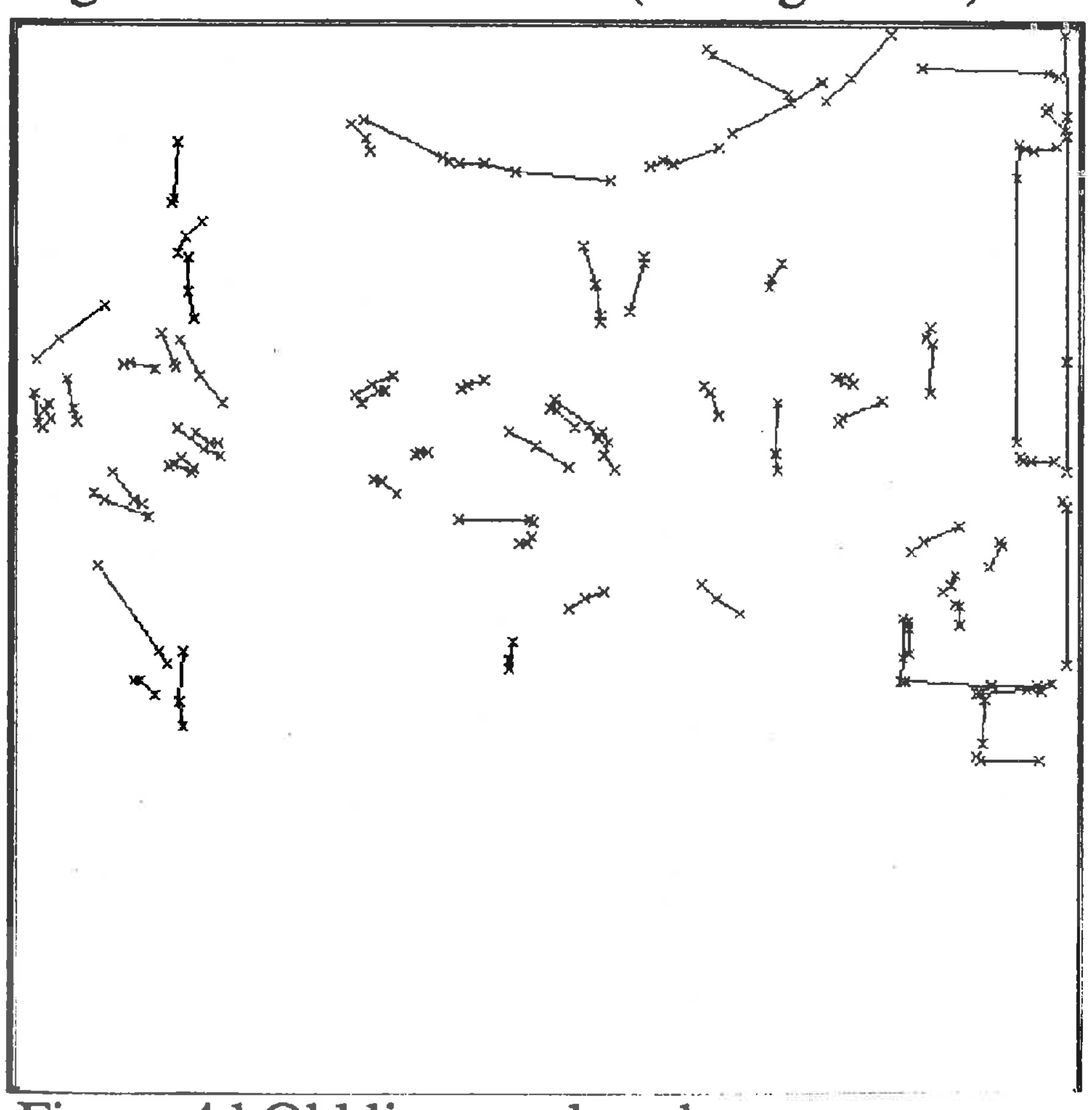


Figure 4d Old lines replaced.

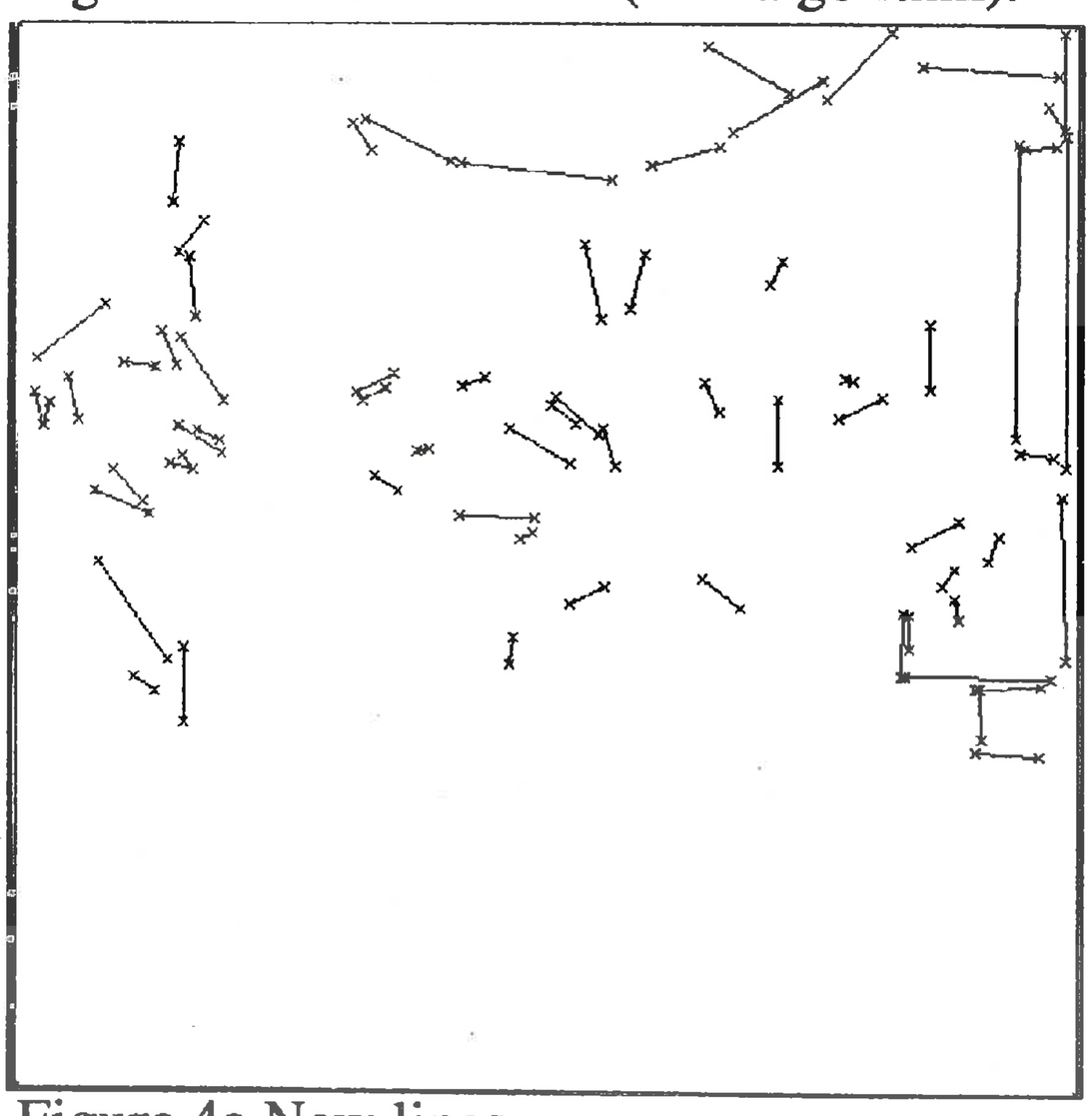


Figure 4e New lines.

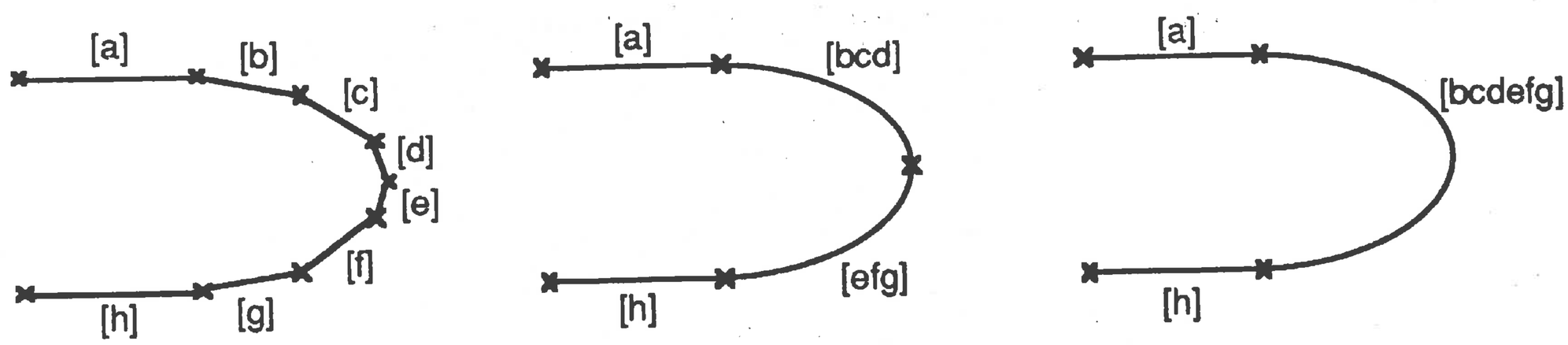



Figure 5. (a) result of line fitting, (b) incorrect result, (c) correct result.

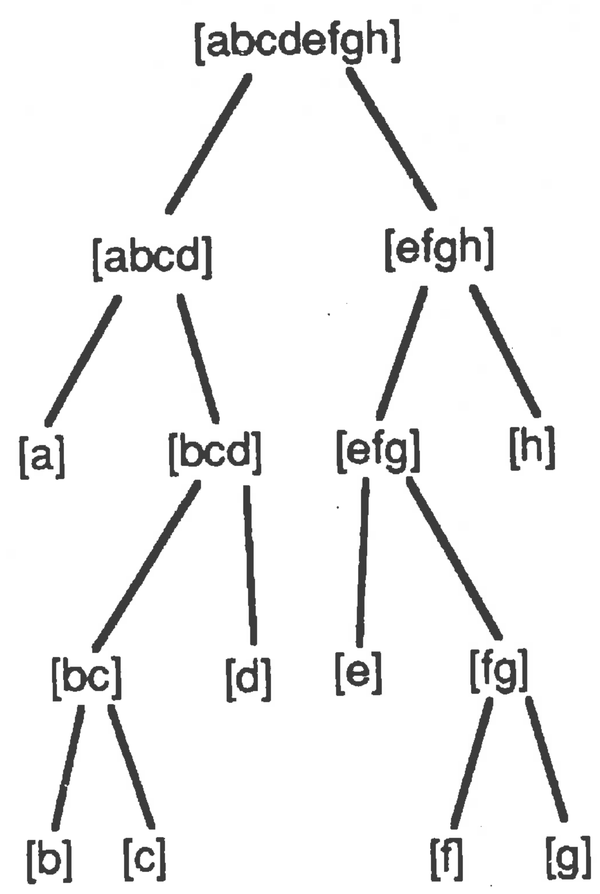



Figure 6. Interpretation tree for figure 5b.

When the length of a list of lines reduces to only 5 lines (or 6 vertices) the ellipse fitting is halted and tail recursion used to choose the combination of lines and ellipses that best describe the list of lines in the sense of significance. At each node in the tree the single ellipse describing the list of lines is chosen if it has a better significance than any of the representations below it in the tree.

A number of limitations occur with the algorithm which can be minimised by the following techniques.

### 3.2 The use of soft breakpoints

Fitting ellipses to line data is more efficient that fitting to pixel data because of the reduction in the number of points for ellipse fitting. However this prevents attempts to fit an ellipse to less that 5 lines (6 vertices) since the fit will be underdetermined. This leads to poor results as ellipses may be missed. It may be that an ellipse would be a better fit to the pixel data than the line. To overcome the problem of not being able to fit an ellipse to less that 5 lines, the concept of soft breakpoints (edge pixel coordinates) is introduced. Consider the hypothesis that an ellipse fits the pixel data better than one line. To confirm the hypothesis 6 data points are required so 4 soft breaks points taken equidistant along the pixel data are used along with the two vertices of the line. If the ellipse is a good fit to the data, a lower significance value should occur. This is the worst case scenario and the algorithm is adaptive such that softbreak points are only used when less that 5 lines are used. With three lines only four vertices are available so one additional soft break point is needed for each line. For two lines, two soft breakpoints per line are required. As such it a variable resolution technique that adapts to the amount of data present using the minimum required for the fitting so reducing computation. The alternative of always fitting an ellipse directly to the pixel data is rejected because of the increased computation in the least mean square conic fitting.

### 3.3 Stage (4): combining ellipses with adjacent ellipses/lines

To overcome the problem of using a binary tree, combinations of adjacent ellipses and lines are tested in a way similar to that described for stage (2) of line fitting. An ellipse will be combined with one or both of its neighbouring lines/ellipses if the significance is better.

Stage (4) is an iterative process that examines each ellipse and determines if it can be extended by combining it with the adjacent representation (line or ellipse). There are four possible outcomes: (i) do not combine, (ii) combine with the left representation, (iii) combine with the right representation and (iv) combine with both representations. For each of these possibilities, the significance measure (as used in the previous three stages) is determined. The resulting representation chosen is the one that results in lower significances (and hence more accurate fits). Iterating over all the ellipses until no further changes can be made results in ellipses growing until further increase in size results in reduced accuracy of fit.

| | Input data | After 3rd stage | After 3rd stage | After 4th stage | After 4th stage |
|---|---|---|---|---|---|
| Image name | Number of lines | Number of lines | Number of ellipses | Number of lines | Number of ellipses |
| mugs6 | 1172 | 505 | 144 | 456 | 139 |
| iccv32 | 815 | 462 | 69 | 442 | 65 |
| can3 | 1019 | 585 | 112 | 561 | 111 |
| cup3 | 496 | 290 | 52 | 274 | 50 |

Table 2 Results for ellipse fitting.

The results of table 2 show that the additional stage of ellipse growing results in reduced numbers of ellipses and straight lines. Longer ellipses are being generated by combining with lines and with some other ellipses. It is interesting to note that there is an appreciable reduction in the number of lines which implies the use of the point of maximum deviation is not an optimal segmentation method for ellipses. However the shortcomings are reduced by the use of the new stage. The effect of the new algorithm is shown in figure 7 for image iccv32. Figure 7a and 7b show the results of the old and improved algorithm respectively. Figures 7c and 7d show the differences between the two algorithms Note that in most cases one ellipse has been grown by combining it with lines. However in some cases, an ellipse has been grown by combining with ellipses and lines.

## 4. RESULTS

To further demonstrate the performance of the new algorithm, results for a number of images are shown in table 3. The results of the new and old algorithms are shown in terms of numbers of lines and ellipses after the line fitting and ellipse fitting stages.

| | Two stage algorithm | | | Four stage algorithm | | |
|---|---|---|---|---|---|---|
| | After stage 1 | After stage 2 | After stage 2 | After stages 1 and 2 | After stages 3 and 4 | After stages 3 and 4 |
| Image name | Number of lines | Number of lines | Number of ellipses | Number of lines | Number of lines | Number of ellipses |
| mugs6 | 1267 | 633 | 128 | 1172 | 456 | 139 |
| cyl21 | 1282 | 925 | 78 | 1185 | 633 | 135 |
| iccv32 | 890 | 574 | 53 | 815 | 442 | 65 |
| can3 | 1103 | 789 | 59 | 1019 | 561 | 111 |
| cup3 | 542 | 377 | 32 | 496 | 274 | 50 |

Table 3 Results for the old two stage algorithm and the new four stage algorithm.

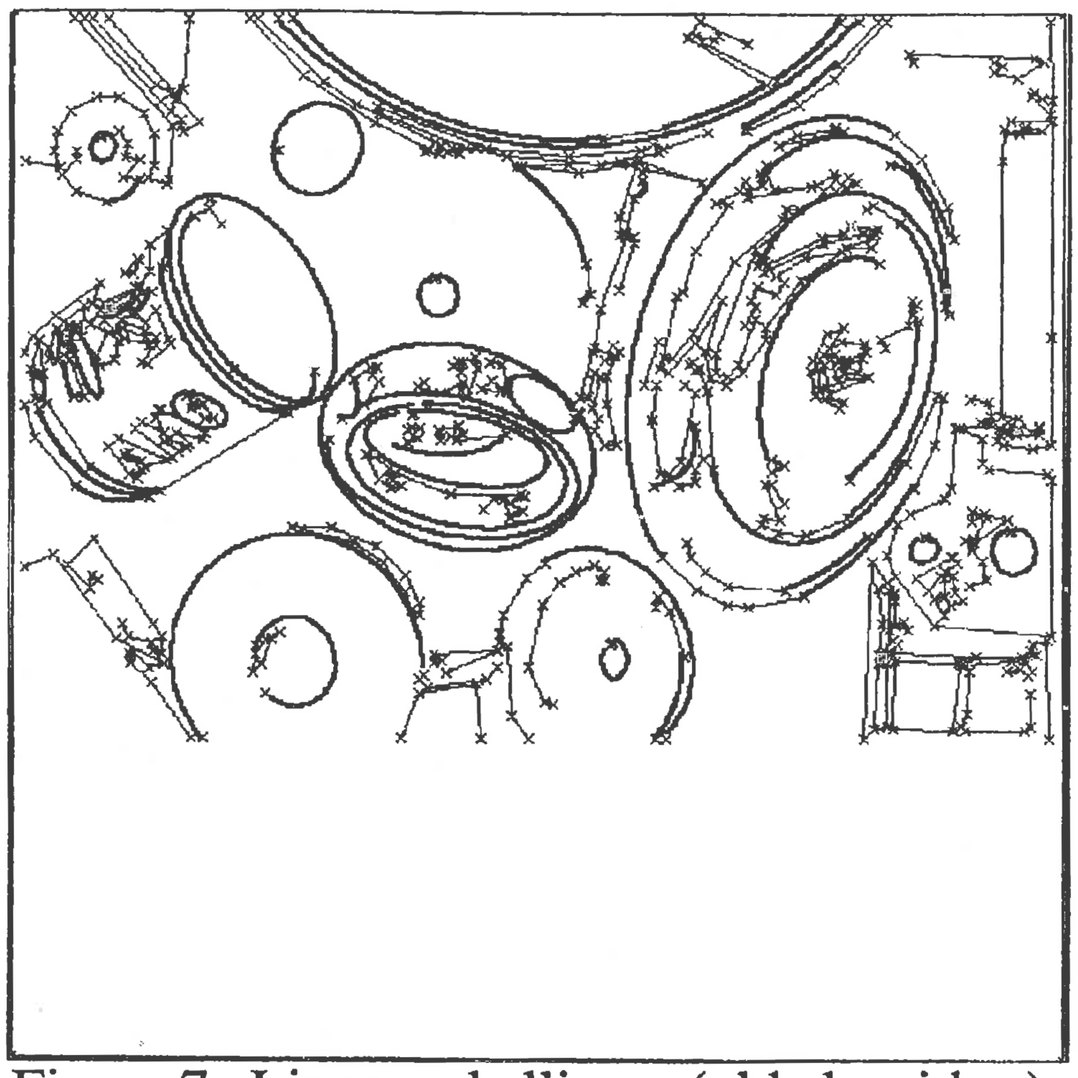


Figure 7a Lines and ellipses (old algorithm).

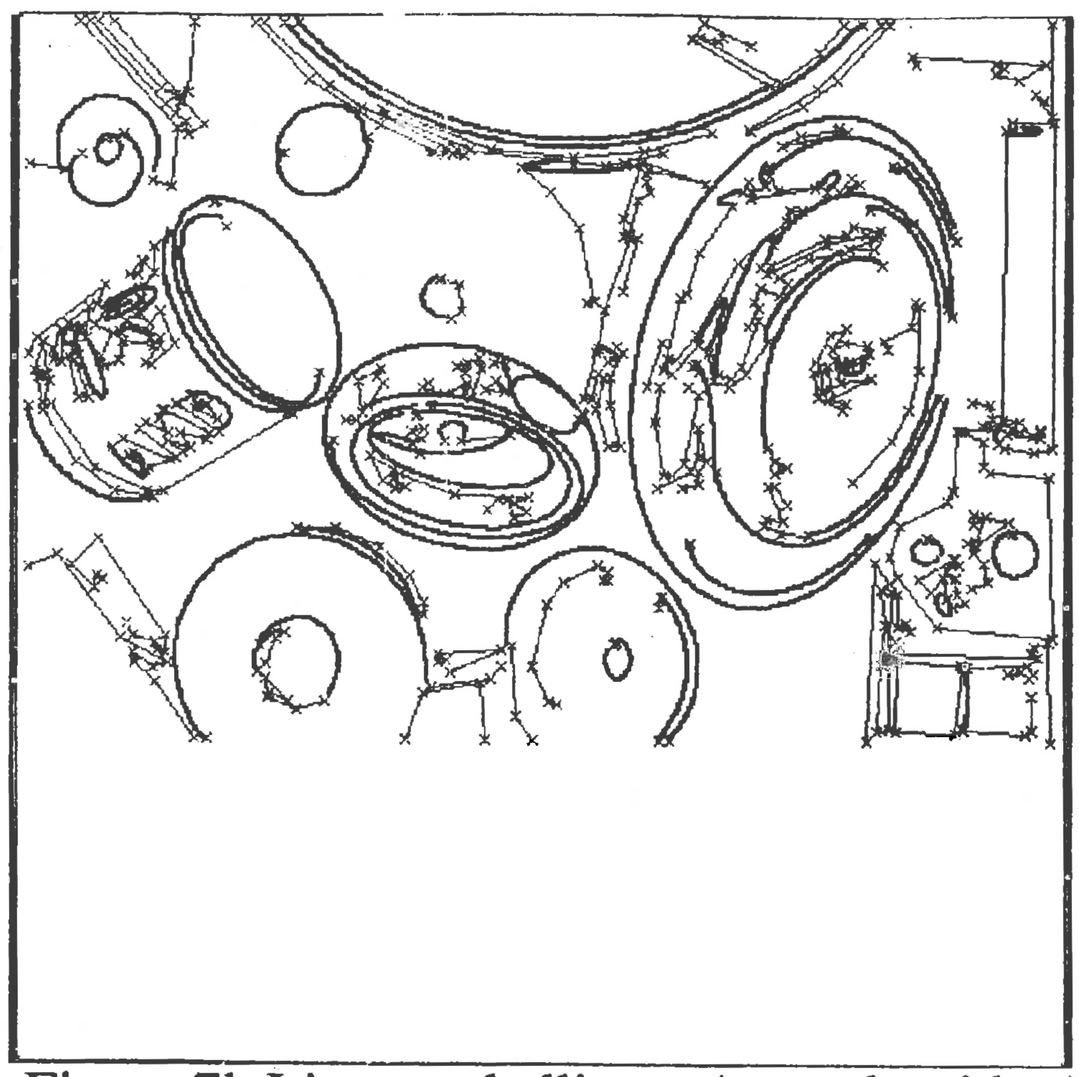


Figure 7b Lines and ellipses (new algorithm).

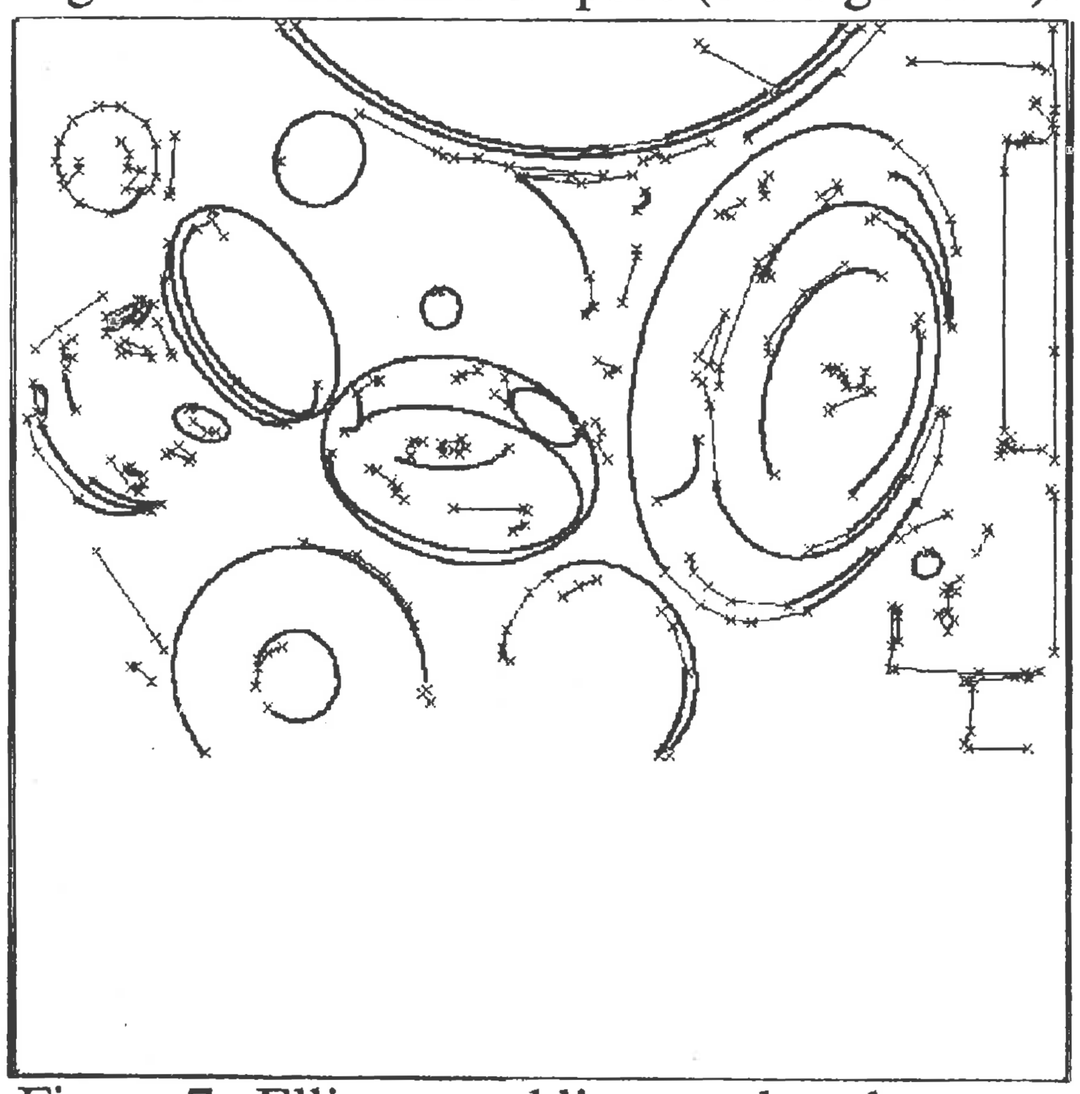


Figure 7c Ellipses and lines replaced.

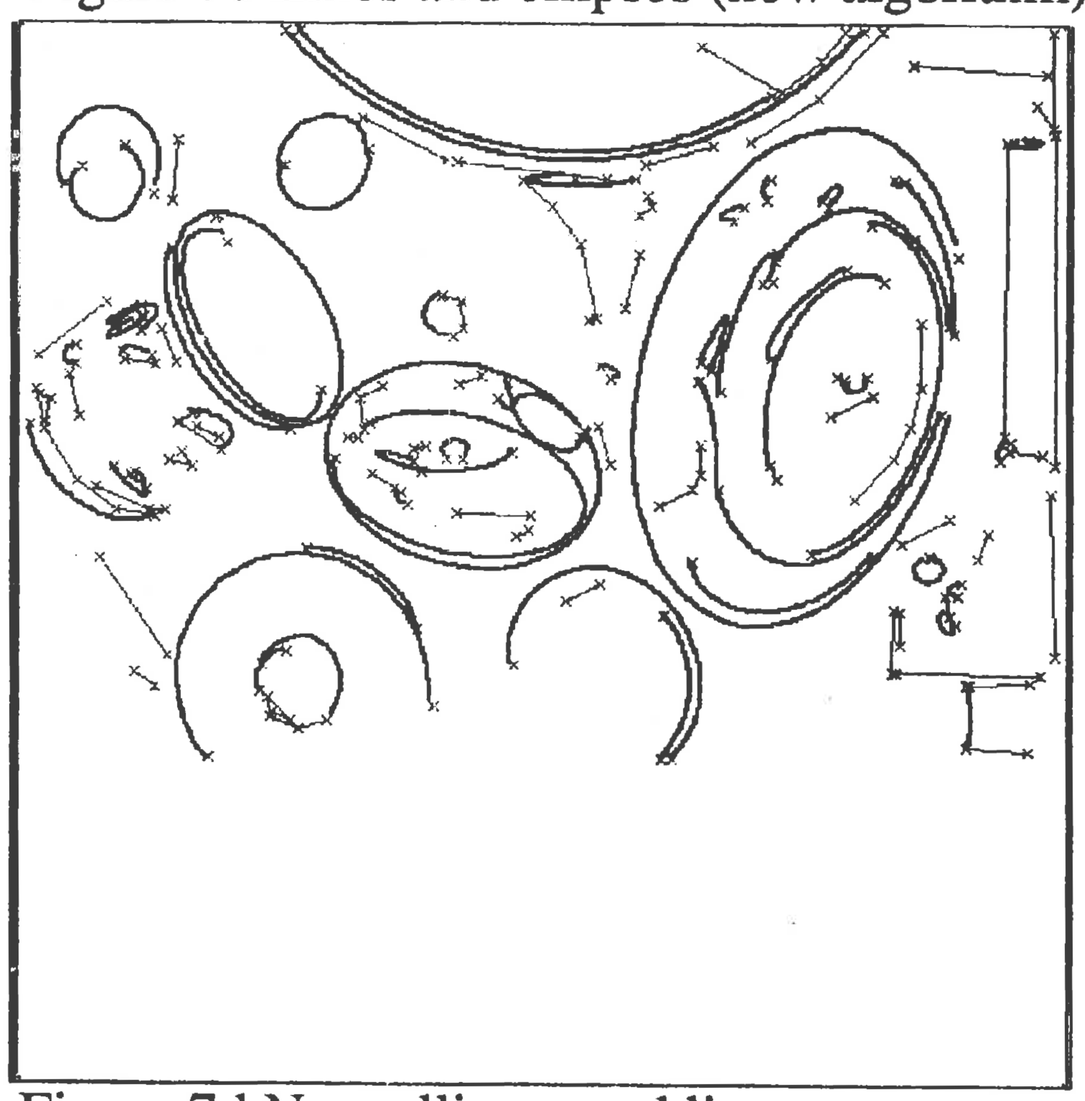


Figure 7d New ellipses and lines.

For all the images, the number of lines has been reduced and the number of ellipses has increased. This is expected because more ellipses are being detected at the expense of lines. It is difficult to quantify the performance based on correct and incorrect classification (as for hypothesis testing). However for each image nearly all the true ellipses were detected and only a small number of false alarms generated. Therefore a small number of line combinations were incorrectly classified as ellipses and a small number of ellipses were incorrectly classified as lines.

## 5. CONCLUSION

The new four stage algorithm shows a significant improvement in performance over the original two stage algorithm [6] for a small increase in computation. There are two reasons for the improvement. The first reason is the use of soft breakpoints meaning that ellipses are adaptively fitted to what can be regarded as the minimum number of points. For a curve consisting of a large number of lines, just the vertices are used. For a part of the curve consisting of less than 5 lines, other points are used to allow an ellipse to be fitted in the overdetermined sense. The effect of soft breakpoints is to enable ellipses to be fitted to more parts of the data than before resulting in more ellipses being detected. The second reason is the use of the extra stages to overcome the shortcomings of using a binary search tree. Ellipses already detected can be extended by combining with other ellipses and/or lines.

The algorithm is made up of a number of stages that can be regarded as splits and merges. The sequence of operations is split and merge (stage 1), merge (stage 2), split and merge (stage 3) and merge (stage 4). Each split uses global information to determine breakpoints and each merge uses local information to remove breakpoints.

The improvements increase the usefulness of the algorithm for many applications which require the detection of ellipses in the 2D image (circles in 3D space) such as detection of generalised cylinders [8] and for bottom up perceptual grouping [7].

## 6. REFERENCES


1. Lowe D.G., "Three dimensional object recognition from single two-dimensional images", Artificial Intelligences, Vol.31, 1987, pp 335-395.
2. Pavlidis T and Horowitz S.L., "Segmentation of plane curves", IEEE Trans. Computing, Vol.23, 1974, pp 860-870.
3. Pavlidis T., "Curve fitting with conic splines", ACM Trans. Graphics, Vol.2, 1983, pp.1-32.
4. Ramer U., "An iterative procedure for the polygonal approximation of plane curves", Comp. Graph and Image Proc., Vol.1, 1972, pp.244-256.
5. Rosin P.L. and West G.A.W., "Segmentation of edges into lines and arcs", Image and Vision Computing, Vol.7, 1989, pp 109-114.
6. Rosin P.L. and West G.A.W., "Segmenting curves into elliptic arcs and straight lines", Proc. IEEE 3rd Int Conf. on Comp. Vision, Osaka, Japan, 1990, pp.75-78.
7. Rosin P.L. and West G.A.W., "Perceptual grouping of circular arcs under projection", Proc. 1st British Machine Vision Conf., Oxford, UK, 1990, pp.379-382.
8. Rosin P.L. and West G.A.W., "Extracting surfaces of revolution by perceptual grouping of ellipses", Proc. IEEE Comp. Soc. Conf. on Comp. Vision and Pat. Rec., Lahaina, Maui, USA, 1991, pp.677-678.
9. West G.A.W. and Rosin P.L., "Techniques for segmenting image curves into meaningful descriptions", Pattern Recognition, Vol.24, 1991, pp.643-652.