Segmenting Curves into Elliptic Arcs and Straight Lines

Paul L. Rosin[§] Geoff A.W. West[§]

Department of Computing Science,
Curtin University of Technology,
Perth,
Western Australia.

Abstract

A method is described for segmenting edge data into a combination of straight lines and elliptic arcs. The two-stage process first segments the data into straight line segments. Ellipses are then fitted to the line data. This is much faster than curve fitting directly to pixel data since the lines provide a great reduction in data. Segmentation is performed in the paradigm suggested by Lowe [5]. A measure of significance is defined that produces a scale-invariant description and allows the replacement of sequences of line segments by ellipses without requiring any thresholds. A method for fitting ellipses to arbitrary curves, essential for this algorithm, has been developed, based on an iterative Kalman filter. This is guaranteed to produce an elliptical fit even though the best conic fit may be a hyperbola or parabola.

Introduction

A key problem area in computer vision is the extraction of meaningful features from images with the most popular approach based on edges. To be useful for model matching, edges must be represented in a more manageable form. The type of description required can be very application dependent but is usually based on a combination of straight line approximations and higher order curves such as arcs, conic sections, splines, and curvature primitives.

The majority of segmentation techniques depend on pre-set parameters to determine the accuracy of fit or the scale at which breakpoints are located. Fixed parameters prevent algorithms from being scale invariant; i.e. a pixel string of a particular shape should have the same description whatever its size. Scale invariance requires smaller thresholds and windows for finer scales. An approach has been suggested by Lowe [5] based on replacing fixed parameters by a measure that accommodates scale invariance. This measure normalizes the maximum error by the length of the representation.

§ Both authors were formerly at City University, London, UK.

Numerous techniques have been proposed for generating straight line approximations. However, less time has been devoted to extracting higher order representations because of the increased number of parameters or degrees of freedom and the ill-conditioned nature of the problem. Lowe's technique has previously been extended [10] to deal with circular arcs. This paper describes a further extension for segmenting curves into straight lines and elliptic arcs. Ellipses are a powerful feature for matching since they can be used to determine the 3D orientation of objects containing circular features. Ellipses may occur in 2D images through the projection (either perspective or orthogonal) of 3D circles onto the image plane.

The method has the advantage that a pixel string is segmented into a combination of lines and ellipses without requiring any thresholds, producing an approximately scale invariant description.

Tree Searching

A simple example of tree searching takes a list of edge pixels which is hypothesised by a straight line passing through its end points. The list is segmented into two, and the process is repeated recursively on each of the two lists. The recursive process is halted when the error between the line and the data is less than some threshold. The result of the recursive process is a tree in which the curve description at each level is a finer approximation of the level above.

There are two issues which arise from such a simple algorithm. These are location of breakpoints and choice of threshold. Breakpoints can be located using a number of criteria such as point of maximum deviation or curvature extrema. Our method uses the maximum deviation criterion which has the advantage that no threshold is required. A better criterion than error thresholds, proposed by Lowe [5], is significance. The significance rating for lines is the ratio of the line segment length divided by the maximum deviation of the curve from the straight line segment. This is based on a pseudo-psychological measure of perceptual significance: the longer the line, the greater the maximum deviation tolerated. This favours long lines provided they fit reasonably well. An advantage of this measure is that it is normalised by the line length, and is therefore scale independent. Segmentation is continued until the line length or maximum deviation is less than three. This termination is necessary to stop small perfect fits which would prevent longer, but imperfect, fits. Tail recursion is then used to choose the best line description at each level by comparing significances.

Lowe's algorithm produces a high quality, general purpose polygonal approximation. No arbitrary error threshold is required, instead the most appropriate values are chosen dynamically throughout the procedure. For instance, curves of different sizes give very similar approximations (but at different scales). The use of a fixed threshold would approximate the smaller curves crudely, and larger curves with unnecessary detail.

The techniques of Lowe's algorithm can be applied to subdivide curves into representations other than straight lines by fitting different functions to the curve. For example, West and Rosin [13] fitted circular arcs to the pixel data in place of straight lines using a least-squares circle fitting approach [12], and segmented curves into sequences of arcs.

Alternatively, the Line and Arc Detection Algorithm [10], segments the curve into a sequence of lines and arcs using a two-stage process. For most environments a combined line and arc representation is a more appropriate description than arcs alone. First, curves are segmented into straight line segments. In a similar manner, circular arcs are fitted to sequences of straight lines and segmented by subdividing at the vertex with the maximum deviation from the arc.

The two-stage process has several advantages over fitting higher-order curves directly to pixel data. Arc fitting is much faster since only the vertices are considered as data points, providing a great data reduction over the edge pixels. In addition, the decision to replace lines by arcs needs no arbitrary threshold. The significance of an arc is calculated in a similar manner to the significance of a line, namely the ratio of the maximum deviation divided by the length of the representation, i.e. the arc. Since their significances are commensurate, an arc replaces a sequence of lines and/or arcs if it is more significant than all of them. The replacement of lines by arcs is straightforward since arc breakpoints (and therefore subarc endpoints) correspond to line endpoints. That is, an arc always replaces a sequence of complete line segments.

Although arcs are a powerful cue for recognition, in the 3D world, circular edges are more likely to appear as ellipses or elliptical sections in an image. Therefore, it is desirable to detect ellipses in images. Two approaches have been investigated along the same lines as the arc detection methods described above in which arcs are fitted to pixel or line data. Crucial to this approach is a robust ellipse fitting algorithm.

Ellipse Fitting

The two main approaches to ellipse fitting are the Hough transform and least-squares error fitting. The Hough transform is not appropriate here since it does not produce a unique solution (the search space can be multi-dimensional and sparse), and requires a large number of data points.

To be incorporated into our approach, we require a method that will fit the best ellipse to any data. That is, both image curves that are not ellipses as well as those that are. Since an ellipse is described by the general equation for a conic:

$$ax^2 + bxy + cy^2 + dx + ey + f = 0 \qquad (1)$$

with the constraint $b^2 - 4ac < 0$, the problem of constraining the fitted conic to be elliptical is non-linear. Thus, most researchers using least-squares error techniques for curve fitting fit general conic sections rather than ellipses. However, this often leads to hyperbolic and parabolic arcs being fitted in place of elliptic arcs. Even for approximately elliptical data a hyperbolic or parabolic arc may give a lower fitting error than an elliptic arc. In particular, sections of low curvature data tend to be fitted best by hyperbolic arcs. The main options when fitting conics are:

- All conic types - hyperbolic, parabolic, and elliptic - are allowed [4,8].

- The possibility of non-elliptic fits is not considered [7,9], often because some heuristic has been used to reject non-elliptic data sets.

- All non-elliptic conic fits are rejected [6,11], and are typically replaced by a simpler representation such as circular arcs or straight line segments.

- The conic fit is forced to be an elliptic arc [1-3].

Cooper and Yalabik [2] attempted to force the best fitting conic to be an ellipse by adding a few artificial data points at appropriate locations. This method only worked in some instances. Another approach [3] added to the error function a penalty function $N(b^2 - 4ac)$ for $b^2 - 4ac > 0$, where N is a positive constant, large compared with the term $b^2 - 4ac$. Unfortunately this often tended to force the curve to be a single branch of a hyperbola with the other branch pushed out towards infinity for large N.

The RANSAC method [1] iteratively selected five data points at random until the resulting computed conic through them was an ellipse coming close to a sufficiently large number of points. A standard least-squares conic fit would then be performed on all non-outlying points. If after a fixed number of trials no acceptable fit was found the procedure terminated without fitting an ellipse.

Our approach fits an ellipse by iteratively applying the Kalman filter. It is straight-forward to employ the Kalman filter - a recursive least-squares filter - to fit a conic (e.g. [9]). For an ellipse the value of $a + c$ in equation (1) can never be zero. To maintain the minimal representation of five parameters for an ellipse the arbitrary scale factor is removed by the normalisation $a + c = 1$. Now rectangular hyperbolae cannot be represented, but all ellipses can still be described. An initial estimate is required, whose degree of influence on the final fit is controlled by the covariance matrix. Fitting is initiated with a circle obtained by the least-squares method [12]. A large covariance value will fit the conic unbiased by the circle fit, while a zero covariance matrix forces the conic to the initial circle. Since a circle is a special instance of an ellipse, we can guarantee an ellipse fit simply by reducing the covariance value. The Kalman filter is first run with a large covariance value to provide an unbiased fit. If the fitted conic is an ellipse the procedure terminates successfully. Otherwise, the Kalman filter is iteratively performed, and the fitted conic forced to an ellipse by lowering the covariance value. The optimal covariance value will be the largest value possible that fits an ellipse. This will give the ellipse fit that is the least biased by the initial circle estimate. The covariance value can be approximately determined in a small number of iterations by performing a binary search of the covariance values between the initial large value and zero for the cross-over value where the conic fits change from hyperbolic or parabolic to elliptic. The search is continued until the range of covariance values is smaller than a threshold, and the ellipse fit arising from the lower covariance value of the range is returned.

Figure 1a shows synthetic data consisting of a section of one arm of a hyperbola. In figure 1b the various

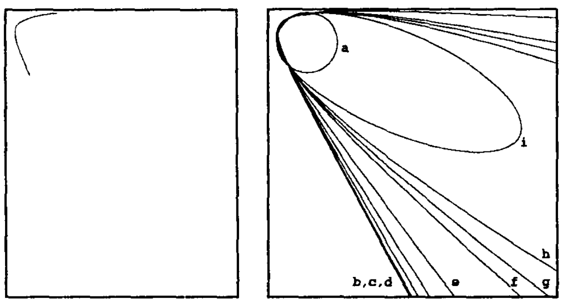

Figure 1 (a) hyperbolic data; (b) sequence of fits to (a).

representations of the data generated by the algorithm are shown. Starting with the initial circle fit (a) the algorithm goes through a number of hyperbolae (b-h) until an ellipse is found (i). This result demonstrates forcing the fit, determined by the Kalman filter, to non-elliptical data to be an ellipse.

Examples

The following examples use edge lists extracted by the Marr-Hildreth edge detector followed by an endpoint linking algorithm. Insignificant edge lists are removed by a threshold on their summed pixel edge strengths.

The image is shown in figure 2a, and the edge data in figure 2b. Figure 2c shows the results of fitting ellipses to the pixel data resulting in a representation consisting purely of ellipses. Most true elliptical features have been represented with reasonable accuracy. For instance the edge of the plate and the rim of the cup. As ellipses are the only representation allowed, straight line segments have been approximated by sequences of elliptical arcs with low curvature. Figure 2d shows the lines extracted by Lowe's algorithm and figure 2e shows the result of fitting ellipses to the line representation. In general the line data has been correctly segmented into ellipses and lines. Large ellipses have been accurately fitted. Small ellipses are typically represented by less lines which gives rise to less accurate ellipse fitting. Some of the obviously elliptical data has been incorrectly classified as lines because there were too few line segments to fit ellipses to. The ellipse fitting requires at least five lines (i.e. six endpoints). With less than five points the fitting process is underconstrained, and for five points there is only a unique, perfect conic fit. Perfect line or ellipse fits are disallowed as they may prevent larger fits which are imperfect, but nevertheless perceptually better. To avoid ellipses being missed due to too few points would require additional points to be inserted. These points could either be interpolated from the line data or be passed on from the line approximation stage as "soft break" points. No obvious straight line data has been misclassified as elliptical. Finally, figure 2f shows the ellipses from figure 2e overlaid onto the original image.

Conclusions

The task of describing edges present in images is an important area of interest and research in the field of computer vision. Many algorithms have been proposed for detecting lines, arcs, and higher order curves. In this paper a method for the detection of lines and ellipses, based on a paradigm proposed by Lowe [5] has been proposed. It has the advantage over other methods in that no thresholds are required. Instead a significance measure is used which is relatively scale invariant and produces perceptually agreeable results. A method for

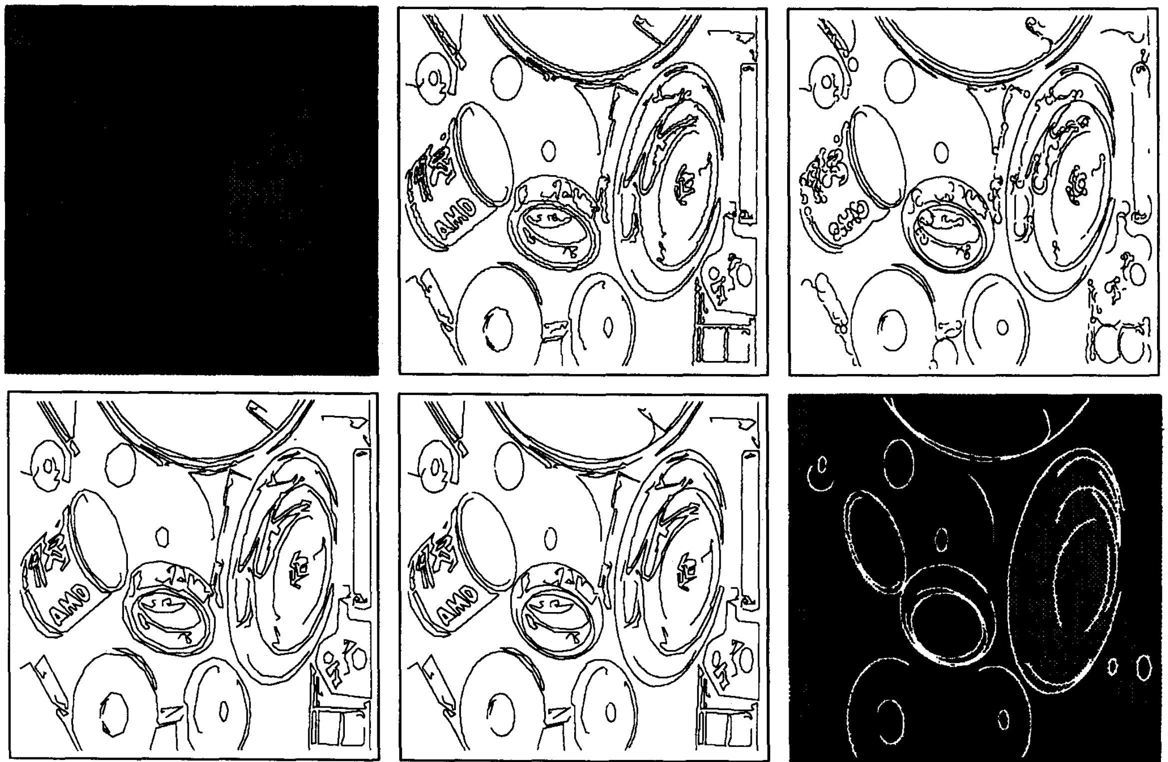

Figure 2 **(a)** original image; **(b)** edge data; **(c)** ellipse fittting to pixel data; **(d)** polygonal approximation; **(e)** ellipse fitting to line data; **(f)** ellipses in (e) superimposed on original image.

fitting ellipses to arbitrary curves, essential for this algorithm, has been developed, based on an iterative Kalman filter. This is guaranteed to produce an elliptical fit even though the best conic fit may be a hyperbola or parabola.

References

1. **Bolles R.C., Fischler M.A.**, "A RANSAC-Based Approach to Model Fitting and its Applications to Finding Cylinders in Range Data", *Proc. 7th, IJCAI*, pp. 637-643, 1981.
2. **Cooper D.B., Yalabik N.**, "On the Cost of Approximating and Recognizing a Noise Perturbed Straight Line or a Quadratic Curve Segment in the Plane", TR. NASA NSG5036/1, Brown Univ., 1975.
3. **Cooper D.B., Yalabik N.**, "On the Computational Cost of Approximating and Recognizing Noise-Perturbed Straight Lines and Quadratic Arcs in the Plane", *IEEE Trans. Computers*, **25**, pp. 1020-1032, 1976.
4. **Liao Y-Z.**, "A Two-Stage Method of Fitting Conic Arcs and Straight-Line Segments to Digitized Contours", *Proc. Pattern Recognition and Image Processing Conf.* pp. 224-229, 1981.
5. **Lowe D.G.**, "3D object recognition from single 2D images", *AI*, **31**, pp. 355-395, 1987.
6. **Nagata T., Tamura H., Ishibashi K.**, "Detection of an Ellipse by Use of a Recursive Least-Squares Estimator", *J. Robotic Sys.*, **2**, pp. 163-177, 1985.
7. **Nakagawa Y., Rosenfeld A.**, "A Note on "Polygonal and Elliptical Approximation of Mechanical Parts", *Pattern Recognition*, **11**, pp. 133-142, 1979.
8. **Pavlidis T.**, "Curve Fitting with Conic Splines", *ACM Trans. Graphics*, **2**, pp. 1-31, 1983.
9. **Porrill J.**, "Fitting ellipses and predicting confidence envelopes using a bias corrected Kalman filter", *Image and Vision Computing*, **8**, pp. 37-41, 1990.
10. **Rosin P.L., West G.A.W.**, "Segmentation of edges into lines and arcs", *Image and Vision Computing*, **7**, pp. 109-114, May 1989.
11. **Shirai Y.**, "Recognition of Real-World Objects using Edge Cue", *Computer Vision Systems*, eds. A.R. Hanson, E.M. Riseman, Academic Press, New York, pp. 353-362, 1978.
12. **Thomas S.M., Chan Y.T.**, "A simple approach for the estimation of circular arc centre and its radius", *CVGIP*, **45**, pp. 362- 370, 1989.
13. **West G.A.W., Rosin P.L.**, "Techniques for segmenting image curves into meaningful descriptions", submitted for publication. Also as Tech. Rep. MIC/MVG/90/04, City University, 1990.