Pattern Recognition, Vol. 23, No. 1/2, pp. 69–79, 1990
Printed in Great Britain.

0031-3203/90 $3.00 + .00
Pergamon Press plc.
Pattern Recognition Society

DYNAMIC TWO-STRIP ALGORITHM IN CURVE FITTING

MAYLOR K. LEUNG and YEE-HONG YANG*
Department of Computational Science, University of Saskatchewan, Saskatoon, Saskatchewan, Canada S7N 0W0

(*Received* 5 *December* 1988; *in revised form* 2 *March* 1989; *received for publication* 23 *March* 1989)

Abstract—In this paper, a new dynamic strip algorithm for fitting a curve with lines is presented. The algorithm first finds the best fitted left-hand side and right-hand side strips at each point on the curve. A figure of merit is computed based on the fitting results. Then, a local maximum detection process is applied to pick out points of high curvature. The aprpoximated curve is one with the high curvature points connected by straight lines. The global structure of a shape, which is insensitive to a scale and orientation changes, is obtained by merging the results from several runs of the algorithm with different minimum strip widths.

Angle measurement Corner Curvature Curve fitting Data compression
Line fitting Polygonal approximation Strip algorithm

1. INTRODUCTION

Line fitting of a plane curve is a process to approximate an arbitrary curve with a sequence of straight lines which satisfy criteria such as minimizing the maximum error between the data points and the fitted lines or requiring the approximated curve to retain the overall shape of the original curve. Such a process is useful in shape representation [8, 10, 11] or in data reduction [14]. Proposed approaches can be classified into two main categories, namely the error tolerance (ET) approach and the feature point (FP) approach.

In the error tolerance approach, the line-fitting problem is formulated as fitting lines to data points such that the perpendicular distance of each point on the curve to the fitted line segment is within a predefined error tolerance. Many algorithms have been proposed in this direction [5, 6, 9, 12, 14–17]. Some aim at minimizing the number of line segments at the expense of time [5, 6, 16] while others aim at minimizing time with less concern on the number of line segments [9, 12, 14, 15, 17]. The time spent in obtaining an optimal solution (e.g. a solution with the least number of line segments) is significant when the number of the input data points goes up [5, 6, 16]. An alternative is to employ non-optimal but fast algorithms which often can provide reasonable reduction comparable to that of the optimal one [9, 12, 15, 17].

The feature point (FP) approach is to use feature points to segment a curve. It was suggested by Attneave [1] that corners or high curvature points of a curve provide important information during the recognition process in the human visual system. Based on this observation, one can argue that a plane curve can be represented by another plane curve and still be recognized to have the same shape by a human observer if the feature points remain invariant. Thus the line fitting problem can be formulated as a feature point detection problem. The approximated curve is one with feature points connected by straight line segments [3, 4, 7]. The study by Blakemore and Over [2] suggests that human perceives side before angle. Based on their observations, the line fitting problem should have the sides determined before the feature points [3].

It is easy to see that the ET approach does not necessarily preserve feature points. For example, points of high curvature in the original curve may not be points of high curvature in the approximated curve. On the other hand, the approximated curve obtained by connecting feature points by straight lines may have large and undesirable deviations from the original curve. The proposed Dynamic Two-Strip algorithm is an attempt to solve these two problems.

The proposed algorithm uses a FP approach. It fits each data point on a curve with two strips, one on the left and one on the right, as the left and right approximated lines. The narrower and longer the strip, the better the approximated line will be. The algorithm is able to derive the best fitted strips by adjusting the orientation and width of the strip dynamically. Based on these two strips, two important properties of the data point can be derived. First, the curvature at the data point can be computed as the angle subtended by these two strips. Then, the figure of merit of the data point as a feature point can be computed. Based on this information, a local maximum detection procedure is employed to pick

* To whom correspondence should be addressed.

out the most representational points. Furthermore, the global feature of a shape which is insensitive to scale or orientation changes can be extracted.

The organization of the paper is as follows. In Section 2 background of the strip algorithm is introduced. In Section 3 the proposed Dynamic Two-Strip algorithm is described in detail. Section 4 contains the procedure for the global structure extraction. Experimental results are shown in Section 5 and conclusions are drawn in Section 6.

2. DYNAMIC STRIP ALGORITHM

Strip algorithms for curve fitting have received much attention recently because of their superior speed performance advantage [14, 9]. Reumann [13] was the first to propose the idea of fitting curve with strips. As shown in Fig. 1, a strip is defined by one critical and two boundary lines. A critical line is defined by two reference points, the first and the second data points (i.e. points 0 and a in Fig. 1) of a curve. Then two boundary lines which are parallel to the critical line and at a distance d from it are defined. The distance d is commonly called the error tolerance. These two boundary lines form a strip to restrict the line fitting process. The curve is then traversed point by point. The process stops and a line segment is generated when the first point which is outside the strip is found (e.g. point e in Fig. 1). A line segment is then defined by the points 0 and c. Point c is used again as the starting point for the next strip fitting mechanism.

One major problem with the strip algorithm is that if the second reference point is positioned in such a way that the third point on the curve is outside the strip, the resulting line segment will then be very short and is often not desirable. An example is shown as strip 1 in Fig. 2. It can be seen in the same figure that strip 2 is a more desirable strip because it contains more data points.

From the above simple observation, Leung and Yang [9] proposed a Dynamic Strip algorithm (DSA) which rotates the strip using the starting point as a pivot. The basic idea is to rotate the strip to enclose as many data points as possible. An example to illustrate the advantage of the Dynamic Strip algorithm can be seen in Fig. 3 where case (a) illustrates the best possible strip without rotation while case (b) illustrates the best possible strip when rotation is allowed. Orientation of the strip is the only parameter to vary in the Dynamic Strip algorithm. In this paper, the Dynamic Two-Strip algorithm includes the width of the strip as a parameter to vary.

3. DYNAMIC TWO-STRIP ALGORITHM

The Dynamic Two-Strip algorithm is a FP approach algorithm but retains the advantages of the ET approach. It has two stages. In the first stage, a generator called the Left–Right Strip Generator (LRSG) is employed to find the best fitted LHS and RHS strips at each data point. In our convention, a point that is traversed $\begin{Bmatrix}\text{before}\\ \text{after}\end{Bmatrix}$ the data point P is said to be on the $\begin{Bmatrix}\text{RHS}\\ \text{LHS}\end{Bmatrix}$ of P respectively. For example, in Fig. 4a point A is on the RHS of point B if the direction of traversal is as shown in the figure. The computed strips are used to compute the figure of merit of the data point. In the second stage, a local maximum detection process is applied to pick out desirable feature points, i.e. points of high curvature. The approximated curve is one with the feature points connected by straight lines. In the following, each stage of the algorithm is described in detail.

3.1. *Left–Right Strip Generator* (LRSG)

LRSG is an extension of the Dynamic Strip algorithm. The strip is allowed to adjust its orientation as well as its width dynamically. To simplify our discussion, we assume our data points are labelled from 0, 1, ..., to $N-1$ and are traversed in either

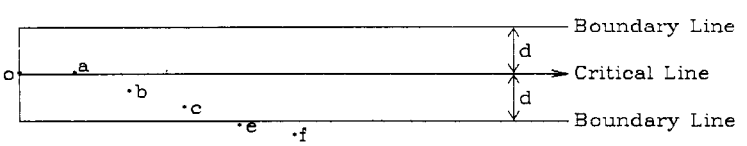


Fig. 1. Definition of a strip.

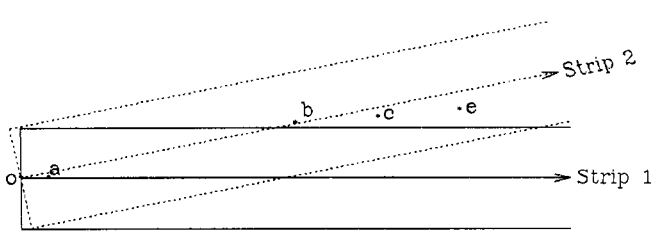


Fig. 2. A badly and a properly chosen strip.

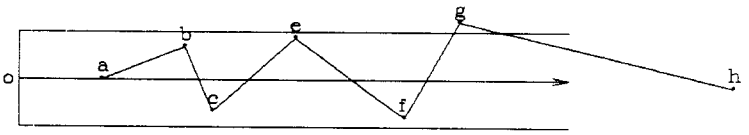


Fig. 3(a). Best fitted strip without rotation.

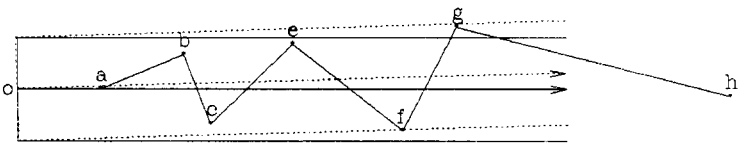
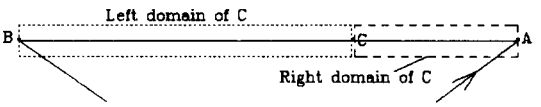
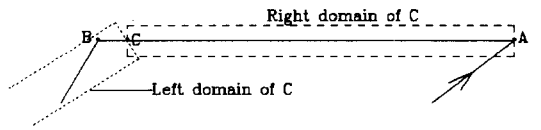


Fig. 3(b). Best fitted strip (in dotted lines) when rotation is allowed.



(a)



(b)

Fig. 4. Examples of two local maximum points (A and B) and one weaker point (C) with its domains.

clockwise or counter-clockwise fashion. Let $\begin{Bmatrix} L_i^{\text{Left}} \\ L_i^{\text{Right}} \end{Bmatrix}$ and $\begin{Bmatrix} W_i^{\text{Left}} \\ W_i^{\text{Right}} \end{Bmatrix}$ be the length and the width of the fitted $\begin{Bmatrix} \text{LHS} \\ \text{RHS} \end{Bmatrix}$ strip at the i^{th} data point. Initially, a strip with the minimum width (i.e. $W_i = w^{\min}$) is used in each direction. When no more data points can be included into the strip, the ratio $\begin{Bmatrix} E_i^{\text{Left}} = L_i^{\text{Left}}/W_i^{\text{Left}} \\ E_i^{\text{Right}} = L_i^{\text{Right}}/W_i^{\text{Right}} \end{Bmatrix}$ is computed. Intuitively, E_i is a measure of the elongatedness of the strip. The longer or narrower the strip, the higher the value of E_i. An enongated strip which is one with large E_i is therefore desirable. The value of the width is then increased by the smallest amount such that the strip fitting operation can resume, i.e. when more data points can be included. This process continues until the maximum allowable width ($W^{\max}$) of the strip is reached. The value of $W^{\max}$ can be set arbitrarily large while the minimum width ($W^{\min}$) cannot be set arbitrary small. This is particularly true if the data are digitized because the length of a strip has an upper bound $L^{\max}$ (bounded by the dimension of the screen) and a lower bound $L^{\min}$ (bounded by the distance between two consecutive vertical or horizontal pixels) but not the width of a strip. The width of a strip does not have a lower bound. If we arbitrarily choose $W^{\min}$ to be less than $1/L^{\max}$, no strip of width ≥ 1 can be chosen since it always gives a smaller E_i. This can be illustrated by considering

$$E_i = \frac{L_i}{W_i}$$

$$E_i' = \frac{L_i'}{W_i'}$$

with $W_i < 1/L^{\max}$ and $1 \leq W_i' < \infty$. In this case, we will have $E_i > L_i \cdot L^{\max}$. Since L_i is bounded from below by 1, E_i would be $> L^{\max}$. On the other hand, E_i' can be at most equal to L_i' with $W_i' = 1$. Since L_i' is bounded from above by $L^{\max}$, E_i would be larger than E_i'. Therefore, under the situation with $W_i < 1/L^{\max}$, no strip of width ≥ 1 will be chosen and none or little data reduction (noise filtering) is done. In practice, data reduction or noise filtering is desirable. In Section 5, a range of the minimum width values is experimented for digitized data.

The result of the above operation is a collection of all the longest possible $\begin{Bmatrix} \text{LHS} \\ \text{RHS} \end{Bmatrix}$ strips with different width at each data point. At each side of the data point, only the strip with the largest E_i is selected.

In Blakemore and Over's study [2], it was suggested that human perceive side before angle. Hence, the LRSG simulates the side detection mechanism. The curvature at a point can then be determined by the angle subtended by the best fitted left and right strips. In order to determine if the i^{th} data point P_i is a feature point, we define a figure of merit (f_i) that measures the worthiness of P_i to be included in the approximation. f_i is defined as:

$$f_i = E_i^{\text{Left}} \cdot S_i^{\theta} \cdot E_i^{\text{Right}}$$

where θ is the angle subtended by the best fitted left and right strips and S_i^{θ} is the angle acuteness measure at point i.

$$S_i^{\theta} = |180^\circ - \theta| \qquad 0 \leq \theta \leq 360^\circ.$$

According to this computation, sharper angles will give a larger value of S_i^{θ}. It can be seen that a sharp angle subtended by long strips will result in a large f_i whereas a blunt angle subtended by short strips will result in a small f_i. The above discussion can be

summarized by the following three steps.

(1) Determine E_i^{Left} and E_i^{Right} for all i.

(2) Determine the angle θ subtended by the left and right strips and also the value of S_i^{θ}.

(3) Determine f_i.

3.2. *Local maximum detection*

The local maximum detection process consists of three stages. First, non-local-maximum points (i.e. points with small f as compared with their neighbours) are eliminated temporarily. The second step is to check if over elimination has occurred. Consequently, some temporarily eliminated points are added back to the result. The final step is to fit narrow strips to the remaining points to eliminate points that align approximately on a straight line. Details of the above steps are described in the following.

(1) Non-local-maximum elimination process: basically, this is a process that allows each data point, P_i, with high f_i to eliminate other points that are in the left and right domain of P_i. A domain is defined by the area or length covered by the best fitted strip of a point. To simplify future discussion, the left and right domains of P_i are denoted by D_i^{Left} and D_i^{Right} respectively. A point Q is in, say, the left domain of P_i is written as:

$$Q \in D_i^{\text{Left}}.$$

An ideal case is shown in Fig. 4 where points A and B are local maxima since all the other points which are between A and B (e.g. C) have strips subtending an angle of approximately 180° (Fig. 4(a)) or they may have strips of wider widths together with wider angles (Fig. 4(b)). In these cases, points between A and B (e.g. point C) are eliminated.

In the algorithm, a point P_j is eliminated if one of the following conditions are satisfied.

(i) There exists m such that

$$P_j \in D_{j-m}^{\text{Left}} \quad \text{and} \quad D_j^{\text{Left}} \subseteq D_{j-m}^{\text{Left}} \quad \text{and} \quad f_j < f_{j-m}.$$

(ii) There exists m such that

$$P_j \in D_{j+m}^{\text{Right}} \quad \text{and} \quad D_j^{\text{Right}} \subseteq D_{j+m}^{\text{Right}} \quad \text{and} \quad f_j < f_{j+m}.$$

In practice, it was found to be difficult to find complete compliance of the domain subsetting conditions, i.e. $D_j^{\text{Left}} \subseteq D_{j-m}^{\text{Left}}$ or $D_j^{\text{Right}} \subseteq D_{j+m}^{\text{Right}}$. Therefore, the conditions are relaxed. We define that the condition $D_j^{\text{Left}} \subseteq D_{j-m}^{\text{Left}}$ is said to hold if half of the left domain of P_{rcrcj}^{**} is covered by D_{j-m}^{Left}. The same can be applied to the right domain.

Another problem that can arise can be understood by considering Fig. 5. In Fig. 5, if the lines AB and FG are long enough, the curve $BCDF$ is comparatively insignificant and can be ignored. On the other hand, if either AB or FG is short, the curve $BCDF$ may be of significance. The classification can be illustrated by considering the angle at point B. At point B, the best fitted right strip would be from point B to A. If the line FG is long, the best fitted left strip of B would be from point B to G. On the other hand, if the line FG is short, the best fitted left strip may be from B to C since a narrower strip, which can give a larger value of f_i^{Left}, can be used. In the first case the angle subtended by the left and right strips of point B is obtuse while in the second case the angle is acute. With this distinction it can be determined whether a portion of the curve should be eliminated. For example (see Fig. 5), if the best fitted left strip of point A is from A to G, the process will examine the points (e.g. B, C, D and F) in between before eliminating any of them. If the lines AB and FG are long enough, all the points in between will have obtuse angles and are eliminated. Otherwise, those which have acute angles will be retained. For example, if point B has an acute angle, only the points between A and B will be eliminated by A. Consequently, the left domain of A is reduced to be from A to B only.

(2) Bridging process: in the first process, weak and insignificant points are eliminated. In practice, some weak points may be of significance. A check is made to determine the possibility of over elimination. In case of over elimination, some temporarily eliminated points are added back to the result. Ideally, neighbouring feature points are supported by domain strips which bridge neighbouring feature points as shown in Fig. 6(a) where the left domain of point A covers point B and the right domain of point B covers point A. If two selected neighbouring points A and B are not bridged, we say over elimination has occurred. Bridges can be broken in the following ways:

(i) $A \notin D_B^{\text{Right}}$ and $B \notin D_A^{\text{Left}}$ (see Fig. 6b)

(ii) $A \notin D_B^{\text{Right}}$ and $B \in D_A^{\text{Left}}$ (see Fig. 6c)

or

$$A \in D_B^{\text{Right}} \quad \text{and} \quad B \notin D_A^{\text{Left}}.$$

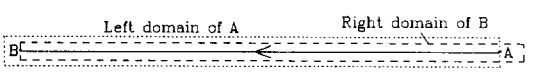


Fig. 6(a). An ideal relationship between two local maximum points (A and B) and their domains.

Fig. 6(b). An example of bridges broken with condition (i).

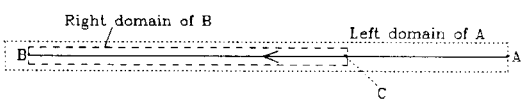


Fig. 6(c). An example of bridge broken with condition (ii).

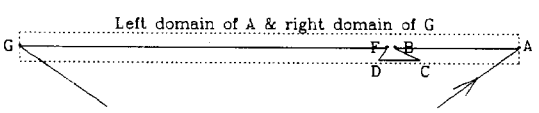


Fig. 5. A possible but undesirable chosen strip (AG).

In either case, additional feature points are sought and the points involved are reexamined iteratively (or recursively) until all neighbouring points are bridged together. The additional feature points are sought at the end of the shortened domains by selecting immediate local maximum points in the neighbourhood. For example, in Fig. 6(b) at the end of the shortened right domain of B, the process looks for the first local maximum starting from point C to B. For the shortened left domain of A, the process starts from point D to A.

In short, the bridging process checks for the termination condition (i.e. all neighbouring points are bridged together) in each iteration. If the condition is satisfied, the process terminates. Otherwise, additional feature points are sought and the iteration continues.

(3) Strip fitting process: it is a data reduction process to fit narrow strips to the remaining points. The reason behind this process is that some consecutive feature points may align approximately on a straight line and it is desirable to eliminate the points in between. For example, if points A, B, C and D are chosen as the feature points after the first two processes as shown in Fig. 6(b), it is desirable to eliminate points C and D and let the more prominent points A and B to represent the curve $ADCB$. In practice, the process first locates the most outstanding points, the local maximum points (e.g. A) among the remaining points, as starting points. Then two narrow strips of fixed widths (one half of the minimum width) is fitted to the LHS and RHS of the data point, eliminating any points within the strips with smaller values of merit (e.g. C and D). The fitting stops whenever the last point can be fitted within the strip is found or a point of a larger value of merit is met. In either case, the last point examined is not eliminated.

4. GLOBAL STRUCTURE EXTRACTION

Current line-fitting algorithms are able to generate coarse or fine line-approximation. However, it is also desirable to have an algorithm that can generate the same approximation irrespective of scale or orientation changes. Unfortunately, current algorithms do not pay much attention to this need. An attempt is made here to extract the global structure of a shape by merging the results from running the algorithm with different minimum width values.

The idea can be illustrated by an example shown in Fig. 7(a) where the asterisks (i.e. points A–O) represent the output from a run with a smaller minimum width value and the crosses represent the output from a run with a larger minimum width value. It is clear from this figure that the crosses can serve as indicators in merging line segments in between. In this case, points B–F and points J and M are eliminated.

On the other hand, indefinite increment of the minimum width will create invalid merging indicators

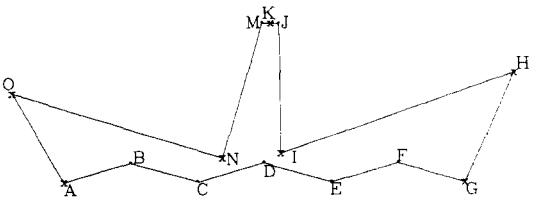


Fig. 7(a). An example of ideal merging indicators.

Fig. 7(b). An example of invalid merging indicators.

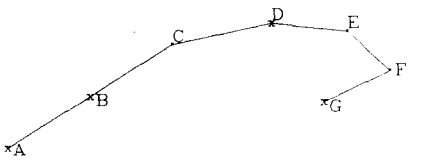


Fig. 7(c). An example of invalid merging indicators.

because large minimum width tends to ignore details of the shape and produces unexpected effect. Three undesirable cases are shown from Figs 7(b) to 7(c). In Fig. 7(b), crosses B and F are merging indicators and an attempt is made to merge the line segments from point B–F. However, it is undesirable to eliminate point C since it is outside the representational area of the line BF. This representational area is the area between two parallel lines; each goes through one end point (i.e. B or F) of the line BF and is perpendicular to BF. This case can be detected by noting that the angle subtended by the lines BF and BC is greater than 90°. Another case, shown in Fig. 7(c), has a cross at B appearing in the middle of the asterisk points A and C. An attempt is made to eliminate any asterisk point between A and B. However, there is no asteriks point between A and B. In this case, cross point B is ignored. The final case, shown again in Fig. 7(c), is the elimination of the asterisk points E and F between the cross points D and G. The result would be a very fat strip fitted from D to G. In this case, a strip is fitted to all the points involved using line DG as the critical line. Then a check is made about the narrowness of the strip. All this is done by comparing the strip elongatedness E against a predetermined $E^{\min}$. No merging is made if E is smaller than $E^{\min}$. Currently, $E^{\min}$ is set between the values of 2.5 and 5.

5. EXPERIMENTAL RESULTS

The algorithm was implemented in the language C on a SUN 4/280 computer running Unix 4.2. Three sets of data were used to demonstrate results obtained by the proposed algorithm.

(1) Hexagon data set: 102 data points from a deformed hexagon are generated with noise as shown in Fig. 8(a).

(2) Circular data set: 352 data points (Fig. 8(b))

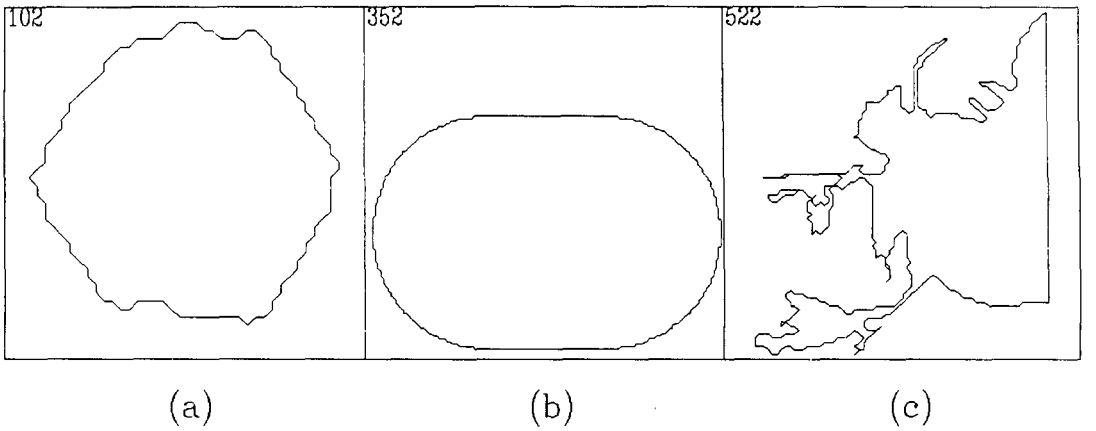


Fig. 8. Input data sets. (a) Hexagon data. (b) Circular data. (c) Shadow data. The number shown at the upper left corner of each figure is the number of data points in the data set.

from two straight lines and two circular arcs are generated to test the algorithm response on circular arcs.

(3) Shadow data set: 522 data points from a segmented shadow region of a digitized picture, Fig. 8(c).

Since the data sets are from different sources, they are of different sizes. The smaller figure is magnified for viewing by showing it in a larger scaled window. For example, the scales of the windows in Figs 8(a), 8(b) and 8(c) are 43 × 43, 140 × 140 and 0.69 × 0.69 respectively.

Two input values were needed to run the program, i.e. (i) the minimum width and (ii) the maximum width. The width value is measured in terms of unit length which is the distance between two consecutive horizontal (or vertical) pixels. Since each diagram in Fig. 8 has a different size, their unit lengths are different too. They are 1, 1 and 0.008333 with respect to Figs 8(a), 8(b) and 8(c).

The results of applying the Dynamic Two-Strip algorithm without the global feature extraction module on the input data with different minimum width values are shown in Figs 9, 10 and 11. In each figure, maximum width value of 10 units are used and the minimum width values of 0.5, 1, 1.5, 2, 4, 6, 8 and 10 are applied on the sub-diagrams, a, b, c, d, e, f, g and h respectively. The number at the upper-left corner gives the resulting number of line segments while the line under the number represents the minimum width of the strip used. It appears that the result is good for small minimum width values and relatively poor for the larger minimum width value. This is because (i) larger minimum width value tends to ignore details of the shape and (ii) the last four minimum width values attempted were unrealistically large with respect to the scale of the diagrams. Despite these, the results generated were good and the overall feature of the shape was preserved.

A comparison is made with the DSA algorithm [9] (an error-tolerance approach algorithm) by applying DSA on the same input data set shown in Fig. 8(c). The DSA results are shown in Fig. 12. It can be seen that the new algorithm generates less number of line segments than the DSA when the minimum width is small but the most important result is that it preserves well the overall feature of the diagram whereas DSA does not. Because of these, the new algorithm is relatively insensitive to the choice of the minimum width value. This alleviates the difficulty in determining a suitable thresholding value of error tolerance. It is also important to know that it is easy to choose a suitable maximum width value. Figure 13 shows the results of applying the new algorithm on the input data Fig. 8(c) with minimum width of 1 unit and maximum widths of 2, 3, 4, 5, 6, 7, 8 and 10 units. It is shown from the results that unless the maximum width is too small (in this case it is 2), the results are similar.

The results of the global structure extraction are shown from Figs 14–16 with $E^{\min} = 5$ (Figs 14(a), 15(a) and 16(a)) or $E^{\min} = 2.5$ (Fig. 14(b), 15(b) and 16(b)), and the maximum width values of 10 units. Each diagram is generated from combining 19 results with minimum widths from 0.5, 1, 1.5, ..., to 9.5. The diagrams show that the algorithm does find the global structure of the shape with very good result.

In order to find out if scale or any other changes would affect the algorithm, the diagram in Fig. 8(c) was reprocessed and redigitized with a video camera. The shape was painted in black and then was magnified, rotated and sheared before it was redigitized. The new input data which has 2302 data points is shown in Fig. 17 with a window of 400 by 400 units. The results, shown in Fig. 18, were generated from combining 20 results with minimum width values from 5, 6, 7, ..., to 24 units. The maximum width is set equal to 25 units while $E^{\min}$ is set to 5 in Fig. 18(a) and 2.5 in Fig. 18(b). By comparing with Fig. 16, it is easy to be convinced that the results demonstrate that the new algorithm does work well.

On the other hand, the proposed algorithm runs

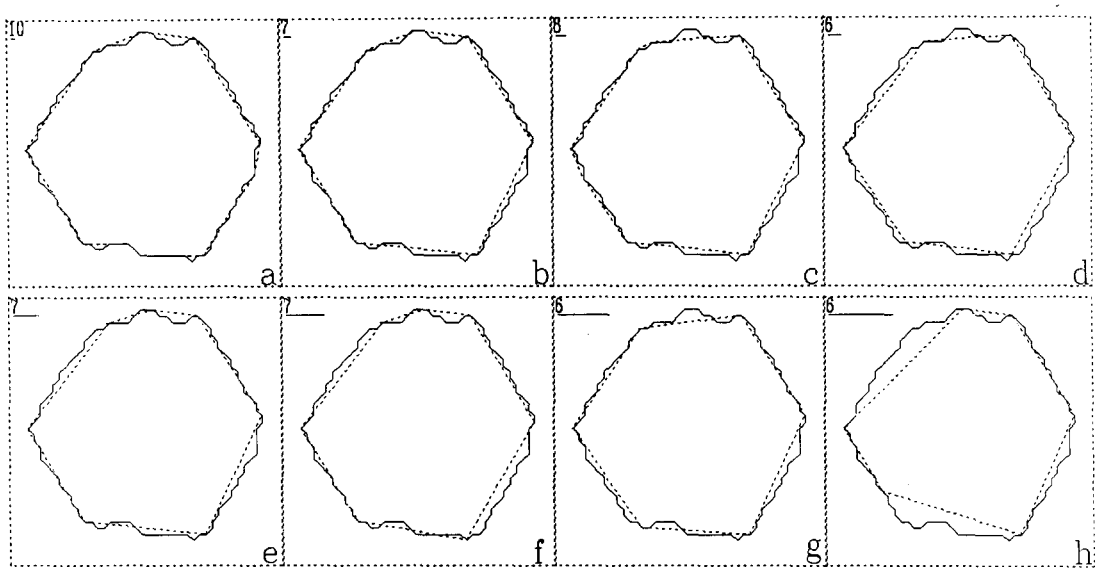


Fig. 9. Line fitting of the deformed hexagon with 102 points. The number shown at the upper left corner of each figure is the number of the approximated lines. The line under the number is the length of the minimum width of a fitting strip. —— Original line. - - - Approximated line.

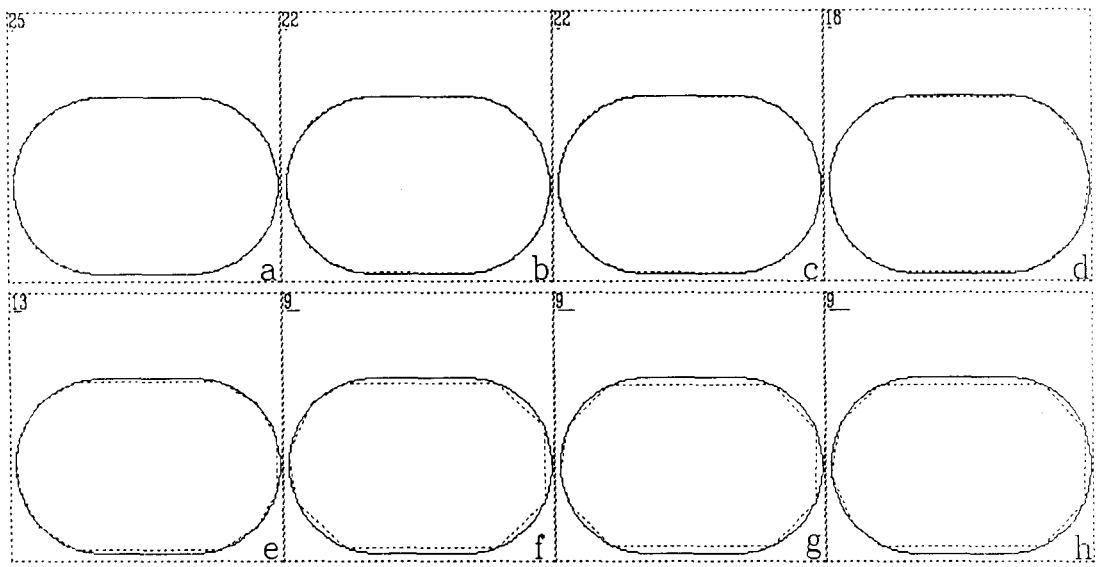


Fig. 10. Line fitting of the circular shape with 352 points. The number shown at the upper left corner of each figure is the number of the approximated lines. The line under the number is the length of the minimum width of a fitting strip. —— Original line. - - - Approximated line.

much slower than the DSA (see Table 1). Let *Tf* and *Tg* be the times to fit lines to the shape and to extract global structure respectively. Note that *Tg* includes *Tf*. The timing order is $O(N^2)$ compared to $O(N)$ of the DSA. Since the implementation was done in a straightforward manner, it is by no means optimal in terms of execution time. Reformulating the implementation could probably improve the timing. Furthermore, it appears that the algorithm provides opportunities of parallelism which can further reduce the execution time if implemented on a parallel machine. Referring to Table 1, it is interesting to note that the most regular shape, i.e. the circular data set, requires the longest execution time. It is because of its symmetry that the strip fitting process will go on until all cases have been exhausted.

6. CONCLUSION

In this paper, a new algorithm for fitting curve with lines using a feature point approach is proposed. It fits each data point on a curve with two strips, one

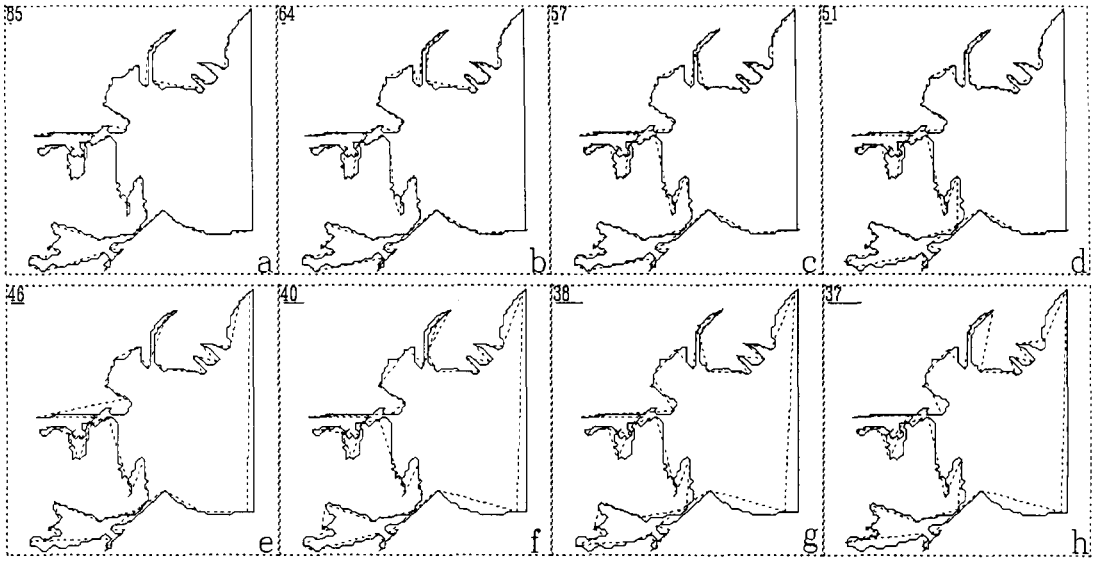


Fig. 11. Line fitting of the shadow region with 522 points. The number shown at the upper left corner of each figure is the number of the approximated lines. The line under the number is the length of the minimum width of a fitting strip. —— Original line. - - - Approximated line.

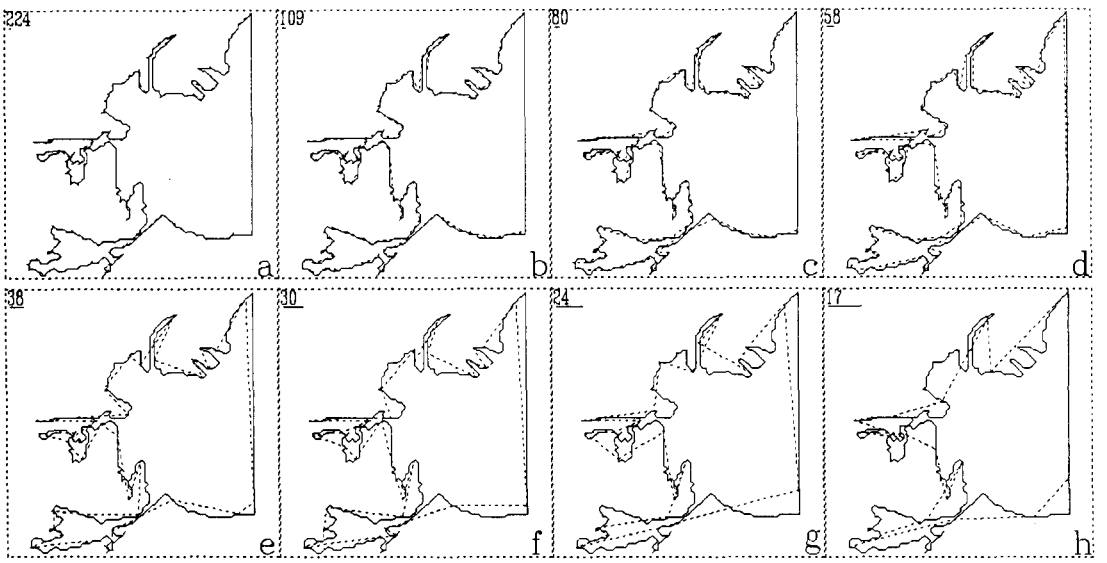


Fig. 12. DSA line fitting of the shadow region with 522 point. The number shown at the upper left corner of each figure is the number of the approximated lines. The line under the number is the length of the width of a fitting strip. —— Original line. - - - Approximated line.

on the left and one on the right, as the left and right approximated lines. The narrower and longer the strip, the better the approximated line will be. The algorithm is able to derive the best fitted strips by adjusting the orientation and width of the strip dynamically. Based on these two strips, two important properties of the data point can be derived. First, the curvature at these points is computed as the angle subtended by these two strips. Then, the figure of merit of the data point as a feature point can be computed. Based on this information, a local maximum detection procedure is employed to pick out the most representational points. Initially, strips with a minimum width are used. It was found that the smaller the minimum width, more details of the curve can be extracted.

In shape matching or shape analysis, it is often desirable to have an algorithm that can generate the same or roughly the same approximation irrespective to scale or orientation changes. A merging process was developed to extract the global structure of a shape. It was found experimentally that the result was stable with respect to scale, rotation and shearing changes. These encouraging results show the possibility of wider applications of the proposed algorithm in the future.

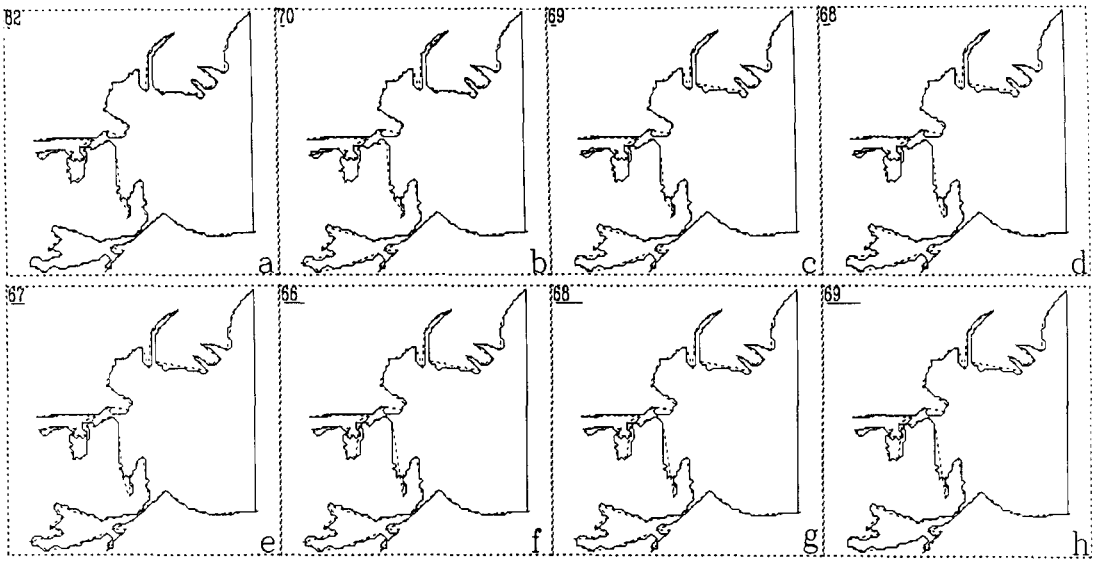


Fig. 13. Line fitting of the shadow region with varying maximum width value. The number shown at the upper left corner of each figure is the number of the approximated lines. The line under the number is the length of the maximum width of a fitting strip. —— Original line. - - - Approximated line.

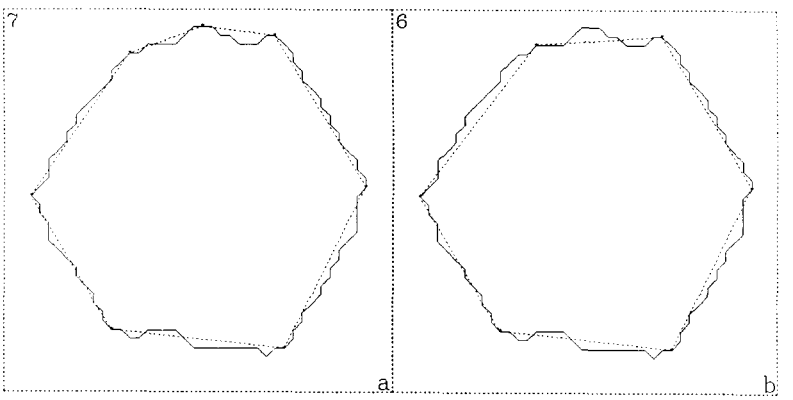


Fig. 14. Global structure extraction of the hexagon with minimum elongatedness of 5 (a) and 2.5 (b). The number shown at the upper left corner of each figure is the number of the approximated lines. —— Original line. - - - Approximated line.

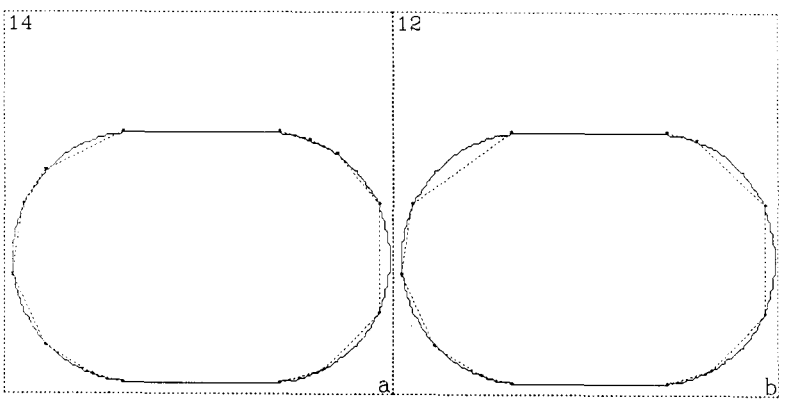


Fig. 15. Global structure extraction of the circular shape with minimum elongatedness of 5 (a) and 2.5 (b). The number shown at the upper left corner of each figure is the number of the approximated lines. —— Original line. - - - Approximated line.

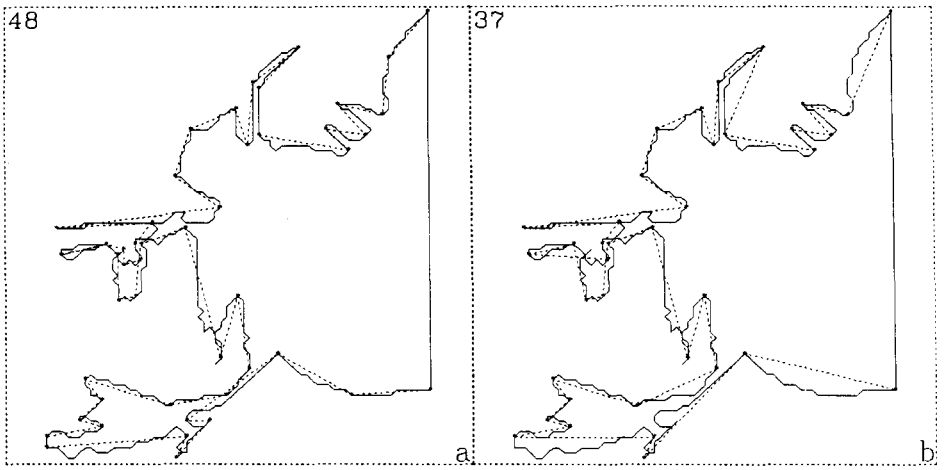


Fig. 16. Global structure extraction of the shadow region with minimum elongatedness of 5 (a) and 2.5 (b). The number shown at the upper left corner of each figure is the number of the approximated lines. —— Original line. - - - Approximated line.

Table 1. Timing information of the algorithm

Timing information on the three data sets with minimum width = 1						
Maximum width	Hexagon data 102 Points		Circular data 352 Points		Shadow data 522 Points	
	Tf(s)	Tg(s)	Tf(s)	Tg(s)	Tf(s)	Tg(s)
5	1.7	1.8	7.9	8.2	4.5	4.9
10	4.1	4.2	23.8	24.6	10.7	11.4
15	6.8	7.9	42.1	43.6	17.6	18.9
20	9.7	10.2	62.3	63.9	25.8	28.3

Fig. 17. Picture of the redigitized region with 2302 data points.

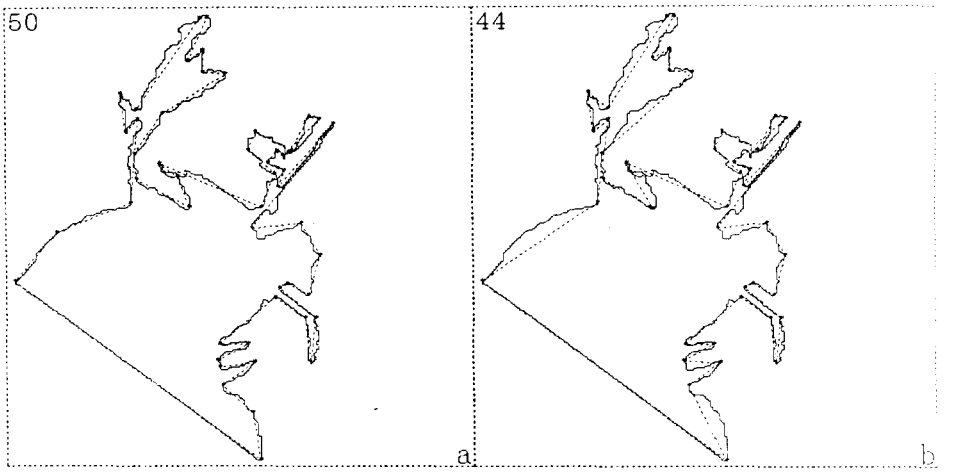


Fig. 18. Global structure extraction of the redigitized figure with minimum elongatedness of 5 (a) and 2.5 (b). The number shown at the upper left corner of each figure is the number of the approximated lines. —— Original line. - - - Approximated line.

Acknowledgements—The first author would like to acknowledge the graduate scholarship provided by the University of Saskatchewan and the research assistantship as well as the teaching assistantship provided by the Department of Computational Science. Financial support to the second author provided by the Natural Sciences and Engineering Research Council of Canada under grant number A0370 as well as an equipment grant is gratefully acknowledged. Finally, the authors are indebted to the referee for the constructive comments.

REFERENCES

1. Attneave, F. Some informational aspects of visual perception. *Psychol. Rev.* **61:** 183–193; 1954.
2. Blakemore, C.; Over, R. Curvature detectors in human vision. *Perception* **3:** 3–7; 1975.
3. Davis, L. S. Understanding shape: angles and sides. *IEEE Trans. Comput.* **C-26:** 236–242; 1977.
4. Davis, L. S. Shape matching using relaxation techniques. *IEEE Trans. Pattern Analysis Mach. Intell.* **PAMI-1:** 60–72; 1979.
5. Dunham, J. G. Piecewise linear approximation of planar curves. *IEEE Trans. Pattern Analysis Mach. Intell.* **PAMI-8:** 67–75; 1986.
6. Imai, H. Computational-geometric methods for polygonal approximations of a curve. *Comput. Vision Graphics Image Process.* **36:** 31–41; 1986.
7. Johnston, E.; Rosenfeld, A. Angle detection on digital curves. *IEEE Trans. Comput.* **C-22:** 875–878; 1973.
8. Leung, M. K.; Yang, Y. A region based approach for human body motion analysis. *Pattern Recognition* **20:** 321–339: 1987.
9. Leung, M.; Yang, Y. H. Dynamic strip algorithm in curve fitting. Research Report No. 88-4, Dept. of Computational Science, University of Saskatchewan, Saskatoon, Sask, Canada; 1988.
10. Pavlidis, T. A review of algorithms for shape analysis. *Comput. Graphics Image Process.* **7:** 243–258; 1978.
11. Pavlidis, T. Algorithms for shape analysis of contours and waveforms. *IEEE Trans. Pattern Analysis Mach. Intell.* **PAMI-2:** 301–312; 1980.
12. Ramer, U. An iterative procedure for the polygonal approximation of plane curves. *Comput. Graphics Image Process.* **1:** 244–256; 1972.
13. Reumann, K.; Witkam, A. P. Optimizing curve segmentation in computer graphics. *Int. Comput. Symp.*; New York; 467–472; 1974.
14. Roberge, J. A data reduction algorithm for planar curves. *Comput. Vision Graphics Image Process.* **29:** 168–195; 1985.
15. Sklansky, J.; Gonzalez, V. Fast polygonal approximation of digitized curves. *Pattern Recognition* **12:** 327–331; 1980.
16. Toussaint, G. T. On the complexity of approximating polygonal curves in the plane. *Proc. IASTED Int. Symp. Robotics Automation*; Lugano, Switzerland 1–4; 1985.
17. Williams, C. M. An efficient algorithm for the piecewise linear approximation of planar curves. *Comput. Graphics Image Process.* **8:** 286–293; 1978.

About the Author—YEE-HONG YANG received the B.Sc. degree (with first class honours) in Physics from the University of Hong Kong in 1974, the M.S. degree in Physics from Simon Fraser University in Canada in 1977, the M.S.E.E. and the Ph.D. in electrical engineering degrees from the University of Pittsburgh in 1980 and 1982 respectively. From 1980 to 1983, Dr Yang was a research scientist in the Computer Engineering Center of Mellon Institute, a division of Carnegie-Mellon University, where he was involved in the Very High Speed Integrated Circuit (VHSIC) program sponsored by the United States Government. He joined the Department of Computational Science at the University of Saskatchewan in 1983 and is currently an Associate Professor. Dr Yang's research interest includes image processing, computer vision, motion analysis, computer graphics and computer architectures.

About the Author—MAYLOR K. H. LEUNG received the B.Sc. degree in Physics from the National Taiwan University in 1979 the B.Sc. and the M.Sc. degrees in Computational Science from the University of Saskatchewan in 1983 and 1985 respectively. He is currently working towards the Ph.D. degree in Computational Science at the University of Saskatchewan. His research interest is in analyzing and interpreting human body motion using computational techniques.