

SHAPE DESCRIPTION VIA THE USE OF CRITICAL POINTS*†

HERBERT FREEMAN

Rensselaer Polytechnic Institute, Troy, NY 12181, U.S.A.

(Received 22 July 1977; received for publication 3 January 1978)

Abstract – A key element in pattern recognition is the description of shape. For two-dimensional objects (blobs), shape is conveyed by the curving of the boundary line and is normally considered independent of scale and orientation. The curving may be regarded as a concatenation of arcs of varying instantaneous radii of curvature, possibly interspersed occasionally by discontinuities. The description of shape is facilitated by segmenting the boundary line at so-called critical points – corners (discontinuities in curvature), points of inflection, and curvature maxima. Additional critical points are intersections and points of tangency. Algorithms are described for extracting such critical points in the presence of noise. An illustration is given showing how the critical points may be used in the development of a shape description system.

Shape description Incremental curvature Line-drawing processing Curvature discontinuities
 Map matching Centroidal profile Critical-point determination Pattern recognition

1. INTRODUCTION

The recognition of two-dimensional spatial patterns is intimately linked with the classification of shape. Shape is the composite effect of curvature; it is a property of a curve, independent of scale and orientation, and refers to the manner in which the curve deviates from some norm, such as a straight line or a circle. We shall use the terms “line” and “curve” interchangeably; a line is not presumed to be straight unless explicitly so identified. Neither must a curve be non-straight. The kinds of curve topologies of interest to us are shown in Fig. 1.

A closed curve traces a single path without end. An open curve has two distinct end points. A closed curve is always considered to be simply connected; a multiply-connected curve is regarded as a set of two or more distinct, closed curves, one of which encloses all others. Both open and closed curves may intersect other open or closed curves, as well as themselves. A closed curve has a definite sense of direction, which is such as to place the interior toward the right. For an open curve, the direction is arbitrary.

The points at which curves intersect or touch will be called junctions. Junctions are characterized by their radially, which is the count of the number of lines entering the junction. We speak of triradial, quadradial, quintradial, etc. junctions. Except in very rare cases, pictures of natural processes exhibit only tri-radial junctions. The existence of a quadradial jun-

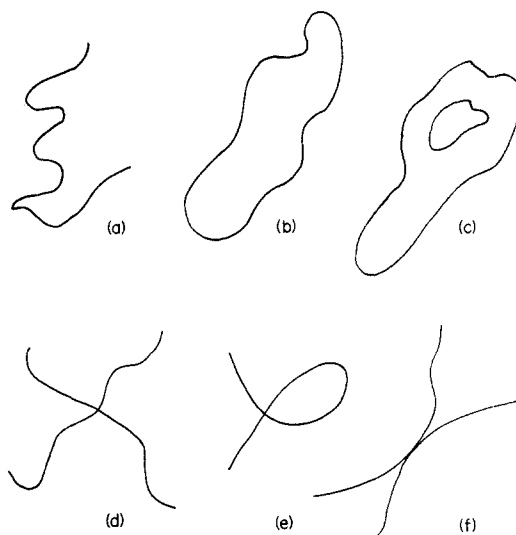


Fig. 1. Curve topologies of interest. (a) Open curve, (b) closed curve, (c) pair of closed curves, (d) intersecting curves, (e) self-intersecting curve, (f) tangent curves.

ction is almost always evidence of a man-made feature. Junctions of radially greater than four are most rare.

At a junction of radially n , pairs of adjacent curves will form n distinct, non-crossing paths through the junction. At least $n - 2$ of these paths must exhibit a slope discontinuity (corner) at the junction.

We shall regard all two-dimensional patterns as occurring in the form of maps. Maps are special kinds of line drawings that occur in the following four types – blob maps, contour maps, flow maps, and region maps. The four types are illustrated in Fig. 2. Different types may be overlaid to form composite maps.

Blob maps are maps containing only closed curves, none of which may contain another closed curve. Some of the curves may be closed via the map boundary. The

* Copyright © 1977 by The Institute of Electrical and Electronics Engineers, Inc. Reprinted, with permission, from Proceedings of the IEEE Computer Society Conference on Pattern Recognition and Image Processing, 6–8 June 1977, p. 168.

† This work was supported by the Air Force Office of Scientific Research, Directorate of Mathematical and Information Sciences, under Grant AFOSR 76-2937.

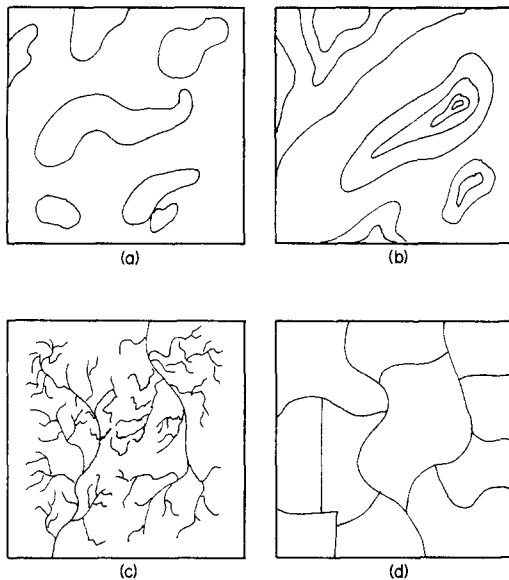


Fig. 2. Four kinds of maps. (a) Blob map, (b) contour map, (c) flow map, (d) region map.

closed curves (which each bound a blob) may touch but not intersect each other. There are no end points.

Contour maps are like blob maps except that one closed curve may contain one or more other closed curves. Two curves may be tangent to each other provided one of the curves is contained within the other. There are no end points.

Flow maps contain only open curves and these are connected to form one or more tree structures. There are no loops. All curves terminate either at junctions or at end points. For each connected tree structure, precisely one end point coincides with the map boundary.

In region maps, the space is divided into mutually exclusive areas. All lines begin and terminate at junctions or are closed (rare). There are no end points.

We identify three kinds of problem tasks that involve shape – “matching”, “fitting”, and “classifying”. In “matching” we desire to establish equivalence (up to a certain level of precision) between two curves which may be open or closed. The equivalence may be on the basis of (1) shape alone, (2) shape and scale, (3) shape and orientation, or (4) shape, scale, and orientation.

In “fitting” we must establish that two open curves (which are usually parts of two different closed curves) are equivalent in scale and complementary in shape. Two curves are said to be complementary in shape if they are geometrically similar but are traced out in the opposite sense. (The direction of tracing is determined by the area bounded, which by convention, is always assumed to lie toward the right.)

“Classifying” refers to the process of assigning given curves to predetermined categories. The assignment may be based solely on shape, or it may involve scale and orientation as well.

2. SHAPE DESCRIPTION

One of the difficulties in any shape processing task is the lack of a definitive way of describing shape. Many investigators have devoted attention to this task and a variety of approaches have been proposed.⁽¹⁻⁷⁾ We shall follow a method originally proposed by Attneave and Arnoult and divide a curve into segments and then use relatively simple features for characterizing the segments. Key to this method is an effective segmentation scheme. Later we shall also consider a scheme, applicable to closed curves only, that does not require segmentation.

In all that follows it will be assumed that the curves have been quantized into a square lattice of sufficient fineness to preserve the desired detail⁽⁸⁾ and then encoded in terms of the 8-direction chain code.⁽²⁾

We shall first examine the shape matching problem. A tried and effective technique for matching two chains (open or closed) is the chain correlation scheme, so called because of its analogy to conventional signal correlation.⁽⁹⁾ It is given by

$$\phi_{ab}(j) = \frac{1}{n} \sum_{i=1}^n \cos(a_i - b_{i+j})\pi/4,$$

where the a_i and b_i are, respectively, the ordered links in the two chains. Chain correlation has the drawback, however, of being rigidly dependent on scale and orientation. To use it for matching the shape of two chains of different scale and orientation, one of the chains must first be rotated and scaled to bring it into alignment with the other. To determine the amounts of rotation and scaling required, some further information about the two chains must be available.

To take a very simple example, suppose two open chains are to be checked for a shape match independently of scale and orientation. One can connect the two end points of each chain with a straight line segment. The relative lengths of the two segments then provides a basis for adjusting the scale of one chain to that of the other; the angle between the two segments establishes the amount of rotation required to bring the two chains into alignment. (Note that such scaling and rotating introduces additional quantization noise.) After the scale and orientation adjustment, the two chains can be correlated against each other to determine the exact degree of match.

The simple scheme just described has serious deficiencies. First of all, it will not work for closed curves. Second, the scheme is overly dependent on the precise locations of the end points. Relatively slight variations here will cause erroneous scale and orientation normalizations, and result in failure of the correlation test, even for chain pairs that are otherwise very similar in shape. Finally, the scheme is inapplicable to the “fitting” problem, where a section of one chain is to be shown to be of complementary shape to a section of another chain – without any *a priori* knowledge of orientation (and, possibly, scale as well). We shall now

develop a more powerful method which overcomes these difficulties.

Even if we expect two curves to match in their entirety, it would be unwise to test them for match over their full length at once. Part of one chain may be missing or severely distorted by noise, or perhaps a part may have been compressed. Or a chain may have been bent, placing its two parts at a wrong angle relative to each other. Although such effects clearly reduce the quality of a match, they should not be permitted to overwhelm evidence of good matching in other parts of the chains. The only solution to this is to segment the chains, and to proceed with the match on a section-by-section basis.

If the chains to be matched are supposed to have the same scale and orientation, and cross-correlation is to be used, segmentation presents no problem. Since in the correlation process one chain (or chain section) is anyway shifted relative to the other, selection of the segmentation points is virtually arbitrary. A totally different approach, however, is called for if the relative scale and orientation of the two chains are not *a priori* known.

3. CRITICAL POINTS

If we wish to compare two chains independently of their relative scale and orientation, we must find features in them that are invariant under these parameters. For this purpose we shall call upon the so-called "critical points" of the chains.

In classical geometry, the "critical points" of a curve are its maxima, minima, and points of inflection. Our interest here in curves is more far-reaching, and we shall expand the concept of critical points to include also discontinuities in curvature, end points, intersections (junctions), and points of tangency. To a varying degree, these points are all well-defined and their character is unaffected by the transformations of scale and rotation.

Let us consider the two curves shown in Fig. 3. Visual inspection shows them to be of apparent similar shape, though, of different size, different orientation, and different sense of direction. We note that each curve has two fairly sharp discontinuities in curvature (DICs). If the two curves are indeed of the same shape and there is a pairwise correspondence between the DICs, then there must be point-for-point shape correspondence in the sections between the DICs. And, of course, all shape-related features that are independent of scale and orientation will be the same for both.

We first connect the two DICs in each curve with a straight line segment and determine its length. Suitable features to be computed next are (1) the length of the curve section between the DICs, (2) the total "bay" area lying between the curve section and the straight line segment (the "bays" lie toward the left of the curve), (3) the total "peninsula" area (lying toward the

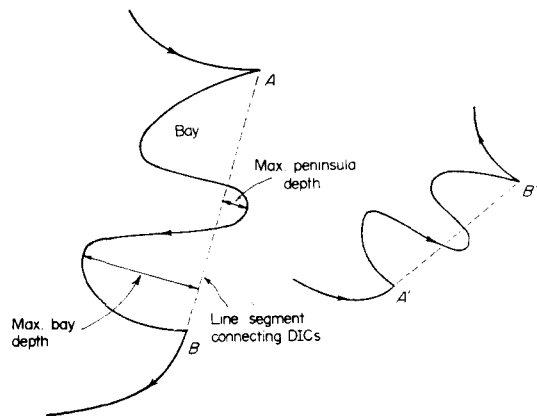


Fig. 3. Two curves considered for matching.

right of the curve), (4) the maximum "bay" depth, and (5) the maximum "peninsula" depth. All these features are, of course, independent of orientation. To make them also independent of scale, we divide each linear feature by the segment distance AB , and each area feature, by AB^2 . The features are illustrated in Fig. 3.

Simple, efficient algorithms exist for computing the foregoing features if the curves are represented in the form of chains.⁽¹⁰⁾ Observe that if for two curve sections the features are pairwise in close agreement, then we have strong indication that the curves are of the same shape; though, of course, a correlation or other point-by-point check is needed to establish this for certain. Conversely, any major discrepancy in a feature is indicative of a difference in shape. If for two chains all features match except that the bay features of one match to the peninsula features of the other, then the two chains are likely to be of complementary shape and thus potential candidates for fitting.

In the foregoing example we selected DICs to define curve sections for matching. Of all the critical points, DICs (as well as junctions) are the most sharply and most reliably defined. End points are also sharply defined; however, their location is less dependable since occlusion of part of a curve will cause a false end point to be generated. The next best critical points to use (if DICs and junctions are not available) are points of inflection and points of maximum curvature (maxima or minima). We shall discuss these further in the next section in connection with the determination of critical points.

4. DETERMINATION OF CRITICAL POINTS

Junctions and end points require no special effort for their determination; they are available directly from the chain data structure. Discontinuities in curvature, however, must be extracted by computation. A scheme that has been found to be effective for this purpose is the *line segment scan*,⁽¹¹⁾ described as follows.

We consider a chain $\{a_i\}_1^n$, $a_i \in \{0, \dots, 7\}$, and define a

straight line segment L_i^s that connects the initium of a_{i-s+1} to the terminus of a_i . The length of L_i^s can vary from a maximum of $s\sqrt{2}$ to a minimum of 1; however, for a well-quantized curve, it should never be less than $s/2$. The actual length of L_i^s is, of course, given by

$$\ell_i^s = [(X_i^s)^2 + (Y_i^s)^2]^{1/2},$$

where

$$X_i^s = \sum_{j=i-s+1}^i a_{ix}$$

$$Y_i^s = \sum_{j=i-s+1}^i a_{iy},$$

and where the a_{ix} and a_{iy} are, respectively, the x and y components of the chain links, $a_{ix}, a_{iy} \in \{-1, 0, 1\}$.

The angle between the x axis and the forward direction of the line segment is

$$\begin{aligned} \theta_i^s &= \tan^{-1} Y_i^s / X_i^s & \text{if } |X_i^s| \geq |Y_i^s| \\ &= \cot^{-1} X_i^s / Y_i^s & \text{if } |X_i^s| < |Y_i^s|, \end{aligned}$$

and can vary from 0 to 360° . Of particular interest is the variation in this angle as L_i^s scans over the chain. Note that for a closed chain, $a_i = a_{i-n}$. We define

$$\delta_i^s = \theta_{i+1}^s - \theta_{i-1}^s,$$

as the *incremental curvature* of the chain; it is equal to twice the mean over two successive angular differences. The incremental curvature is a smoothed measure of curvature; the greater s , the heavier the smoothing. For a well-quantized⁽⁸⁾ curve, s will range normally from a minimum of 5 to a maximum of *ca.* 13. This appears to be the optimum range of removing the effect of chain quantization noise and yet preserving significant fine detail. An illustration of line-segment scanning is given in Fig. 4.

A plot of δ_i^s for a given chain-encoded curve provides insight into the shape of the curve. The plot is, of course, independent of the orientation of the curve, and if the abscissa is normalized, can be made independent of scale as well. For a fairly straight section of a curve, δ_i^s will hover about zero. For a gentle curving to the left or right, δ_i^s will maintain a small positive or negative value, respectively. At a point of inflection, δ_i^s will change sign. At a significant curvature discontinuity, δ_i^s will take on a succession of relatively large values, all of the same sign, and all compressed into a span of precisely $s + 1$ links. This is readily apparent from an inspection of Fig. 4. The sum of the δ_i^s values over the $s + 1$ steps at a discontinuity is given by

$$\begin{aligned} \Phi_i^s &= \sum_{j=i-s}^i \delta_j^s \\ &= \theta_{i+1}^s + \theta_i^s - \theta_{i-s-1}^s - \theta_{i-s-2}^s, \end{aligned}$$

and is equal to twice the net angular change at the discontinuity.

The detection of curvature discontinuity in the presence of noise is, of course, a somewhat subjective

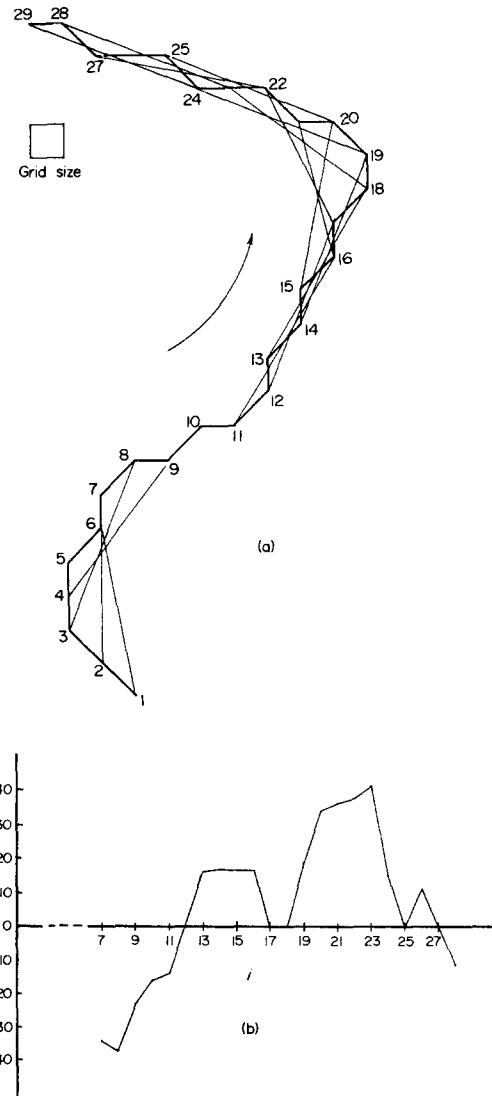


Fig. 4. Illustration of line-segment scan. (a) Chain being scanned ($s = 5$); for sake of clarity, not all scan segments are shown, (b) plot of incremental curvature as a function of i .

process. A chain is a special kind of straight-line approximation of a curve in which all segments are of length 1 or $\sqrt{2}$, and all angular changes are multiples of 45° . To get above the chain quantization noise level, a value of s of at least 4 is required. Heavier smoothing ($s > 4$) is desirable, but at some increased value, valid curvature discontinuities will be smoothed over. In the context of a chain representation (where true curvature continuity does not exist), a discontinuity in curvature can only be defined as the combination of three concatenated sequences of δ_i^s values – two during which δ_i^s varies about a small mean value (or about zero), separated by one precisely of length $s + 1$ during which it differs significantly from zero and accumulates a substantial total variation Φ_i^s . In Fig. 4(b), this condition is satisfied in the region $i=19-24$. For this region the accumulated incremental curvature Φ_{24}^5 equals 189.2, indicating a new CCW curvature of

94.6°. The curvature discontinuity is clearly evident; its location is fixed at the value of i just preceding the first significant value of δ_i^s , that is, at $i = 18$.

In general, we shall refer to the prominence of a curvature discontinuity as its "cornerity" and assign the following quantitative measure⁽¹¹⁾:

$$K_i = \sqrt{t_1 t_2} \Phi_{i+s+1}^s,$$

where t_1 and t_2 are the lengths of the δ_i^s sequences, to either side of the discontinuity, over which δ_i^s remains at a small mean value.

For the purpose of shape matching (or fitting) using scale- and orientation-invariant features, only discontinuities with large cornerity values should be used. Use of discontinuities with small cornerity values may lead to erroneous segmentation.

Examination of Fig. 4(b) also shows that δ_i^s changes sign at $i = 12$, indicating the presence of a point of inflection. The actual location of the point of inflection is, of course, $s/2$ positions earlier since δ_i^s is labelled according to its leading node. In the example shown, with $s = 5$, we place the point of inflection at $i = 10$.

If the point of inflection is prominent, that is, if there is a clear and abrupt shift of the radius of curvature from one side of the chain to the other, the preceding method will locate it. Not infrequently, however, a curve changes from convex to concave curvature in a slow and meandering fashion. Then we cannot determine the point of inflection in this way. If the two regions of opposing curvature (for which we are seeking the point of inflection) are both prolonged and strong, we may find their maxima. We then connect these two points of maximum curvature with a straight line and take its intersection with the chain, lying between the two regions of curvature, as the point of inflection. If the two curvature regions are not sufficiently strong, or are spaced too far apart, the point of inflection cannot be reliably determined and we must seek other critical points elsewhere.

5. CENTROIDAL PROFILE

An alternate scheme for representing the shape of a blob in an orientation- and scale-invariant manner is provided by the centroidal profile. It is applicable only to closed curves and does not involve any segmentation. The centroidal profile is simply a normalized plot of the distance from the boundary to the centroid of the blob, as a function of distance along the boundary. Since the centroid's coordinates are determined by the ratio of first-order moments to enclosed area, they are relatively insensitive to noise, and the centroid thus tends to be a stable point of reference for the blob. If a blob is described in the form of a chain, its centroid can be rapidly computed by means of a simple algorithm.⁽¹⁰⁾

To obtain the centroidal profile for a chain-coded blob, one first computes the location of the centroid. Next one determines the distances from the centroid to each chain node in sequence, beginning with the

initium of the chain. The Euclidean distance measure could be used; however, a simpler and equally effective measure for the purpose at hand is provided by the residue-chain length. A residue chain between two points is the standardized chain of minimum length connecting the two points.⁽¹⁰⁾ The computation of residue-chain length involves only the operations of addition and branching, and because of its simplicity is the preferred distance measure for the centroidal profile. Its use does require, however, that the coordinates of the centroid be quantized to the nearest grid node, thereby slightly increasing the quantization noise in the blob description.

For a closed chain, the choice of the initium is normally arbitrary. To provide some uniformity in the selection of the origin of the centroidal profile, the initium is cyclically shifted to the chain node maximally distant from the centroid. If the maximum is not unique, any of the maxima may be chosen. This shifting of the initium provides some orientation invariance for the centroidal profile. However, if there is more than one maximum, there is an ambiguity of order equal to the number of maximally-distant chain nodes. This is only a minor drawback, as we shall see shortly.

The dependence on scale can be removed by dividing all profile values by the maximum value. The actual maximum value will be called the vertical normalization factor. Every centroidal profile thus will have a value of unity at the origin and be less than or equal to unity elsewhere.

One additional normalization step remains. There is no fixed rule concerning the number of chain links that are needed to describe a given blob. Guidelines exist for selecting the quantization grid fineness that are based on the value of the minimum radius of curvature that is to be preserved in the resulting chain.⁽⁸⁾ However, the selection of the smallest radius of curvature is subjective and a shape comparison scheme should not be too rigidly dependent on such a loose criterion. Accordingly we must also normalize the abscissa of the centroidal profile. This is done by setting the maximum abscissa value – corresponding to one complete trace around the blob contour (or, equivalently, one pass over the complete chain) – arbitrarily to 100. All other values are then adjusted in proportion. The factor required to yield the uniform abscissa scale will be called the horizontal normalization factor. A centroidal profile, together with the blob whose shape it represents, is shown in Fig. 5.

For the blob of Fig. 5, the vector from the centroid to successive chain nodes rotates uniformly about the centroid in the clockwise sense. This will, of course, not be the case in general. To avoid possible ambiguities in the interpretation of a centroidal profile, we shall take profile values corresponding to clockwise angular rotation as being positive, and those corresponding to counterclockwise rotation, as being negative. An illustration of this is provided by Fig. 6.

The centroidal profile represents the blob without

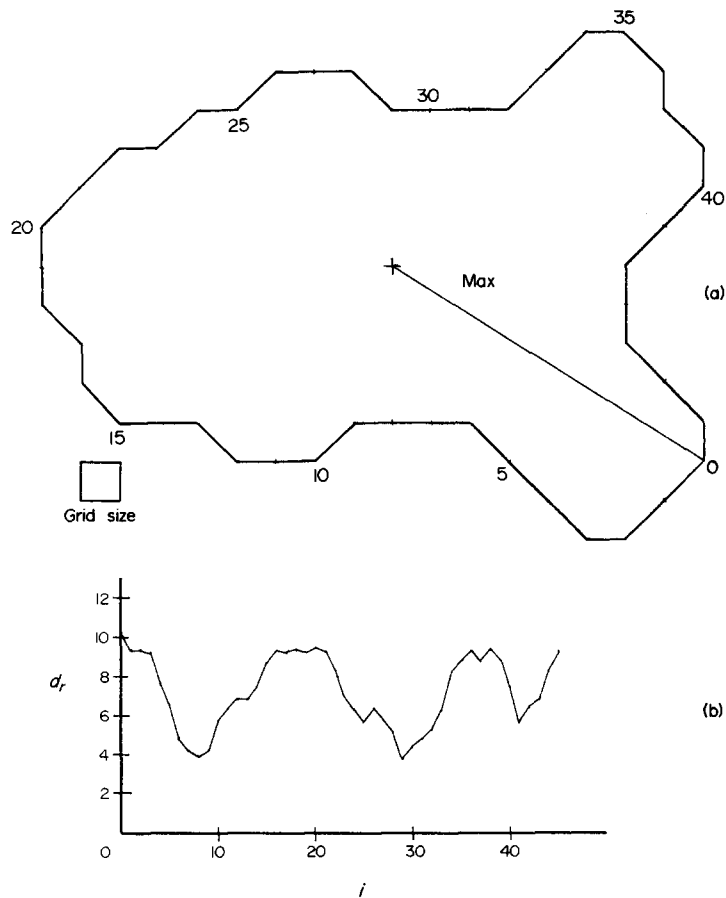


Fig. 5. Illustration of centroidal profile. (a) Chain-coded closed curve, (b) corresponding profile based on chain-residue distance.

regard to scale, orientation, and (in a limited sense) quantization fineness. In effect only the shape remains. Note that the profile generation process is reversible; if the origin shift and the vertical and horizontal normalization factors are known, it is possible to reconstruct the blob chain from its centroidal profile.

Two blobs may be compared as to their shape by comparing their centroidal profiles. If both profiles have a single, well-defined maximum (which, of course, will be at the origin), the comparison is straightforward. Suitable measures of shape dissimilarity are the integral of the differences between the two profile functions and the integral of the squared differences.

If one or both of the profiles have a maximum that is not well defined, or have multiple maxima, a better means for comparison is provided by conventional signal cross-correlation. This will yield a measure of the similarity as well as identify possible errors in the origin selection.

It should be pointed out that the abscissa of the centroidal profile represents the (normalized) link-index parameter i of the boundary chain. It does *not* represent uniform angular rotation about the centroid; nor, in fact, is it a precise measure of distance along the boundary because no account is taken of the actual lengths of the links (1 or $\sqrt{2}$). This variation is

link length, together with the initial chain-generating quantization and the node-quantization of the centroid mentioned earlier are the chief sources of noise. Unless held to a low value by using a sufficiently fine grid size, the effect of this noise will be to reduce the quality of shape matching. Some good "feel" for adequate quantization can be gotten by selecting a blob that is representative of those of interest, quantizing it for two different orientations, computing the corresponding centroidal profiles, and then comparing these. If the quality of match is satisfactory, the quantization fineness is likely to be adequate.

The precision of the centroidal profile can be improved though at considerable increase in processing cost — by plotting the distance from centroid to chain node against the angle this vector makes with the x axis, and using the Euclidean distance from the true centroid (rather than from the node nearest the centroid). The same origin shift and vertical normalization are then applied. The abscissa range now will, of course, be precisely equal to 360° and no horizontal normalization is necessary. Although relatively simple algorithms exist for calculating the Euclidean distances between points and the angles relative to the x axis¹⁰, indications thus far are that these refinements are not worth the extra effort.

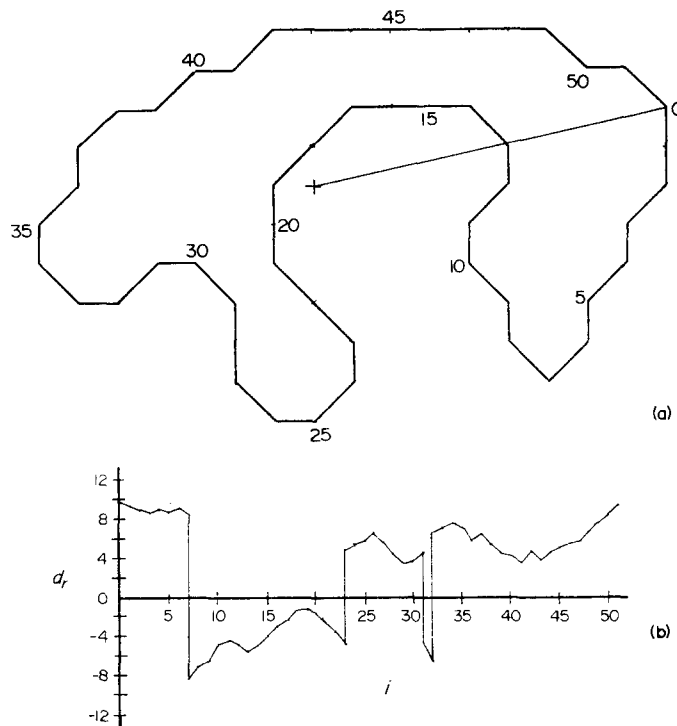


Fig. 6. Illustration of centroidal profile for a blob rotating both positively and negatively about centroid.
(a) Blob chain, (b) centroidal profile.

REFERENCES

1. F. Attneave and M. D. Arnoult, The quantitative study of shape and pattern perception, reprinted in *Pattern Recognition*, (L. Uhr, ed.). Wiley, New York, pp. 123-141 (1966).
2. H. Freeman, On the encoding of arbitrary geometric configurations, *IRE Trans.* **10** (2), 260-268 (1961).
3. H. Blum, A transformation for extracting new descriptors of shape, in *Models for the Perception of Speech and Visual Form*, (W. Wathen-Dunn, ed.). M.I.T. Press, Cambridge, MA, pp. 362-380 (1967).
4. C. T. Zahn and R. Z. Roskies, Fourier descriptors for plane closed curves, *IEEE Trans. Comput.* **21**, (3), 269-281 (1972).
5. Eric Persoon and King-Sun Fun, Shape discrimination using Fourier descriptors, *IEEE Trans. Comput.*, *IEEE Trans. Syst., Man, Cybern.* **7**, (3), 170-179 (1977).
6. M. J. Eccles, M. P. C. McQueen and D. Rosen, Analysis of the digitized boundaries of planar objects, *Pattern Recognition* **9**, (1), 31-41 (1977).
7. Larry S. Davis, Understanding shape: angles and sides, *IEEE Trans. Comput.* **26**, (3), 236-242 (1977).
8. H. Freeman and J. Glass, On the quantization of line drawing data, *IEEE Trans. Syst., Sci., Cybern.* **5**, (1), 70-79 (1969).
9. J. Feder and H. Freeman, Digital curve matching using a contour correlation algorithm, *IEEE Int. Conf. Record*, Pt. 3, 69-85 (1966).
10. H. Freeman, Computer processing of line-drawing images, *ACM Comput. Surv.* **6**, (1), 57-97 (1974).
11. Herbert Freeman and Larry S. Davis, A corner-finding algorithm for chain-coded curves, *IEEE Trans. Comput.* **26**, (3), 297-303 (1977).

About the Author - HERBERT FREEMAN (A'47-M'52-SM'54-F'67) received the B.S.E.E. degree from Union College, Schenectady, N.Y. in 1946, and the M.S. and Dr.Eng.Sc. degrees in electrical engineering from Columbia University, New York, N.Y., in 1948 and 1956, respectively.

He joined the Sperry Gyroscope Company, Great Neck, N.Y., in June, 1948, as a project engineer. In the early 1950's he designed Sperry's first digital computer, the SPEEDAC. In February, 1957 he was promoted to head of the advanced studies department, and in December, 1959, to head of the military data processing department.

In September, 1960 he joined the electrical engineering department at New York University, and served as chairman of the department from 1968 to 1973. Since 1975 he has been professor of computer engineering at Rensselaer Polytechnic Institute, Troy, N.Y. He has been a visiting professor at M.I.T. (1958-59), the Swiss Federal Institute of Technology in Zurich (1966), and the Computer Science Research Institute in Pisa, Italy (1973). He is the recipient of a NSF Senior Post Doctoral Fellowship (1966) and a Guggenheim Fellowship (1973).

Dr. Freeman is the author of over 50 technical papers and the book *Discrete Time Systems*, and currently also serves as editor of the international journal *Computer Graphics and Image Processing*, and associate editor of the journals *Calcolo* and *Pattern Recognition*.

He is a registered professional engineer in the State of New York, a Fellow of the IEEE, and member of ACM, Sigma Xi, the Society for Information Display, the Pattern Recognition Society, and the New York Academy of Sciences. He has been active in the International Federation for Information Processing, was programme chairman for IFIP Congress 74, held in Stockholm in August 1974, and has received the IFIP Silver Core Award. Since 1976 he has been treasurer of the International Association for Pattern Recognition. He was a general chairman for the IEEE Computer Society Conference on Pattern Recognition and Image Processing (1977), and is currently chairman of the IEEE Computer Society Committee on Machine Intelligence and Pattern Analysis.