copyedit

XEROX

Identifying Salient Circular Arcs on Curves

Eric Saund

SPL-93-017

P93-00005

Systems & Practices Laboratory
Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, California 94304
415-812-4334 (fax)

Identifying Salient Circular Arcs on Curves

Eric Saund
Xerox Palo Alto Research Center
3333 Coyote Hill Rd.
Palo Alto, CA. 94304
(415) 812-4474
(415) 812-4241 (fax)

April, 1992

Identifying Salient Circular Arcs on Curves

Abstract

This paper addresses the problem of identifying perceptually significant segments on general planar curvilinear contours. Lacking a formal definition for what constitutes perceptual salience, we develop subjective criteria for evaluating candidate segmentations (such as might be delivered by an algorithm), and formulate corresponding objective measures. An algorithm is presented attempting to meet these criteria. The segments delivered have the following properties: (1) each segment is well-approximated by a circular arc, (2) each pair of segments describe different sections of the contour, and (3) the curve either terminates or changes in orientation and/or curvature beyond each end of every segment. The result is a description of the contour at multiple scales in terms of circular arcs that may overlap one another.

1 Introduction

How many distinguished pieces or segments comprise the contour in figure 1? Under different interpretations, this figure can be viewed as a rectangle with *four* roughly straight sides, an encircling with *eight* segments (four sides and four rounded corners), or a figure with *fourteen* subsegments as shown in figure 1d. One is entitled to quibble with these counts on the basis of his own perceptual intuition and judgement, but no one would decompose this object in terms of the *seven* arbitrarily chosen parts shown in figure 1e. What are the ingredients leading to a natural partitioning of a planar contour into smaller, perceptually salient pieces?

This question is important to computer vision because vision algorithms typically operate by combining local measurements, e.g., edge features, into more global interpretations, e.g. object recognition by matching parts of a model object to features in the image. Effective interpretation of local measurements relies upon the identification of appropriate sized units in which to describe structure in the visual world [36]. When local edge or line features on intensity images have been linked into extended (chain coded) contours, the problem becomes one of breaking the contour into smaller pieces that are likely to correspond to useful units of later interpretation. For

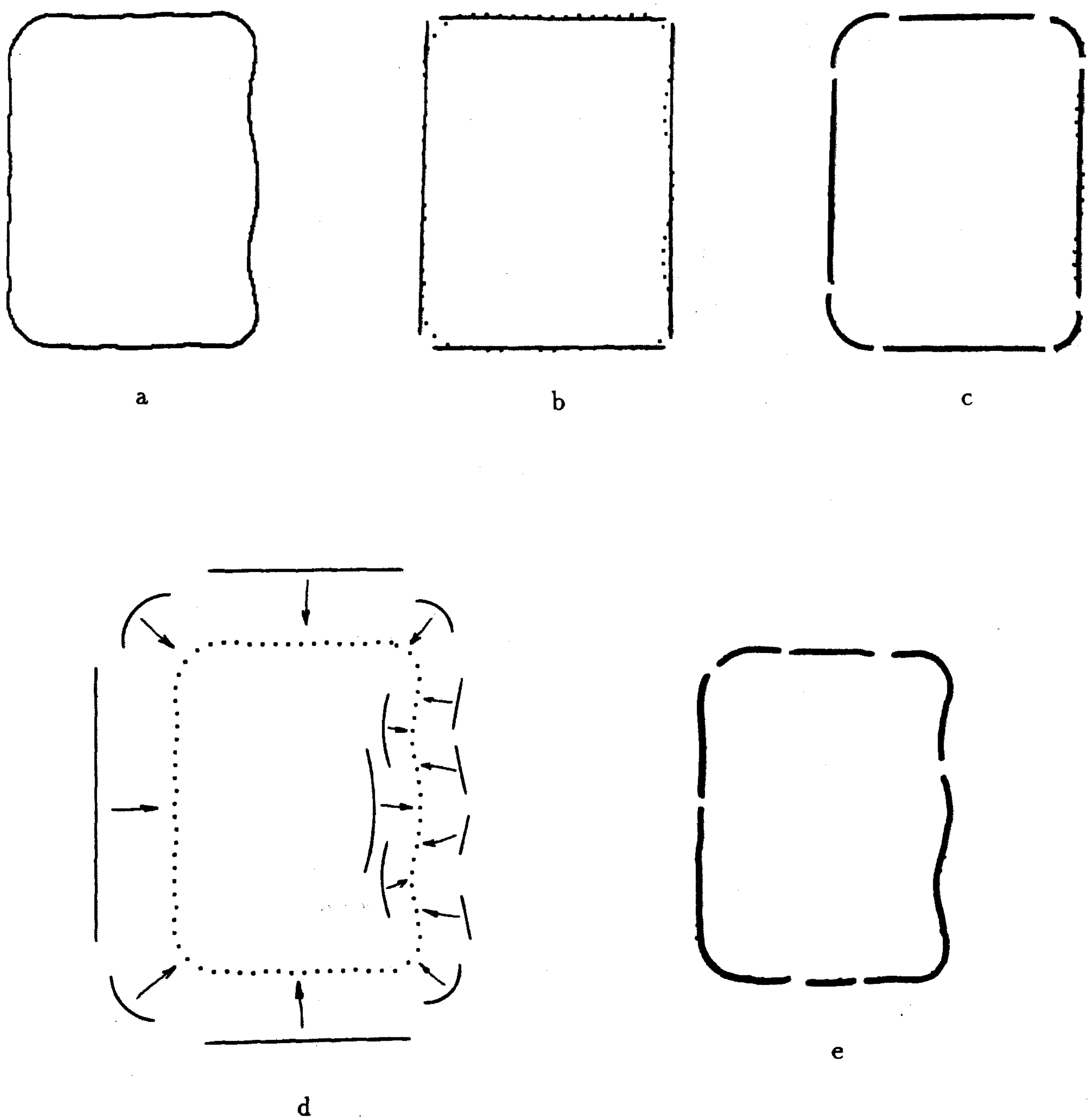


Figure 1: (b), (c) and (d) are perceptually satisfying interpretations of (a); (e) is not. The algorithm presented returns the union of interpretations (b), (c), and (d).

example, in an industrial setting with rectilinear objects, it is natural to decompose visual contours into straight line segments (sides) and circles (holes) [3].

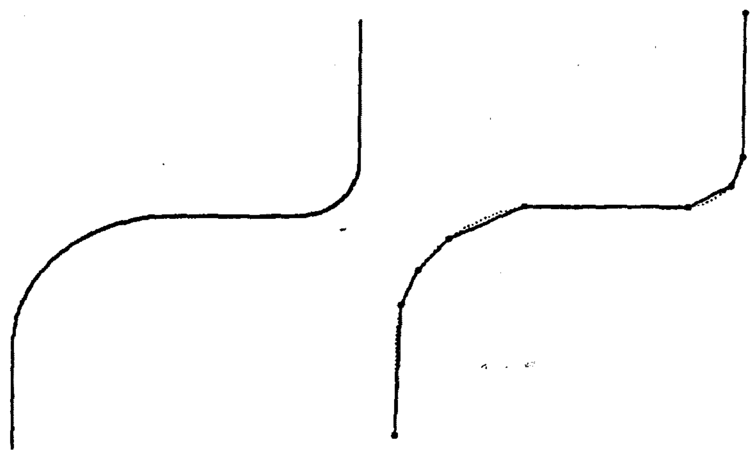


Figure 2: Straight line segments approximate curved sections of a contour but do not describe them as significant units in their own right.

While straight line segments can be used to approximate any contour e.g. [1, 7, 11, 15, 23], they do not lend themselves to the description of curving segments of contour as units unto themselves, as seen in figure 2. For this reason, workers in geometric modeling as well as computer vision have turned to more complex parametric models, including circular arcs [6, 10, 16, 24], more general conics [5, 27], and splines [12, 14, 19, 25]. In general, the analytic form selected for describing contour segments should be matched to the domain-dependent processes that generate the contours; for example, if all images for a given task are oblique views of circular objects, then elliptical models are appropriate for describing the contours which will be found in the resulting images. However, for general purpose image analysis tasks in which a priori knowledge about contour shape is not available, a domain-independent curve descriptor must be used. The present work explores curve segmentations in terms of circular arc models, which are in a sense the next most simple form beyond simple straight line approximations by virtue of adding one degree of freedom (curvature).

Most previous work with curve segmentation, including segmentation in terms of circular arc approximations, treats the problem as one of finding some optimal set of "knot" points which decompose the contour into disjoint segments that meet end to end [1, 7, 10, 12, 15, 19, 30]. This approach is well suited to the problem of *reconstructing* the original contour from an information-compressed representation. However, for purposes of visual interpretation, it can be important to identify very different but equally perceptually significant segments that may overlap one another [2, 6, 11, 16, 18]. The right side of figure 1a can be viewed with equal validity in terms of a single approximately straight line, or in terms of a number of arcs and short oblique lines. Either of these decompositions might be important natural units for performing later visual tasks, for example, answering the questions, "Is this a square?," or responding to the command, "Count the wiggles."

Thus the problem we pose is to identify *all* segments of a contour that can be interpreted as a "natural" or "perceptually salient" section to approximate using a circular arc model. Unfortunately, what it means to be perceptually salient is not specified by any formal definition. Therefore, section 2 of this paper attempts first to articulate subjective criteria for what constitutes a perceptually natural contour segment by examining a number of prototypical situations that occur on curvilinear contours and appealing to the reader's own judgements. Objective formulations corresponding to these criteria are developed, and these may be applied in evaluating any algorithm that purports to decompose contours into salient segments. Next, section 3 presents an algorithm attempting to meet these criteria.

Perceptual salience in image curves has been linked with contour curvature by many investigators. Curvature measures can be used either in ranking the "salience" of chain-coded segments themselves [6, 16, 33], or to establish breakpoints between segments [2, 8, 13, 17]. Alternatively, low curvature change is reflected directly in certain parametric models (most notably a circular arc), which have been sought by filter-based detection directly in the image [9], by Hough techniques [29], by iterative optimization [22, 37] or by token grouping [5, 21, 26, 31]. Although some techniques distinguish between straight lines and circular arcs returned as output, any such technique is subsumed by pure circular arc detection because the amount of angular extent deemed to be a "straight" arc is simply a matter of setting an application-specific threshold. Detection of salient structure at multiple scales is facilitated in many approaches by incorporating a smoothing step using kernels of different widths [35]. By developing a more subtle candidate selection technique and elucidating the segment saliency criteria, this paper builds upon an overall strategy outlined by Lowe [16] which consists in assembling a set of candidate model fits to a contour and then pruning this set on some heuristic basis.

2 Criteria for Perceptually Salient Circular Arc Segments

Let us assume that a contour is specified by a linked list of points equally spaced in the plane[1]. Let us also assume that we are provided with means for approximating with a circular arc the segment between any two points on the contour. In practice, a standard method for fitting a circle to a set of points [4, 32] works well for smooth contours and for contours deeper than approximately 70° in angular extent, but breaks down for noisy shallow arcs as shown in figure 3b. In the latter case a good approximation can be found by fitting first a straight line, then simultaneously least-squares adjusting the y'-offset and curvature (x'^2 coefficient $a = \frac{\kappa}{2}$ under Taylor expansion of a circle of radius $= \frac{1}{\kappa}$ tangent to the $x-$axis at the origin) of a parabolic arc as shown in figure 3c.

For a contour of length N points, the number of possible segments is (1 of length N) + (2 of length $N-1$) + ... + ($N-1$ of length 1) = $\frac{N(N+1)}{2}$. The problem at hand is to select out a relatively small number of these, preferably without examining each of them explicitly. Because of the combinatorics, at the outset we exclude from consideration situations such as shown in figure 4 in which a good circular arc approximation is found by combining discontiguous sections of the contour; detection

[1] An algorithm for conversion to this representation from a linked list of four-connected or eight-connected pixels sampled from a square grid is presented in Appendix A.

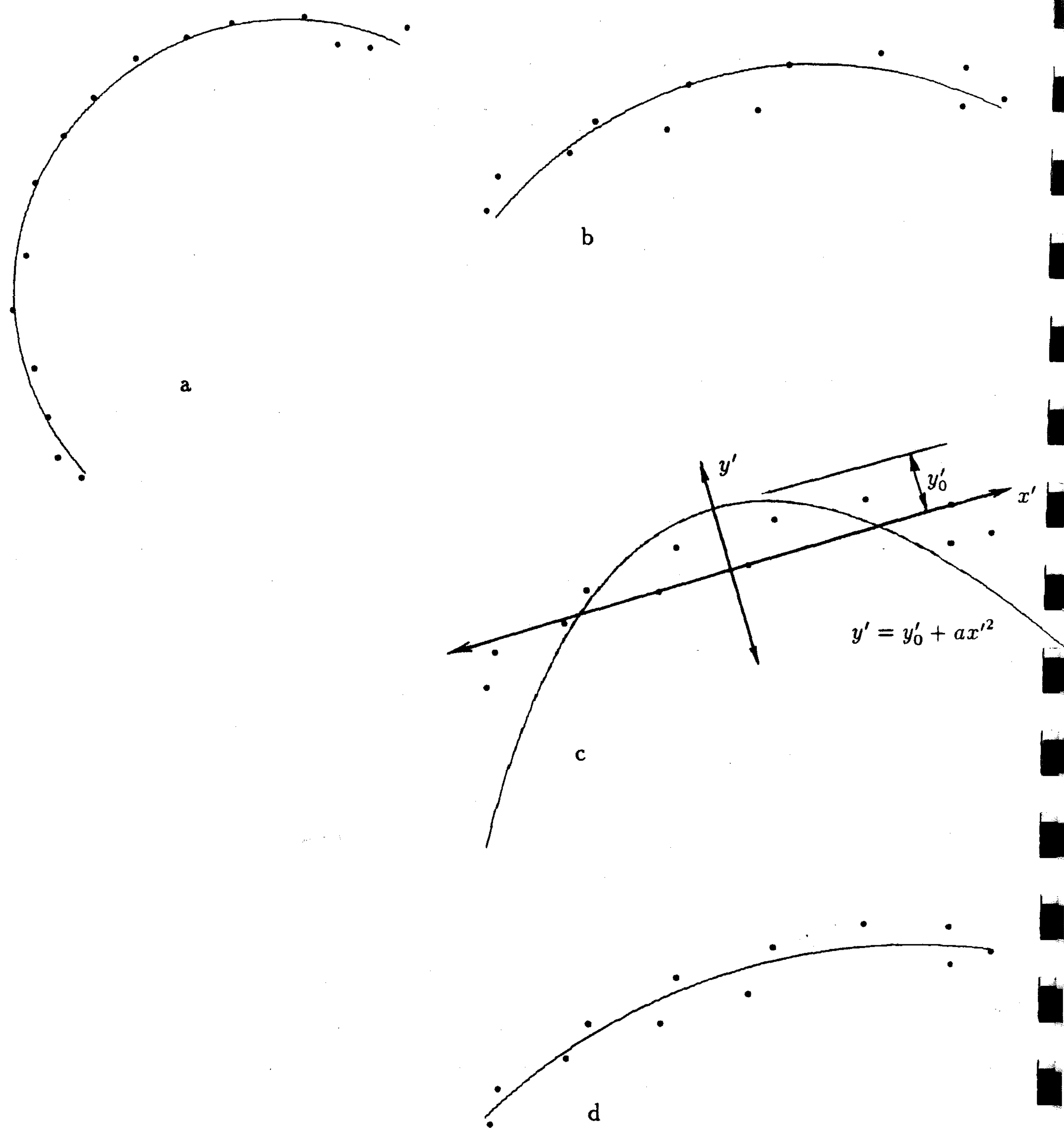


Figure 3: Standard circle fitting method [4, 32] on a deep arc (a) and a noisy shallow arc (b). (c) Parameters of a parabolic fit to the points in (b). (d) resulting circular arc fit.

of this kind of visual structure is envisioned as occurring at a later stage at which information arising from more than one contour is combined.

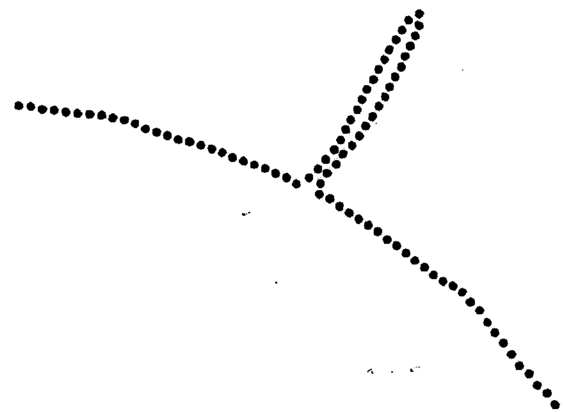


Figure 4: A single circular arc fits two discontiguous sections of this curve.

We now proceed to develop criteria useful in assessing the efficacy of any algorithm purporting to identify perceptually salient segments of a curvilinear contour. These criteria are first stated in subjective terms as motivated by telling prototypical examples, and are then expressed in terms of objective formulations engineered to reflect their respective subjective properties. The objective measures are applied either individually to a single contour segment approximated by a circular arc, or else pairwise. In general, these criteria permit a set of curve segments to be labeled with various properties indicating perceptual salience along a number of dimensions. After describing the properties, we will illustrate their use in evaluating hypothetical segmentations of an interesting test contour.

2.1 Criterion 1: Goodness of Fit

One obvious property for any contour segment approximated by a parametric model is that the model should achieve a good fit to the curve. Figure 5 illustrates this point with best-fit circular arcs for a number of equal length subsegments of a test contour. Clearly, segments A and C better correspond to natural subpieces of the contour than do B or D.

In seeking an objective measure of goodness of fit, at the outset we demand the property of self-similarity across scales: the goodness of fit measure for a given segment of a given contour must remain unchanged if the contour and segment are scaled uniformly in size. One convenient measure of goodness of fit, or *fit-quality*, that meets this criterion is the following ratio

$$\textit{fit-quality} = \frac{l}{e} \tag{1}$$

where l is the circular arc's length and e is the maximum nearest distance between the arc and the curve, as depicted in figure 5b. This measure implements a tradeoff between two competing objectives found in many previous studies of contour segmentation: (a) maximize the size of parametric models returned while (b) minimizing the error between the the models and the contour itself. While we happen to judge expression (1) preferable to, say, the ratio of arc length to *average* distance between contour and arc model, variations in an objective measure of goodness of fit are not critical as long as they preserve its overall form, including the self-similarity property. The goodness of fit criterion comports with the notion that for a given curve, equally valid contour segments, as approximated by circular arcs, may be found to overlie one another at multiple scales, as shown in figure 5c.

2.2 Criterion 2: Uniqueness

While Criterion 1 applies to curve segments individually, Criterion 2 applies to a *set* of segments attempting to label all perceptually salient segments of a given curve. This criterion states that each member segment of such a set should be unique in the sense that no other segment in the set should describe essentially the same piece of contour. As illustrated in figure 6, circular arc models of segments whose endpoints are very near one another's will in general have very similar measures of *fit-quality*. Intuitively,

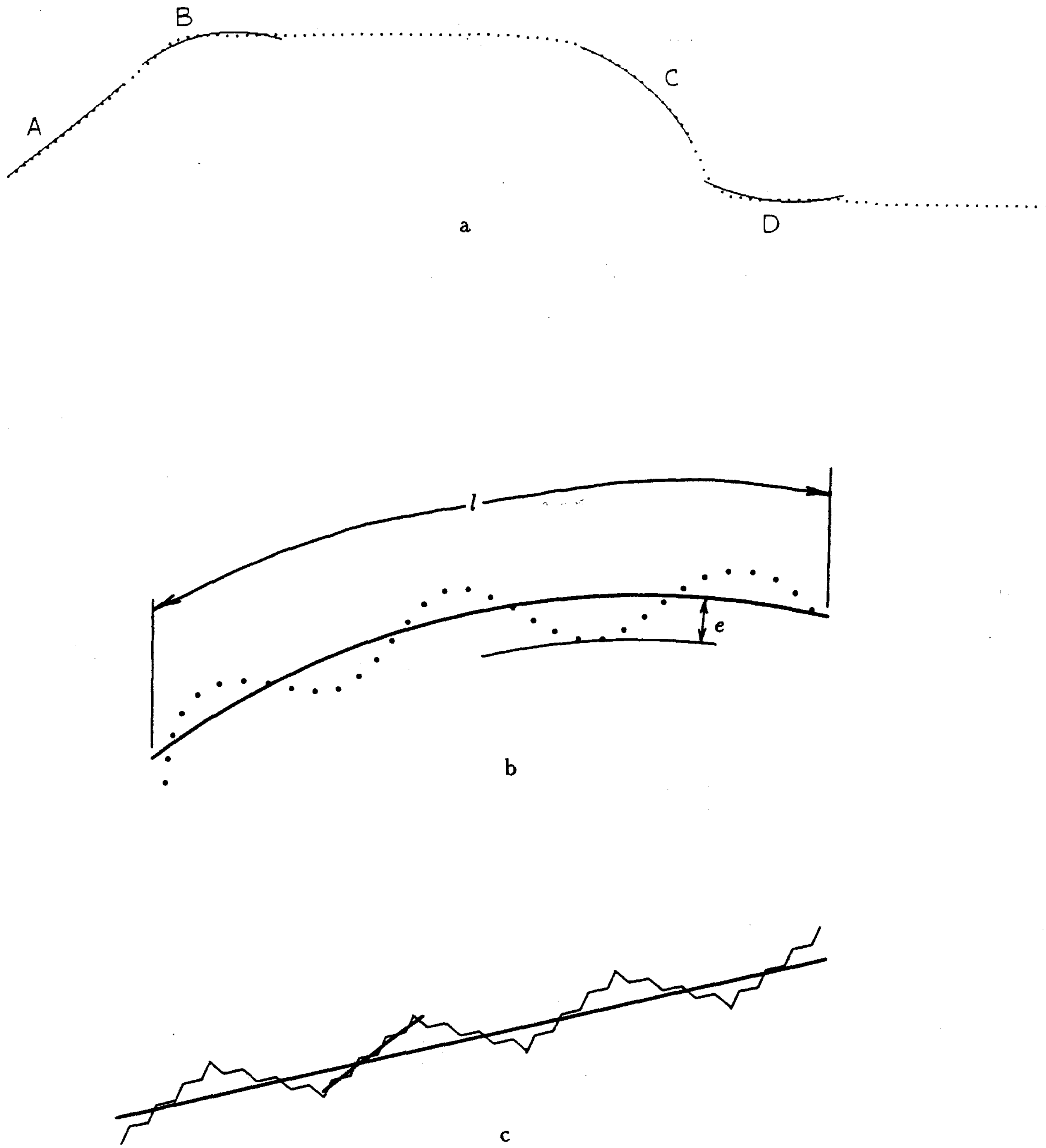
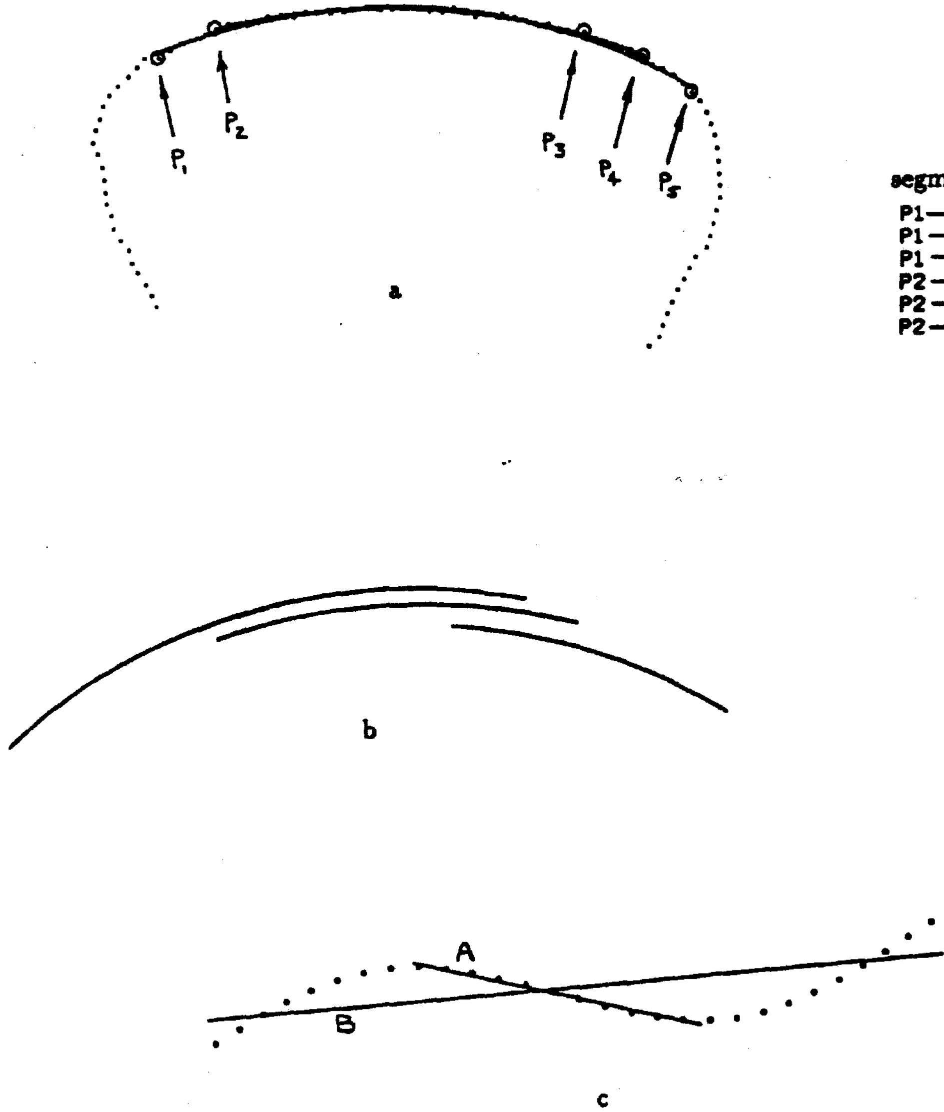


Figure 5. (a) Circular arc fits to equal length segments at different locations on a curve. Segments A and C are well fit by a circular arc, while B and D are not. (b) Geometry of a *fit-quality* measure based on the ratio of circular arc length l to the distance e between the arc and the point on the sample curve farthest from the arc. (c) A curve with similar structure at two different scales. Our *fit-quality* measure returns similar values for similarly shaped segments regardless of scale.

we are inclined to interpret the presence of but one unified "piece" or "chunk" of the curve between two roughly specified locations, not several. It is sufficient to label this section of the curve by returning only one of these segments' circular arc fit. From the standpoint of computational efficiency, some form of uniqueness criterion is necessary because, were the distance between sample points on the curve to decrease, the resulting multitude of essentially redundant segments would overwhelm any later processes.

Although the uniqueness criterion applies with respect to a set of curve segments, it is possible to formulate this condition in terms of a *pairwise* measure on the degree to which the portion of a curve described by one segment is already sufficiently described by another segment. In figure 6a, it is apparent that the circular arc fit to the segment P_2–P_3 is completely subsumed by the circular arc fit to segment P_2–P_4. The pairwise measure is cast in these terms, assigning a number indicating the degree to which some curve segment, SEGMENT-A, modeled by circular arc, ARC-A, is subsumed by another segment, SEGMENT-B, modeled by circular arc, ARC-B. Figures 6b and 6c indicate that two factors must enter into this subsumedness measure. The first, an *overlap factor*, takes into account the degree to which SEGMENT-B covers the same section of the curve as does SEGMENT-A. If there is no overlap between the two segments, then SEGMENT-A cannot be at all subsumed by SEGMENT-B, and conversely, if SEGMENT-A is completely covered then it could potentially be interpreted as totally subsumed by SEGMENT-B. The overlap factor plays against a second, *fit-quality factor*. In 6c, SEGMENT-A is not subsumed by SEGMENT-B because SEGMENT-A fits the section of the curve between its endpoints much more closely than does SEGMENT-B. Even though SEGMENT-A is completely overlapped by SEGMENT-B, it provides information about a distinct and significant section of the curve that SEGMENT-B does not.

These considerations suggest that a numeric measure of subsumedness may be expressed as a product relation between a term expressing degree of overlap between two segments and a term expressing the relative *fit-quality*; both factors must be



segment	*fit-quality*
P1— P3	79.0
P1— P4	90.4
P1— P5	95.0
P2— P3	77.6
P2— P4	91.9
P2— P5	81.9

Figure 6: (a) Six curve segments describing essentially the same section of contour have very similar measures of *fit-quality*. (b) A segment can be subsumed by another to the degree that its span is completely overlapped by it. (c) A segment is not subsumed by another segment, even if completely overlapped, if its fit quality is substantially greater.

present for the degree of subsumedness to be considered high:

$$
\begin{aligned}
&\textit{is-subsumed-by}(\text{SEGMENT-A}, \text{SEGMENT-B}) = \\
&\qquad \max(0, \min(1, \frac{\textit{overlap-fraction}(\text{SEGMENT-A},\text{SEGMENT-B}) - \gamma_1}{1.0 - \gamma_1})) \quad (2) \\
&\qquad \times \max(0, (\text{min}\, 1, \frac{\textit{relative-fit-quality}(\text{SEGMENT-A},\text{SEGMENT-B}) - \gamma_2}{1.0 - \gamma_2}))
\end{aligned}
$$

where *overlap-fraction*(SEGMENT-A,SEGMENT-B) is simply the proportion of the original curve within SEGMENT-A that is also within SEGMENT-B (a number between 0 and 1), and

$$
\textit{relative-fit-quality}(\text{SEGMENT-A}, \text{SEGMENT-B}) = \frac{\textit{fit-quality}(\text{SEGMENT-B}) \times \frac{\textit{arc-length}(\text{ARC-A})}{\textit{arc-length}(\text{ARC-B})}}{\textit{fit-quality}(\text{SEGMENT-A})}. \quad (3)
$$

The quotients in expression (2) implement linear interpolations between points at which it is specified that each term supports total subsumedness of SEGMENT-A by SEGMENT-B (*is-subsumed-by* = 1) or no subsumedness (*is-subsumed-by* = 0). The parameters γ_1 and γ_2 control, respectively, the degree of overlap considered negligible, and the degree of *fit-quality* required to consider some segment which is totally overlapped, significant nonetheless. (Values for free parameters γ used in the current implementation are listed in Appendix B.) Lowe [16] mentions an analogous mechanism that addresses the uniqueness consideration but treats overlap categorically and does not trade this factor off against fit-quality.

Figure 7 demonstrates our *is-subsumed-by* measure for a number of pairs of curve segments. Note that this measure is not commutative: in general *is-subsumed-by*(SEGMENT-A, SEGMENT-B) $\neq$ *is-subsumed-by*(SEGMENT-B, SEGMENT-A). Given an ensemble of segments, any segment can be compared against nearby segments to determine the degree to which it is subsumed by another, and may therefore be considered superfluous. Since this is a continuous-valued measure, different applications may tailor the use of the *is-subsumed-by* measure to particular ends, e.g. by choosing a threshold on when to remove segments from a set as discussed in Section 3.3.

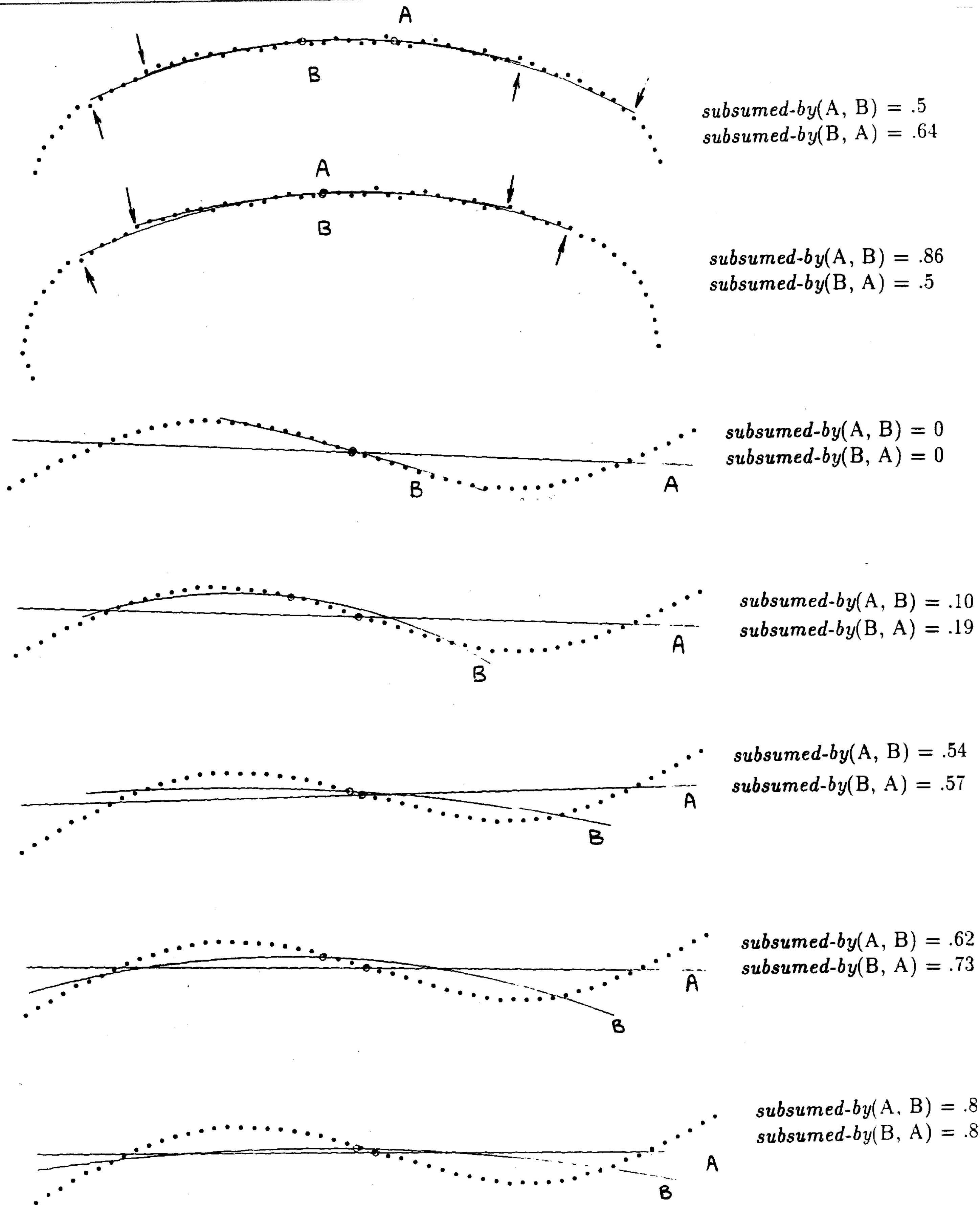


Figure 7: Examples of the *subsumed-by* measure.

2.3 Criterion 3: End Abruptness

A third criterion governing the apparent perceptual significance of a curve segment as modeled by a circular arc is motivated in figure 8. This factor takes account of the locus of the curve extending beyond the segment's ends. Roughly speaking, if the curve proceeds in a continuation of the circular arc fit to the segment in question, then that segment is seen as less salient than when the curve changes abruptly in direction or curvature immediately beyond the segment's bounds. This ***end-abruptness*** factor applies independently to each of the two ends of a curve segment. The ***End-abruptness*** factor is necessarily high if the segment boundary occurs at a termination of the curve, as shown in figure 8c.

Based on these considerations, we may formulate an expression for ***overall-end-abruptness*** as the minimum of the independent measures of ***end-abruptness*** at each of the ends of a segment, where ***end-abruptness*** $= 0$ indicates perfect continuation of the arc model and is considered not at all salient, and ***end-abruptness*** $= 1$ indicates severe change in direction and/or curvature and is considered maximally salient. In other words, smooth continuation of the circular arc model beyond ***either*** end of a curve segment is sufficient to veto that segment's overall salience.

For the measure of ***end-abruptness*** at one end of a segment (not coinciding with a curve termination), an initial heuristic formulation consists in tracing points (x, y) along the curve for some extension distance, h, ($h = \frac{\textit{segment-length}}{2}$ proves satisfactory) beyond the end of the segment, recording the maximum, $\textit{end-abruptness}_{c,max}$, of the expression (see figure 9a):

$$\textit{end-abruptness}_c(x, y) = \max(0, \min(1, \frac{\max(0, (\mid y' \mid - e))}{e + x' \tan \gamma_3})) \tag{4}$$

where e is the maximum deviation between the curve segment and the circular arc model (as in (1)). In practice it is advantageous for this purpose to employ an arc fit not to the entire segment but to the half of the curve segment containing this end. Expression (4) implements a linear interpolation between ***end-abruptness*** $= 0$ occurring when curve point (x, y) on the extension of the curve lies on the curve

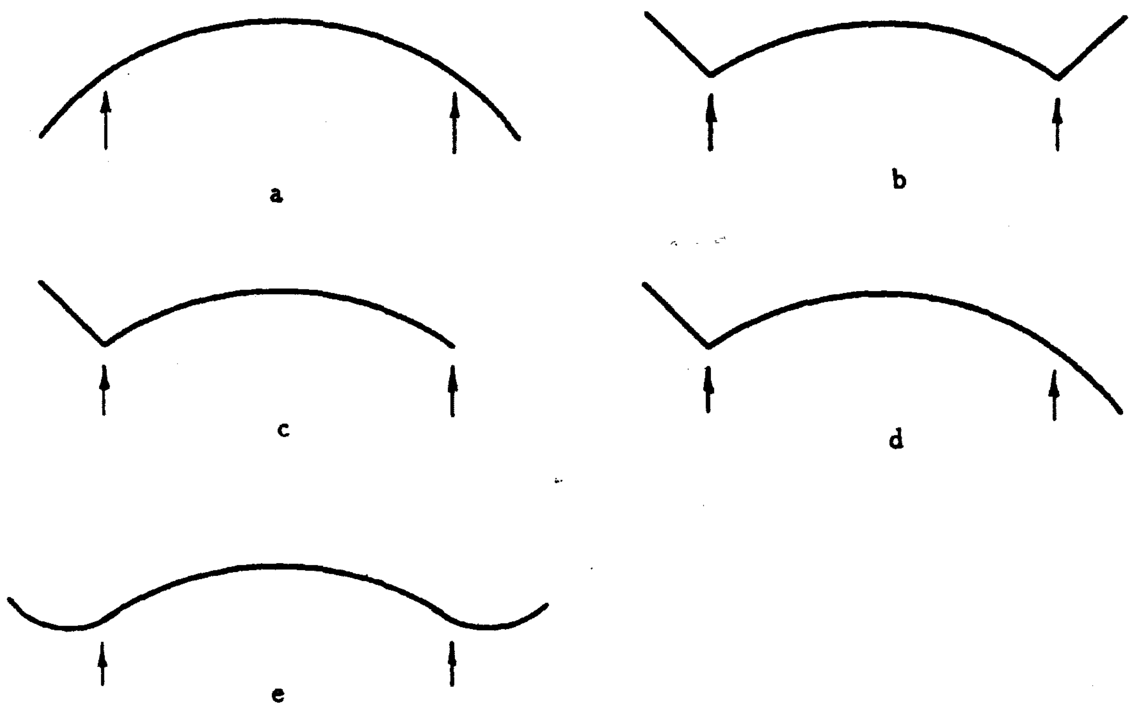


Figure 8: The perceptual salience of a segment depends not only on the properties of the curve within that segment but also on the locus of the curve beyond the segment's endpoints.

segment's circular arc model, and *end-abruptness* $= 1.0$ occurring when the curve veers off sharply to the left or right of the arc. The term e in this expression is introduced to permit more tolerance in deviating from a low *fit-quality* circular arc approximation than a perfectly fitting model.

The geometric simplicity of this formulation notwithstanding, empirical observation of many curves indicates that an additional factor must be taken into account in developing an expression adequately capturing the intent of *end-abruptness.* Figure 9b shows that rather shallow arcing curve segments appear to merge smoothly, without large apparent change in direction or curvature, with certain appropriately oriented nearly straight (very low curvature) curve extensions. We model this effect by introducing a second auxiliary term (see figure 9c):

$$\textit{end-abruptness}_s(x,y) = \min\left(1, \left(v + (1.0 - v) \left| \frac{y'}{e + x' \tan w} \right| \right)\right) \qquad (5)$$

where v and w represent the center value and angular width, respectively, of a "trough" centered on the linear extension of the circular arc in the direction indicated in the figure.

$$v = \begin{cases} \gamma_4 & \text{if } w < \gamma_5 \\ 1.0 & \text{otherwise} \end{cases} \qquad (6)$$

γ_5 is set so that the auxiliary term only takes effect for small w. As with *end-abruptness*$_{c,max}$, the value *end-abruptness*$_{s,max}$, is taken as the maximum of *end-abruptness*$_s$ over all points (x, y) on the curve out to distance, h, beyond the end of the segment. Finally,

$$\textit{end-abruptness} = \min(\textit{end-abruptness}_{c,max}, \textit{end-abruptness}_{s,max}). \qquad (7)$$

Figure 10 illustrates *end-abruptness* measures for a number of representative curves/curve-segments by displaying the points where *end-abruptness*$_{c,max}$ and *end-abruptness*$_{s,max}$ occur. As with the *fit-quality* salience criterion, *end-abruptness* is a unary property on individual curve segments that can be used, for example, to rank curve segments in order of significance along this dimension.

a

b

c

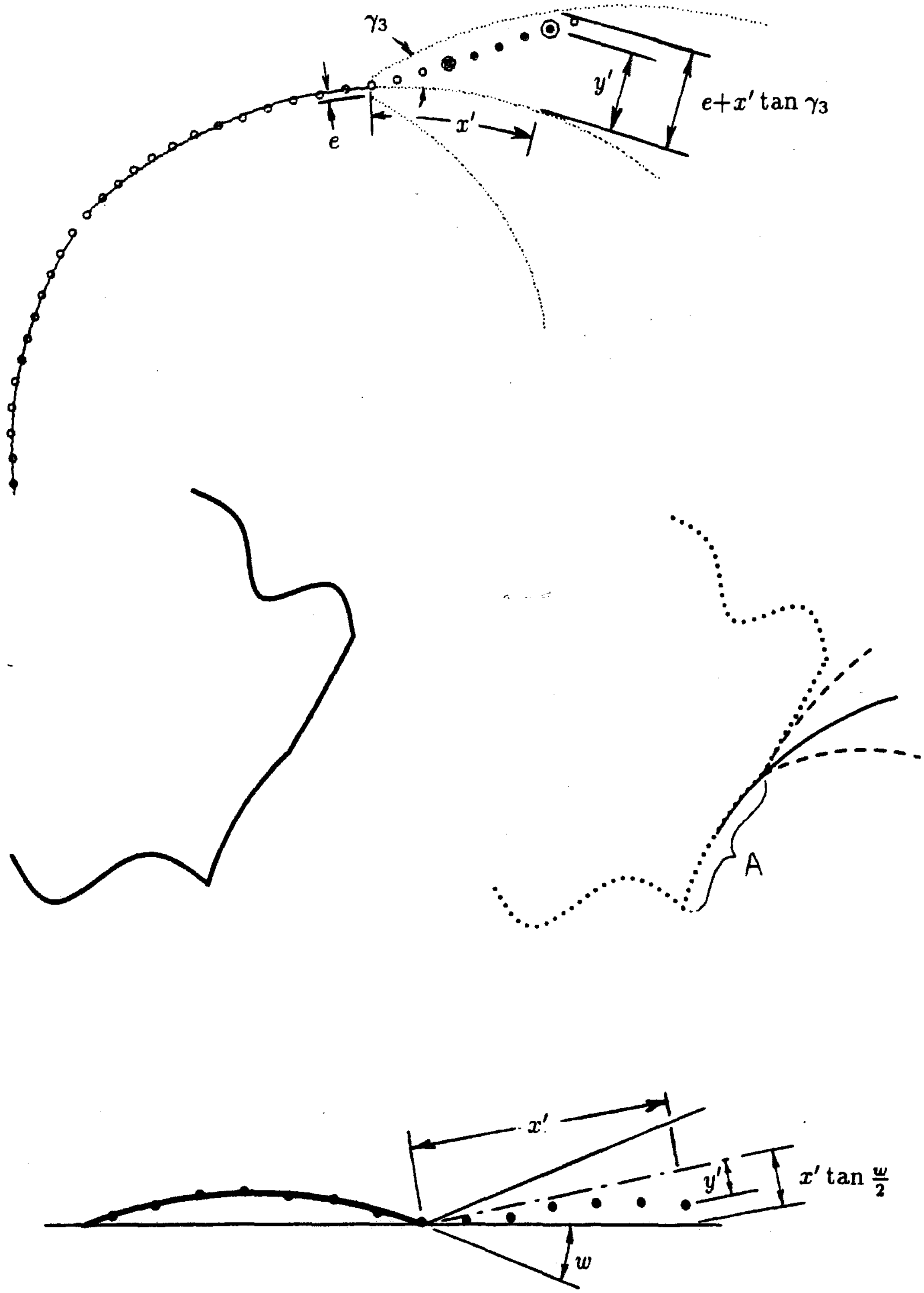


Figure 9: (a) Geometry of the basic ***end-abruptness*** measure, which assess the rapidity with which the sampled contour (open circles) deviates from a circular arc model beyond the end of a fit segment. (b) Under some conditions a curve can deviate strongly from continuation of a segment's circular arc fit (segment A) without that segment appearing to end abruptly. (c) An auxiliary measurement accounts for this effect by estimating the curve's deviation from a straight line (centerline notation) with orientation as shown.

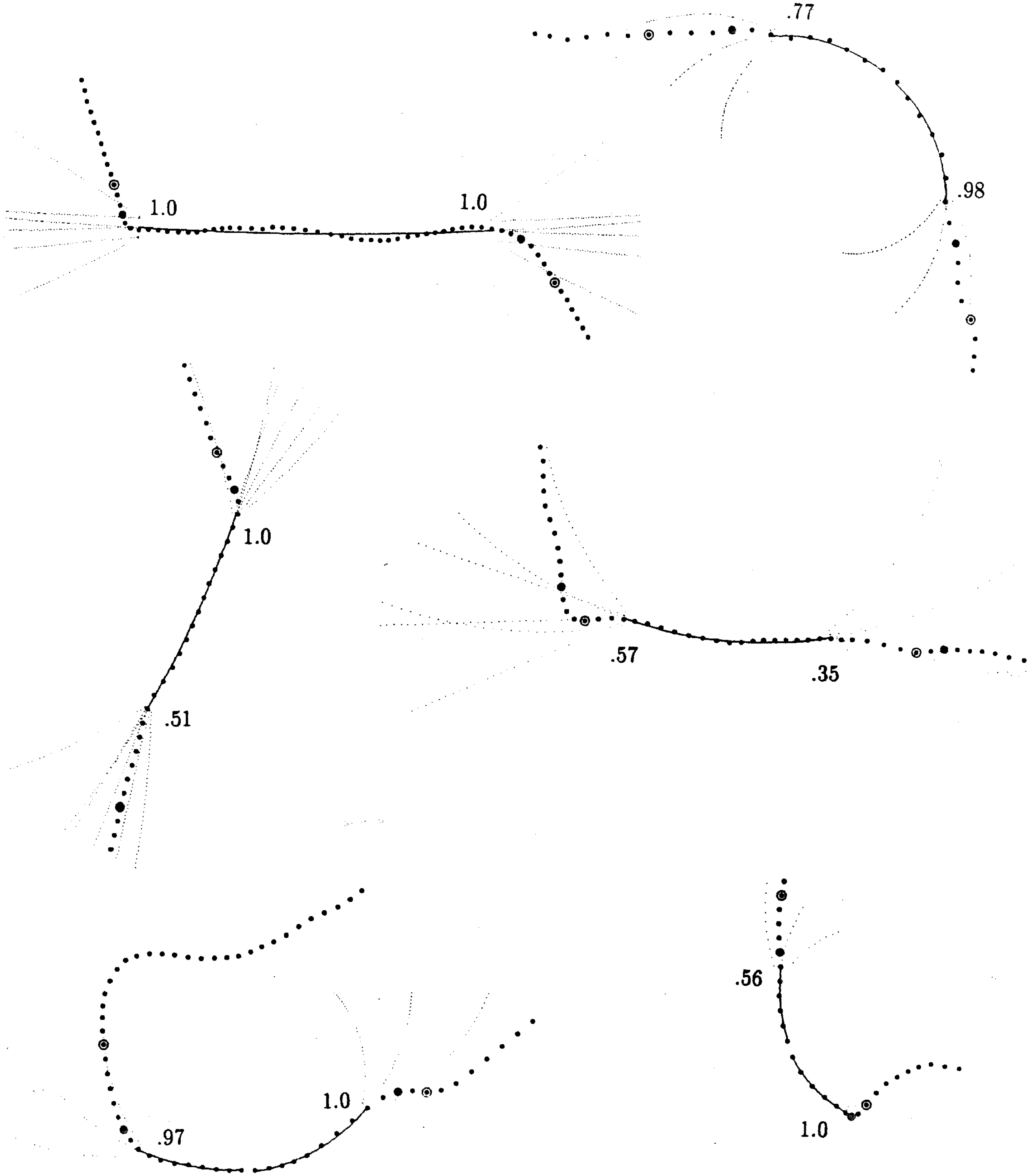


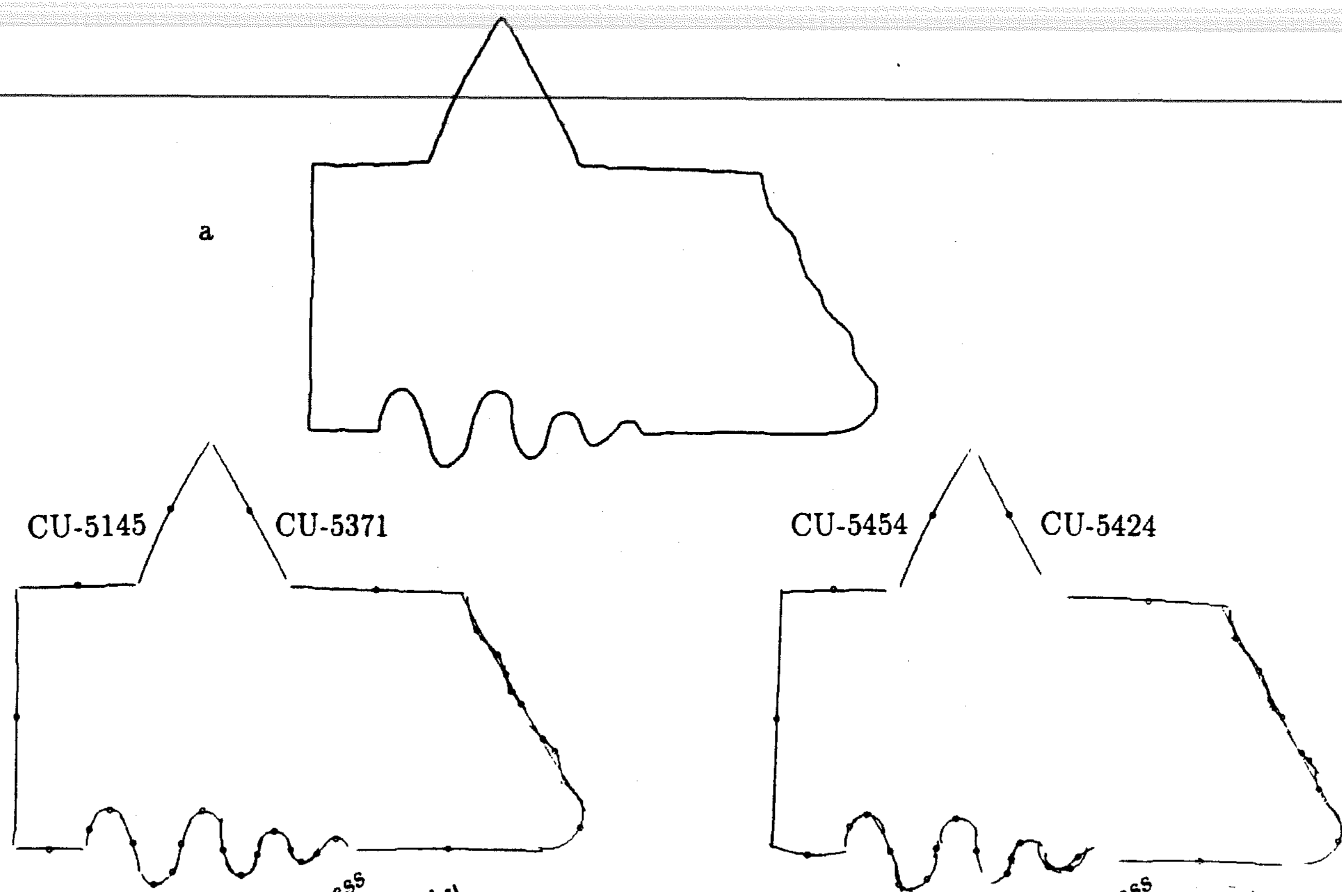
Figure 10: Numbers indicate ***end-abruptness*** measure at each end of the segment shown. ***End-abruptness*** ranges from 0 (not at all abrupt) to 1.0 (very abrupt). Circles indicate points on the curve where maximum ***end-abruptness*** occurs.

2.4 Usage and Examples

It is important to consider the above mathematical expressions in an appropriate context. These are not strict derivations, nor are they fits to sampled data, but they are merely attempts to engineer formal counterparts to subjective criteria judged important to assessing three dimensions of perceptual salience of curve segments. The effectiveness of these devices must be assessed ultimately by the builders of systems relying on the identification of significant segmentations of planar curves found in images. It lies beyond the scope of this paper to delve into issues of shape recognition or other particular applications using these segments, so for the time being it must be left for the reader to judge the degree to which the formal measures correspond with his or her perceptual intuition.

In order to see how these three criteria may be deployed, consider figure 11. Here, a test curve is presented along with four sets of segments (displayed as their associated circular arc fits) that might be selected to describe the curve. For each set, the figure indicates values of *fit-quality* and *end-abruptness* independently for each segment, plus degree of *is-subsumed-by* factor for pairs of overlapping segments. Segments in figure 11b were chosen such that each segment has high *fit-quality* and large *end-abruptness*, and so that no segment significantly subsumes any other. Segments in figure 11c may be viewed as slightly sloppy versions of the segments in 11b; *fit-quality* or *end-abruptness* are degraded, but pairwise *is-subsumed-by* factors are relatively unaffected. Figure 11d can be seen as an overly redundant representation of the original curve; extra segments are present that display strong *is-subsumed-by* factors with one another. Finally 11e is a too-sparse depiction of the curve; certain sections of the curve are not identified by circular arcs that, if they were, would have high degree of *fit-quality* and *end-abruptness* and low subsumedness with other segments.

A formal test of a candidate set of curve-segments thus amounts to using expressions (1), (2), and (7), to assess where there are redundant segments, where "good" segments are missing, and where segments are present but could be better positioned. None of these is an all-or-none decision; on the contrary, by supplying continuous val-



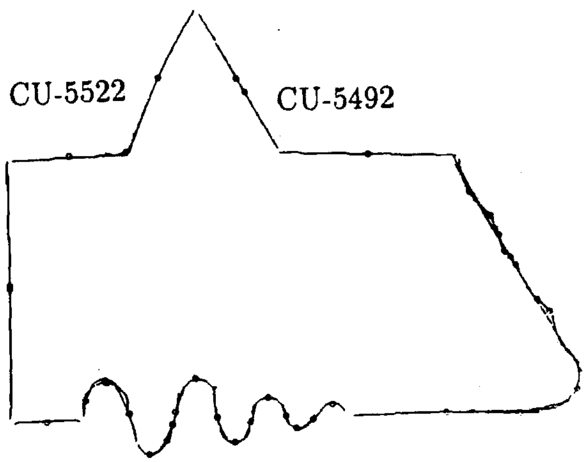
b

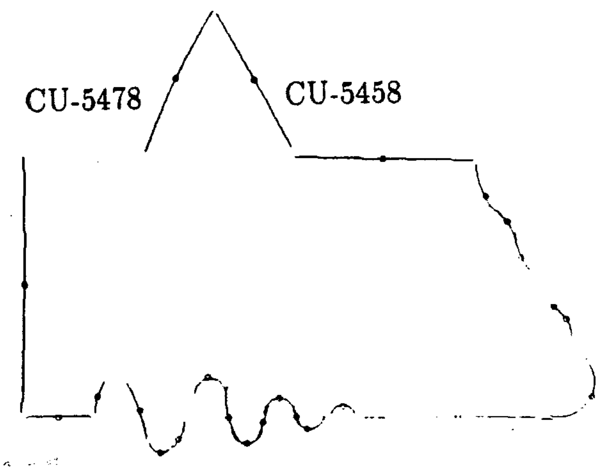
segment	fit-quality	end-abruptness	subsumed-by	
CU-5371	122.	1.0	-	
CU-5351	211.	1.0	-	
CU-5340	41.8	.935	CU-5311	0.21328814
CU-5339	75.4	.728	CU-5336	0.039061952
CU-5336	41.6	.674	CU-5333	0.4666667
CU-5333	59.1	.476	CU-5336	0.20175037
CU-5311	44.7	1.0	-	
CU-5326	35.4	.579	CU-5311	0.10027016
CU-5322	47.2	.695	CU-5311	0.053362675
CU-5313	35.1	1.0	CU-5311	0.19480809
CU-5299	43.9	.635	-	
CU-5286	137.	.800	-	
CU-5276	36.9	1.0	-	
CU-5274	66.0	1.0	CU-5269	0.06453106
CU-5269	25.5	1.0	-	
CU-5267	42.7	1.0	CU-5269	0.12656625
CU-5262	72.8	1.0	-	
CU-5257	27.5	1.0	-	
CU-5250	29.6	.513	-	
CU-5243	41.1	1.0	-	
CU-5236	69.8	.552	-	
CU-5230	59.8	1.0	-	
CU-5227	49.9	.145	-	
CU-5224	36.6	.461	-	
CU-5214	82.6	1.0	-	
CU-5210	47.7	.424	-	
CU-5203	62.7	0.7	-	
CU-5194	82.5	1.0	-	
CU-5174	223.	1.0	-	
CU-5154	92.8	1.0	-	
CU-5145	154.	1.0	-	

c

segment	fit-quality	end-abruptness	subsumed-by	
CU-5424	130.	1.0	-	
CU-5425	24.7	1.0	-	
CU-5426	59.2	1.0	CU-5433	0.07343065
CU-5427	31.0	.111	CU-5433	0.4222645
CU-5428	51.1	.316	CU-5429	0.14217523
CU-5429	22.0	0.2	CU-5433	0.46924755
CU-5433	45.4	1.0	-	
CU-5430	18.5	0.0	CU-5433	0.4737943
CU-5431	40.3	.202	CU-5430	0.17330183
CU-5432	37.6	.221	CU-5433	0.117307715
CU-5434	63.6	.793	-	
CU-5435	223.	.832	-	
CU-5436	19.5	.241	CU-5437	0.17469695
CU-5437	7.73	.061	CU-5438	0.1509679
CU-5438	7.12	.263	CU-5437	0.13043475
CU-5439	15.2	0.55	CU-5438	0.14972273
CU-5440	10.2	.414	-	
CU-5441	20.3	1.0	CU-5442	0.31877935
CU-5442	27.5	.243	CU-5441	0.125
CU-5443	46.8	.934	-	
CU-5444	16.0	.336	-	
CU-5445	26.7	0.0	CU-5446	0.0031796770
CU-5446	14.0	.102	-	
CU-5447	24.9	.315	CU-5448	0.071062304
CU-5448	15.0	1.0	-	
CU-5449	26.0	0.51	CU-5448	0.044480447
CU-5450	34.0	.267	CU-5449	0.113349095
CU-5451	0.39	.281	-	
CU-5452	23.3	1.0	-	
CU-5453	25.3	1.0	-	
CU-5454	150.	1.0	-	

Figure 11: (a) Test curve from [8]. (b) through (e) Sets of curve segments (indicated by their circular arc fits) that could be selected to describe the curve. Tables present *fit-quality* and *end-abruptness* measures for each segment, plus the name and *subsumed-by* measure for the most strongly subsuming other segment. *Fit-quality* $< \sim 15.0$ may be considered poor, and *fit-quality* $> \sim 40.0$ is good. *end-abruptness* $= 1.0$ is high, and *end-abruptness* $< \sim .5$ is low. *subsumed-by* measure $< \sim .5$ indicates low redundancy, *subsumed-by* measure $> \sim .8$ indicates high redundancy. Segments are presented in order proceeding clockwise around the curve. (c), (d), and (e) are, respectively, sloppy, redundant, and overly sparse versions of the segments in (b).





d

segment	fit-quality	end-abruptness	subsumed-by	
CU-5492	122.	1.0	-	
CU-5490	65.2	1.0	CU-5492	0.91397846
CU-5493	211.	1.0	-	
CU-5494	41.8	.897	CU-5501	0.21320814
CU-5495	75.4	.707	CU-5496	0.039061952
CU-5496	41.6	.635	CU-5497	0.4666667
CU-5497	59.1	.767	CU-5496	0.20175037
CU-5484	41.3	1.0	CU-5501	0.9374272
CU-5501	44.7	1.0	-	
CU-5498	35.4	.503	CU-5501	0.10027016
CU-5499	47.2	.573	CU-5501	0.053362675
CU-5500	35.1	1.0	CU-5501	0.19400809
CU-5502	43.9	.926	-	
CU-5485	36.0	0.3	CU-5503	0.39999998
CU-5503	137.	.946	-	
CU-5504	36.9	1.0	-	
CU-5505	52.6	1.0	CU-5506	0.09226164
CU-5506	25.5	1.0	-	
CU-5507	35.0	.963	CU-5506	0.17299056
CU-5508	72.8	1.0	-	
CU-5509	27.5	1.0	-	
CU-5510	29.6	.808	-	
CU-5511	41.1	1.0	-	
CU-5512	69.8	.577	-	
CU-5513	59.8	1.0	CU-5486	0.2572218
CU-5486	31.2	.708	CU-5513	0.47826087
CU-5514	49.9	.337	CU-5486	0.23955834
CU-5515	36.6	.686	-	
CU-5516	82.6	1.0	CU-5487	0.007783541
CU-5487	18.3	.641	-	
CU-5517	47.7	0.21	CU-5487	0.06460365
CU-5518	55.1	.723	CU-5487	0.0231473
CU-5519	82.5	1.0	-	
CU-5520	223.	1.0	-	
CU-5488	202.	1.0	CU-5520	0.97416216
CU-5521	92.8	1.0	-	
CU-5489	21.5	.187	-	
CU-5522	154.	1.0	-	

e

segment	fit-quality	end-abruptness	subsumed-by	
CU-5458	122.	1.0	-	
CU-5459	211.	1.0	-	
CU-5460	41.8	.897	-	
CU-5461	75.4	.707	-	
CU-5462	59.1	.767	-	
CU-5463	35.4	.503	CU-5464	0.08333337
CU-5464	47.2	.573	-	
CU-5465	43.9	.926	-	
CU-5466	137.	.946	-	
CU-5467	36.9	1.0	-	
CU-5468	25.5	1.0	-	
CU-5469	35.0	.963	CU-5468	0.17299056
CU-5470	72.8	1.0	-	
CU-5471	27.5	1.0	-	
CU-5456	29.6	.808	-	
CU-5455	41.1	1.0	-	
CU-5472	69.8	.577	-	
CU-5457	49.9	.337	-	
CU-5473	36.6	.686	-	
CU-5474	82.6	1.0	-	
CU-5475	55.1	.723	-	
CU-5476	82.5	1.0	-	
CU-5477	223.	1.0	-	
CU-5478	154.	1.0	-	

ued measures of their respective properties they reflect the ambiguity resultant from applying idealized models. Even so, they serve as effective guidelines for developing algorithms intended to pick out segments of curves that stand out clearly to any human observer yet to which machines have heretofore been oblivious.

3 Algorithm

A computer algorithm attempting to return a set of curve segments with high *fit-quality*, high *end-abruptness*, and low pairwise *is-subsumed-by* factors must be designed to optimize not only quality of results, but also computational efficiency in time and memory. The algorithm presented here reflects this tradeoff by first preselecting, then further pruning, a set of candidate segments that is typically several times larger than the number of segments finally returned, but very much smaller than the $\frac{N(N+1)}{2}$ segments potentially available to check. The basic procedure is outlined here, elaboration follows below:

1. Derive smoothed and subsampled versions of the original curve, forming representations suited to analysis at all spatial scales.

2. Discover candidate segments by fitting straight lines to the orientation vs. arc-length representations at all scales; calculate *fit-quality* parameter for each of the candidate segments.

3. Compare overlapping candidate segments pairwise and remove segments in order of the degree to which they are subsumed by another candidate, up to a chosen threshold degree of subsumedness.

4. Remove remaining segments falling below threshold values of *fit-quality* and/or *end-abruptness*. Return all surviving candidates.

Two crucial considerations guide the design of this algorithm. First, in order detect circular arcs of all lengths and radii in curves of any magnification, the algorithm must prefer no spatial scale, but instead operate uniformly across all scales. Thus motivates the scale-space [35] computation of step 1. Second, as much work as possible is done in the orientation vs. arc-length representation. Since a circular arc in the plane corresponds to a straight line in this representation, the algorithm can search for a particularly simple type of structure in this signal; furthermore, it is computationally much less work to deal with a single one-dimensional array than the geometry of curves and circular arcs in two dimensions.

3.1 Step 1: Smooth and Subsample the Curve

After preprocessing by the algorithm in Appendix A to remove any city block pixel sampling effects, the original curve is smoothed with a truncated Gaussian kernel ($\sigma = \gamma_6$). We employ Lowe's [17] methods for performing smoothing out to the end of a curve and for correcting for curve shrinkage, although straight Gaussian smoothing in lieu of the latter seems not to impact the results a great deal. The result is labeled as the Scale 0 signal and stored for later use, and is then subsampled by a factor of two to yield the input for another Gaussian smoothing pass (identical kernel as above) which in turn yields a Scale 1 signal. Subsampling and smoothing steps alternate until the number of samples remaining in the signal is less than the kernel width of the truncated Gaussian. This procedure results in a scale-space representing the original curve at different resolutions. The tangent direction between successive points is computed to yield orientation vs. arc-length θ-S representations at all scales. See figure 12.

The subsampling step serves two functions. First, it greatly reduces processing load compared to applying increasingly large Gaussians to the original signal (for an input curve of 1000 points the entire procedure takes only about 3 times as long as the first Gaussian smoothing pass). Second, it serves to normalize first difference in orientation (curvature) so that different magnifications of the same contour give rise to virtually identical representations that vary only by a shift along the scale index of the contents of the orientation vs. arc-length arrays. This allows the remaining processing steps to operate essentially uniformly across all scales.

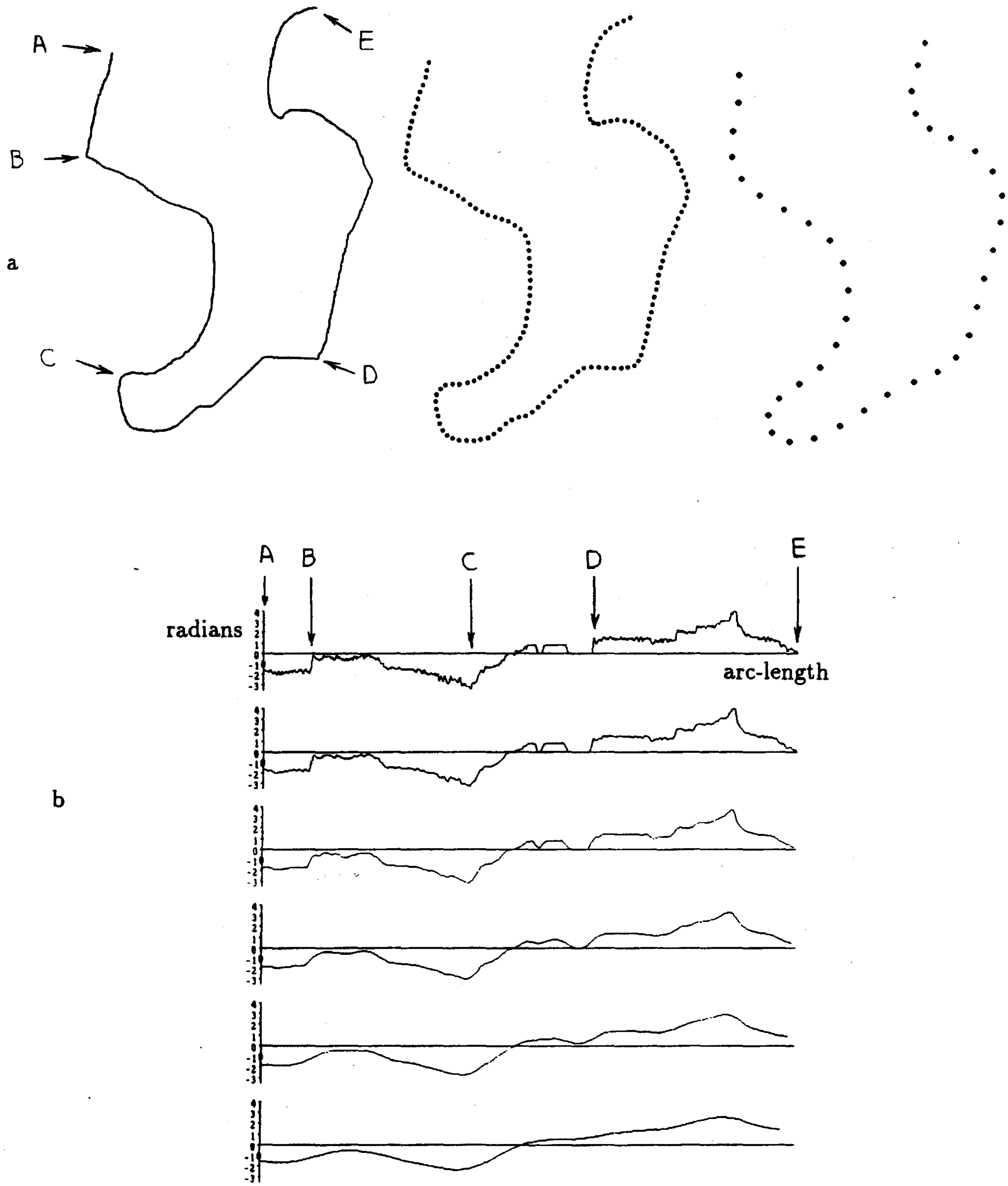


Figure 12: (a) Test curve, plus smoothed and subsampled versions of this curve at two coarser scales. (b) θ-S representation of the test curve across scales.

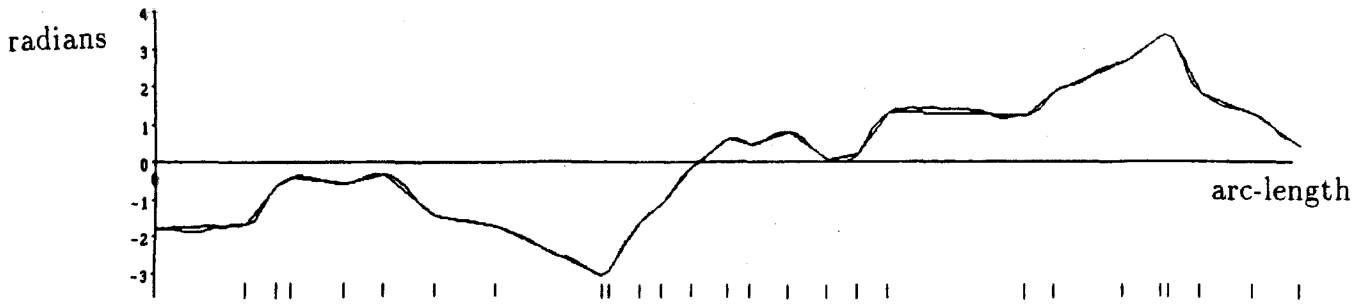
3.2 Step 2: Fit Straight Lines in Orientation vs. Arc Length

For every scale orientation vs. arc-length array, a two-step process is used to identify locally straight lines which will correspond to circular arcs in the original curve. These are (see figure 13):

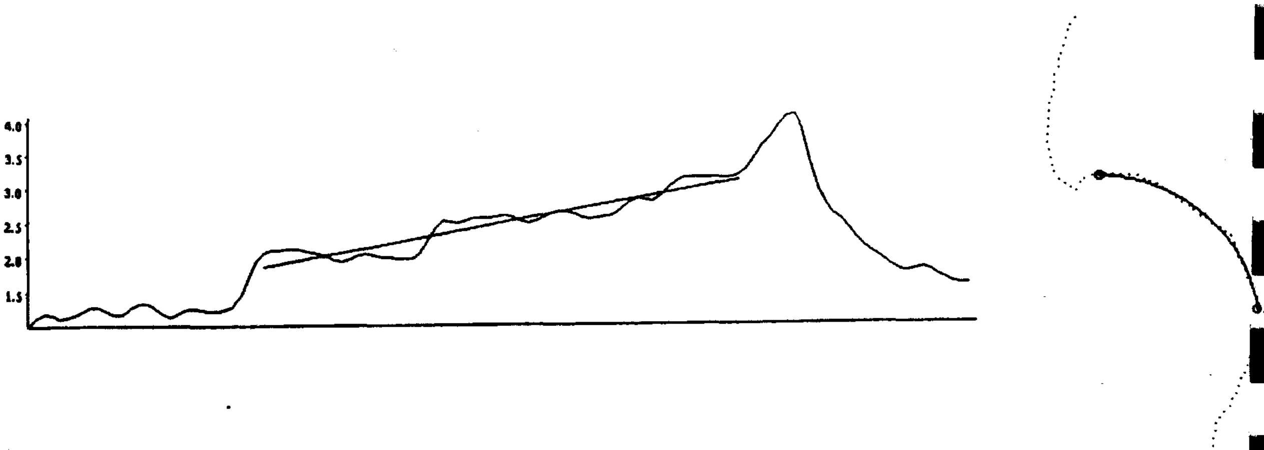
2a. Perform split-and-merge segmentation [23] to locate initial candidates for end-points of straight lines in θ-S space as first suggested by Grimson [10]. This is done with a very tight error tolerance $\epsilon = \gamma_7$ so as to ensure that every conceivable line break point is found. These segments are combined (union operation) with a second set of segments delivered by next performing just the merge step of the split-and-merge algorithm, this time backing off to a somewhat looser tolerance $\epsilon = \gamma_8$ so as to detect longer, more imperfectly straight lines. Each resulting segment is adjusted by least-squares to optimally fit its section of the θ-S space curve. A final merge step detects pairs of segments forming a shallow upright or inverted "v," corresponding to inflections in the 2D input curve.

2b. For each line segment in θ-S space, a segment-growing procedure is performed. First, an estimate is made of the degree of match or fit between between the straight line segment and the corresponding segment of the θ-S signal. From this match estimate is derived an adaptive threshold, t. The segment is tentatively extended one point to the left, and one point to the right, and, after updating the least-squares fit including these tentative extensions, the match estimates are recomputed. If both the left-extended and right-extended match estimates exceed the threshold t, then the segment-growing procedure terminates. Otherwise, the segment is extended permanently in the direction of best match estimate, and the procedure repeats. Finally, all segments longer than a threshold ($2\gamma_6$) number of (subsampled) points are returned.

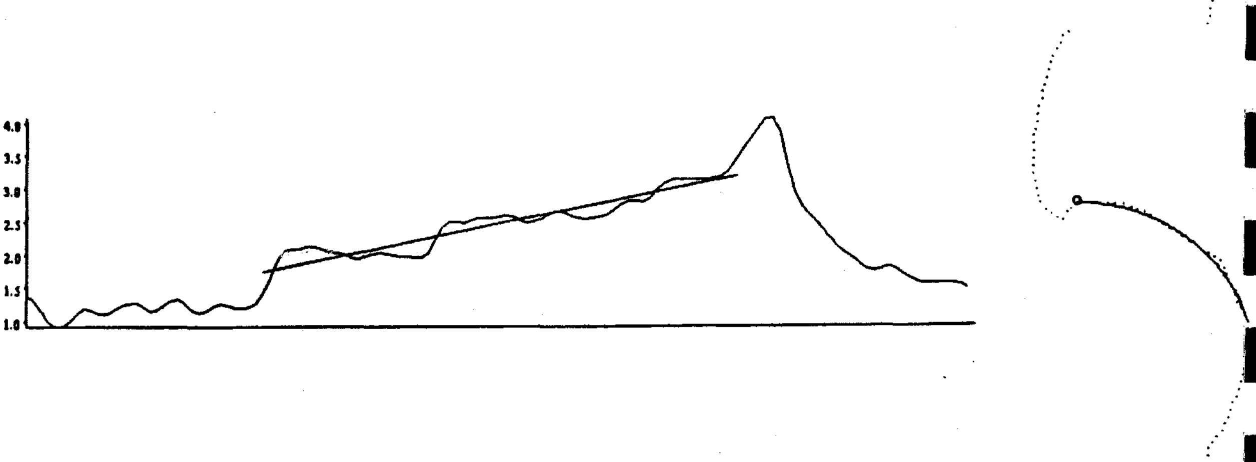
The result is a set of line segments in θ-S space, each of which delimits a candidate segment in the input curve, and each of which is computed so as to correspond to an approximately circular arc in this curve. Because this procedure is performed



a



b



c

Figure 13: (a) Line segments in θ-S space after split-and-merge segmentation (b) Least-squares fit to θ-S curve for one of these segments, and corresponding circular arc on the 2D test curve. (c) This segment after segment-growing.

independently at all scales of θ-S, a given approximately circular arc section of an input curve will quite often give rise to several more or less equivalent candidate segments. For rather precise circular arcs, the segments found by finer scales of analysis will typically deliver the best estimates for the endpoints of the segments, while roughly formed arcs will be detected only at the larger scales.

It is worth considering why it is necessary to perform candidate segment detection on the θ-S spaces at different resolutions of the input curve in scale-space, and not, say, simply to smooth the finest scale θ-S representation with different size kernels. Figure 14a shows that spatially small blips in a curve can nonetheless give rise to large bumps in θ-S space that obscure true large scale spatial events such as gradual changes in curvature. A different shortcut might consist in attempting to bypass the split-and-merge step by detecting sudden changes in orientation or curvature directly in θ-S space. While this strategy cannot be completely ruled out, we point out the difficulty it faces in detecting segments such as figure 14b, for which one end is delimited by change in curvature at a coarse scale while the other end is defined by a change in orientation at a fine scale which is absent in the coarse scale signal. Both the smooth and noisy versions of this type of segment score highly on all three perceptual salience criteria developed in Section 1, but to identify both is beyond the capacity of all previously reported algorithms we are aware of.

3.3 Step 3: Prune Redundant Segments

At this stage all candidate segments found through analysis of θ-S space are transformed into circular arc fits of the original input curve. Curve fitting is performed as as discussed in section 2, and the *fit-quality* parameter is measured. In practice, because of pixel quantization effects in images it is not uncommon to encounter perfectly straight curve segments. To avoid complications due to the corresponding interpretation that these have infinite *fit-quality*, we compute this parameter using the following

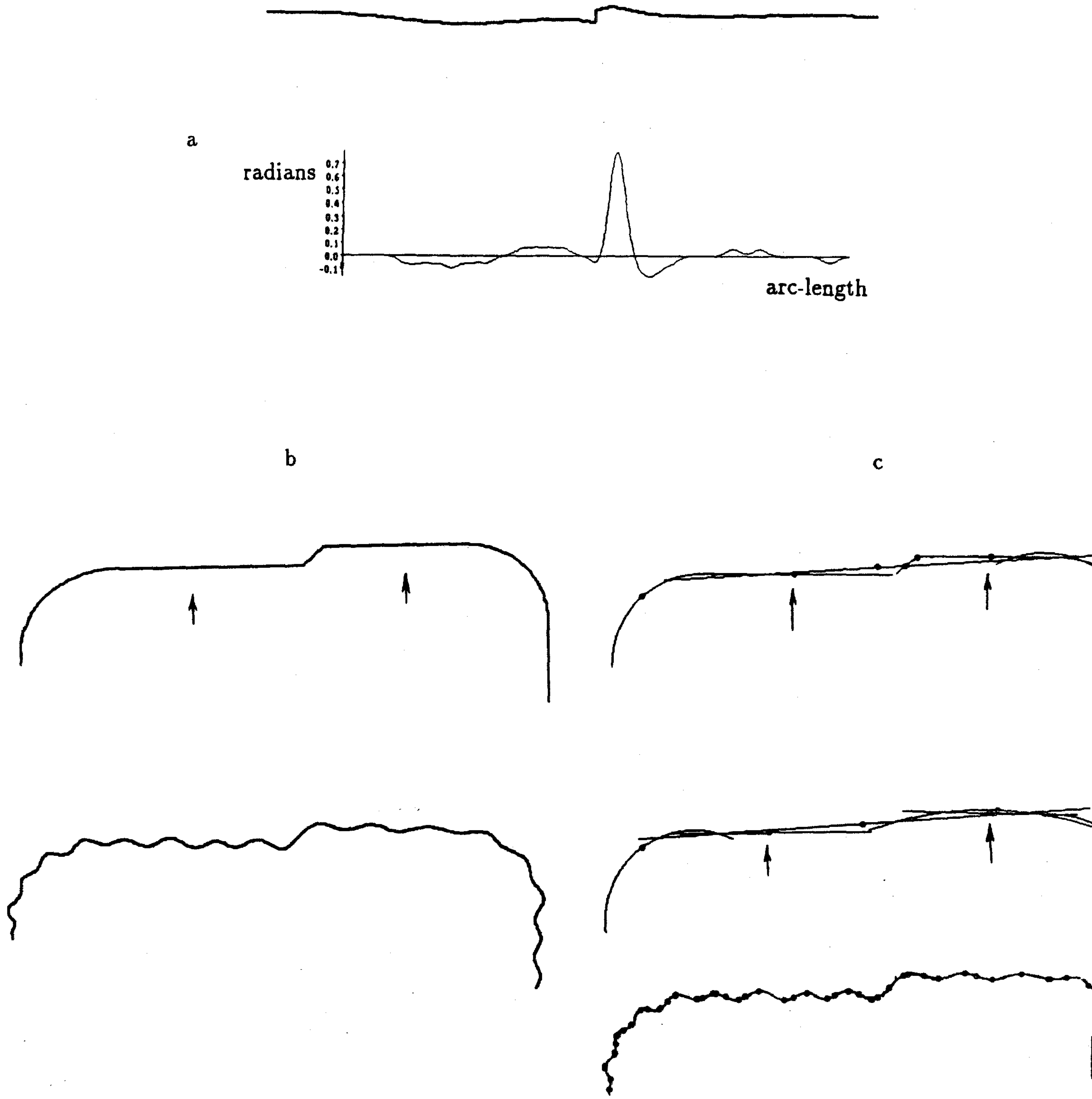


Figure 14: (a) Curve for which a spatially small feature gives rise to a large feature in θ-S space. (b) Curves containing salient segments (indicated by arrows) bounded by a purely large scale feature (smooth join) on one end and a purely small scale feature (ramp step) on the other. (c) Segments returned by the algorithm. For the wiggly curve, circular arcs representing small scale and large scale segments are displayed separately for clarity.

modification of expression (1):

$$\textit{fit-quality} = \frac{l}{\max(\gamma_9, e)} \tag{8}$$

Next, overlapping segments are detected after sorting by location from one end of the input curve, for efficiency. For each pair of overlapping segments, ***is-subsumed-by*** factors are computed as described in Section 2.2. All segments are removed that are completely subsumed by another segment (***is-subsumed-by*** factor = 1). Remaining segments are rank-ordered by the maximum amount they are subsumed by some other segment. Segments are removed in order of decreasing max(***is-subsumed-by***) factor, taking care to note that any segment removed from consideration can no longer subsume any other segment still under consideration. Segment removal proceeds until some threshold γ_{10} on subsumedness is reached, at which point no remaining segment is subsumed by any other to a degree greater than this threshold.

3.4 Step 4: Remove Poor Quality Segments

End-abruptness is measured as described in Section 2.3, with one minor modification. In executing expressions (4) and (5), we sample points (x, y) not from the input curve, but instead from the smoothed version of the input curve in which the segment was detected. This modification removes artifacts in the estimate of ***end-abruptness*** that arise on very wiggly or noisy curves. The smoothed input curve also supports computation of a ***smoothed-fit-quality*** parameter, analogous to the previously described ***fit-quality*** measure, that is useful later in assessing the degree to which the segment's gross shape fits a circular arc independent of small scale wiggles or noise it may possess. Finally, a coarse thresholding step removes curve segments whose ***fit-quality*** or ***end-abruptness*** falls below respective thresholds γ_{11} and γ_{12}.

4 Results

Because curve segments possess graded values of ***fit-quality***, ***end-abruptness***, and *subsumed-by* factors, the algorithm is liberal in returning nearly all conceivably salient segments. Alternate ways are possible of further refining these. For example, in figure 15 the full cadre of 130 even minimally salient curve segments returned by the algorithm are reduced to the 42 shown in 15b and 15c by applying a threshold on a *composite-salience* measure:

$$\textit{composite-salience} = \textit{smoothed-fit-quality} \times \textit{end-abruptness} \tag{9}$$

It is characteristic of heuristic algorithms such as this one that results are affected by parameters whose settings are adjusted in a seat-of-the-pants fashion. What seems to work well on one image may yield suboptimal results on another. Furthermore, because of the intuitive nature of "perceptual salience," judgment of the quality of results may differ from person to person. All results presented in the paper employ the same fixed internal parameter settings for the algorithm. Figures 16 through 20 are presented for the reader to judge the degree to which the results match his or her perceptual intuition.

Failures of the algorithm can be of two types. First, the algorithm can fail to identify curve segments that would be considered qualified with regard to *fit-quality*, *end-adjustment*, and *subsumed-by* measures. This occurs for example at the location indicated by the arrow in figure 16. In this case the algorithm for finding candidate segments by locating straight lines in orientation space delivers the segment shown in figure 16c, whose endpoints extend not quite far enough to bring *fit-quality* above the threshold which was set on the basis of testing the algorithm on other images. One could change the threshold or further tweak the segment-growing module of the algorithm, but it is impossible to ensure that other failures will not occur in other marginal situations.

This type of mistake can be considered a failure in *implementation* of a "theory" of perceptual salience reflected in the three criteria of section 2. The second type of failure is due to weakness of this theory itself; cases may be found where the *fit-quality*, *end-abruptness* and *subsumed-by* measures do not correspond with perceptual intuition. Such a situation occurs in figure 17, which illustrates the fact that sometimes curve segments are made salient due to their "texture," which is not accounted

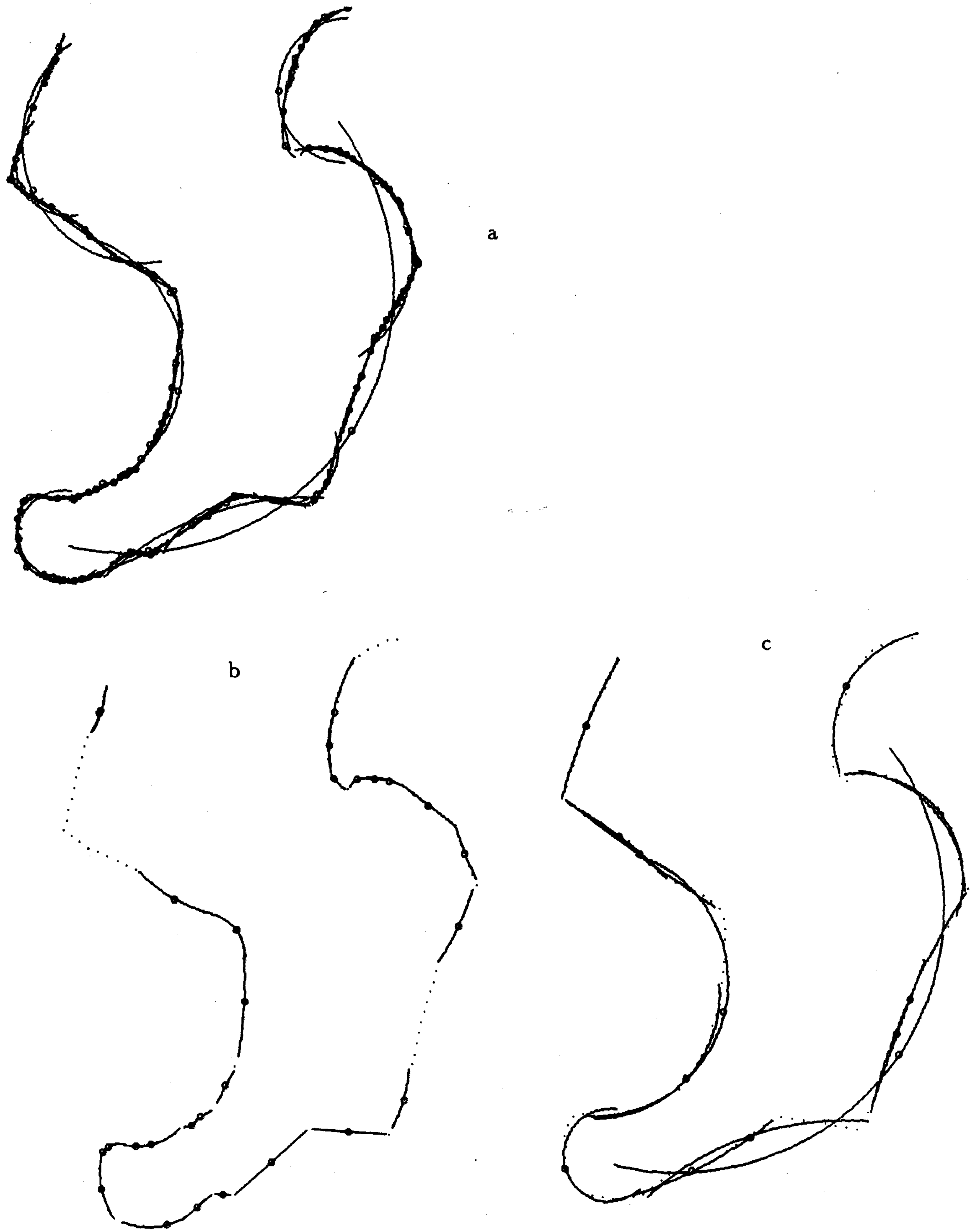


Figure 15: (a) Circular arcs depicting the 130 curve segments returned by the algorithm on the test shape of figure 12. (b) and (c) Arcs depicting smaller and larger scale segments, respectively, resulting from thresholding the segments in (a) at a level of 30.0 in *composite-salience* (see text).

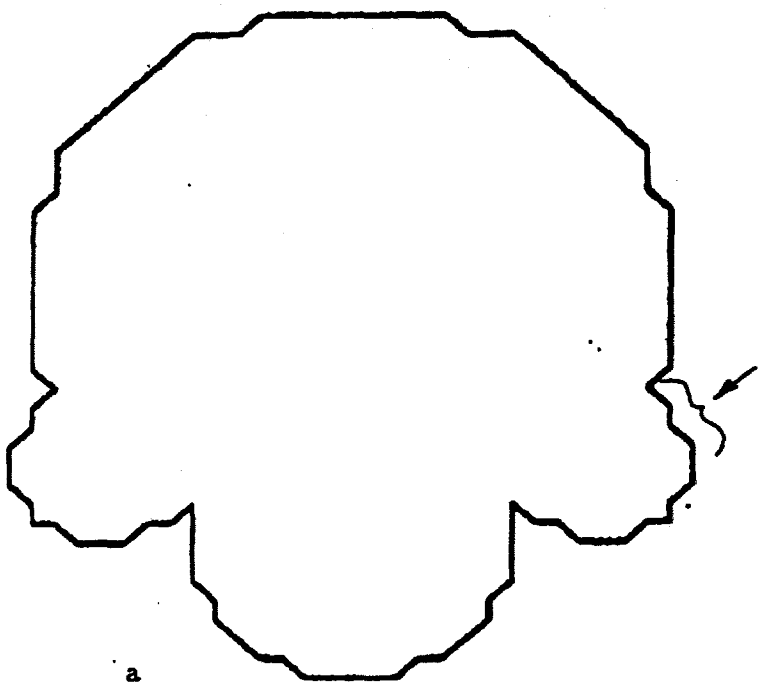
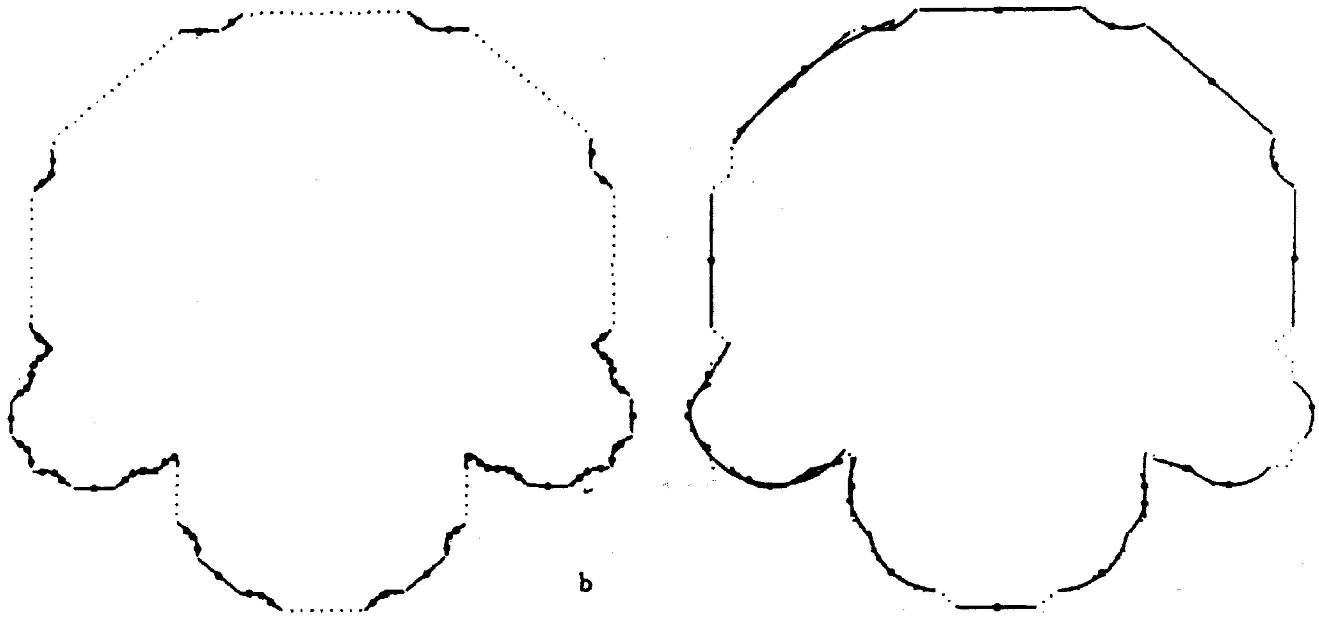
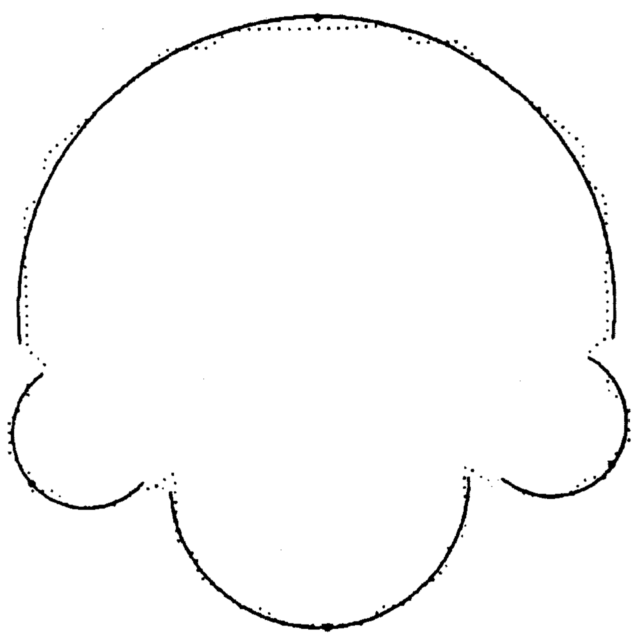
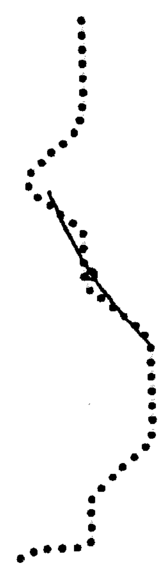


Figure 16: (a) Test curve used by Teh and Chin (see figure 6 of [30]). (b) Circular arcs depicting curve segments with *end-abruptness* > .5 returned by the present algorithm. Small, medium, and large scale segments are shown separately to highlight significant structure found at different scales. (c) Magnified section of the curve where the algorithm rejected the potentially salient segment shown because its *fit-quality* fell below a previously set threshold.



b





c

for by our criteria. Perhaps another criterion could be added to account for this phenomenon, but we do not attempt to do so here.

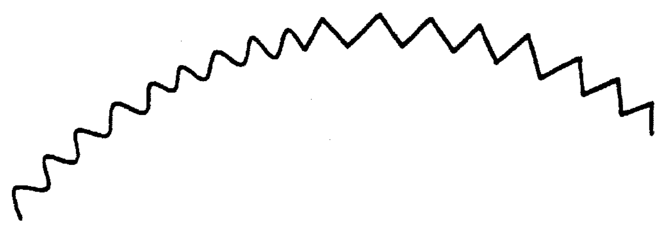
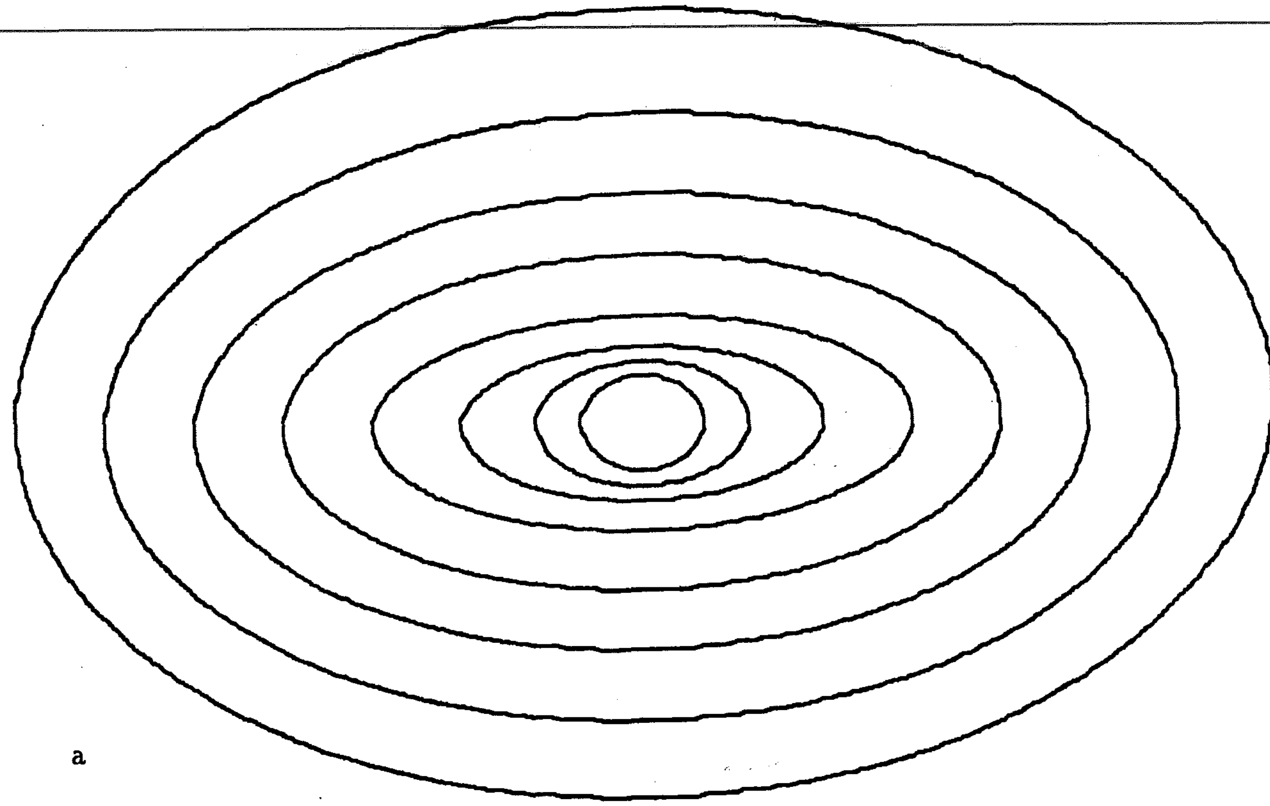


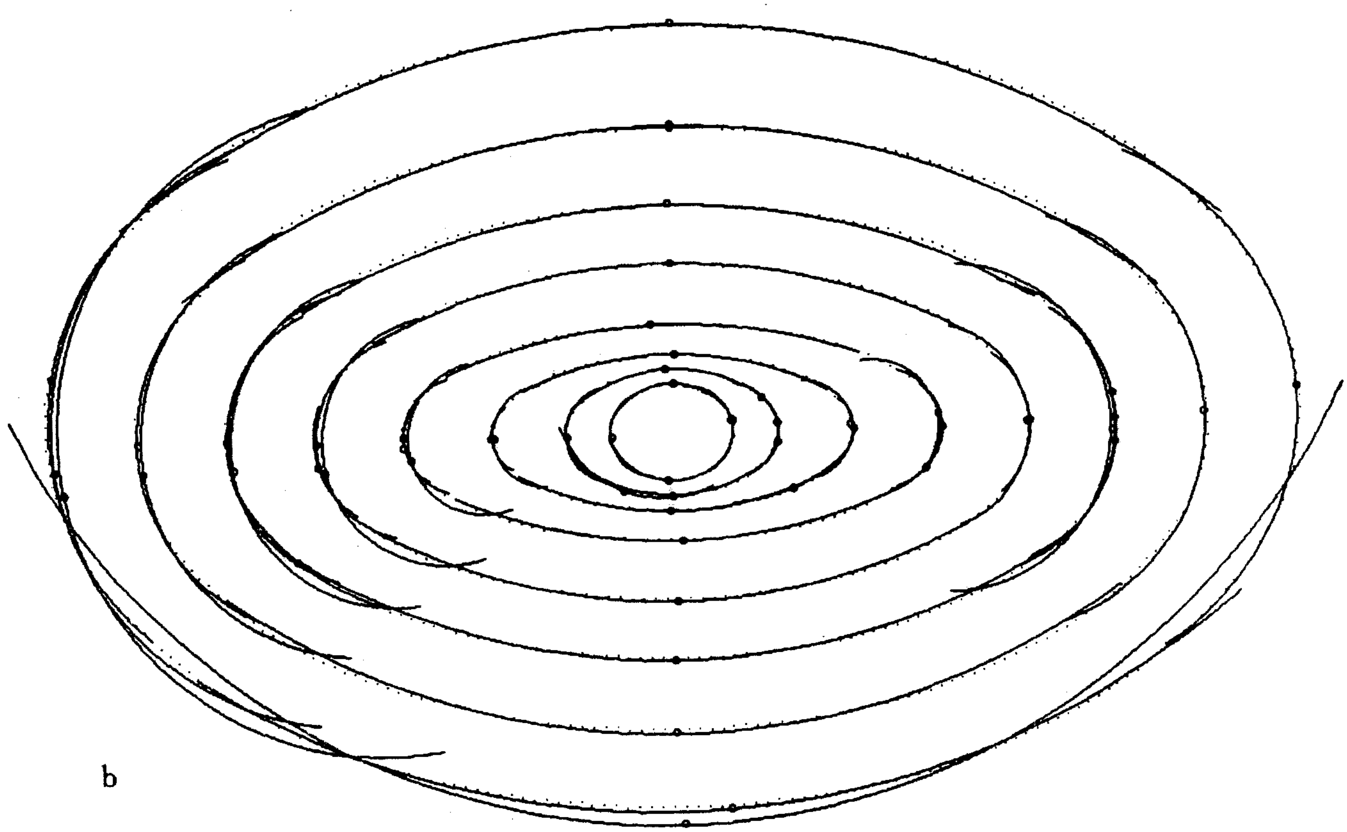
Figure 17: The three criteria of Section 2 do not account for segments made perceptually salient due to their contour texture properties.

Figure 18 illustrates the algorithm's self-similarity with respect to scale; arcs are found in comparable places on ellipses of all sizes. Finally, figures 19 and 20 show curve segments found on edge maps derived from real images.

The algorithm's computational requirements are dominated by the effort required to fit circular arcs to candidate curve segments and compute their ***fit-quality*** and ***end-abruptness*** measures. Total compute time is 70 seconds for the test curve of figure 12 (1270 image curve points) on a mid-1980's era workstation running Lisp.

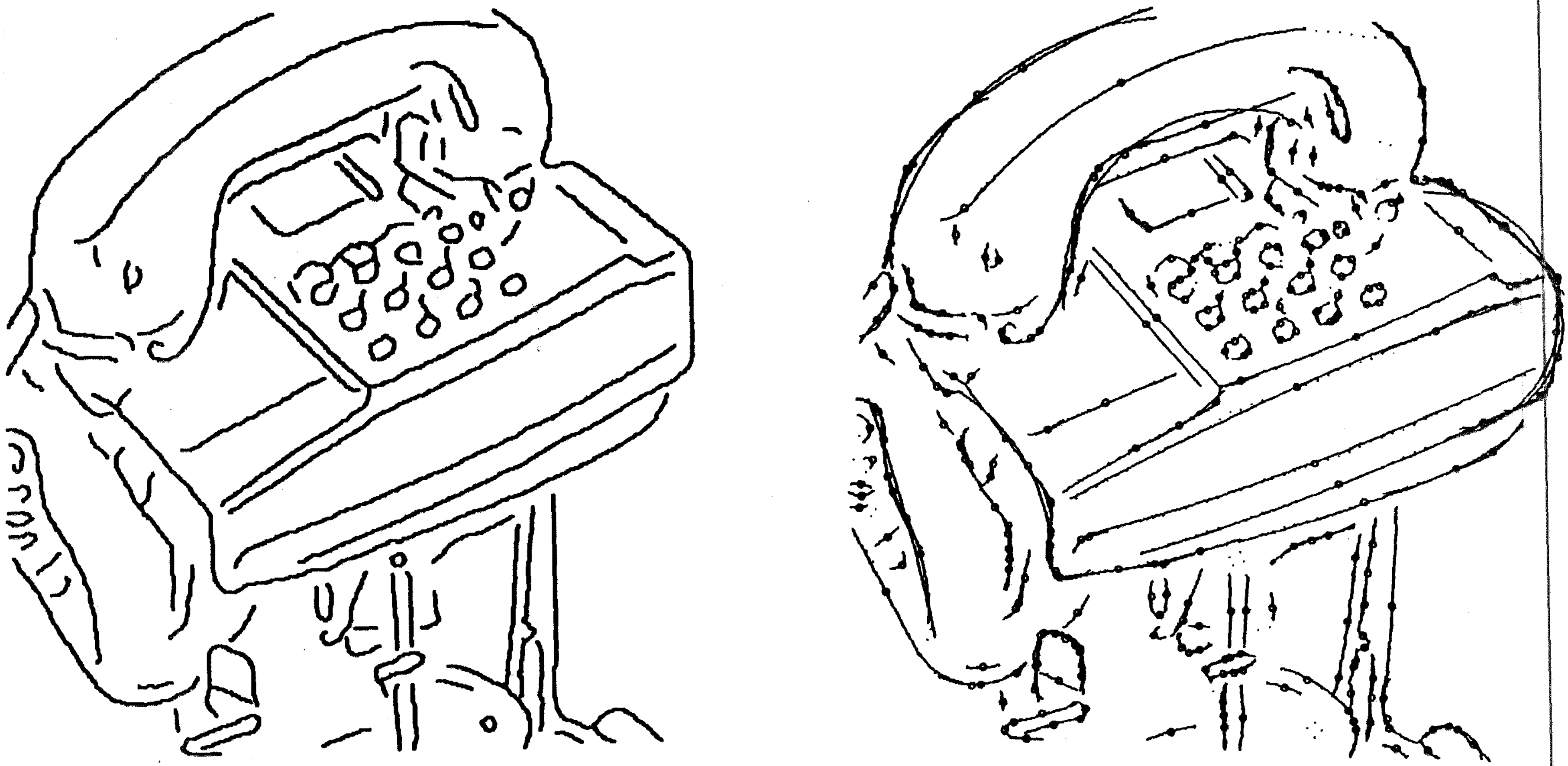


a



b

Figure 18: (a) "Ellipses" test image due to A. Etemadi. (b) Circular arcs depicting curve segments with overall-salience > 15.0 returned by the algorithm. Note similarity of results for ellipses of different sizes.



a

b

Figure 19: (a) "Phone" test image due to A. Etemadi. (b) Circular arcs depicting all curve segments returned by the algorithm (312 segments).

a

b

c

Figure 20: (a) Test image due to Lowe [17]. (b) and (c) Circular arcs depicting salient curve segments with *end-abruptness*> .5 at smaller and larger scales, respectively (769 segments).

5 Appendix A: Curve Preprocessing Algorithm

This algorithm converts a linked list of points spaced unevenly along a plane curve into a list of evenly spaced points approximating the original curve. This preprocessing step removes city block pixel quantization effects of four-connected or eight-connected curves prior to working with the θ-S representation. See figure 21.

```
initialize output-curve(0) ← input-curve(0)              ;first point of output curve
initialize i ← 1                                           ;input curve index
initialize o ← 0                                           ;output curve index
initialize d ← euclidian-distance(output-curve(o),input-curve(i))
loop until (i > curve-length) doing
    loop until ((d > γ13) or (i > curve-length)) doing
        i ← (i + 1)
        d ← euclidian-distance(output-curve(o), input-curve(i))
    endloop
    loop until (d < 1.0) doing
        output-curve(o+1) ← [unit-step from output-curve(o) in direction from
                                            output-curve(o) to input-curve(i)]
        d ← euclidian-distance(output-curve(o+1), input-curve(i))
        o ← o +1
    endloop
endloop
```

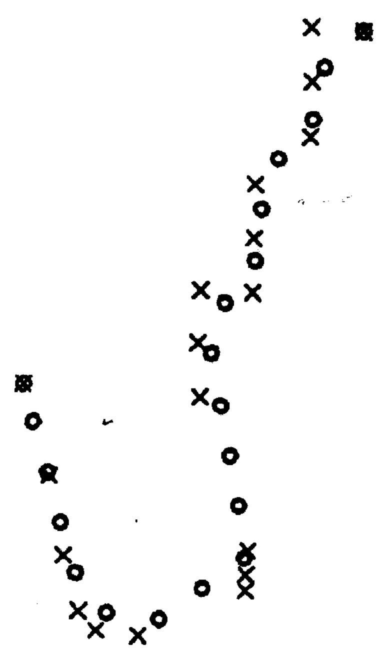


Figure 21: X's: sample points on a test curve. Open circles: points returned by the curve preprocessing algorithm.

6 Appendix B: Values of Free Parameters

The following are values of the free parameters of the perceptual salience criteria and curve segment identification algorithm used in the current implementation.

parameter	value	use
γ_1	.5	*subsumed-by* measure
γ_2	.1	*subsumed-by* measure
γ_3	30°	*end-abruptness* measure
γ_4	.2	*end-abruptness* measure
γ_5	20°	*end-abruptness* measure
γ_6	.2	Gaussian smoothing kernel width
γ_7	.15	candidate segment generation
γ_8	.4	candidate segment generation
γ_9	.25	*fit-quality* measure
γ_{10}	.6	*subsumed-by* threshold for pruning candidate segments
γ_{11}	15.0	*fit-quality* threshold for pruning candidate segments
γ_{12}	.1	*end-abruptness* threshold for pruning candidate segments
γ_{13}	1.3	curve preprocessing

References

[1] Aoyama, H., and Kawagoe, M.; [1991], "A Piecewise Linear Approximation Method Preserving Visual Feature Points of Original Figures," *CVGIP: Graphical Models and Image Processing*, **53**:5, 435-446.

[2] Asada, H., and Brady, M., [1984], "The Curvature Primal Sketch," *IEEE Trans. PAMI*, **8**:1, 2-14.

[3] Bolles, R., and Cain, R., [1982], "Recognizing and Locating Partially Visible Objects: The Local-Feature-Focus Method," *Int. J. Robotics Res.*, 1:3, 57-82.

[4] Bookstein, F.; [1979]; "Fitting Conic Sections to Scattered Data," *CGIP*, **9**, 56-71.

[5] Dolan, J., and Weiss, R.; [1989]; "Perceptual Grouping of Curved Lines," *Proc. DARPA IU Workshop*, 1135-1145.

[6] Dudek, G., and Tsotsos, J. [1990]; "Recognizing Planar Curves Using Curvature-Tuned Smoothing," *Proc. 10th International Conf. Pattern Recognition*, Atlantic City, June, 1990, 130-135.

[7] Dunham, J.; [1986]; "Piecewise Linear Approximation of Planar Curves," *IEEE TPAMI*, **8**:1, 67-75.

[8] Fischler, M., and Bolles, R., [1986]; "Perceptual Organization and Curve Partitioning," *IEEE TPAMI*, **8**:1, 100-105.

[9] Gigus, Z., and Malik, J.; [1991]; "Detecting Curvilinear Structure in Images," *Technical Report UCB/CSD 91/619*, UC Berkeley.

[10] Grimson, W.E.L., [1989]; "On the Recognition of Curved Objects," *IEEE TPAMI*, **11**:6, 632-642.

[11] Gunther, O., and Wong, E.; [1990]; "The Arc Tree: An Approximation Scheme to Represent Arbitrary Curved Shapes," *CVGIP*, **51**, 313-337.

[12] Hemminger, T., and Pomalaze-Raez, C., [1990]; "Polynomial Representation: A Maximum Liklihood Approach," *CVGIP*, **52**, 239-247.

[13] Hoffman D., and Richards, W., [1984], "Parts of Recognition," *Cognition*, **18**, 65-96.

[14] Imai, H., and Ira, M., [1986]; "Computational-Geometric Methods for Polynomial Approximations of a Curve," *CVGIP*, **36**, 31-41.

[15] Leung, M., and Yang, Y.; [1990]; "Dynamic Strip Algrithm in Curve Fitting," *CVGIP*, **51**, 146-165.

[16] Lowe, D.; [1985]; *Perceptual Organization and Visual Recognition*, Kluwer, Boston.

[17] Lowe, D.; [1988]; "Organization of Smooth Image Curves at Multiple Scales," *Proc. ICCV*, Tampa, Fl., 558-567.

[18] Marimont, D., [1984], "A Representation for Image Curves," *Proc. National Conf. on Artificial Intelligence,* Austin, Texas, 237-242.

[19] Medioni, G., and Yasumoto, Y., [1987]; "Corner Detection and Curve Representation Using Cubic *B*-Splines," *CVGIP* **39**, 267-278.

[20] Merlin, P., and Farber, D., [1975], "A Parallel Mechanism for Detecting Curves in Pictures," *IEEE Trans. Computer* **24**, 96-98.

[21] Mohan, R., and Nevatia, R.; [1989]; "Segmentation and Description Based on Perceptual Organization," *Proc. IEEE CVPR*, 333-341.

[22] Parent, P., and Zucker, S.; [1989]; "Trace Inference, Curvature Consistency, and Curve Detection," *IEEE TPAMI*, **11**:8, 823-839.

[23] Pavlidis, T., and Horowitz, S, [1974], "Segmentation of Plane Curves," *IEEE Trans. Computers*, **23**:8, 860-870.

[24] Perkins, W., [1978], "A Model-Based Vision System for Industrial Parts," *IEEE Trans. Computers*, **27**:2, 126-143.

[25] Plass, M., and Stone, M., [1983]; "Curve-Fitting with Piecewise Parametric Cubics," *ACM Computer Graphics (SIGGRAPH)* **17**:3, 229-239.

[26] Saund, E., [1992]; "Putting Knowledge into a Visual Shape Representation," *Artificial Intelligence*, in press.

[27] Shirai, Y.; [1978]; "Recognition of Real-World Objects Using Edge Cues," in Hanson, A., and Riseman, E., *Computer Vision Systems*, Academic Press.

[28] Shneier, M., [1982], "Extracting Linear Features from Images Using Pyramids," *IEEE SMC*,**12**:4, 569-572.

[29] Sklansky, J., [1978], "On the Hough Transform for Curve Detection," *IEEE Trans. Computers*, **27**, 923-926.

[30] Teh, C., and Chin, R., [1989]; "On the Detection of Dominant Points on Digital Curves," *IEEE TPAMI* **11**:8, 859-872.

[31] Tuceryan, M., Jain, A., and Lee, Y., [1987], "Extracting Perceptual Structures in Line Patterns," *5th Scandinavian Conference on Image Analysis*, 531-538.

[32] Thomas, S., and Chan, Y., [1989], "A Simple Approach for the Estimation of Circular Arc Center and its Radius," *CVGIP* **45**, 362-370.

[33] Ullman, S., and Shaashua, A.; [1988]; "Structural Saliency: The Detection of Globally Salient Structures Using a Locally Connected Network," MIT AI Memo 1061.

[34] Weiss, R., and Boldt, M., [1986], "Geometric Grouping Applied to Straight Lines," *Proc. IEEE CVPR,* 489-495.

[35] Witkin, A., [1983], "Scale-Space Filtering," *Proc. IJCAI* 1019-1022.

[36] Witkin, A. and Tenenbaum, J., [1983], "On the Role of Structure in Vision," in Beck, J., Hope, B., and Rosenfeld, A., eds, *Human and Machine Vision*, Academic Press, New York.

[37] Zucker, S., David, C., Dobbins, A., and Iverson, L.; [1988]; "The Organization of Image Curve Detection: Coarse Tangent Fields and Fine Spline Coverings," *Proc. ICCV*, Tampa, Fl., 568-577.