Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters

CHARLES T. ZAHN

Abstract—A family of graph-theoretical algorithms based on the minimal spanning tree are capable of detecting several kinds of cluster structure in arbitrary point sets; description of the detected clusters is possible in some cases by extensions of the method. Development of these clustering algorithms was based on examples from two-dimensional space because we wanted to copy the human perception of gestalts or point groupings. On the other hand, all the methods considered apply to higher dimensional spaces and even to general metric spaces. Advantages of these methods include determinacy, easy interpretation of the resulting clusters, conformity to gestalt principles of perceptual organization, and invariance of results under monotone transformations of interpoint distance. Brief discussion is made of the application of cluster detection to taxonomy and the selection of good feature spaces for pattern recognition. Detailed analyses of several planar cluster detection problems are illustrated by text and figures. The well-known Fisher iris data, in four-dimensional space, have been analyzed by these methods also. PL/1 programs to implement the minimal spanning tree methods have been fully debugged.

Index Terms—Clustering, data structure analysis, feature space evaluation, gestalt psychology, graph theory, minimal spanning trees, nearest neighbor methods, numerical taxonomy, pattern recognition.

I. Introduction

WE SHALL address ourselves to the problem of detecting inherent separations between subsets (clusters) of a given point set S in a metric space governed by a distance function $\rho(x, y)$.

The phrase "inherent separations" is used to emphasize that any separation we detect will depend solely on interpoint distances within the set S. We shall strive for cluster methods which are "determinate" in the sense that detection of a given cluster does not depend on random choices in the detection algorithm and is not sensitive to the order in which points of S are scrutinized. In short we want an answer to the question, "What does the set S look like in terms of the structure of the space in which it is imbedded?"

To illustrate more concretely what is meant we refer to the point sets of Fig. 1(a), (e), (f), and (j). In Fig. 1(a), (e), and (f) we would like to be told that the set falls naturally into two distinct point clusters. In Fig. 1(j) we would like to be told that one cluster is present which can be separated into two clusters at a small "neck."

The cluster detection methods in this paper were motivated by our own personal perception of two-dimensional point sets as separate groupings or "gestalts." The principle of grouping will be "proximity" as described by Wertheimer [26]. We have investigated the behavior of these clustering algorithms on point sets in the plane where the set can be "seen" and the reader can get an intuitive feel for what the algorithm is doing. It is our hope that the same methods will be useful in higher dimensional spaces and this is why we have posed the cluster problem in a general metric space.

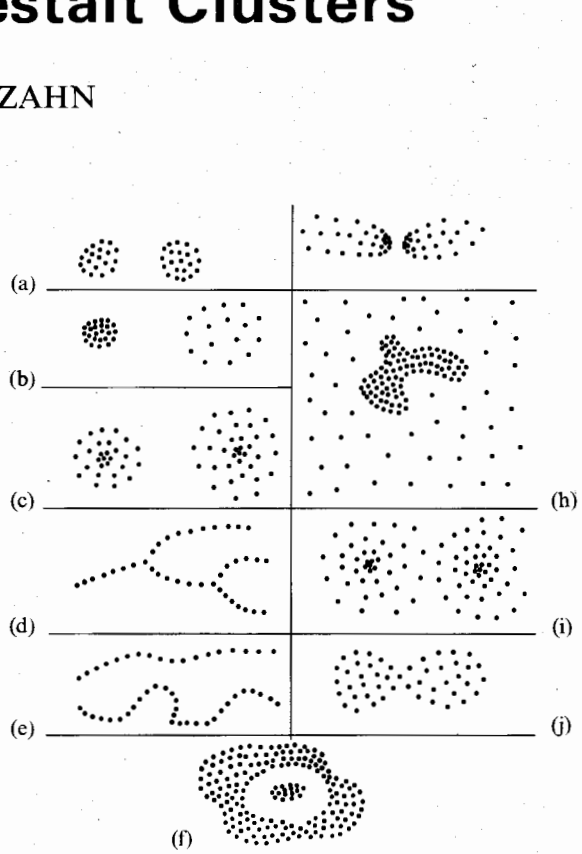


Fig. 1. Sample cluster problems.

II. Applications

The field of taxonomy in biology and the problem of organizing library materials into groups in a meaningful manner—these are two good examples where cluster detection is useful. In biology one makes measurements on a set of organisms and then attempts to group them in a way which reflects similarity based on these measurements. An interesting example cited by Bonner [1] is encountered in medicine; a set of 350 patients were measured with regard to 18 symptoms. All these patients had been diagnosed as having the same disease, whose lengthy Hellenic name shall remain unidentified. This disease classification is known to be rather loose and hence it is of interest to know if it consists of several smaller groupings which can be observed directly from the measurement space by cluster detection. Then we will have found more natural disease categories to replace or subdivide the original disease.

It must be emphasized that cluster detection will depend in a very sensitive way on the particular imbedding of objects in a metric space and this choice will almost certainly be made on information outside the scope of this paper.

Manuscript received November 3, 1969; revised June 30, 1970. This work was supported by the U. S. Atomic Energy Commission.

The author is with the Stanford Linear Accelerator Center, Stanford University, Stanford, Calif.

Cluster detection and description find their way into pattern recognition in two rather surprisingly different ways. In the first the points of a two-dimensional pattern constitute the point set to be clustered after which it is desired to describe the shapes of whatever clusters emerge. An example of this is shown in Fig. 1(h). Rosenfeld [2] calls this the smoothing of quantum-limited pictures, the point being that connected shapes appear only after some local smearing has occurred. This problem is related to the work of Pizer and Vetter [3] on visual enhancement of quantum-limited pictures. Rosenfeld and Pfaltz [4] consider algorithms for performing such smearing on sets which are subsets of a uniform planar lattice (binary matrix patterns).

Another application which can be considered an example of quantum-limited smearing is described by Clark and Miller [5] and involves the correct linking together of spark images from a physics spark-chamber photograph. It is desired to observe the line-like structure of the point set in Fig. 1(d) and detect the branch points as well as the order of points along each track.

The second way that cluster detection is applicable to pattern recognition is by providing an answer to the question whether a given set of features constitutes a good feature space in which to discriminate a given set of pattern classes. It has been observed recently by more than one researcher that the crucial problem in pattern recognition is the selection of "good" features rather than sophisticated attempts to separate classes in a feature space which may be poorly chosen. We feel that a reasonable definition for good feature space in this connection is a space and a metric in which the clusters are identical (or nearly so) to the classes to be discriminated. It should be clear that the metric space clustering algorithms developed in this paper *assure* good class discrimination via the nearest neighbor [6] classifier whenever the "classes" are "clusters" in the given feature space with its metric.

Since clustering algorithms give us a yardstick by which to measure the efficacy of a given feature space we could in principle attack the problem of feature-space selection via the learning approaches which previously have been used to select the parameters of the classifier for a given feature space.

Another application for clustering is Hough's scheme [32] for recognizing the existence of approximately straight dotted lines in a two-dimensional picture (bubble-chamber particle tracks, for example). First he transforms each point into a line in another plane in such a way that a set of collinear points goes into a pencil of lines through a common point. Such a transformation takes "nearly" collinear points into a set of lines whose mutual intersection points form a relatively dense cluster—hence the need for a cluster detection algorithm.

Finally, we mention the use of clustering in visual scene analysis referred to by Rosenfeld *et al.* [38] and Minsky and Papert [37]. When local gradient detectors are passed over a digital picture and a threshold is used the picture points thus extracted usually form dotted curves which trace boundaries between faces of objects in the scene. Point clustering algorithms are needed to reconstruct the curvilinear boundaries formed by these points.

III. Motivation

Our interest in the general problem of cluster detection was aroused by a reading of the survey paper by Nagy [7] and the little book on pattern recognition by Arkadev and Braverman [8]. It was further awakened by the survey paper of Ball [17] on clustering methods. In particular we were challenged by Nagy's assertion that few clustering methods could handle problems like that in Fig. 1(f) successfully. It seemed to us that the problems in Fig. 1(a), (e), and (f) are not substantially different in difficulty for the following reason: in each case we have two disjoint point sets P and Q whose distance apart (P, Q) is substantially larger than the average distance from a point to the nearest neighbor in its particular point set. This observation suggested that if we connected each point to all other points within a suitably chosen radius then the resulting graph would consist of two connected components whose points were P and Q, respectively. It also occurred to us that the algorithm just outlined depends only on the point set belonging to some metric space and hence is general enough to extend to E^n.

We were also challenged by the statement in [8] that no algorithm for the problem in Fig. 1(j) was known although it posed no difficulty for human perception. Arkadev and Braverman call this problem "learning without reward" and ascribe considerable importance to its solution. Our perception of two clusters in this case can possibly be explained by the fact that the boundary of the point set contains a narrow portion or "neck." If we construct a graph as we suggested for Fig. 1(a) there will almost certainly be a single connected component but it will be possible to disconnect it in a nontrivial fashion by deleting an edge-set of small size. In the simplest case the graph might contain a single bridge or cut-vertex. Once again this line of attack was generalizable to high dimensional spaces.

Throughout this trend of thinking we were influenced either consciously or otherwise by Julesz's experiments on humanly perceivable texture differences [9] and a report by Narasimhan [10]. Both these references stress the importance of the gestalt principles of "proximity," and "similarity" in perceptual groupings of points. Julesz's paper is especially important since the experiments lead to the conclusion that "clusters or lines formed by proximate points of uniform brightness play a decisive role" in human discrimination of visual texture. Pattern textures of a purely statistical nature (except point density) tend to be impossible to discriminate unless such clusters or lines are present. The ability to perceive areas of different point density is exemplified by Fig. 1(h).

The use of the "minimal spanning tree" of a graph as an aid to detecting and describing the structure of point clusters was suggested to us by the processing of spark-chamber photographs reported by Clark and Miller [5]. We have discovered this to be a most powerful and general tool for these tasks and have discovered some theorems which indicate that its success is no accident. It is also rather interesting to

note that Arkadev and Braverman ([8], pp. 109–110) sketch an algorithm for cluster detection which essentially constructs a minimal spanning tree (MST) and Johnson's "minimum method" of hierarchical clustering ([28], p. 248) is one of the algorithms for constructing an MST. Gower and Ross [36] have independently observed the connection between the MST and cluster analysis; their paper exhibits ALGOL programs to construct the MST and use it to perform the single linkage cluster analysis of Sneath [35].

IV. Sample Cluster Problems

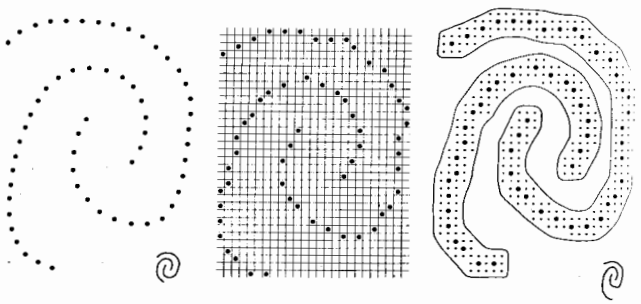
We shall briefly describe a set of problems typical of those which will be treated in this paper. We think the range of problems indicates the power of graph-theoretical methods in the context of detecting and describing inherent cluster structure in arbitrary point sets with a distance function. Fig. 1(a) represents a pair of well-separated clusters each having the same relatively homogeneous point density. Not unexpectedly, most existing clustering methods do well on this problem [7]. The next problem, Fig. 1(b), is almost identical to Fig. 1(a) except that the two point densities are not equal. In Fig. 1(c) the point density varies smoothly within each cluster but the separation is still substantial. The problem in Fig. 1(d) is to describe the cluster as composed of linear pieces with a definite branching structure. Fig. 1(e) is essentially like Fig. 1(a) but the shape of clusters is quite different.

The problem in Fig. 1(g) represents smoothly varying nonhomogeneous cluster densities similar to Fig. 1(c) but here we show that the separation is dependent only on the point densities near where the clusters approach each other. Solving this problem appears to require some adaptive mechanism. The problem in Fig. 1(h) involves the ability to detect sharp gradients in point density as well as to describe the boundary of the cluster thus detected. Examples like this show the close relation between the processing of grey-scale digital pictures [2] and the processing of two-dimensional point distributions [3]. As pointed out by Rosenfeld [2] distributions like those in Fig. 1(h) can be considered as quantum-limited versions of grey-scale pictures where local point density becomes grey-scale. The converse is also true since photographs and especially halftone images are really only point distributions which our eyes average to compute grey values. Rosenfeld and Pfaltz [4] discuss methods for transforming a quantum-limited picture to a grey-scale picture on a square lattice.

Fig. 1(i) is like Fig. 1(c) except that the clusters touch and are not really well separated. We seem to perceive the separation by noticing that the point density is at a local minimum near where the clusters touch. In Fig. 1(j) we actually have a single cluster but we notice it contains a very thin section ("neck") whose removal separates it into two distinct clusters.

V. Gestalt Principles of Perceptual Organization

To create the proper setting for the methods to be described in this paper we shall give a brief resume of those principles of gestalt psychology which we have borrowed and attempted to mechanize. In 1923 Wertheimer [26] enunciated several principles which were claimed to govern



(a) (b) (c)

Fig. 2. Gestalt principles exemplified. (a) Proximity. (b) Pattern on a discrete lattice. (c) Smeared lattice pattern.

the way in which our perceptual processes organize the raw sensory data presented to our eyes. Fig. 2(a) illustrates *proximity* which is the most basic of all gestalt principles. The point is that human subjects perceive the set of dots as if they were organized into two curves as shown in the lower right corner of Fig. 2(a). The only reasonable analysis of this observation is that perceptual organization favors groupings which represent smaller interpoint distances. This principle was the starting point of all our methods for cluster detection and in some sense our methods attempt a precise mathematical formulation of this principle.

A related principle suggests that organizations which are *simple* or *minimal* in some sense are preferred as are those organizations which take *smaller amounts of information* to encode the picture. Our nervous system seeks the most economical encoding of the data presented. This is discussed more fully by Koffka ([21], ch. IV) and Hochberg [20], [27].

As we mentioned earlier we owe to Narasimhan [10] our first acquaintance with these gestalt principles as well as the suggestion that they have a strong connection with possible algorithms for pattern description. He incorporates these principles into a *syntactical model* of picture description and suggests its use to investigate on a more rational basis the separation between "innate" and "learned" perceptual organization. A very suggestive idea indeed was the way that the proximity principle was incorporated through a smearing algorithm applied to points on a square lattice. Fig. 2(b) shows an approximate version of Fig. 2(a) on a discrete square lattice. Fig. 2(c) depicts the smeared version in which each black point is replaced by an entire 3×3 neighborhood of black points. After this the connected sets of points (whose boundaries can be efficiently computed by the "curvaturepoint" method of Zahn [29], [30]) represent the two distinct line patterns that correspond to human perceptual organization.

VI. Intuitive Methods for Cluster Detection

We shall begin our solution to cluster detection by attacking three problems representing the types exemplified in Fig. 1(a), (g), and (j). Our methods will be highly intuitive and directed at each specific problem type. Later we develop a single method to solve all these but the development of this general technique resulted from our realization of several deficiencies in these intuitive and somewhat brute-force methods. In spite of their deficiencies these techniques are a

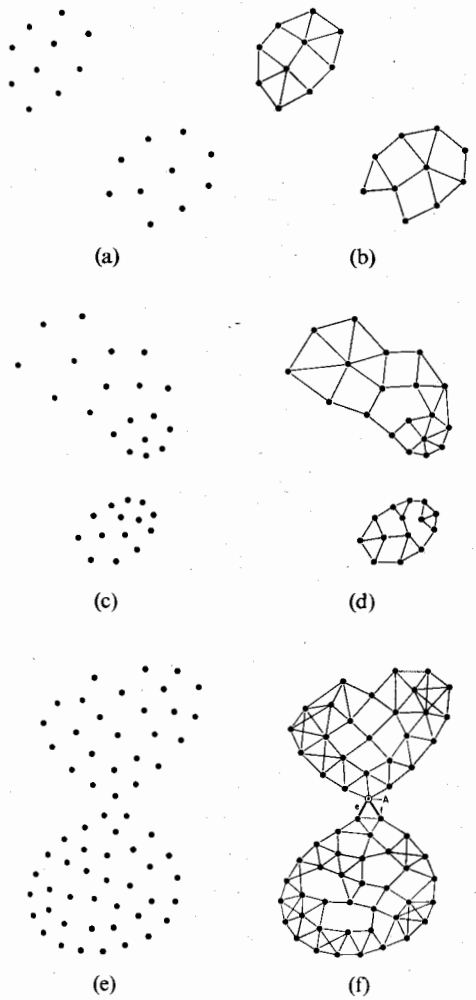


Fig. 3. Intuitive methods. (a) Homogeneous clusters. (b) Nearest-neighbor graph with $t=\mu+3\sigma$. (c) Nonhomogeneous clusters. (d) Three-nearest-neighbor graph for (c). (e) Touching homogeneous clusters. (f) Four-nearest-neighbor graph for (e) showing the cut point A and the cut set (e, f).

good introduction to the latter more powerful methods and they illustrate difficulties in cluster detection rather well.

We begin with the simplest case of all—two well-separated clusters each with approximately the same homogeneous point density. This is illustrated in Fig. 3(a). Computing the distance from each point to its nearest neighbor we get a mean value μ of 20.5. The standard deviation σ of these distances is 1.66 and the distance between the clusters is 58. The range of values of nearest distance is (17.5, 24). The relatively small value for σ is what we mean by the term "homogeneous" and the size of intercluster distance relative to μ implies "well-separated." It seems reasonable to form a graph from the points of Fig. 3(a) by connecting any pair whose distance is smaller than a threshold value depending on μ and σ. Fig. 3(b) shows the graph that results using $(\mu+3\sigma)$. This graph has exactly two connected components representing the clusters as we perceive them. There might possibly be cases in which the values for μ, and intercluster distance are known accurately enough *a priori* and the threshold could be precomputed (such a method was used by Abraham [31]).

Fig. 3(c) illustrates two clusters with smoothly varying but nonhomogeneous densities. They are well-separated as we perceive them because the regions of closest approach between the two clusters are regions where the density is high compared to the distance between clusters. The fact that the left most point of the upper cluster is further from its nearest neighbor than the distance between clusters does not confuse us in the slightest; it does, however, mean that we could never hope to detect these clusters with the method used on the previous example. We obviously require some sort of adaptive connecting algorithm and the simplest idea is to connect each point to its K-nearest-neighbors regardless of the absolute distances involved. This idea is closely related to the variable-aperture method for measuring density at a point found by Pizer and Vetter [3] to be statistically more suitable than fixed-aperture methods. Fig. 3(d) shows how the graph constructed from edges with 3-nearest-neighbors detects the two clusters perfectly.

The "touching clusters" example shown in Fig. 3(e) will appear as a single cluster under the previous methods as indeed it should. By the criterion of connectivity it is a single cluster; we see it as two because it is only connected at a small "neck." Our intuitive approach is to construct a K-nearest-neighbor graph (we use $K=4$ but the choice is probably not critical) as shown in Fig. 3(f) and look for *cut points* (whose removal disconnects the graph) or *bridges* (edges whose removal disconnects the graph). Harary [33] has described an algorithm to detect cut points and Zahn [34] has recently developed some alternative methods suggested by Pohl's method [15] for detecting bridges. In Fig. 3(f) there is a single cut point at A which reveals the "neck" at once. In general the neck may not be so small and it may be necessary to find a small set of edges (a *cut set*) whose removal disconnects the graph like the two edges (e, f) in Fig. 3(f). This is not a trivial problem but Pohl [15] describes a reasonable heuristic for approaching it.

VII. Minimal Spanning Trees

The previous section indicated the need for a "locally adaptive interconnecting mechanism" for a point set—one which favors nearest neighbors. We have found the minimal spanning tree to be a powerful mechanism in this regard and this section will serve to define, exemplify, and characterize this graph-theoretical construct. Basic definitions of graph theory may be found in Ore [40] but we shall attempt to make the discussion as self-contained as possible. Detailed proofs of theorems may be found in the Appendix.

An *edge-weighted linear graph* is composed of a set of points called *nodes* and a set of node pairs called *edges* with a number called a *weight* assigned to each edge. Graphs are easier to think about in their geometric form so we shall use an example to describe the concepts needed for the theorems of this section. Fig. 4(a) depicts a weighted graph with six nodes and nine edges. A *path* in a graph is a sequence of edges joining two nodes as $(ABCFE)$ or $(BADF)$. A *circuit* is a closed path as $(ABCA)$ or $(ACFEDA)$. A *connected* graph has paths between any pair of nodes. A *tree* is a connected graph with no circuits and a *spanning tree* of connected graph G is a tree in G which contains all nodes of G. Fig. 4(b) and (c) represents spanning trees of the graph in Fig. 4(a). If we define the weight of a tree to be the sum of the weights of its constituent edges then a *minimal spanning*

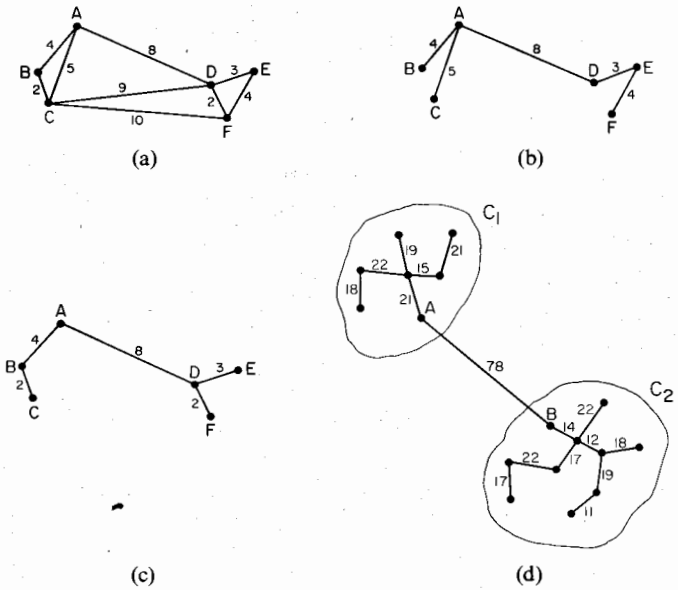


Fig. 4. Graphs and minimal spanning trees. (a) Weighted linear graph. (b) Spanning tree. (c) Minimal spanning tree. (d) Illustration of Theorem 3 showing clusters C_1 and C_2 and the "inconsistent" cluster-joining edge (A, B).

tree of graph G is a spanning tree whose weight is minimum among all spanning trees of G. Fig. 4(c) is the MST for Fig. 4(a). The computational problem of constructing an MST has been treated by several authors [11]–[13] and is briefly discussed in the Appendix. It is a surprisingly simple computation.

A *partition* of the nodes of graph G is a division into two disjoint nonempty subsets (P, Q). For the graph of Fig. 4(a), $P=(A, B, C)$ and $Q=(D, E, F)$ constitute a partition. The *distance* $\rho(P, Q)$ across a partition is the smallest weight among all edges which have one end node in P the other in Q. The distance $\rho((A, B, C), (D, E, F))=8$ for the example above since the other two edges which span across P and Q are CD and CF with greater weight than edge AD. The set of edges $C(P, Q)$ which span a partition will be referred to as the *cut-set* of (P, Q) and a *link* is any edge in $C(P, Q)$ whose weight is equal to the distance $\rho(P, Q)$. The set of all links in $C(P, Q)$ is called the link-set $\lambda(P, Q)$. For the sample partition above $C(P, Q)=(AD, CD, CF)$ and $\lambda(P, Q)=(AD)$.

Looking at the graph of Fig. 4(a) it seems plausible to expect that the minimal spanning tree would choose edge AD as the bridge spanning from set (A, B, C) to (D, E, F) since that edge does the job at minimal expense. This is in fact true as is shown in the following.

Theorem 1: Any MST contains at least one edge from each $\lambda(P, Q)$.

Furthermore, it is true that the following theorem holds.

Theorem 2: All MST edges are links of some partition of G.

The following theorem is important because it reveals the inherent relationship between the MST and cluster structure.

Theorem 3: If S denotes the nodes of G and C is a nonempty subset of S with the property that $\rho(P, Q)<\rho(C, S-C)$ for all partitions (P, Q) of C, then the restriction of any MST to the nodes of C forms a connected subtree of the MST.

The significance of Theorem 3 for cluster detection is illustrated in Fig. 4(d) which depicts the MST for a point set consisting of two clusters C_1 and C_2. No partition (P_2, Q_2) of C_2 is such that $\rho(P_2, Q_2)>22$ and therefore the hypothesis of Theorem 3 holds since $\rho(C_2, C_1)=78>22$. This assures us that the subgraph of the MST which spans only the nodes of C_2 will be a connected subtree as we see in Fig. 4(d). The same is also true for C_1.

It is quite helpful that the MST does not break up the real clusters in S, but on the other hand neither does it force breaks where real gaps exist in the geometry of the point set. A spanning tree is forced by its very nature to span all the points but at least the MST jumps across the smaller gaps first. Theorem 2 says that any MST edge is the smallest jump from some set to the rest of the nodes. We still have the problem of deleting edges from an MST so that the resulting connected subtrees correspond to the observable clusters. In the example of Fig. 4(d) we need an algorithm which can detect the appropriateness of deleting the edge AB and no others.

The following criterion is suggested for this type of two-dimensional clustering observable by humans. A tree edge XY, whose weight $W(XY)$ is significantly larger than the average of nearby edge weights on both sides of the edge XY, should be deleted. We call such an edge *inconsistent*. There are two natural ways to measure the significance referred to. One is to see how many sample standard deviations separate $W(XY)$ from the average edge weights on each side. The other is to calculate the *factor* or ratio between $W(XY)$ and the respective averages. See Section XVII for details.

Edge AB in Fig. 4(d) has a length (weight) of 78. There are four edges which are within two steps of A and their average length is $(21+22+19+15)/4=19.25$. The sample standard deviation for these four edge lengths is approximately 2.7 so that the length of edge AB is more than 20 standard deviations in excess of the average lengths at A. If we assumed a normal distribution for edge lengths, then one exceeding three or four standard deviations would occur less than one percent of the time and hence may be regarded as significant. A similar situation exists for the neighborhood of node B. The definition of edge inconsistency depends on several factors—the size of neighborhood explored for each end node, the number of standard deviations and the factor considered as significant, and whether or not inconsistency is required at both ends. A later section discusses our computational experience with these factors including some difficulties encountered near the fringes of the MST where small sample sizes can give distorted results. Finally, we should mention that AB is the only edge in Fig. 4(d) which meets our criterion at a significance level of two standard deviations.

Occasionally we shall refer to a *factor* of inconsistency which is the ratio between edge weight and the average of other nearby edge weights. A factor of 2 usually means the separation is quite apparent. The example above suggests

to us that any uniformly dense point pattern in the plane which is separated from other points by an amount significantly larger than the average nearest distance within the pattern can probably be detected by humans as a distinct cluster. It also appears that human perception may be sensitive to the condition stated as hypothesis for Theorem 3. We suggest this condition as a first approximation to a precise geometrical statement of what makes a point set distinctly observable to humans.

Before stating the next important theorem we need some definitions. It is sometimes useful to assign a *cost* to each path in a weighted graph by taking the maximum edge weight of the path. The path (*CADE*) in Fig. 4(a) has edge weights (5, 8, 3) and hence a cost of 8. We may think of this as the cost of going from *C* to *E* along the path *CADE*. It is natural to ask what path between a pair of nodes has the least cost and such a path is called a *minimax path* because it minimizes over all paths the cost, which is the maximum weight in the path. In Fig. 4(a) there are four minimax paths from *C* to *F*; they are (*CADF*), (*CADEF*), (*CBADF*), and (*CBADEF*), all of cost 8. We see immediately that minimax paths are far from unique in general. We might wonder if there exist minimax paths each of whose subpaths are also minimax. Among the four minimax paths from *C* to *F* only the path (*CBADF*) has this stronger property which we shall refer to as *strongly minimax*. The fact that the strongly minimax path (*CBADF*) lies within the MST (see Fig. 4(c)) is more than mere coincidence as shown by the following theorem of Kalaba [14].

Theorem 4: If *T* is an MST for graph *G* and *X*, *Y* are two nodes of *G*, then the unique path in *T* from *X* to *Y* is a minimax path from *X* to *Y*.

This result has some of the flavor of Theorem 3 because the preference for minimax paths in the MST forces it to connect two nodes *X* and *Y* belonging to a tight cluster without straying outside the cluster. Under the condition of Theorem 3 the change in cost incurred by routing a path out of a cluster is an increase and hence, by Theorem 4, is avoided by the MST. It can also be seen that each link of *G* is a single-edge minimax path joining its end nodes.

We have taken the trouble to introduce these theorems because they help to characterize the behavior of the MST and they indicate why the MST can be a good starting point for cluster analysis. The proofs of Theorems 1 through 3 are in the Appendix along with discussion of algorithms for constructing the MST. The reader is referred to [14] for the proof of Theorem 4 and a more detailed discussion of minimax paths.

We remark in passing that in one-dimensional space the algorithm for calculating the MST becomes a sort algorithm and relative compactness measures the tendency toward longer "runs" of the same class. The actual computation could be performed more efficiently in this special case by using more traditional sort algorithms.

VIII. A Composite Cluster Problem

We shall now apply methods using the MST to several interesting cluster problems. To begin we attack a point set

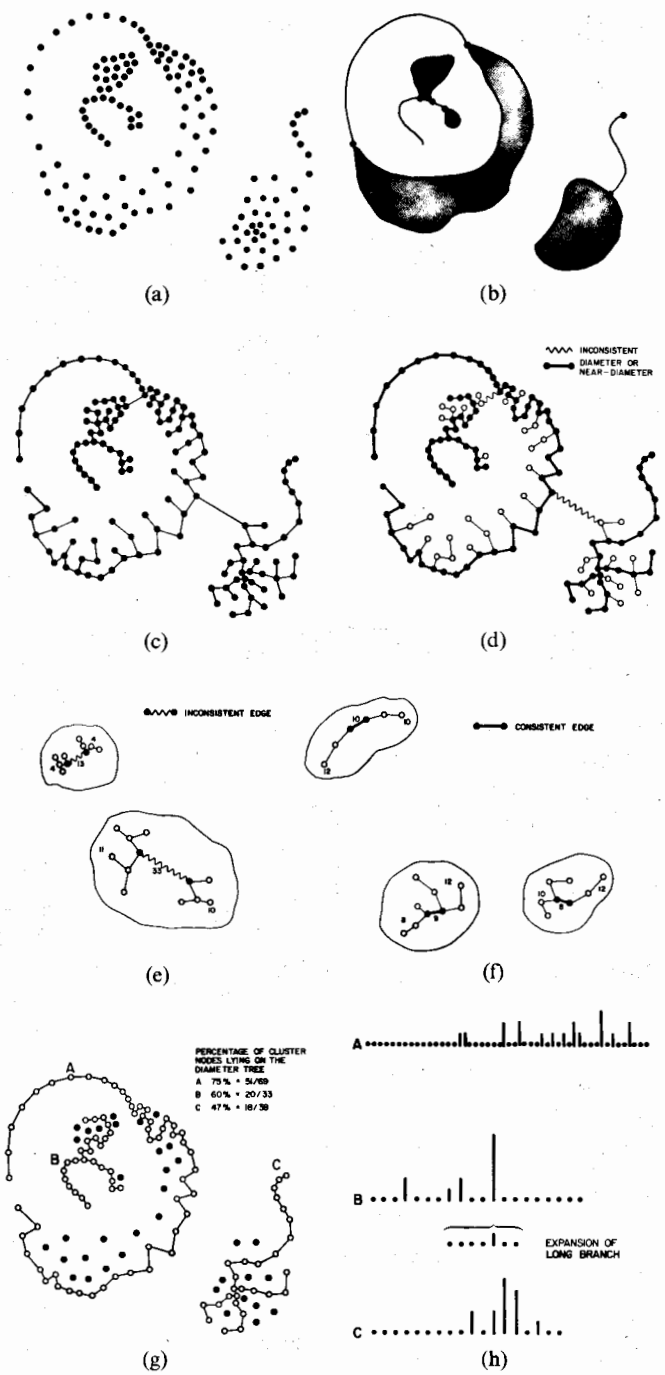


Fig. 5. Composite cluster problem. (a) Planar point set. (b) Perceptual gestalt. (c) Minimal spanning tree for (a). (d) Inconsistent and diameter edges. (e) Closer look at inconsistent edges. (f) Consistent edges. (g) Near-diameter trees. (h) Histograms of tree depth along diameters.

encompassing the peculiar difficulties associated with the problems in Fig. 1(a) through (g). After careful study of this single example the reader should be able to convince himself that each of these seven problems will succumb to the same MST technique we shall employ for the composite problem.

The point set is shown in Fig. 5(a) and a possible conceptualization of Fig. 5(a) is depicted in Fig. 5(b). Fig. 5(c) shows the MST for the point set calculated by visual inspection but still quite accurate. Using a factor of 2 as the measure of significant edge inconsistency we can delete the two edges shown in Fig. 5(d) and then determine the

diameter (path with most number of edges) of each connected tree remaining. The *near-diameter* edges are on a path whose number of edges is fairly close to that for a true diameter. Fig. 5(e) gives a closer look at the two inconsistent edges and Fig. 5(f) shows three edges which are consistent. Fig. 5(g) depicts how well the remaining trees of diameter and near-diameter edges reflect the cluster separation and geometrical shapes of the three clusters. The percentages shown are a crude indicator of the noncompactness or line-likeness of the cluster. By calculating the maximum depth of branching from each node of a single diameter path we can produce histograms which reflect geometrical structure more specifically. Long runs of zero suggest well-defined line clusters. Line branching can be detected by histogramming any deep branch in the midst of an otherwise linear portion of the main diameter as is shown in Fig. 5(h).

Even the doubly connectedness of the larger cluster could be detected by looking for ends of the diameter tree which are very close in the plane relative to their distance apart in the MST. If we visualize the histogram for this cluster as being wrapped around to form a cycle, then we obtain an extremely accurate description of the cluster's geometrical and topological properties. The reader is reminded that everything we have done here can be done in higher dimensional spaces or, in fact, general metric spaces.

IX. Particle Track Description

The next example we shall try is an artificial bubble-chamber photo with gaps in the tracks and noise points. Fig. 6(a) depicts a point set like one that might emerge from the digitization of a simple bubble-chamber photograph. The "gaps" and "noise" are readily visible. The physical interpretation of "perception" of this point set as particle tracks and interaction vertices is depicted in Fig. 6(b); this is the correct structure of the image and we would hope to find an algorithm which would reveal this connectivity structure without any serious prompting. Notice that this is not so much a question of cluster detection as it is one of description. A very similar problem for spark-chamber imagery was solved by Clark and Miller [5] employing a subtle combination of graph theoretical and geometrical concepts. They initially construct a graph from the points (sparks) by including edges between pairs of points satisfying a criterion which is based primarily on distance but also emphasizes the degree to which the edge in question is parallel to the expected direction of tracks. The minimal spanning tree is then computed and "hairs" are removed; these are nodes of degree 1 connected directly to nodes of degree 3 or greater. Finally, certain isolated nodes are incorporated into the tree on an angular criterion.

We shall show the effect of a somewhat stripped-down version of this algorithm on our admittedly simpler example and argue why the MST can be expected to do the right thing except for localized anomolies. Fig. 6(c) contains the MST of the complete graph constructed from the point set. The "noise" points L, M, N, and O appear as "hair" on the MST and "gaps" like (A, B) and (D, E) are bridged as one would have hoped. It is striking how paths in the MST between pairs of points follow the basic connectivity perceived in the point set in spite of these "gaps" and noise points. The path from A to C goes through the vertex V_2 in spite of the gap (A, B); the path from I to T (which has a physical meaning in time) traverses the interaction vertices V_1 and V_2 in the correct order in spite of the potential distractions represented by noise points L, M, N and gaps (A, B), (D, E), and (F, G). These facts are simple results of Theorem 4 of the previous section which states that the minimal spanning tree contains minimax paths between pairs of points.

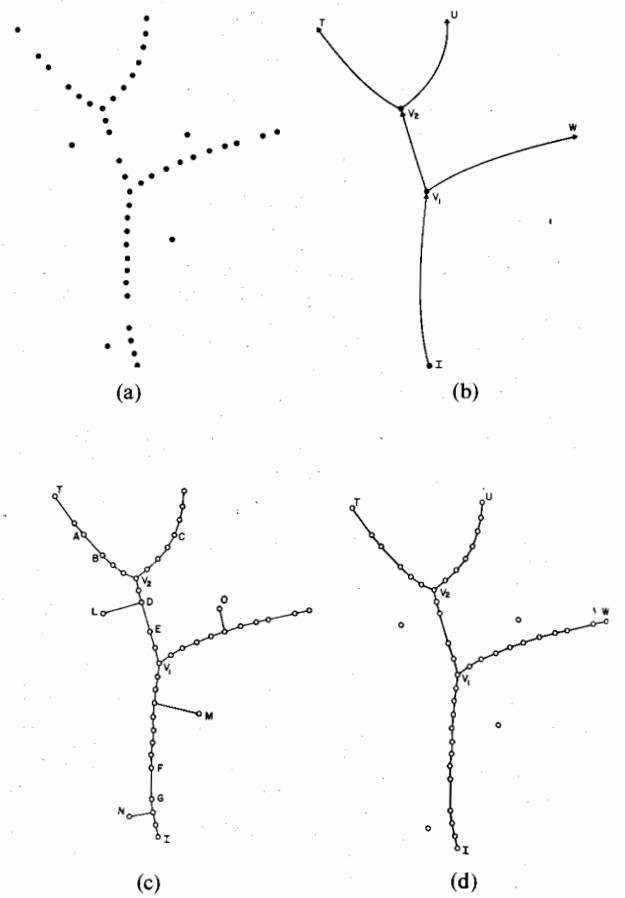


Fig. 6. Particle tracks problem. (a) Artificial bubble-chamber photograph with gaps and noise points. (b) Particle tracks and interaction vertices. (c) Minimal spanning tree for (a) showing "hairs." (d) MST without hairs.

The MST can be pruned by eliminating "hairs" as was done by Clark and Miller or alternatively by the diameter and near-diameter path techniques used on the composite cluster problem of Fig. 5(a). The resulting tree (shown in Fig. 6(d)) contains precisely the right connectivity to be interpretable along the lines of Fig. 6(b). When sequences of degree-2 nodes are interpreted as a single edge we are left with a tree whose nodes of degree ≥ 3 are interaction vertices and whose nodes of degree $= 1$ are track starts or track ends.

Deletion of inconsistent edges does not play a part here since the "gap" edges would almost invariably be so deleted. The method will not work effectively if there are close parallel tracks or crossing tracks, both of which phenomena are of frequent occurrence in bubble-chamber photographs. The latter problem can probably be handled by special techniques applied to the MST using edge directions.

X. Touching Clusters

In our next example we shall use MST methods on the touching clusters problem referred to by Arkadev and Braverman [8] as "learning without reward." Fig. 7(a) is a copy of the cluster set on p. 109 of [8]. The authors have the following to say about this set.

> If asked to draw a line separating two isolated groups of points (Fig. 7(a)) one would do it without difficulty. But it would be quite difficult, even impossible, to tell how one did it, i.e., to describe the algorithm for constructing the separating line. If such an algorithm for the separating of compact groups of points could be formulated in a sufficiently clear and detailed form, the problem of learning to recognize images without rewards would probably be solved.

We feel our treatment of this example constitutes a rather general answer to their challenge particularly since our method does not depend on performing the desired separation with a line (hyperplane in higher dimensional spaces).

The human perception of Fig. 7(a) as two clusters joined by a small neck may be related to our ability to imagine a boundary for the point set as depicted in Fig. 7(b). The existence of the neck is clearly visible as a relatively small area with opposing concavities in the boundary on two sides. Unfortunately, this description of the neck depends on concepts such as boundary concavity and opposite directions—concepts involving spaces in which angular measure makes sense. If we are to have any success applying graph theoretic and metric space techniques to this problem we shall need more general concepts of neck and so we propose the following.

A *neck in a graph* is any small connected subgraph whose deletion disconnects the graph into components at least two of which are substantially larger than the neck itself. The measure of size for the subgraphs (neck and components) will be the length of a diameter. The diameter can be defined using edge lengths if they exist or size of path otherwise. The idea of neck is a generalization of the concept of cut point in a graph.

For a point set imbedded in a metric space we shall define *neck* to be a small localized subset of the points whose deletion leaves a set consisting of at least two large clusters separate in the sense of the hypothesis to Theorem 3 and such that our normal MST method will detect a significantly inconsistent edge in the MST for the reduced point set. The idea of neck in a graph is an appropriate one when we have constructed a nearest-neighbor graph from a metric point set but the idea of neck in a point set applies if MST techniques are being used. Hence, in this section we use the latter.

The MST for the point set is depicted in Fig. 7(c) and one of the diameters is drawn in Fig. 7(d) with the depth of branching off the diameter indicated. Fig. 7(e) shows a near-diameter subtree of the MST. The *associated number* of a node in a graph is the number of edges in the longest path emanating from that node. Any one of these longest paths will be called a relative diameter for the node and the node at the other end of a relative diameter will be called an *antipode* of the node. A *diameter* of a graph is of course a relative diameter whose associated number is maximum. The tree of Fig. 7(e) consists of all relative diameters whose length is within 4 of the diameter length. The associated numbers of the end nodes are indicated. The path (unique in a tree) from D to B is a diameter and contains the path (E, F); an antipode for A is the node B at distance 21 from A and an antipode for C is node D at distance 22. The interesting thing is that the path (E, F) is the only subpath common to the relative diameters (A, B) and $C, D)$. It turns out to be the intersection of *all* relative diameters in this near-diameter tree. It seems quite plausible that if there exists a constriction in the original point set then it will occur somewhere around the segment (E, F). The idea of intersecting near-diameter paths was suggested by a similar method used by Pohl [15] to detect bridges and small cut-sets in a graph.

In Fig. 7(f) we have depicted the histogram of branching depths along the diameter shown in Fig. 7(d); also shown is the common section (E, F) of relative diameters and within this the best local minimum. Not only do we expect a constriction to be in (E, F) but more specifically we expect it where branch depths are small and at a local minimum. According to this reasoning we should test the hypothesis that the points G, H, and I along with their branches (none in this example) represent a "neck" between two otherwise well-defined clusters. To perform this test we delete the points G, H, and I at the local minimum and try to detect clusters in the remaining point set shown in Fig. 7(g). It is visually clear that the hypothesis is correct in this case and, therefore, when we construct the MST for the new point set (Fig. 7(h)) and then check for inconsistent edges, the edge (Q, R) shows up. A more detailed look at this new inconsistent edge is provided by Fig. 7(j) in which edge lengths are noted. The edge (Q, R) has a factor of inconsistency somewhat larger than 2. The geometrical nearness of the inconsistent edge to the deleted points provides further confirmation that a neck exists.

XI. Touching Gaussian Clusters

Now we attempt to solve a clustering problem which we call "touching Gaussian clusters" because each cluster has a point density which varies from high values at the center to low values at the boundary as if density had a Gaussian distribution. Such clusters can be separated even if they are touching over a larger segment of boundary than the previous example. Fig. 8(a) depicts such a cluster pair and Fig. 8(b) shows the centers and boundaries of the two clusters. A cluster center is a point of high density and can be detected by its having a low average edge length for edges incident to it in the MST or a nearest-neighbor graph. The point should also be a relative maximum of density. Fig. 8(c) shows the MST for the cluster set and Fig. 8(d) depicts the path in the MST joining the two cluster centers selected by this criterion. The division between the two clusters is effected by breaking this path between centers near the point of sparsest density which will once again be a relative minimum also. The "hysteresis smoothing" technique should probably be employed in looking for this local minimum. When two Gaussian clusters overlap slightly rather than just touching, then the point density along the path of centers will have two local minima on either side of a relative

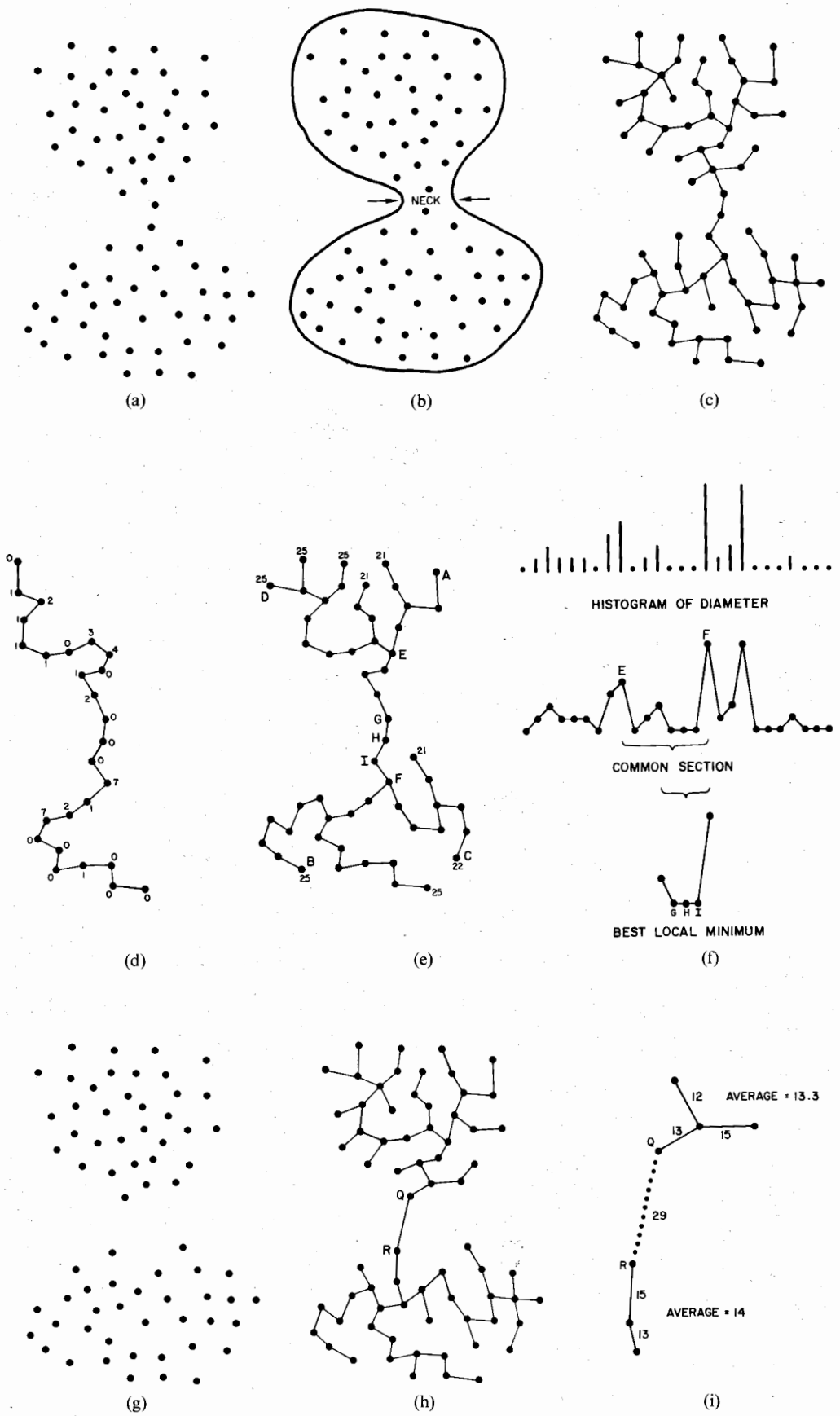


Fig. 7. Touching clusters problem. (a) Touching clusters. (b) Boundary for (a) showing the "neck" between two clusters. (c) Minimal spanning tree for (a). (d) Diameter of MST showing branching depth. (e) Near-diameter tree with "associated numbers" of end nodes. (f) Detection of neck using diameter histogram. (g) Clusters after deletion of points near local minimum of branching depth. (h) MST for (g) showing new "inconsistent" edge (Q, R). (i) Closeup of inconsistent edge.

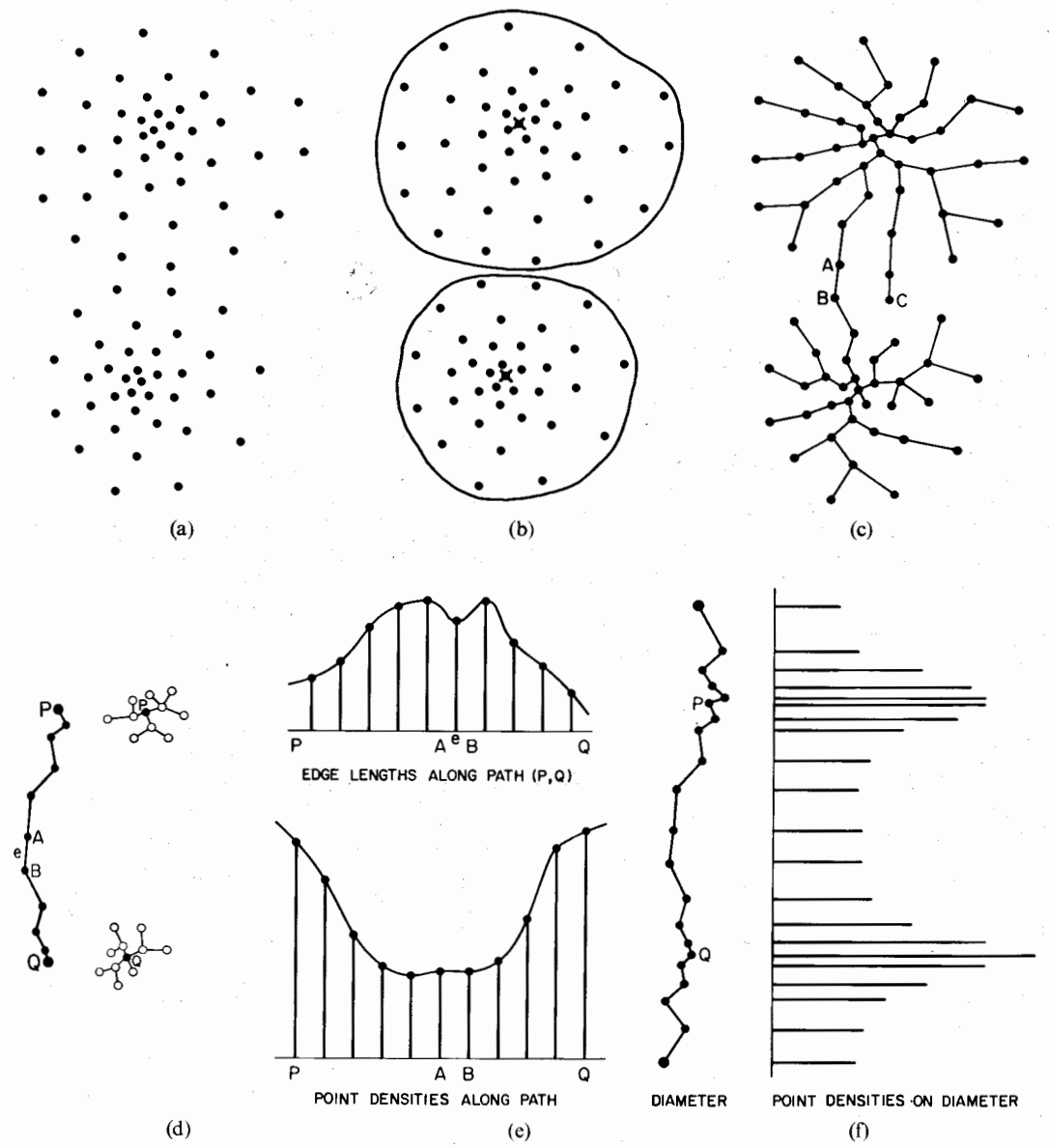


Fig. 8. Touching Gaussian clusters problem. (a) Point set. (b) Centers and approximate boundaries of two clusters. (c) Minimal spanning tree for (a) with cluster-joining edge (A, B). (d) Cluster centers P and Q with path joining them in the MST. (e) Edge lengths and point densities along path (P, Q). (f) Variation of point density along an MST diameter.

maximum because the overlap means density increases.

Fig. 8(e) shows a plot of the edge lengths along the path between centers and the dip at edge (A, B) is a slight overlap effect. The point densities plotted in Fig. 8(e) are gotten by taking the reciprocal of the average length of the two edges of the path incident to the point whose density is being calculated. Edge (A, B) represents the flattest portion of the relative minimum of point densities and edge $e=(A, B)$ is also the relative minimum of edge length between two local maxima; when this edge is deleted from the MST the resulting two connected subtrees correspond almost exactly to the division in Fig. 8(b)—the only error is the misclassification of point C at the periphery.

For those who still doubt the authenticity of this cluster analysis, Fig. 8(f) shows a diameter of the MST with the point densities plotted along side to indicate how well they reveal the cluster structure. The branching structure of the MST reveals the Gaussian-like point density distribution in the following way. Each cluster has a tree structure exhibiting a radial growth away from the cluster center. This radial outward growth is essentially a corollary of Theorem 3 along with the monotonic decrease in point densities along radial lines from the center.

As a concrete use for this variety of cluster detection we paraphrase Sokal and Sneath ([35], p. 174) who point out that two species (the points are vectors of measured features of biological specimens) can be recognized as clusters even though all intermediate forms are present because the hybrids being less frequent represent a saddle point between two mountains (see Fig. 8(f)). This is clearly a description of the touching Gaussian cluster problem. These methods would probably have great difficulty recognizing substantially overlapping Gaussian clusters. See Section XVII for details.

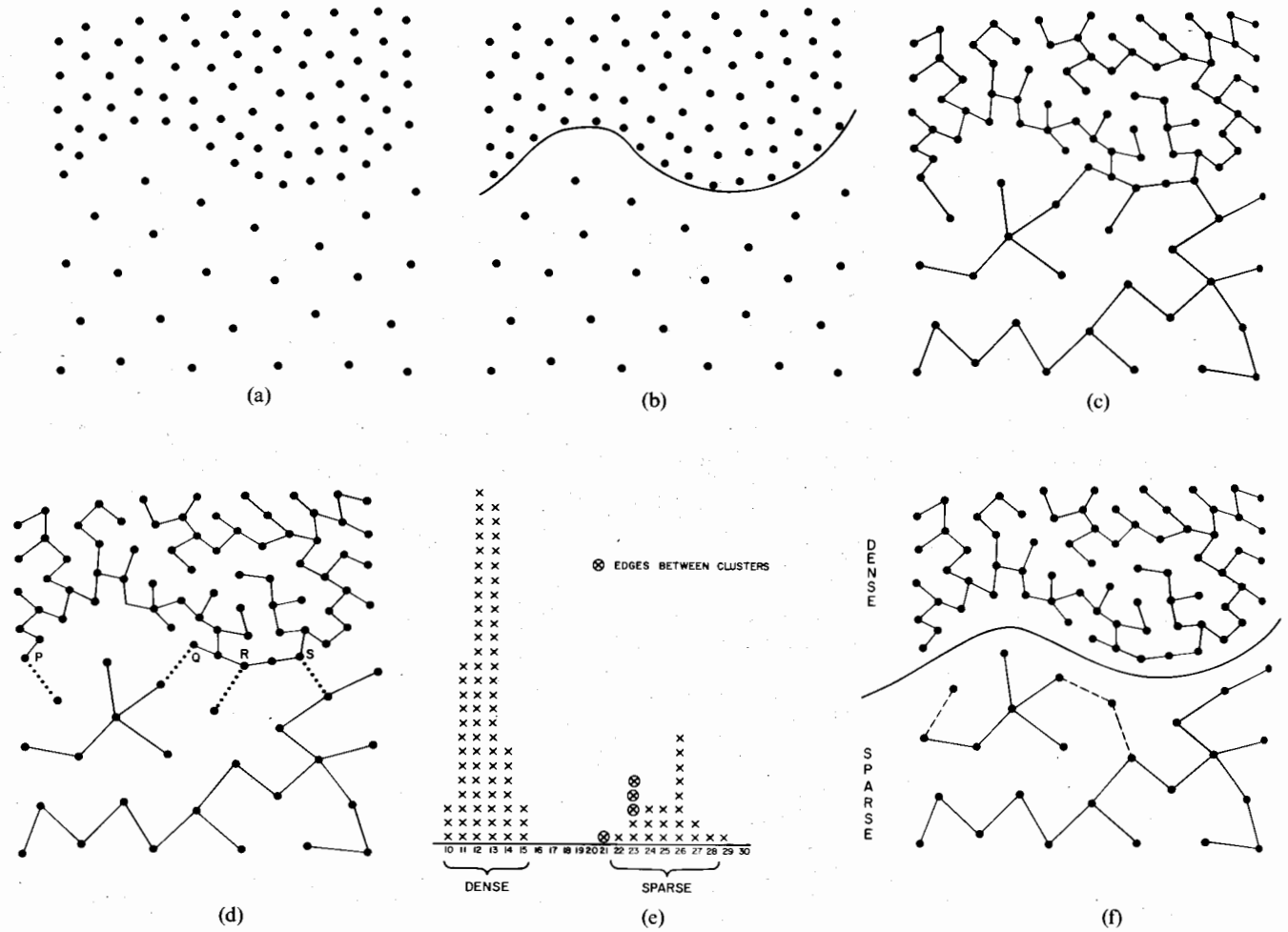


Fig. 9. Density gradient problem. (a) Point set with gradient. (b) Approximate separating boundary for (a). (c) Minimal spanning tree for (a). (d) Gradient edges in the MST. (e) Histogram of edge lengths in the MST showing bimodality. (f) Final cluster trees showing sparse and dense clusters.

XII. Density Gradient Detection

In our next example we face the problem of detecting a sharp gradient in point density between two fairly homogeneous areas of different density as in Fig. 9(a). The boundary that one readily perceives in this point set is drawn in Fib. 9(b). The minimal spanning tree shown in Fig. 9(c) is an excellent example of the effect of Theorem 3 because the upper denser cluster as defined by the boundary in Fig. 9(b) satisfies the hypothesis of Theorem 3 and, indeed, the restriction of the MST to points of the upper cluster is a "connected" subtree. This MST shows how a sparse cluster near a dense one can be severely fragmented by the MST; the restriction of the MST to the lower cluster consists of four connected subtrees, two of which are isolated nodes. One answer to this problem is to detect and delete the densest clusters and then repeat with the remaining point set. Detecting the limits of the denser cluster involves designing an algorithm which can single out the four dotted MST edges in Fig. 9(d). These are the intercluster edges connecting points in different clusters as defined by Fig. 9(b). By histogramming the values of edge length for the MST (see Fig. 9(e)) we immediately recognize that there is a definite and narrow peak around 12.5 and another less definite at about 25.

Especially significant is the well-defined valley or local minimum in the range 16 to 20. We have designated the four intercluster edges by a different symbol in the histogram. In general we would expect them to occupy the range of values between cluster peaks. The idea that intercluster edge lengths would likely be found at significant local minima of the edge length histogram is derived from Prewitt and Mendelsohn [16] who use an analogous idea to determine the best quantization levels for grey-scale digitizings of cytological imagery.

In any case the four intercluster edges can be distinguished from the two sets of intracluster edges in the following way. Select nodes of the MST which have incident edge lengths from both the dense and sparse set determined by Fig. 9(e). Such a criterion singles out the nodes (P, Q, R, S) of Fig. 9(d). Now select all edges incident to these nodes whose edge lengths are not in the dense set. We get precisely the dotted edges of Fig. 9(d). In this example it appears that the edge inconsistency criterion applied to one end of an edge would achieve the same purpose but this is only because the density change is extremely abrupt here.

When the dense cluster has been determined by the deletion of the edges joining it to the sparse cluster, then the nodes of the dense cluster should be erased and the MST recalculated for the remaining points. Fig. 9(f) depicts the subtree of the original MST which serves to define the dense cluster and the new MST for remaining points which defines the sparse cluster. The dashed edges are the three edges

required to construct the new MST from the four fragment subtrees left over when the upper cluster is removed.

XIII. Rationale for MST as Cluster Description

We would like to discuss the capability of the MST of a point set in E^n (or general metric space) to describe the shape (or topology of near connectedness) of the set. Brief glimpses of this descriptive ability have been afforded us by Theorem 3 and several earlier examples, particularly the particle track problem. Here we shall try to develop a deeper understanding of what sort of cluster structure is embodied in the minimal spanning tree and learn how to detect this information in an efficient way.

Fig. 10(a) shows a planar figure with a central blob and four arm-like protuberances. A sample point set from this region is depicted in Fig. 10(b), and the MST in Fig. 10(c). In Fig. 10(d) we show the effect of several iterations of "hair removal" as explained for the particle track problem. As can be observed this has the effect of deleting most shallow branching subtrees off the main global paths in the MST. In this case we continued the iteration until the number of hairs was the same for two successive iterations. The edges remaining are shown as very wide in Fig. 10(d). An obvious defect of the subtree remaining is the fact that the branching at the end of arm-like structures has been deleted. Happily, however, these portions can be reintroduced by adding back sequences of hairs starting from the end nodes of the remaining subtree. We call the resulting tree an *MST skeleton* which seems an appropriate name. It is shown in Fig. 10(e) enclosed by the original figure boundary to show how it reveals some of the geometrical structure of the point set.

It should be clear from this example that arms in the point set will become arms in the MST skeleton but that some arms which show up in the skeleton are not true revelation of arm structure. Measuring branch depths and weights (number of nodes) along the MST skeleton as we did earlier for near-diameter trees will probably reduce this ambiguity somewhat but no general guarantee can be made. A somewhat more promising idea is to calculate an MST for a slightly perturbed version of the original point set and then any arm structures which show up twice in the same area are not likely to be spurious. Fig. 10(f) shows such a perturbed version of the set in Fig. 10(b) with its MST skeleton. The skeleton arms show up in Fig. 10(f) and (e) and are the more outstanding features of the original region.

There are situations where the structure of the MST reveals the geometry of the point set in a more reliable way and that is when the point density is greater near the internal portions of the region as was the case for the Gaussian clusters. Indeed, when point density is inversely proportional to distance from the boundary the MST skeleton is analogous to the "medial-axis" skeleton of Blum [18].

To see the relation between MST and cluster shape under the above condition we describe a type of region which we call a "Gaussian worm" in E^n and discuss what an MST for a sample point set would look like. Let Γ be a smooth rectifiable curve in E^n and $W_r(\Gamma)$ the union of all spherical neighborhoods of radius r centered at points of Γ. Now consider a probability density imposed on $W_r(\Gamma)$ in such a manner that the probability of a point is greater the nearer

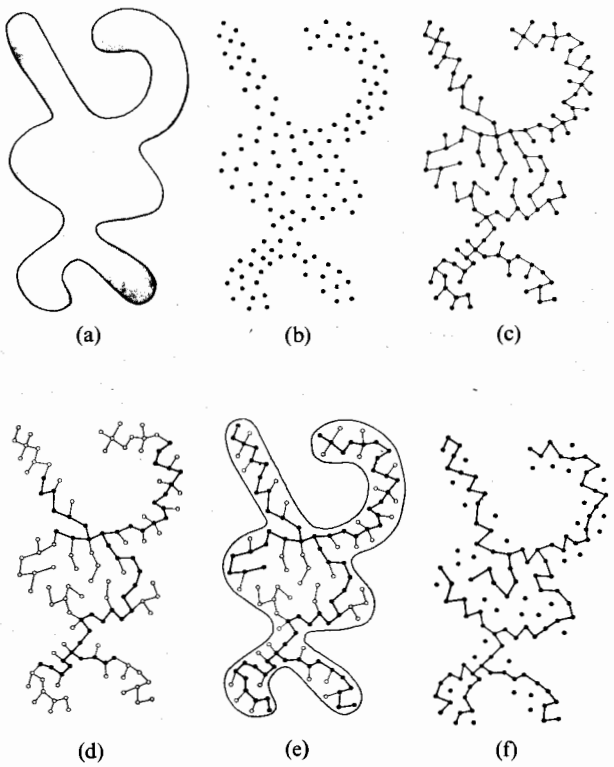


Fig. 10. Cluster description by MST. (a) Planar region. (b) Sample point set from region of (a). (c) Minimal spanning tree for (b). (d) MST after iterated hair removal. (e) MST skeleton. (f) MST skeleton for slightly perturbed version of (b).

to Γ the point is located. We call this a *Gaussian worm with axis* Γ. A sample point set S from such a distribution has an MST whose structure serves to delineate the axis rather well. Take any point p on Γ and pass a plane (hyperplane) through p perpendicular to the direction of Γ at p. This cut divides the sample set S into two pieces P and Q, a partition of S. Theorem 1 tells us that the MST for S contains at least one edge from the link set $\lambda(P, Q)$; now that edge is most likely to occur near the point p on the axis Γ because point density is higher on the axis making nearest distances shorter there than at the periphery of the worm. If we took a transverse wafer slice of the worm and projected the sample points in the wafer onto the plane of the slice it should look something like one of the two clusters in Fig. 8(a). This would suggest a tendency toward radial MST edges pointing toward the axis, as in Fig. 8(c). Combining these two observations we should expect an MST whose diameter path closely approximates the axis Γ with radially branching subtrees off this axial path.

The radial edge phenomenon is quite a bonus; it occurs because edges in the MST tend to follow steepest gradients in point density. Connecting a point to its nearest neighbor is indeed tantamount to selecting a point where the density is most likely higher.

Any strong tendency for a point set to satisfy the Gaussian worm conditions will reflect in the MST and can be detected there.

XIV. Advantages of MST for Cluster Detection

We feel the principal advantage of the MST is its close conformity to the "proximity" principle of perceptual organization enunicated by Wertheimer [24], [26]. Theorems

3 and 4 have a very "gestalt" flavor because the hypothesis of Theorem 3 and the idea of minimax path both represent a specific formalization of the proximity principle. Another widely held principle of perceptual organization states ([20] p. 87) that, "Our nervous systems organize the perceived world in whatever way will keep changes and differences to a minimum."

The MST is a configuration which satisfies a "*minimum principle*" and so there is an analogy between the organization effected by the MST and that effected by our perceptual mechanisms whatever they may be. The MST methods described here represent a possible model to certain perceptual mechanisms which should be tested in psychological experiments. The minimum principle is made more attractive by the demonstrated usefulness in the physical sciences of "principles of least effort" associated with stable configurations.

Another advantage of MST methods is determinacy. This means that the results of applying the method do not depend on random choices in the algorithm or the order in which points are scrutinized but are affected solely by the point set given as input.

The MST of a point set in the plane is *invariant under similarity transformations* (translations, rotations, and changes in size). More generally, it is unchanged under any transformation which preserves the ordering of the edge lengths. All this implies that a point set can retain the same MST under some fairly nonlinear distortions.

Finally, the MST is relatively insensitive to small amounts of noise widely and randomly spread over the field. We have seen this in the particle track problem but the principle applies much more generally. The noise points will very often be end nodes of the MST and inconsistent at that.

In a recent paper on cluster techniques Johnson [28] argues that good clustering algorithms should satisfy the following three properties.

1) Input data should consist solely of a point set and a matrix of similarities.
2) The method should be such that a clear, explicit, and intuitive description of what the clustering accomplishes is possible.
3) The method should be invariant under monotone transformations of similarity measure.

MST methods for the most part satisfy these principles. We accommodate 1 by treating cluster problems in the context of a general metric space. The reader may in fact have noticed that the triangle inequality is never needed so even a metric space is unnecessarily restrictive. Our concentration on two-dimensional examples answers 2 and "monotone invariance" applies because as we have mentioned the MST depends only on the ordering of the lengths of edges. This can be seen from Kruskal's algorithm [11] immediately.

XV. Hierarchical Clusters in Taxonomy

In the application of cluster detection methods to the objective classification of biological specimens it is usually appropriate to be able to detect what amounts to a hierarchy of clusters. For example, specimens tend to be grouped into species and these groups are themselves grouped into genera, etc. The point distribution in Fig. 11(a) is reproduced from the book by Sokal and Sneath on numerical taxonomy ([35] p. 172) and shows graphically what they mean by hierarchical clusters. The two levels of clustering apparent to most observers of Fig. 11(a) is illustrated in Fig. 11(b).

Once again the MST reveals the hierarchical structure of the clustering in Fig. 11(a). In Fig. 11(c) we show the minimal spanning tree for Fig. 11(a). The dashed and dotted edges in this figure are all inconsistent so we immediately obtain the correct inner level of clustering depicted in Fig. 11(b). If we now coalesce the points in each cluster we obtain a shrunken version of the MST (a homomorphic image in the language of graph theory) consisting of only dashed and dotted edges and whose nodes correspond to the inner clusters of Fig. 11(b). In this new MST the inconsistent edges are the dotted ones and the clusters obtained are the outer clusters of Fig. 11(b) as desired. This technique is similar to but somewhat more sensitive than the dendrograms of [35], [36].

The degree to which this hierarchy is explicit in the original MST is shown by histogramming the edge lengths (see Fig. 11(d)) and observing the one-dimensional clustering that occurs.

As an example of what real data may look like we have graphed the 50 specimens each of Iris Setosa and Iris Versicolor from data tabulated by Fisher [39]. We have for this example used only two of the four variables found there. The MST for this point distribution is shown in Fig. 11(e) and the speciation is reflected in this structure. A later section treats the four-dimensional case in full.

XVI. Clustering, Linear Separability, and Relative Compactness

In dealing with problems related to the separability of an n-dimensional point set into distinct classes we have found that there are three different criteria for separability which it is useful to carefully distinguish. The first two are "cluster detection" and "hyperplane separation;" Fig. 12(a) illustrates the distinction. In the hyperplane case we are given the classes (solid and open points) *a priori* and the question is whether or not a hyperplane (line) exists which separates the points into two subsets identical (or approximately so) to the given classes. In the cluster case the question is to detect the existence of cluster structure (dashed line) based only on the interpoint distances. This single example shows clearly that the two questions are quite different. The third criterion is "relative compactness" as introduced by Arkadev and Braverman ([8] pp. 20–26) and is important for pattern recognition in feature spaces. In this case we are given a set of points divided into classes and the question is to determine to what extent the classes are intermixed in space. The upper half of Fig. 12(b) shows two classes which are relatively compact, whereas in the lower half the two classes are badly mixed up. It is not an easy matter to formulate a rigorous definition of this notion but it seems to be related to how smooth a curve (hypersurface) can be drawn separating the classes or the ratio between boundary and interior points of the two classes. An *interior* point of a class of points is one which is not very near points of the other class. A *relative-boundary* point is one which is near

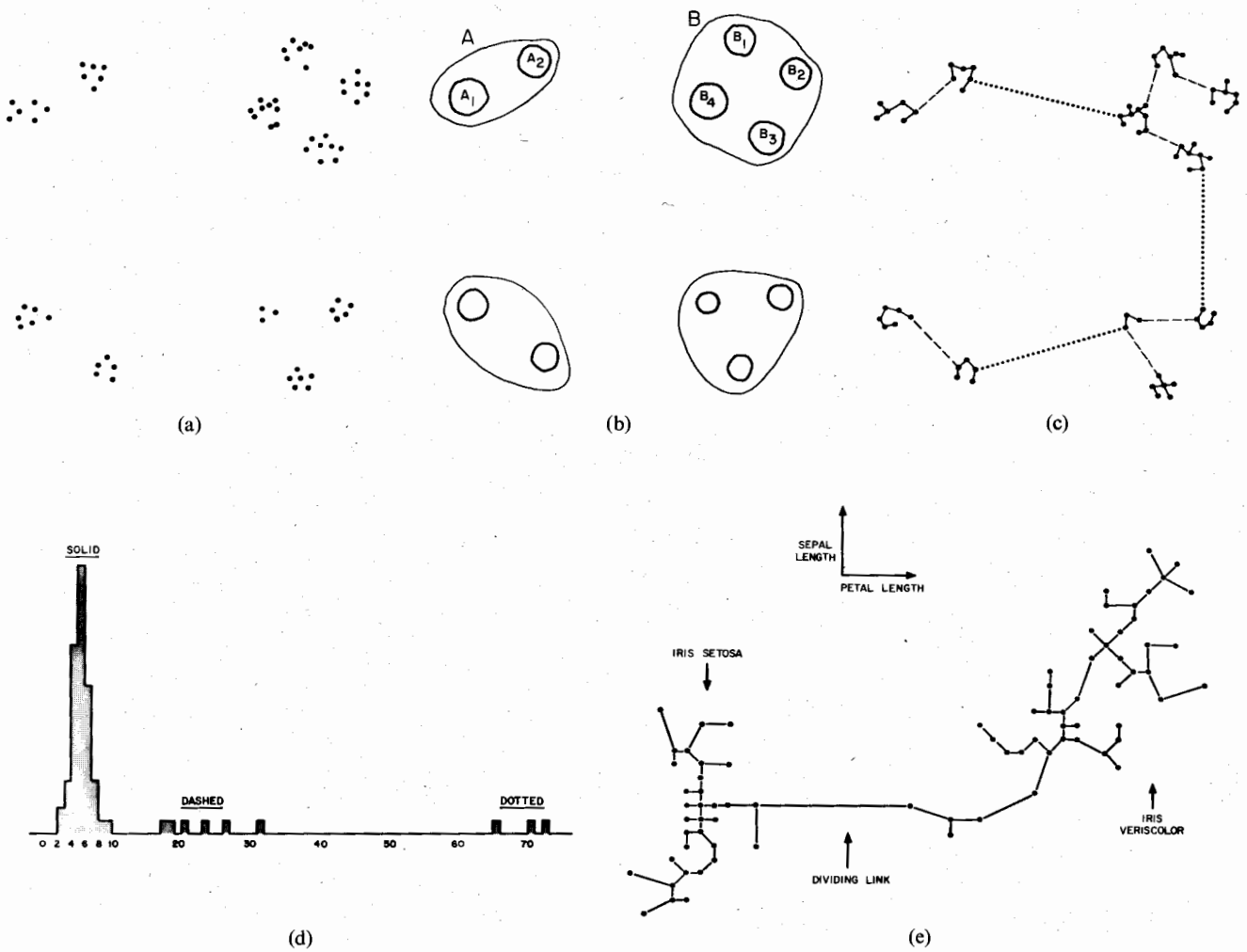


Fig. 11. Hierarchical clusters and biological speciation. (a) Hierarchical clusters from [35] p. 172, Fig. 7(1-c). (b) Two levels of clustering present in (a). (c) MST for (a) showing two separate levels of inconsistent edges. (d) Histogram of MST edge lengths. (e) MST for Iris species from Fisher [39].

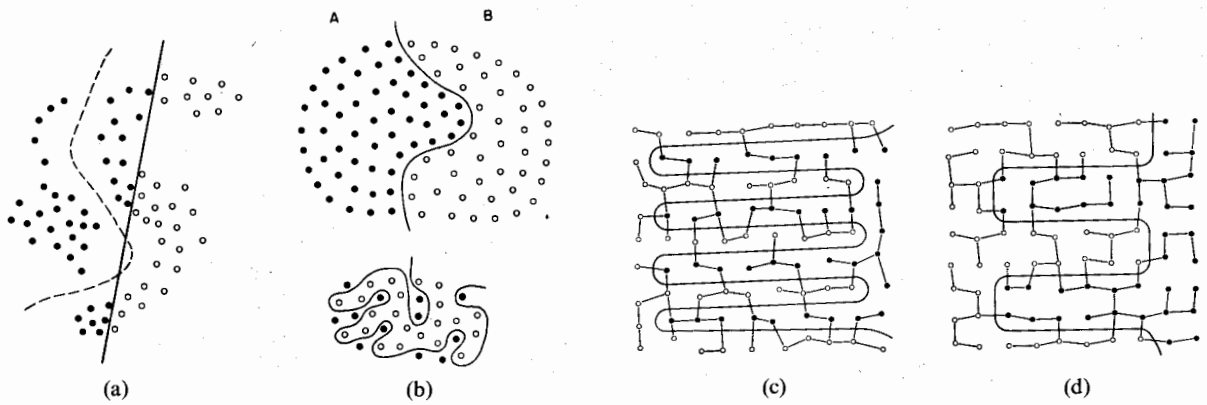


Fig. 12. Separation criteria compared. (a) Illustration of the distinction between "linearly separable" and "cluster separation." (b) Relative compactness exemplified. (c) MST for relatively noncompact set. (d) MST for relatively compact set.

some point of the other class. Once "near" has been solidly defined the above definition partitions each class into interior and boundary points and the degree of relative compactness is reflected in the ratio between the number of boundary and interior points. This can be seen very quickly in Fig. 12(b) where the mixed set contains no clearly interior points.

The MST can be used to approximate the degree of "relative compactness" of the two classes in a point set. Just construct the MST (which takes no account of classes) and then count the proportion of MST edges joining points in different classes. This approximates the ratio of boundary to interior points.

Fig. 12(c) and (d) show MSTs for two point sets each consisting of two classes of points. We calculate the relative compactness by counting the number of MST edges which join similar points and dividing by one less than the total number of MST edges. We take one less because the MST

must have at least one edge between dissimilar points and we want relative compactness to be 1 for separated clusters. In Fig. 12(c) the relative compactness is 59 percent while in Fig. 12(d) it is 77 percent.

The next section has several more examples of this relative-compactness analysis via the MST. The results obtained there suggest that the simple count of crossover edges may be somewhat too crude as a measure of relative compactness. Occasionally a single node which falls into the area of another class will generate two or three crossover edges and hence is weighted too heavily.

XVII. Computational Experience with MST Clustering

Programs have been written in PL/1 to calculate the distance matrix for a set in n space, construct the minimal spanning tree in a plex-structured format, compute the relative compactness assuming each point has been given a class designation, calculate approximate point density at each point, determine a set of inconsistent edges, and partition the point set into clusters based on deletion of inconsistent edges from the MST. In its present form the determination of edge inconsistency requires that the length of the candidate edge exceed the average local edge length on each end by σ_T units of the respective sample standard deviation and furthermore that the ratio of edge length to each average exceed f_T; the statistics are taken from a subtree of depth d. To get slightly better statistics we use a first pass with $\sigma_T=3$ and $f_T=2$ and eliminate these oversize edges from the statistics at the second pass. This particular determination of inconsistency obviously does not detect one-way gradients however steep.

Several point sets have been analyzed with the help of the above programs and we shall discuss our conclusions briefly. Fig. 13(a) depicts a point set gotten by adding small amounts of "random jitter" to a subset of lattice points. Using fairly low thresholds $\sigma_T=2$, $f_T=1.3$, we found two inconsistent edges. One bridges the quite visible vertical gap and would have been judged inconsistent with $\sigma_T=2.9$, $f_T=1.71$. The other joins the two points in the lower left corner to the larger adjacent cluster and would have been inconsistent with $\sigma_T=2.1, f_T=1.47$. The local neighborhood depth used was $d=3$.

The most extensive analysis was done on the Iris data from Fisher [39] mentioned earlier. This involved all 149 (there is a repetition which we deleted) points in four-dimensional space representing three species of iris. To help visualize this point set Fig. 13(d) shows the original four-dimensional set mapped into a two-dimensional space in such a way as to distort the original interpoint distances in a minimal way [42].

Using the current inconsistency algorithm with $d=3$, $\sigma_T=2$, and $f_T=2$, we obtain eight inconsistent edges, five of which are end edges separating single points from the larger clusters; a sixth edge separates a two-point cluster. The final two edges separate the Iris Setosa species and a small four-point cluster at one end of the Iris Versicolor species. Most of this can be seen in Fig. 13(e) where part of the MST has been drawn in the plane with its edge lengths to scale. In spite of the fact that Versicolor and Virginica do not separate on the basis of our inconsistency measure, the

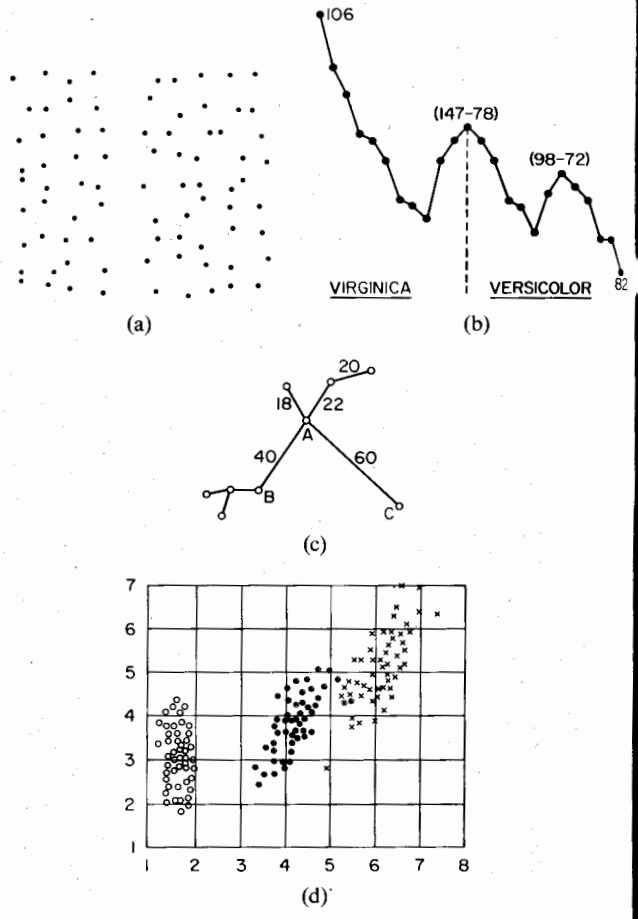
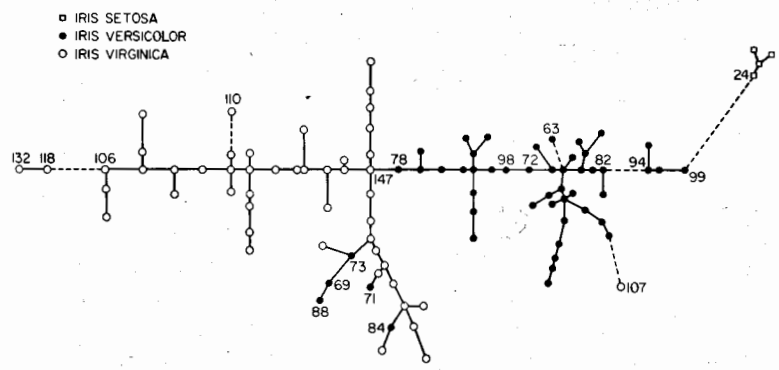


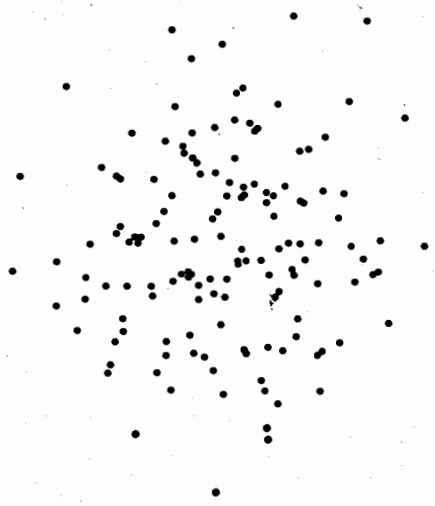
Fig. 13. Results of computer experiments. (a) Randomly jittered point clusters. (b) Smoothed edge lengths along MST path from (e). (c) Example showing failure of edge-inconsistency criterion. (d) Diagram of Iris species from Sammon [42] illustrating "separated clusters," "touching clusters," and "relatively compact sets." (Fig. 13 continued on facing page.)

two species are relatively compact at a ratio of 94 percent since there are only six MST edges between the two species out of a total 98. Furthermore, it is not hard to see from the MST that edge (147–78) is the main bridge between the two species and may warrant more detailed investigation. Fig. 13(b) is a smoothed plot of edge lengths along the MST path fron node 106 to node 82. Each point represents the average of three adjacent edge lengths. According to this data the edge (147–78) is a local minimum of point density along the path as is the edge (98–72). The edge (147–78) would be judged inconsistent for $d=2$, $\sigma_T=1.4$, and $f_T=1.4$. We conclude that there is some evidence in the MST indicating a possible division at (147–78) without any use of the iris class information. At the very least this information reinforces the assumed speciation. It also suggests that Iris Versicolor may be subdivided into three subspecies (one fairly small) as determined by the inconsistent edge (82–94) and the relative minimum edge (98–72). Evidence of this sort from the MST may be quite helpful in suggesting where further investigations might be made.

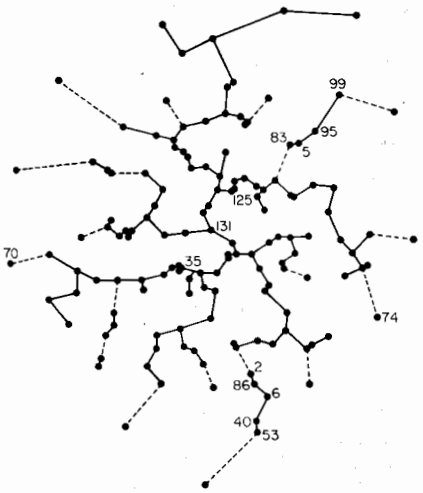
Since the MST for all three Iris species together put five Versicolors on the part of the tree belonging to Virginica and vice versa for one Virginica, we wondered if some mistake could have been made in the original tabulating of the data. To test these doubts we performed MST cluster analysis on the two species separately. The only clearly signifi-



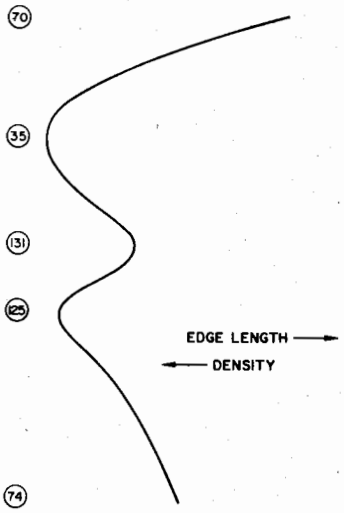
(e)



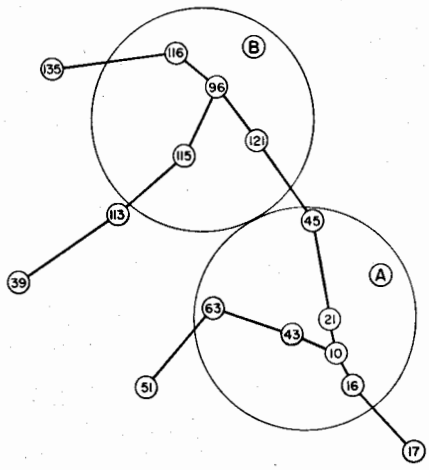
(f)



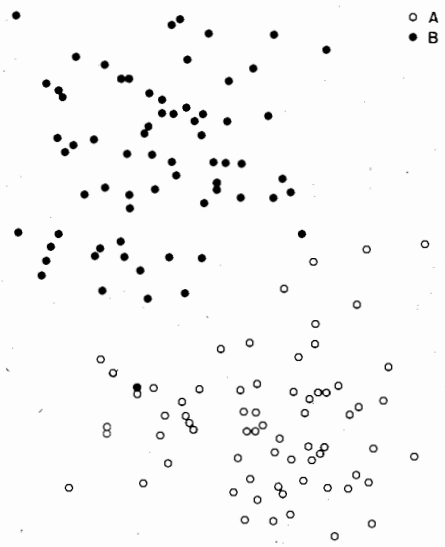
(g)



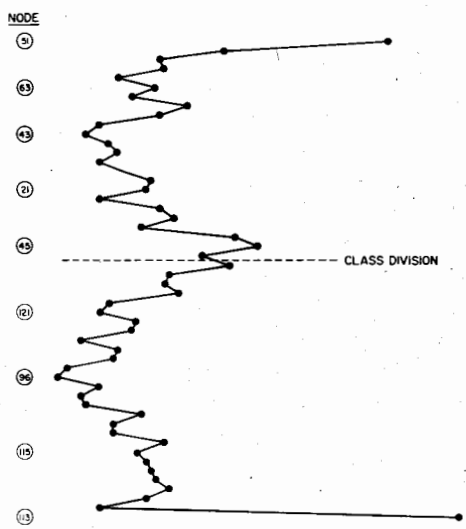
(h)



(i)



(j)



(k)

Fig. 13. (continued) (e) MST for touching Iris classes with edge lengths to scale. (f) Random Gaussian cluster generated by computer. (g) MST of (f) showing inconsistent edges. (h) Smoothed values of point density along MST path in (g). (i) Global aspect of MST for (j). (j) Slightly overlapping clusters. (k) Smoothed edge lengths along a major MST path for point set in (j).

cant result was that node 107 is well separated (factor=2.7) from the species it is supposed to belong to.

Two problems were encountered in analyzing the Iris data. The first is the crudeness of measurements from [39]. In many cases the distance from a point to its nearest neighbor is no more than two or three times the least significant digit in feature measurement. Fig. 11(e) shows this phenomenon quite graphically. The second problem is a deficiency in the current determination of inconsistency. There are cases where an edge which should probably be judged inconsistent misses detection because of the phenomenon depicted in Fig. 13(c). If edge AC is not considered then edge AB will be inconsistent with a factor=2. When AC is part of the local neighborhood of edge AB at node A then the factor drops to 1.33—a result we find unsatisfying.

Comparing Fig. 13(d) from Sammon [42] with the results of MST cluster analysis above we can see several points of fairly detailed agreement. Setosa is clearly a separate cluster whereas Versicolor and Virginica are touching. Node 107 which was separated from both species seems to have an analog in Fig. 13(d) and several Versicolors seem to be imbedded in the Virginica. There is a set of four points at the lower left extremity of the middle cluster in Fig. 13(d) that probably corresponds to the four-point cluster we found but the gap is not clear. This may be explained by the fact that Sammon's measure of distortion of a point set is based on the average distortion of individual points and this allows fairly large distortions in one or two points of the set. Point-set distortions based on the maximum of individual distortions are usually avoided because they are very noise sensitive and mathematically intractable.

We have also tested our methods on Gaussian clusters. Fig. 13(f) shows 144 points (X, Y) with each X and Y chosen independently from a normal distribution with 0 mean and unit standard deviation. Fig. 13(g) is the MST for this point set showing inconsistent edges for $d=2$, $\sigma_T=2$, and $f_T=1.5$. In spite of the generously low factor threshold the main cluster contains 113 points and the next largest only five points. Disregarding the single-point clusters and raising f_T to 2.0 we would get only two small clusters (83, 5, 95, 99) and (2, 86, 6, 40, 53). With the exception of end edges no inconsistent edge would be found for $f_T=2.6$. Fig. 13(h) shows a smoothed plot of point density along the path joining node 70 to node 74. There are two relative maxima of point density and this bimodal phenomenon can be observed in the point set which is unexpectedly sparse at the center. This bimodality and cluster fragmentation at the periphery would probably occur less frequently for substantially larger sample sizes. As can be seen from Fig. 13(g) the smaller fragments (including isolated nodes) tend to be near the periphery of the cluster and the directions of MST edges tend to be toward the center of the cluster in a fashion discussed earlier in conjunction with more homogeneous Gaussian clusters.

Finally, we tried several cases of overlapping Gaussian "classes" to see what could be detected from the MST analysis. In each case there were 72 points in each of two classes labeled A and B. Each class was a translated version of a sample from the distribution used above. In each case relative compactness was calculated and seemed to be well correlated with the degree of overlap of the clusters. The following table gives relative compactness as a function of the distance between the centers of the two classes.

Distance of Centers	0	1.5	2.0	2.5	3.0	3.35
Relative Compactness	57	(70, 70)	(80, 83)	89.0	93.0	97.00

An investigation of the point density along major paths of the cases with distance=2 revealed no information that could warrant separating the two classes. A similar investigation of the case with distance=3.35 revealed enough to divide the MST at a definite relative minimum of point density which division fit the class designations quite well. Fig. 13(j) depicts most of the point set and Fig. 13(i) shows in condensed form several of the major branches in the MST. The longest MST path joins node 39 to node 51 and Fig. 13(k) plots point density along this path. The apparent relative minimum around edge (11, 45) would break the MST only two steps away from edge (8, 74) which represents the class separation.

We conclude that fairly inhomogeneous overlapping Gaussian clusters cannot be detected as separate unless the overlap is minor.

XVIII. Descriptions for Further Research

We see three major directions in which this paper suggests further investigation. One is in psychology, one in the area of cluster description, and one relates to feature space determination for pattern discrimination.

For the psychologist interested in visual perception we feel the MST along with the idea of "edge inconsistency" and other techniques described earlier afford a quantitative tool for characterizing patterns of points in the plane. It seems natural to want to investigate in detail the possible correlation between the clusters of human perception and those determined by our quantitative tools. A program of research similar to the work of Attneave and Arnoult [41] is what we have in mind. We feel our work strongly suggests that the inconsistent edges of the MST of a point set are correlated with cluster separations seen by human visual perception. A more precise statement of this correlation awaits appropriate perceptual experiments; for example, we conjecture that the *percentage* of subjects who see a separation between two clusters is monotonically related to the *ratio* measuring the degree of inconsistency of the edge bridging the two clusters. We also suspect that the ease of separation of two clusters (visually) is somewhat dependent on the sizes of the clusters and the homogeneity of cluster density. There are any number of questions that one can pose in this context and the answers may provide new hints to psychophysicists.

The second matter of unfinished business is a more extensive study of what quantitative structure of a point set can be extracted from its MST. We have made several simple probes in this paper (for example, path histograms and iterated hair removal) but these can only be considered as a beginning. Other techniques will probably be motivated by the need to describe cluster structure that has not occurred to us. We have concentrated on point sets embedded in metric spaces because there was a great deal we could say at this level of generality; nevertheless, point sets in more

structured spaces (e.g., E^n where direction is defined) can probably be better described by using the additional structure. For example, the radial structure of density in the Gaussian clusters of Fig. 8(a) is only hinted at by the topology of its MST but can be confirmed by observing the consistency of edge direction in the radial paths and the uniform distribution of radial directions around the unit circle (see Fig. 8(c)).

The third direction for research suggested by this paper is to determine if a particular n-dimensional feature space is good for the discrimination of two classes of points. The optimum situation is when the two classes are actually two clusters as well separated as those in Fig. 1(a). Touching clusters as in Fig. 1(i) or (j) would still be a useful result. If the situation were like that in the upper half of Fig. 12(e) then discrimination can probably be accomplished even though the classes are not clusters. Relative compactness of two classes is enough to warrant that a "nearest-neighbor" discrimination algorithm [6] will be moderately successful. If the two classes are mixed up like the bottom half of Fig. 12(b) then we claim there is no hope for this feature space and another set of measurements should be tried. The methods for determining what situation actually applies are contained here and this gives us a handle on the very important problem of judging the efficacy of a given feature space for separating a pair of classes.

Appendix

Theorem 1: Any MST for G contains at least one edge from each $\lambda(P, Q)$.

Proof: We show that a spanning tree T^* containing no edge from $\lambda(P, Q)$ can be improved by switching one of its edges for one in $\lambda(P, Q)$. In fact, select any edge $(X, Y) \in \lambda(P, Q)$ and add it to T^* to produce a new graph with precisely one circuit. The portion of this circuit which lies in T^* must have at least one edge (U, V) in the cut set $C(P, Q)$ because X and Y are in P and Q, respectively. The edge (U, V) is not in $\lambda(P, Q)$ by definition of T^*. The spanning tree $T = \{T^* - (U, V)\} \cup (X, Y)$ has smaller weight than T^* because by definition of $\lambda(P, Q)$, $w(X, Y) < w(U, V)$. Thus any minimal spanning tree must have at least one edge from $\lambda(P, Q)$.

Lemma 1: Each edge (X, Y) of a spanning tree T determines a unique partition (P, Q) of the nodes of G in a natural way so that $C(P, Q)$ contains exactly one edge in T.

Proof: Let $T' = T - (X, Y)$ be the graph obtained by deleting edge (X, Y) from the spanning tree T. Since every edge of a tree is a bridge it follows that T' has two connected components T'_1 and T'_2 and the nodes X and Y are in different components (say $X \in T'_1$, $Y \in T'_2$). Now let P denote the nodes of T'_1 and Q the nodes of T'_2. (P, Q) certainly are disjoint and they contain all nodes of G since no nodes were deleted from T which spans G. Also the only edge in T joining P to Q is the deleted edge (X, Y) and so the lemma is proved.

Theorem 2: All MST edges are links of some partition of G.

Proof: Let T be any MST for G and (X, Y) any edge in T. Let (P, Q) be the unique partition assured by Lemma 1. From Theorem 1 we see that T must contain at least one edge from $\lambda(P, Q)$, but since T contains only one edge from $C(P, Q)$ it certainly contains only one edge of $\lambda(P, Q)$. The edge (X, Y) must therefore be *the* edge which belongs to $\lambda(P, Q)$ and so (X, Y) is a link of G.

Corollary 1 (Kruskal [11]): If all edge weights of G are different then the MST is unique.

Proof: In this case each $\lambda(P, Q)$ is a single edge which must belong to each MST. Thus the set of all links of G, $L(G) \subseteq T$ for any MST. According to Theorem 2, $T \subseteq L(G)$ for any MST. This means $T \subseteq L(G) \subseteq T$ for any MST and hence any MST must be identical to $L(G)$ and so unique. We have in fact proved the stronger.

Corollary 2: If all edge weights of G are different then the MST is unique and identical to $L(G)$.

The following theorem is the most important we shall derive because it relates the MST to the problem of cluster detection.

Theorem 3: If S denotes the nodes of G and C is a nonempty subset of S with the property that $\rho(P, Q) < \rho(C, S - C)$ for all partitions (P, Q) of C then the restriction of any MST to the nodes of C forms a connected subtree of the MST.

Proof: Select an arbitrary partition (P, Q) of C and let $R = S - C$. We must show that any MST contains at least one edge in the cut set $C(P, Q)$. To do this we need to show that $\rho(P, Q) < \rho(P, R)$ for then $\lambda(P, S - P) \subseteq C(P, Q)$.

First notice that

$$\rho(C, S - C) = \rho(P \cup Q, R) = \min \{\rho(P, R), \rho(Q, R)\}$$

and hence

$$\rho(P, R) \geq \rho(C, S - C).$$

By the hypothesis of the theorem

$$\rho(C, S - C) > \rho(P, Q)$$

and therefore

$$\rho(P, Q) < \rho(C, S - C) \leq (P, R).$$

As indicated earlier this implies the link set $\lambda(P, S - P)$ is a subset of the cut set $C(P, Q)$. Invoking Theorem 1 we conclude that any MST has at least one edge from $\lambda(P, S - P)$ and hence from $C(P, Q)$ whenever (P, Q) partitions C. This means the restriction of an MST to C cannot fall into two or more components.

Algorithm 1 (Kruskal [11]): Arrange the edges of G in order from smallest to largest weight and then select edges in order making sure to select only edges which do not form a circuit with those already chosen. Stop when $(n - 1)$ edges have been selected where n is the number of nodes in G. The set of edges is then an MST for G.

Example: The following table shows how Algorithm 1 would work on the graph in Fig. 4(a).

Edge	Weight	Circuit	MST Edges
BC	2		*
DF	2		*
DE	3		*
EF	4	$(DEFD)$	
AB	4		*
AC	5	$(ABCA)$	
AD	8		* (5th edge)
CD	9		
CE	10		

Algorithm 2 (Prim [12]): Begin with an arbitrary node of G and add the edge with smallest weight connected to this node. This edge with its two end nodes constitutes fragment tree T_1. The kth fragment tree is gotten by adding the shortest edge from T_{k-1} to the nodes of G not in T_{k-1}. This continues until T_{n-1} is the desired MST.

In this algorithm the MST is grown from a single node by adding the closest node to the current tree at each stage along with the edge corresponding to that closest distance (smallest weight).

Example: For the graph of Fig. 4(a) starting with node A we get the following.

Fragment Nodes	New Edge	Weight
A	AB	4
A, B	BC	2
A, B, C	AD	8
A, B, C, D	DF	2
A, B, C, D, F	DE	3
A, B, C, D, F, E		

From the point of view of computational efficiency Algorithm 2 is the best when done by computer program. Algorithm 1 requires presorting of all edges and must test for existence of circuits at each step, both of which are nontrivial tasks computationally. Algorithm 2, on the other hand, looks at each edge exactly once and can be programmed in such a way that only n edges need be in the computer memory at one time where n is, as before, the number of nodes in G. Since the total number of edges in our graphs will be approximately $n^2/2$ this is important.

Gower and Ross [36] give ALGOL programs to construct an MST using Algorithm 2 and to print out the MST. They discuss storage and time requirements for their programs in some detail. We have implemented Algorithm 2 in PL/1 but the MST is represented by a plex-structure so that neighborhood explorations in the MST can be programmed in a more straightforward and efficient manner.

Acknowledgment

The author thanks one of the referees whose suggestions led to substantial expository improvements in the paper, and the actual implementation and testing of Section XVII.

References

[1] R. E. Bonner, "On some clustering techniques," *IBM J.*, pp. 22–32, January 1964.
[2] A. Rosenfeld, *Picture Processing by Computer*. New York: Academic Press, 1969.
[3] S. M. Pizer and H. G. Vetter, "Perception and processing of medical radioisotope scans," in *Pictorial Pattern Recognition*. Washington, D. C.: Thompson Book Co., 1968, pp. 147–156.
[4] A. Rosenfeld and J. L. Pfaltz, "Distance functions on digital pictures," in *Pattern Recognition*, vol. I. New York: Pergamon Press, 1968, pp. 33–61.
[5] R. Clark and W. F. Miller, "Computer based data analysis systems at Argonne," *Methods in Computational Physics*, vol. V.
[6] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 21–23, January 1967.
[7] G. Nagy, "State of the art in pattern recognition," *Proc. IEEE*, vol. 56, pp. 836–862, May 1968.
[8] A. G. Arkadev and E. M. Braverman, *Computers and Pattern Recognition*. Washington, D. C.: Thompson Book Co., 1967.
[9] B. Julesz, "Visual pattern discrimination," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 84–92, February 1962.
[10] R. Narasimhan, "Syntactic descriptions of pictures and gestalt phenomena of visual perception," Digital Computer Lab., University of Illinois, Urbana, Ill., Rept. 142, July 1963.
[11] J. B. Kruskal, Jr., "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proc. Amer. Math. Soc.*, no. 7, pp. 48–50, 1956.
[12] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Sys. Tech. J.*, pp. 1389–1401, November 1957.
[13] E. W. Dijkstra, "Some theorems on spanning subtrees of a graph," *Kon. Ned. Akad. Wetensch., Versl. Gewone Vergad. Afd. Natuurk.*, series A, vol. 63, no. 2; also *Indag. Math.*, vol. 22, no. 2, pp. 196–199, 1960.
[14] R. Kalaba, "Graph theory and automatic control," in *Applied Combinatorial Mathematics*. New York: Wiley, 1964, ch. 8.
[15] I. Pohl, "Bi-directional and heuristic search in path problems," Stanford Linear Accelerator Center, Stanford University, Stanford, Calif., Rept. SLAC-104, pp. 111–115, May 1969.
[16] J. M. S. Prewitt and M. L. Mendelsohn, "The analysis of cell images," *Ann. N. Y. Acad. Sci.*, no. 128, pp. 1035–1953, 1966.
[17] G. H. Ball, "A comparison of some cluster-seeking techniques," Stanford Res. Inst., Stanford, Calif., Tech. Rept. RADC-TR-66-514, November 1966.
[18] H. Blum, "A transformation for extracting new descriptors of shape," *Models for the Perception of Speech and Visual Form*. Cambridge, Mass.: M.I.T. Press, 1967, pp. 362–380.
[19] W. Köhler, *Gestalt Psychology*. New York: Mentor Books, 1947.
[20] J. E. Hochberg, *Perception*. Englewood Cliffs, N. J.: Prentice-Hall, 1964.
[21] K. Koffka, *Principles of Gestalt Psychology*. New York: Harcourt, 1935.
[22] W. Ellis, *A Source Book of Gestalt Psychology*. New York: Harcourt, 1938.
[23] D. Beardsley and M. Wertheimer, *Readings in Perception*. Princeton, N. J.: Van Nostrand, 1958.
[24] M. Wertheimer, "Laws of organization in perceptual forms," in *A Source Book of Gestalt Psychology*, W. Ellis, Ed. New York: Harcourt, 1938.
[25] K. Gottschaldt, "Gestalt factors and repetition," in *A Source Book of Gestalt Psychology*, W. Ellis, Ed. New York: Harcourt, 1938.
[26] M. Wertheimer, "Principles of perceptual organization," in *Readings in Perception*, D. Beardsley and M. Wertheimer, Eds. Princeton, N. J.: Van Nostrand, 1958.
[27] J. E. Hochberg, "Effects of the gestalt revolution: The Cornell symposium on perception," in *Readings in Perception*, D. Beardsley and M. Wertheimer, Eds. Princeton, N. J.: Van Nostrand, 1958.
[28] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, pp. 241–254, September 1967.
[29] C. T. Zahn, "A formal description for two-dimensional patterns," *Proc. Int. Joint Conf. on Artificial Intelligence*, pp. 621–628, May 1969.
[30] ——, "Two-dimensional pattern description and recognition via curvaturepoints," Stanford Linear Accelerator Center, Stanford University, Stanford, Calif., Rept. SLAC-70, December 1966.
[31] C. Abraham, "Evaluation of clusters on the basis of random graph theory," IBM Corp., Yorktown Heights, N. Y., IBM Res. Memo, November 1962.
[32] P. V. C. Hough, "Method and means for recognizing complex patterns," U. S. Patent 3 069 654, December 1962.
[33] I. C. Ross and F. Harary, "Identification of the liaison persons of an organization using the structure matrix," *Management Sci.*, pp. 251–258, 1955.
[34] C. T. Zahn, unpublished manuscript, 1969.
[35] R. R. Sokal and P. H. Sneath, *Principles of Numerical Taxonomy*. San Francisco, Calif.: Freeman, 1963.
[36] J. C. Gower and G. J. S. Ross, "Minimum spanning trees and single linkage cluster analysis," *Appl. Statistics*, vol. 18, no. 1, pp. 54–64, 1969.
[37] M. Minsky and S. Papert, *Perceptrons*. Cambridge, Mass.: M.I.T. Press, 1969, p. 233.
[38] A. Rosenfeld, R. B. Thomas, and Y. H. Lee, "Edge and curve enhancement in digital pictures," Computer Sci. Center, University of Maryland, College Park, Md., Tech. Rept. 69–93, May 1969.
[39] R. A. Fisher, "Multiple measurements in taxonomic problems," *Contributions to Mathematical Statistics*. New York: Wiley.
[40] O. Ore, *Theory of Graphs*, Amer. Math. Soc. Colloquium Publ., vol. XXXVIII, 1962.
[41] F. Attneave and M. D. Arnoult, "The quantitative study of shape and pattern perception," *Pattern Recognition*, L. Uhr, Ed. New York: Wiley, 1966.
[42] J. W. Sammon, Jr., "A nonlinear mapping for data structure analysis," *IEEE Trans. Computers*, vol. C-18, pp. 401–409, May 1969.