Dot Pattern Processing Using Voronoi Neighborhoods

NARENDRA AHUJA

Abstract—A sound notion of the neighborhood of a point is essential for analyzing dot patterns. The past work in this direction has concentrated on identifying pairs of points that are *neighbors*. Examples of such methods include those based on a fixed radius, k-nearest neighbors, minimal spanning tree, relative neighborhood graph, and the Gabriel graph. This correspondence considers the use of the region enclosed by a point's Voronoi polygon as its *neighborhood*. It is argued that the Voronoi polygons possess intuitively appealing characteristics, as would be expected from the neighborhood of a point. Geometrical characteristics of the Voronoi neighborhood are used as features in dot pattern processing. Procedures for segmentation, matching, and perceptual border extraction using the Voronoi neighborhood are outlined. Extensions of the Voronoi definition to other domains are discussed.

Index Terms—Clustering, computational complexity, dot patterns, Gabriel graph, k-nearest neighbors, matching, minimal spanning tree, neighborhood, neighbors, perceptual boundary extraction, relative neighborhood graph, Voronoi tessellation.

I. Introduction

In processing visual information one often encounters dot patterns instead of gray level or color images. For example, objects in an image are often represented by the locations of some of their spatial features, such as spots, corners, etc. This is done to reduce the sensitivity of the analysis to changes in lighting conditions, scale, orientation of the camera, sensor characteristics, geometrical distortions, etc. The locations of spatial features also serve as a natural choice as landmarks in relating multiple views of real world scenes. Differences in images of the same scene may be induced by the relative motion of the camera and the scene, by the relative displacements of the cameras, or by the motion of objects in the scene. The spatial relationships between corresponding points in different dot patterns are used in deducing depth, velocity, and shape information about the scene. In pattern recognition dot patterns are ubiquitous. Representation of images by the values of certain features measured on the image parts provides dot patterns in the feature space. Pattern recognition procedures such as classification and clustering operate on such dot patterns. The features used may not be spatial, but may be, for example, values of certain transform coefficients. In cartography points denote locations of landmarks detected by airborne sensors. The nighttime sky is a natural dot pattern. Operations commonly performed include identifying certain star clusters and matching stored star patterns against repeated observations. An air traffic situation may be represented by specifying the point locations of aircrafts. This representation may then be used to detect potential collisions. Research on visual perception has made extensive use of dot patterns to investigate human image understanding. Dot patterns with controlled characteristics have been used in experiments evaluating models of certain visual processes such as texture, motion perception, and stereopsis.

Manuscript received October 6, 1980; revised November 17, 1981. This work was supported in part by the National Science Foundation under Grant ECS-8106008 and the Joint Services Electronics Program (U.S. Army, Navy, and Air Force) under Contract N00014-79-C-0424.

The author is with the Coordinated Science Laboratory and the Department of Electrical Engineering, University of Illinois, Urbana, IL 61801.

While regions in a gray level image can be described in terms of their geometrical properties, this clearly does not extend to points, which are the structural units of dot patterns. Any characterization or comparison of dot patterns must be in terms of the relative spatial arrangements of points. Inferences about the structure of the pattern can be made, if it is produced by a point process of a known type. However, in general, such *a priori* knowledge or model of the point process is unavailable. Any inferences about the structure must be data-driven.

Global features of point patterns (such as the gestalt law of good continuation) are important in the perception of their structure. However, taking global features into account necessitates a top-down approach, and hence the availability of a model. Without a model, a bottom-up approach is required. Structural descriptions must be built using the relative positions of neighboring points. Therefore, a sound notion of neighborhood is necessary.

Past work on defining the neighborhood of a point has concentrated on identifying its *neighbors*. The analysis of dot patterns incorporates joint properties of neighbors. In this correspondence we examine the use of *neighborhood* of a point, which associates with a point not only other points as its neighbors, but also a part of the Euclidean plane around it. The analysis makes use of the shape features of the neighborhood, in addition to the neighbors themselves. In particular, we examine the use of (the region enclosed by) the Voronoi polygon as a point's neighborhood. It should be pointed out that the general idea of making use of geometric properties of the Voronoi polygons has been mentioned before (see Section III), although it has not been explored for any specific application.

In Section II we review briefly the definitions of the neighbors of a point used in the past. Section III discusses the Voronoi neighborhood approach and its salient features that we believe make it more promising than the other methods. Section IV discusses applications of the Voronoi neighborhood to dot pattern segmentation, pattern matching, and perceptual border extraction. Section V presents concluding remarks.

II. Approaches to Defining the Neighborhood of a Point—A Review

The past work on defining the neighborhood of a point has been concerned with the following question: given an arbitrary point in a dot pattern, which other points should be treated as its neighbors? For identifying clusters of dots, Koontz and Fukunaga [17] do not allow two points to be assigned to two different clusters if they are closer than a distance R apart. The value of R is globally specified; it defines the extent of the neighborhood of a point anywhere in the pattern. Patrick and Shen [22] start a cluster with a single point and then iteratively consider for inclusion every noncluster point within a given distance of any cluster point. Sneath [31] uses the same neighborhood criterion to detect points along a curve. Koontz and Fukunaga [17] also mention the possibility of assigning a point and its k-nearest neighbors to the same cluster. However, they argue that such a criterion may be inappropriate if the neighbors are unsymmetric (two points P_1 and P_2 are unsymmetric neighbors if P_1 is a neighbor of P_2, but P_2 is not a neighbor of P_1). In a relaxation formulation [26] of the clustering problem, Zucker and Hummel [43] use the k-nearest neighbors criterion with success. They consider three different types of dots: cluster interior points, cluster edge points, and noise points. Koontz and Fukunaga's objection to the k-nearest neighbors criterion may not be serious for this formulation since neighbors are not required to be of the same type. Jarvis and Patrick [12] also use a k-nearest neighbors approach. Two points are called neighbor if each belongs to the other's set of k-nearest neighbors and the two sets share at least k_t points. This group similarity concept of neighborhood favors clusters with a globular structure when k_t is large. In their experiments they only use point pair similarity ($k_t = 1$). Velasco [37], [38] also uses a k-nearest neighbors approach for cluster analysis.

O'Callaghan [19] presents examples showing when the k-nearest neighbors method fails. He points out that problems arise when a dot pattern contains clusters consisting of fewer than k points or when a sparse cluster or an isolated point is close to a dense cluster. He suggests [19], [20] the following alternative method to determine the neighborhood of a point. Let p_{i1} be the nearest neighbor of a given point P_i. Any point P_k is a neighbor of P_i if: 1) $d(P_i, P_k)/d(P_i, P_{i1})$ is no greater than a specified threshold T_r, and 2) P_k is not "behind" any neighbor P_j of P_i, i.e., the angle $P_iP_jP_k$ differs from 180° by more than a threshold T_θ. O'Callaghan [21] uses this definition of neighbors to compute the perceptual boundaries of dot patterns.

All the definitions of neighborhood described so far define a point's neighbors in terms of certain parameters. Also, the performance of these methods depends upon the values of the parameters, and hence there may always be a need to tune the parameters for best performance on a given pattern. The following approaches make use of only the locations of the points.

Zahn [40] uses the minimal spanning tree (MST) of a point set to define neighborhoods. An MST spans all the points such that the sum of the Euclidean edge lengths is less than that for any other spanning tree. Interaction is considered between only those points connected by the edges of the tree. Thus, although the choice of the neighbors of a point is from its nearest neighbors, the exact number of neighbors that a point has depends upon the global optimum connectivity of the points. Johnson [13] and Gower and Ross [9] discuss the use of an MST-like approach for hierarchical clustering. For a good review of graph theoretic concepts in clustering, see Hubert [11].

Toussaint [34] defines the relative neighborhood graph of a set of planar points. Let $d(P_1, P_2)$ denote the Euclidean distance between any two points P_1 and P_2. Points P_i and P_j are connected in the relative neighborhood graph if $d(P_i, P_j) \leqslant \max\,[d(P_i, P_k), d(P_j, P_k)]$ for all points $P_k \neq P_i, P_j$. Thus, P_i and P_j are connected whenever there is no point within their lune. A related criterion is used to define Gabriel graph [35]. Points P_i and P_j are connected in the Gabriel graph if there is no point within the circle whose diameter is the line P_iP_j. That is, P_i and P_j are connected if $d^2(P_i, P_j) \geqslant d^2(P_i, P_k) + d^2(P_k, P_j)$, for all points $P_k \neq P_i, P_j$. Both these criteria for connectivity between points are not global as for MST, and therefore allow more edges.

Neighbors of a point have also been defined by constructing the Voronoi tessellation of the point pattern. This definition will be explained in the next section, after we describe the Voronoi tessellation of the plane.

III. Voronoi Neighborhoods

Our motivation for this correspondence came from the observation that humans find it easy to identify the neighborhood and neighbors of a point in a wide variety of dot patterns, including those having varying density. This would suggest the existence of a general parameter-free concept of neighborhood.

In this section we present a definition of the neighborhood of a point based upon the Voronoi tessellation defined by the points. First, we review the definition of the Voronoi tessellation.

A. Voronoi Tessellation

Suppose that we are given a set S of three or more points in the Euclidean plane. Assume that these points are not all collinear and that no four points are cocircular. Consider an arbitrary pair of points P and Q. The bisector of the line joining P and Q is the locus of points equidistant from both P and Q and divides the plane into two halves. The half plane $H_P^Q(H_Q^P)$ is the locus of points closer to $P(Q)$ than to $Q(P)$. For any given point P, a set of such half planes is obtained for various choices of Q. The intersection $\cap_{Q \in S, Q \neq P} H_P^Q$ defines a polygonal region consisting of points closer to P than to any other point. Such a region is called the Voronoi [39] (Dirichlat, Wigner-Seitz, Theissen [32] or "S" [23]) polygon associated with the point.

Voronoi polygons may be viewed as the result of a growth process. Assume that all the points (nuclei) simultaneously start a uniform outward growth along a circular frontier. At some later time they reach a tightly packed state in which the number of points of contact between the circle centered at a given point P and other circles is determined by the configuration of points in the vicinity of P. The growth stops at the points of contact. As remaining points on the circles continue to expand, the points of contact become midpoints of growing straight line segments along which growth frontiers meet and freeze. Since all circles expand at the same rate, the first point of contact between two circles must occur at the midpoint between their nuclei. Likewise, the growing line segments must be equidistant from the two nuclei. These points are on the common edge of two developing Voronoi polygons. An edge continues elongating until it encounters the border of a third expanding circle. The point of contact of this edge and the border of the third circle must be equidistant from the growth centers of all three circles. It is therefore, the circumcenter of the triangle defined by the three nuclei. Eventually, only the circles whose nuclei are on the convex hull of S are still expanding. Each of the remaining nuclei is contained in exactly one convex polygon. The set of complete polygons is called the Voronoi diagram of S [25], [28]. The Voronoi diagram, together with the incomplete polygons on the convex hull define a Voronoi tessellation of the entire plane. The collection of edges obtained by joining each point with its neighbors is the dual of the Voronoi tessellation and is called the Delaunay tessellation.

An example of Voronoi tessellation is shown in Fig. 1. $O(N\log N)$ algorithms to construct the Voronoi tessellation of N points are given by Shamos and Hoey [28] and Lee [18].

Voronoi tessellation has also been used to define neighbors in a point pattern [3], [4], [29], [33], [36]. As discussed previously, the points whose polygons share edges with the polygon containing a given point P are called P's Voronoi neighbors. Besag [3], [4] mentions the possibility of using the Voronoi neighbors in the formulation of conditional probability models of interaction among random variables associated with irregularly distributed coplanar sites. For classifying point data, Toussaint and Poulsen [36] use the Voronoi neighbors to extract a reduced set of data points, which when used with a nearest neighbor decision rule for classification, implements the original decision boundary. Tobler [33] also suggests the use of the Voronoi neighbors in processing geographical data available at irregular locations. In the next section we will discuss application of geometrical characteristics of the Voronoi neighborhood to processing dot patterns.

B. Neighborhood of a Point

We will consider as the neighborhood of a point P (the region enclosed by) the Voronoi polygon containing P. Considering the way a Voronoi polygon is constructed, this is an intuitively appealing approach. Geometric features of the

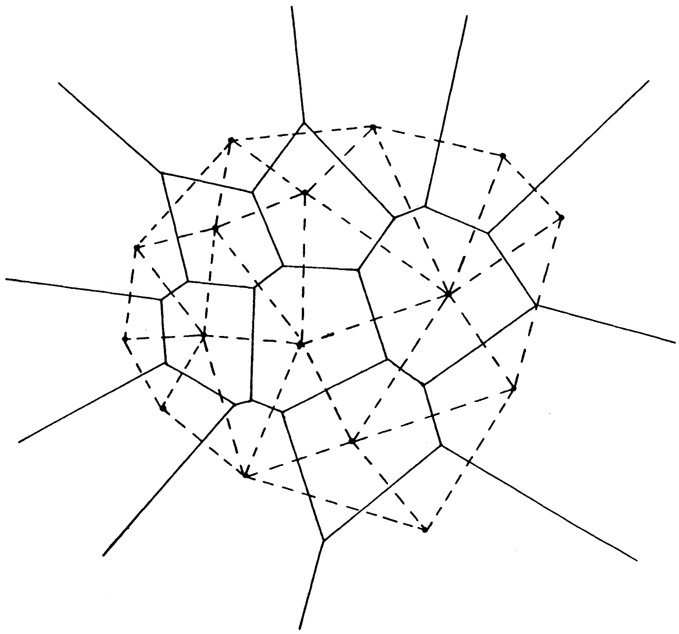


Fig. 1. Voronoi tessellation defined by a given set of points. Dotted lines show the corresponding Delaunay tessellation.

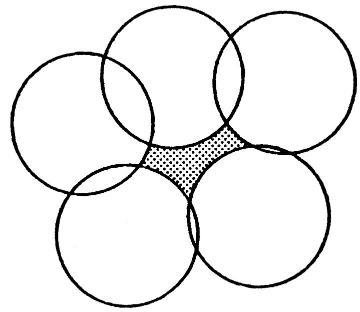


Fig. 2. Fixed radius neighborhoods of a given set of points. The hatched region does not belong to any of the neighborhoods.

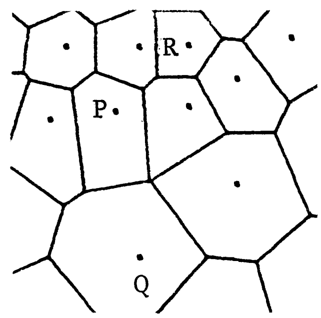


Fig. 3. Voronoi neighbors of a point may be farther from it than non-neighbors. Here Q is a neighbor of P, R is not, and $d(P, Q) > d(P, R)$.

Voronoi polygons, in addition to the Voronoi neighbor relation between points, will be used to analyze dot patterns. First, we will compare the Voronoi neighborhood and the Voronoi neighbors with the other definitions.

A crucial difference between the Voronoi and most traditional definitions of neighbors is that the former makes use of the Euclidean plane. The Voronoi construction actually goes through the intermediate stage of assigning a unique Euclidean neighborhood to each point. The assignment of neighbors follows quite naturally, being based upon the adjacency characteristics of Euclidean neighborhoods. Among the approaches mentioned in the last section, only the fixed radius method involves a Euclidean neighborhood (although the objective is only to identify the neighbors). The rest determine the neighbors of a point directly in terms of interpoint distances. The fixed radius approach is not satisfactory for the following reasons.

1) It is insensitive to variations in the local densities of points. In dense dot areas a point may have a large number of neighbors, whereas it may not have even a single neighbor in sparse regions. The number of neighbors is thus determined by scale, and not by the structure of the pattern alone. Voronoi polygons, on the other hand, reflect local structure.

2) The fixed radius neighborhood is by definition stationary. It does not respond to trends in point density gradients or any other direction sensitive structure. Voronoi polygons assume shapes that reflect the properties of local spatial point distributions.

3) The set of circles of radius R centered at a given set of points will, in general, have regions of overlap. There will also be regions surrounded by various neighborhoods, but not included in any (Fig. 2). Intuitively, such ring-like neighborhoods surrounding "no-man's" land are less than satisfactory. (In Fig. 2 we would expect the central region to be divided among the neighborhoods of the surrounding points.) The Voronoi tessellation assigns each region of the plane to the neighborhood of one and only one point.

A comparison between the Voronoi approach and the k-nearest neighbors approach follows next.

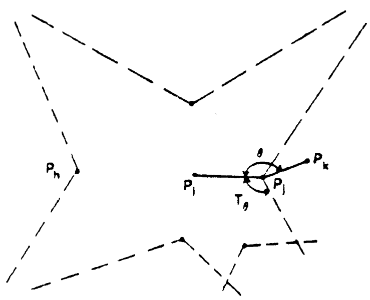
1) The number of Voronoi neighbors of a point varies from point to point. This is not true for the k-nearest neighbor approach.

2) Voronoi neighbors are symmetric by definition. The k-nearest neighbors of a point may or may not be symmetric, depending upon the approach used.

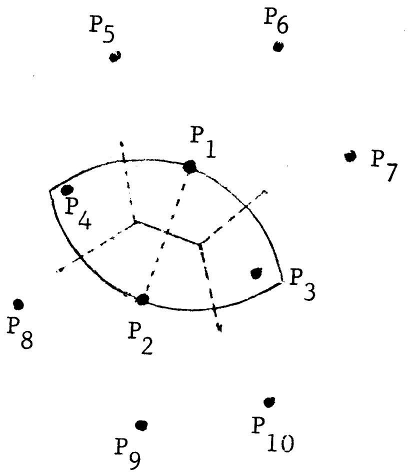
3) The Voronoi neighbors of a point are not necessarily its *nearest* neighbors. In fact, some of a point's neighbors may be farther from it than some other points which are not its Voronoi neighbors (Fig. 3). The Voronoi neighbors of a point must "surround" it. Hence, distant points may be accepted as neighbors on the sparsely populated side of a point whereas relatively close points may not be accepted as neighbors on the dense side if they occur "behind" other closer points (Fig. 3).

O'Callaghan's [19] approach is an attempt to impart part of this last attribute to the neighborhood of a point. The parameter T_θ determines which points should be excluded from a given point's neighbors, regardless of how close they are to it. An appropriate value of T_θ would depend upon the point configuration under consideration; using a fixed value as suggested in [19] may not be optimal for all the points in an arbitrary pattern. O'Callaghan [19] suggests that T_θ be regarded as a constant for many kinds of dot patterns. Fig. 4(a) shows the shape of a typical neighborhood of a point (from [19]). Intuitively, one would expect the neighborhood to be spatially compact. The star shaped region is a result of the fixed, configuration-independent parameter values. The Voronoi neighborhood and be viewed as resulting from a context sensitive choice of both T_r and T_θ for each neighbor of a point.

The points linked in Zahn's mst [40] may be called "neighbors." The Delaunay triangulation provides an analogous linking for Voronoi nuclei. The concept of neighbor in Zahn's case relies on global considerations, since the mst of N points contains only (N-1) edges. The Delaunay triangulation, on the other hand, has many more edges, the total number being the same as the total number of pairs of neighbors. The total number of edges in the Delaunay triangulation is at most $3N$-6. The set of edges in the relative neighborhood graph is a superset of the set of edges in the MST and a subset of the edges in the Delaunay triangulation [34].



(a)



(b)

Fig. 4. (a) A typical neighborhood of a point P_i, as defined by O'Callaghan [19]. (b) P_1 and P_2 are not connected in the relative neighborhood graph. However, they are Voronoi neighbors.

For the Voronoi approach, two points which have a "clear" zone between them may be labeled as neighbors, irrespective of their separation and cluster-memberships. On the other hand, the fixed radius and k-nearest neighbor approaches label a pair of points as neighbors if they are "sufficiently close." Neighbors are generally required to belong to the same cluster. The relative neighborhood graph may not recognize intuitively obvious neighbors of a point. For example, in Fig. 4(b) point P_1 has points P_2 to P_6 as its neighbors. Similarly, the intuitive neighbor set of P_2 contains points P_1, P_3, P_4, P_8, P_9, and P_{10}. Thus, P_1 and P_2 appear to be neighbors. However, the relative neighborhood graph of points P_1 to P_{10} does not connect P_1 and P_2, since the lune formed by P_1 and P_2 is not empty. But P_1 and P_2 are still Voronoi neighbors, as illustrated by the partial Voronoi tessellation (dotted lines) shown in Fig. 4(b).

The local environment of a point in a given pattern is reflected in the geometrical characteristics of its Voronoi polygon. This presents a convenient way to compare the local environments of different points. Since the perceived structure in a dot pattern results from the relative spatial arrangement of points, the geometric properties of Voronoi polygons may be useful for describing and detecting structure in dot patterns. In addition, such an approach lends a fully two-dimensional character to the problem in that the dot pattern is converted into a planar image or a mosaic. As a result, many common low-level computer vision techniques become relevant.

IV. Applications

Since a Euclidean neighborhood makes possible a continuous image-like treatment of a dot pattern, it permits the use of general image processing techniques. In addition, the performance of the available methods of dot pattern processing that use alternate definitions of neighbors may be examined using the Voronoi neighbors. If the Voronoi definition is more appropriate than the others, these techniques should exhibit improved performance.

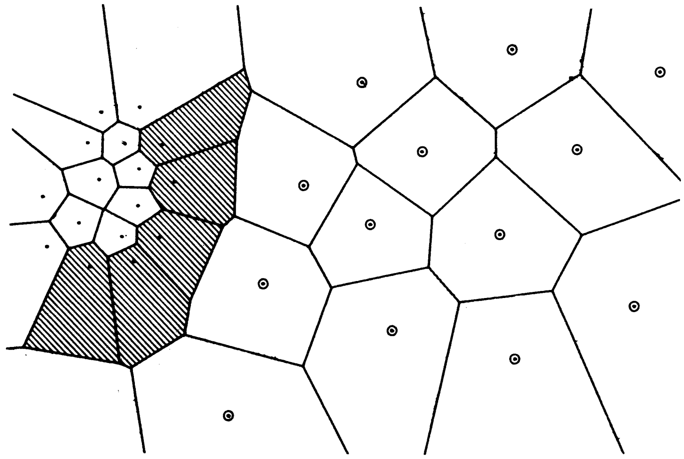


Fig. 5. Two homogeneous clusters nearby. The interior cells of the two clusters have different geometries. Some of the border cells are incomplete, while others (shown hatched) are complete but elongated and have high eccentricity values.

In the rest of this section we discuss three different application areas. In order to evaluate the expected relative performance of the methods, approaches to the problems using alternative neighborhood definitions are also outlined. We will show that solutions using the Voronoi approach appear to have advantages in computational complexity, quality, and conceptual simplicity.

A. Segmentation by Dot Cluster Analysis

Finding clusters in a dot pattern usually means finding a partition of the given set of points into subsets whose in-class members are "similar" in some sense and whose cross-class members are "dissimilar" in a corresponding sense [12]. A formalization of the notion of "similarity" is instrumental in determining the power of any such partitioning algorithm. Traditionally, externally specified similarity measures between pairs of objects in an n-dimensional space have been used to make decisions about grouping the objects. Various criteria using pairwise similarity measures have been developed to measure the goodness of a grouping [1], [7], [8], [13], [17]. However, for planar dot clusters, the desired similarity measure must compare neighborhoods of points. Several such methods using the traditional notions of neighborhood have been reported [9], [11], [12], [40], [43].

The Voronoi neighborhoods of the points which reside within the interior of a homogeneous cluster will have similar shapes and sizes. For different clusters, these interior polygons may differ in their geometrical properties (Fig. 5). The border cells of a cluster will be open if there is no other cluster to bound them. The cells of the border points of a cluster that have neighbors in a nearby cluster will differ from interior cells. For example, they may be elongated if the distance between cross cluster neighbors is larger than within cluster neighbors, or the nucleus of the cell may be located well off its center (Fig. 5). For a cluster having orientation sensitive density, the shapes of the resulting Voronoi polygons will exhibit a corresponding direction sensitivity [Fig. 6(a)]. The Voronoi cells of a cluster whose spatial point density varies will have decreasing area along the direction of increasing density [Fig. 6(b)]. Clearly, a globular cluster will have a larger number of interior cells than will a more elongated cluster (Fig. 7). Detection of a curved cluster and of a cluster with a

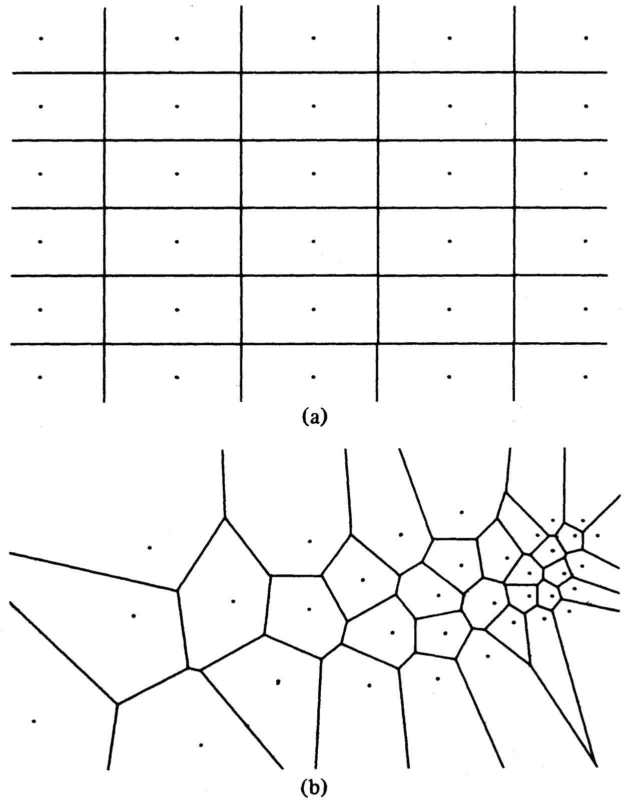


Fig. 6. (a) A simple example of a cluster having direction sensitive point density. Points are more closely packed in the vertical direction. The shape of the cells reflect this property. (b) A cluster with the point density increasing towards the right. The cells get smaller.

neck (Fig. 8) may require the use of the joint characteristics of neighboring cells, e.g., distance between the neighbors, etc. Characteristics of the Voronoi polygons for some additional types of clusters [12], [19], [40], [43] may also be easily inferred in terms of the cluster structure.

A solution to the clustering problem then lies partly in the use of a meaningful set of geometrical features of the individual Voronoi cells, and also of features relating to the joint characteristics of neighbors. Several examples of neighborhood features that may be used to label a point are as follows:

1) area and perimeter,
2) completeness,
3) elongatedness or compactness,
4) direction of principal axis,
5) variance of side lengths,
6) eccentricity.

Examples of some joint properties that may be useful are: distance between neighbors (e.g., for clusters with necks), gradients of cell features (e.g., area gradient for clusters with varying density), and status of neighbors with respect to completeness of cells (e.g., for identifying necks in clusters). We would like to point out that Sibson [29] has suggested the use of areas and nucleus-vertex distances of the Voronoi polygons and the distances between neighboring points as statistics of a point pattern.

If we use n different properties, then the points acquire n-dimensional vector attributes that may be of visual significance. A similarity measure based on the Euclidean distances in n-dimensions may be used. Alternatively, the attributes can be ordered according to their significance and the decision about merging any two points made by a sequential consideration of the individual attribute differences. Only neighbors need be tested for similarity.

A cluster may be grown by starting from a point and itera-

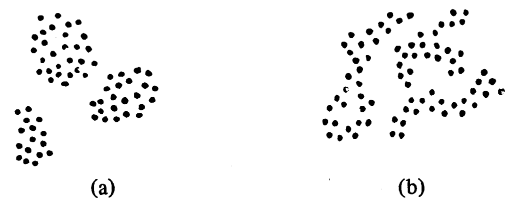


Fig. 7. (a) Globular and (b) nonglobular clusters [12]. A larger fraction of the points in (b) will give rise to border cells.

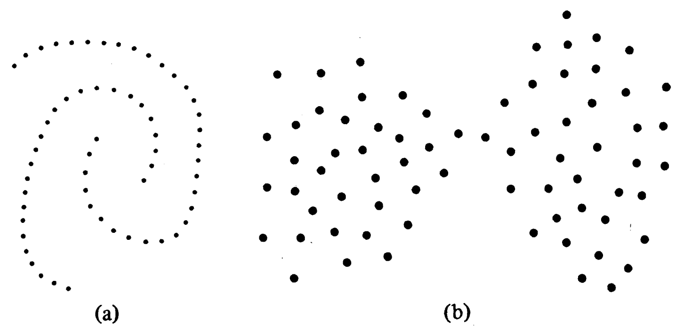


Fig. 8. (a) Curve-like cluster and (b) cluster with neck [40]. A greater use of joint cell characteristics may be desirable in such cases.

tively merging similar neighbors into the cluster. Given the regional attributes assigned to each point, this approach is similar to region growing [27], [42] in the Voronoi tessellation. Alternatively, decisions about the cluster memberships of points may be taken in parallel. Zucker and Hummel [43] describe a relaxation formulation of such an approach. However, they use the k-nearest neighbors of a point to define the relaxation process. Using the more natural Voronoi approach should improve the performance of their methods.

The Delaunay triangulation of a dot pattern resembles Zahn's MST. Actually, the edges of the MST form a subset of the edges of the Delaunay triangulation. Thus, a treatment analogous to Zahn's may be developed for Delaunay triangulation, which on the average has more links than MST. Other graph theoretic clustering methods may also be considered [9], [11], [13]. The actual complexity of a clustering operation depends upon the specific algorithm used. However, since a point's computations involve only its neighbors, any such algorithm should be linear in N for common patterns (given the triangulation). Traditional clustering criteria and algorithms [10] suitable for planar processing may be reexamined using the Voronoi approach.

B. Pattern Matching

When matching pictures of the same scene, the conventional methods based upon correlation between gray levels are impractical when the pictures are taken by different sensors, by the same sensor at different times, or when there are geometrical distortions present [15]. The dot patterns corresponding to the local feature positions of objects (e.g., points of locally maximum radar reflectivity) may prove to be relatively insensitive to such variations. In addition to the differences pointed out above, the objects in the scene may be at different orientation or scale, and may be noisy. Some examples of the noise include missing dots (e.g., due to radar shadowing) and dots with perturbed positions.

The matching problem can be stated as follows: given two dot patterns, we want to know if one is a rotated, translated, and scaled version of the other. We also want to allow for deletions of points and some random perturbations in their relative locations. We desire solutions that are computationally efficient.

Kahl [14], Kahl *et al.* [15], and Ranade and Rosenfeld [24] consider matching with respect to translation, allowing perturbation of a point by at most a given threshold t. They attempt all possible translations that map a pair of points in one pattern onto a pair in the other, within the given tolerance t. The translation resulting in the best overall match is sought. Simon *et al.* [30] and Lavine *et al.* [16] list the interpoint distances for all pairs of points in each pattern and compare the sorted lists of distances to detect potential matches between pairs across patterns. Zahn [41] compares the minimal spanning trees of the patterns in order to determine their degree of match. He attempts a match between points in the two patterns with respect to the degree of the minimal spanning tree at the points, angles formed by the lines joining the points to their neighbors, etc. Good matches are used to establish correspondences between points in the two patterns. Bernard and Thompson [2] discuss a relaxation-based approach to matching for disparity analysis of images.

Using the Voronoi neighborhoods, points may be sorted according to the geometrical feature values of their cells. Scale invariance may be attained by normalizing, say, the total area of the cells in each of the lists to a common value. The sorted lists of cell features may be compared to detect potentially matching pairs of points. Rotation invariance may be attained by retaining only the neighbor pointers between the indices of the points and ignoring their absolute coordinates. The major effort here is in sorting ($O(\log N)$ for N points). Comparison of the sorted lists takes only $O(N)$ time. When dot locations are perturbed, the resulting cell characteristics may change. This may be taken into account by allowing a match between a pair of cells from the two patterns such that their feature values do not differ by more than a certain threshold. This process may suggest multiple mappings of points in one pattern onto those in the other. The best map is that which best preserves the within-pattern spatial adjacency characteristics of the matching pairs of cells, and the distances between their corresponding points. The derivation of the best match may be carried out by a relaxation labeling process [26]. If a point in one pattern has its corresponding point in the other pattern missing, it will either not have a match or it will have a poor match when the relaxation labeling process converges.

A more efficient matching procedure may be obtained by comparing ordered lists of boundary cells [5]. This should have the effect of aligning the borders of the two patterns, thereby suggesting the potentially matching point pairs in the interiors of the patterns. Sorting B boundary points will require $O(B \log B)$ time and the comparison can be carried out in $O(B)$ time.

C. Perceptual Boundary Extraction

A "perceptual boundary" of a dot pattern is a boundary seen because of the given relative locations of dots, without any semantic or cognitive interpretation [21]. Some examples of such boundaries are reproduced from [21] in Fig. 9. As in the case of cluster identification, the global view of the cluster may influence local decisions regarding boundary characteristics. O'Callaghan [21] uses his neighborhood criteria to determine which points lie along the perceptual boundary of a dot pattern. He gives special consideration to necks in boundaries since necks may cause certain points or links to divide a boundary curved into two shorter curves (Fig. 9).

The cells of a Voronoi tessellation which correspond to border points are identifiable as explained in Section III-A. One must trace the border points from neighbor to neighbor to obtain a boundary. Sometimes the Voronoi neighborhood of a point of concavity may not differ much from an interior cell. O'Callaghan makes the decision about including or skipping such concavities by comparing neighbor-angles against certain thresholds. With the Voronoi approach, the following rule may be used to determine when to add a point P_j to the boundary line joining P_i to P_k: if the line joining the point P_i to P_k intersects the edges of P_j's cell [Fig. 10(a)], follow the boundary from P_i to P_k through P_j; otherwise skip the concavity point P_j [Fig. 10(b)]. Equivalently, all pairs of adjacent border points which are Gabriel neighbors are included in the perceptual border. The resulting border is the same as the Gabriel hull proposed by Toussaint [35] as the shape hull of the point pattern. The use of several other thresholds in O'Callaghan's [21] method may also be avoided as a result of the Voronoi treatment. Given the Voronoi polygons, which are labeled "border" or "interior" (Section IV-A), the algorithm to trace the boundary is linear in the number of border points.

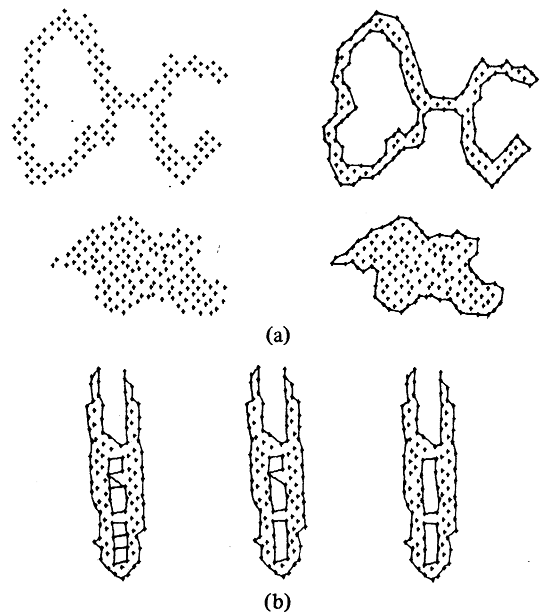


Fig. 9. (a) Clusters and their perceptual boundaries [21]. (b) Successive removal of extraneous boundary links using thresholds on local angles, etc. [21]. The links denote the incorrectly marked neighbor-pairs.

(a) (b)

Fig. 10. Following cluster boundary from A to B: (a) P_j is included on the boundary if the line P_iP_k intersects P_j's neighborhood. (b) P_j is skipped if the line P_iP_k does not intersect P_j's neighborhood.

The Gabriel criterion for a point's inclusion in the border is intuitively appealing. It would be interesting to see if data obtained from experiments with human subjects support the Gabriel criterion or suggest a generalized Gabriel criterion that uses noncircular regions.

Fairfield [6] treats a related but different problem, that of contoured shape generation in point patterns. He suggests a method using the Voronoi tessellation to identify a hierarchy of nested contours as will be seen by a human observer around her center of attention. Fairfield's algorithm obtains the various contours by following Voronoi edges outward from a vertex which is closest to the center of attention.

D. Extensions

The above applications of the Voronoi tessellation concern planar dot patterns. However, the concept of the Voronoi neighborhood can be generalized to higher dimensional spaces. For example, the neighborhood of a point in a three-dimensional dot pattern is a polyhedron. Patterns that represent images by feature vectors often are in high dimensional space. Moreover, clustering procedures are commonly used to process such patterns. The availability of the Voronoi tessellation should greatly influence such processing. Although no efficient algorithms for generating the Voronoi tessellation in three or higher dimensional space are known, it is not clear that the complexity involved is necessarily higher than that of other n-dimensional clustering algorithms.

The notion of the Voronoi neighborhood may also be extended to define neighborhoods of nonpoint objects. This may be done in several ways. For example, objects may be represented by single points; the Voronoi tessellation may be constructed for the specific object shapes involved, etc. This presents a quantitative method of assigning spatial relations to objects. Among other applications, such spatial relations can be used in texture classification based upon second and higher order statistics of the primitives.

V. Concluding Remarks

A sound notion of the neighborhood of a point is crucial for analyzing dot patterns. Past work in this direction has concentrated on identifying pairs of points that are *neighbors.* Many definitions, including the Voronoi definition, of a point's neighbors have been used. In this correspondence we have considered the use of the region enclosed by a point's Voronoi polygon as its *neighborhood.* We have compared the Voronoi approach with the others and argued that the Voronoi neighborhood possesses intuitively appealing characteristics expected from a point's neighborhood. We have explored the use of the geometrical properties of the neighborhood to characterize a point's "local environment," instead of using it just for defining the point's neighbors. Although we have considered only planar dot patterns, the ideas extend to higher dimensions. Thus, for example, the neighborhood of a point in a three-dimensional point pattern will be a polyhedron. Unfortunately, no efficient Voronoi tessellation algorithms in higher than two dimensions are known.

We have outlined procedures for segmentation, matching, and border extraction that use the neighborhood characteristics as features. Although a general idea of the complexity of the algorithms can be obtained from our discussion, an accurate judgment of the complexity and performance of the methods can only be made by working with real data. Experiments involving several aspects of point pattern processing, including the applications discussed in this correspondence, are in progress and the results will be reported in the near future.

Acknowledgment

The author would like to thank the IEEE reviewer and B. Schachter for useful comments on an earlier version of the paper.

References

[1] G. H. Ball, "Data analysis in the social sciences: What about the details?," in *Proc. 1965 Fall Joint Comput. Conf.*, AFIPS, 1965, pp. 533–559.

[2] S. T. Bernard and W. B. Thompson, "Disparity analysis of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, pp. 333–340, July 1980.

[3] J. Besag, "Spatial interaction and the statistical analysis of lattice patterns," *J. Royal Stat. Soc.*, Series B, vol. 36, no. 2, pp. 192–236, 1974.

[4] —, "Statistical analysis of non-lattice data," *Statistician*, vol. 24, no. 3, pp. 179–195.

[5] L. S. Davis, "Shape matching using relaxation techniques," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-1, pp. 60–72, 1979.

[6] J. Fairfield, "Contoured shape generation: Forms that people see in dot patterns," *in Proc. IEEE Conf. Syst., Man, Cybern.*, 1979, pp. 60–64.

[7] K. Fukunaga and W.L.G. Koontz, "A criterion and algorithm for grouping data," *IEEE Trans. Comput.*, vol. C-19, pp. 917–923, Oct. 1970.

[8] I. Gitman, "A parameter free clustering model," *Pattern Recog.*, vol. 4, pp. 307–315, 1972.

[9] J. C. Gower and G.J.S. Ross, "Minimum spanning trees and single linkage cluster analysis," *Appl. Statist.*, vol. 18, no. 1, pp. 54–64, 1969.

[10] J. A. Hartigan, *Clustering Algorithms.* New York: Wiley, 1975.

[11] L. J. Hubert, "Some applications of graph theory to clustering," *Psychometrika*, vol. 39, pp. 283–309, 1974.

[12] R. A. Jarvis and E. A. Patrick, "Clustering using a similarity measure based on shared near neighbors," *IEEE Trans. Comput.*, vol. C-22, pp. 1025–1034, Nov. 1973.

[13] S. C. Johnson, "Hierarchical clustering schemes," *Psychometricka*, vol. 32, pp. 241–252, Sept. 1967.

[14] D. J. Kahl, "Sketch matching," M. S. thesis, Dep. Comput. Sci., Univ. Maryland, College Park, 1978.

[15] D. J. Kahl, A. Rosenfeld, and A. Danker, "Some experiments in point pattern matching," Univ. Maryland, Comput. Sci. Tech. Rep. TR-690, Sept. 1978.

[16] D. Lavine, B. A. Lambird, and L. N. Kanal, "Recognition of spatial point patterns," in *Proc. IEEE Conf. Pattern Recog. Image Processing*, Dallas, TX, Aug. 3–5, 1981, pp. 49–53.

[17] W.L.G. Koontz and K. Fukunaga, "A nonparametric valley-seeking technique for cluster analysis," *IEEE Trans. Comput.*, vol. C-21, pp. 171–178, 1972.

[18] D. T. Lee, "Proximity and reachability in the plane," Ph.D. dissertation, Coord. Sci. Lab., Univ. Illinois, Urbana, Rep. ACT-12, 1978.

[19] J. F. O'Callaghan, "An alternative definition for neighborhood of a point," *IEEE Trans. Comput.*, vol. C-24, pp. 1121–1125, 1975.

[20] —, "Human perception of homogeneous dot patterns," *Perception*, vol. 3, pp. 33–45, 1974.

[21] —, "Computing the perceptual boundaries of dot patterns," *Comput. Graphics Image Processing*, vol. 3, pp. 141–162, 1974.

[22] E. A. Patrick and L. Shen, "Interactive use of problem knowledge for clustering and decision making," *IEEE Trans. Comput.*, vol. C-20, pp. 216–222, Feb. 1971.

[23] E. C. Pielou, *Mathematical Ecology.* New York: Wiley, 1977.

[24] S. Ranade and A. Rosenfeld, "Point pattern matching by relaxation," Univ. Maryland, Comput. Sci. Tech. Rep. TR-702, Oct. 1978.

[25] C. A. Rogers, *Packing and Covering* (Math. Tract No. 54). London and New York: Cambridge Univ. Press, 1964.

[26] A. Rosenfeld, R. Hummel, and S. W. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, pp. 420–433, 1976.

[27] A. Rosenfeld and A. Kak, *Digital Picture Processing.* New York: Academic, 1976.

[28] M. I. Shamos and D. Hoey, "Closest-point problems," in *Proc. 16th Annu. Symp. Foundations of Comput. Sci.*, Oct. 1975, pp. 131–162.

[29] R. Sibson, "The Dirichlet tessellation as an aid in data analysis," *Scandinavian J. Statist.*, vol. 7, pp. 14–20, 1980.

[30] J. Simon, A. Checroun, and C. Roche, "A method of comparing two patterns independent of possible transformations and small distortions," *Pattern Recog.*, vol. 4, pp. 73–81, 1972.

[31] P.H.A. Sneath, "A method for curve seeking from scattered points," *Comput. J.*, vol. 9, pp. 383–391, 1966.

[32] A. H. Thiessen, "Precipitation averages for large areas," *Mon. Weather Rev.*, vol. 39, pp. 1082–1084, 1911.

[33] W. R. Tobler, "Linear operators applied to areal data," in *Display and Analysis of Spatial Data*, J. C. Davis and M. J. McCullagh, Eds. New York: Wiley, 1975.

[34] G. T. Toussaint, "The relative neighborhood graph of a finite planar set," *Pattern Recog.*, vol. 12, pp. 261–268, Aug. 1980.

[35] —, "Pattern recognition and geometrical complexity," in *Proc.*

5th Int. Conf. Pattern Recog., Miami Beach, FL, Dec. 1980, pp. 1324–1347.
[36] G. T. Toussaint and R. S. Poulsen, "Some new algorithms and software implementation methods for pattern recognition research," presented at COMPSAC, Chicago, IL, Nov. 1979.
[37] F.R.D. Velasco, "A method for the analysis of Gaussian-like clusters," Univ. Maryland, Comput. Sci. Tech. Rep. TR-718, Jan. 1979.
[38] F.R.D. Velasco and A. Rosenfeld, "Some methods for the analysis of sharply bounded clusters," Univ. Maryland, Comput. Sci. Tech. Rep. TR-740, Mar. 1979.
[39] G. Voronoi, "Nouvelles applications des parametres continus a la theorie des formes quadratiques. Deuxieme memoire: Recherches sur les parallelloedres primitifs," *J. Reine Angew. Math.*, vol. 134, pp. 198–287, 1908.
[40] C. T. Zahn, "Graph-theoretical methods for detecting and describing Gestalt clusters," *IEEE Trans. Comput.*, vol. C-20, pp. 68–86, 1971.
[41] —, "An algorithm for noisy template matching," *Proc. IFIP*, 1974, pp. 727–732.
[42] S. W. Zucker, "Region growing: Childhood and adolescence," *Comput. Graphics Image Processing*, vol. 5, pp. 382–399, 1976.
[43] S. W. Zucker and R. A. Hummel, "Toward a low-level description of dot clusters: Labelling edge, interior and noise points," *Comput. Graphics Image Processing*, vol. 9, pp. 213–233, 1979.

Repeated Hypothesis Testing on a Growing Data Set

G. V. TRUNK AND J. O. COLEMAN

Abstract—**In many problems, especially when data are collected over a long period of time, hypothesis testing is done repeatedly as new data arrive. It is shown both for one particular problem and for a more general class of problems that if testing is performed each time on the total amount of data accumulated, the true simple null hypothesis will be rejected at least once as the number of tests approaches infinity. Furthermore, it is conjectured that the conclusion holds for most problems of interest in which the null hypothesis is simple.**

Index Terms—**Decision theory, detection theory, growing data set, hypothesis testing.**

INTRODUCTION

In many problems, especially when data are collected over a long period of time, hypothesis testing is done repeatedly as new data arrive. If each test is performed only on data that have arrived since the previous test, all of the tests are independent. Consequently, if binary hypothesis tests with a constant false alarm rate were always performed, the probability of getting at least one false alarm would approach unity as the number of tests approached infinity. People who perform such testing are aware of this behavior and take it into account when setting the false alarm rate and interpreting the results.

There is, however, a related type of testing in which the certainty of an eventual false alarm is not obvious at all. In some systems in which data are accumulated slowly and a need for obtaining maximum performance is perceived, testing is performed each time on the *total* amount of data accumulated, with the total amount increasing with each test. Rather than successive tests being independent as in the first type of testing described, the latter tests grow more and more *dependent* as data are acquired (i.e., the probability of making the same decision when using n and $n + 1$ samples approaches one as n approaches infinity). The purposes of this correspondence are 1) to prove both for one particular problem of this type and for a more general class of problems of this type that the probability of an eventual false alarm is still unity and 2) to suggest that a similar proof could probably be applied to almost any problem of this type as long as the null hypothesis is simple.

Manuscript received February 2, 1981; revised November 18, 1981.

The authors are with the Radar Analysis Branch, Radar Division, Naval Research Laboratory, Washington, DC 20375.

PROBLEM

Given a sequence of independent samples $\{x_i\}$ where the x_i are Gaussian distributed with mean μ and unit variance, consider the following binary hypothesis test:

$$H_0: \ \mu = 0$$
$$H_1: \ \mu \neq 0.$$

While the general proof is applicable to this simple problem, a different proof, which provides insight on why the probability of an eventual false alarm is unity, will be used. Since hypothesis H_1 is two-sided, no uniformly most powerful test exists. However, the accepted procedure is to perform the following test:

$$\text{accept } H_0 \quad \text{if} \quad |S| \leqslant T$$
$$\text{accept } H_1 \quad \text{if} \quad |S| > T$$

where the test statistic S is

$$S = \sum_{i=1}^{n} x_i$$

and the threshold T is set by

$$\int_T^\infty \frac{e^{-z^2/2n}}{\sqrt{2\pi n}} dz = \alpha/2.$$

Solving for the threshold T, it is of the form

$$T = K\sqrt{n}$$

where K is a constant determined by the false alarm rate α.

It will be assumed in the discussion which follows that H_0 is the true hypothesis. Let the probability of rejecting H_0 on or before the mth test be P_m. Then

$$P_m = P_{m-1} + (1 - P_{m-1})\, C_m \tag{1}$$

where C_m is the conditional probability that the null hypothesis is rejected on the mth test given that it has not been rejected by the preceding $m - 1$ tests. We will now show that $P_m \to 1$ as $m \to \infty$. Since P_m is monotonic increasing and bounded by 1, P_m converges. However, if one performs a test every time one accumulates N_0 new samples, and if $C_m \to 0$[1] as $m \to \infty$, then the rate of convergence of C_m determines whether $P_m \to 1$. Unfortunately, it is difficult to calculate C_m to determine its convergence rate. Consequently, a different approach will be taken. Rather than perform a test every time N_0 new samples are accumulated, an increasing number of samples will be accumulated between tests.

Specifically, let m be related to the sample size n by

$$n = N_0^{2^m} \tag{2}$$

where N_0 is any integer greater than 1. That is, if n samples are used for test m, n^2 samples are used for test $m + 1$. By this

[1]If $C_m \to \epsilon > 0$ as $m \to \infty$, then $P_m \to 1$.