M.F. Goodchild (1977) Statistical aspects of the polygon overlay problem. In *Harvard Papers on Geographic Information Systems*. Reading, MA: Addison-Wesley, 6.

Michael F. Goodchild
Department of Geography
The University of Western Ontario
London, Ontario N6A 5C2
Canada

STATISTICAL ASPECTS OF THE POLYGON OVERLAY PROBLEM

The geographic data processing field encompasses a number of fairly clearly defined computational problems which tend to occur as frequent components of larger tasks. Each one is associated with a number of algorithms, having varying degrees of computational efficiency and core storage requirements. Perhaps the most frequently reviewed is the Point in Polygon problem (see for example Nordbeck and Rystedt, 1969). Choice of an optimum algorithm in this case will depend on the number of vertices per polygon, and whether repeated tests are required against a single polygon, or a number of tests of polygons against a single point. Other examples from the same field might include the line culling problem, contouring from a terrain model, or the Thiessen polygon problem. But of all these the polygon overlay problem, of defining a new set of polygons from the superimposition of two independent sets, has probably achieved the greatest notoriety. While it is undoubtedly an essential feature of any vector-based information system for areal data, it contains a number of challenging computational and statistical problems, of which the latter are the main subject of this paper.

The first section reviews example applications, and may be conveniently ignored by readers familiar with the problem. The following section reviews algorithms, as a prelude to subsequent sections on statistical difficulties.

EXAMPLE APPLICATIONS

The term areal data is used to describe any geographical information which is represented by an image consisting of a set of non-overlapping and normally space-partitioning polygons, together with a descriptive list of the attributes of each polygon. It includes maps of political areas, soil types, land uses, trading areas, and socio economic data aggregated above the household level. Contour maps can also be regarded as areal with a little willing imagination.

The simplest form of overlay arises when any two such maps of the same area, at the same scale, are superimposed. This creates a new, more complex image, in which every unique combination of some polygon A on one map and some polygon B on the other becomes a new polygon. This is of course the intersection of A and B, $A \cap B$, or A.AND.B. The descriptions of A and B are concatenated to form a combined description of each new polygon. This simple form of logical overlay is applied to answer information system queries about the area of land having various combinations of characteristics, such as land use x in county y.

Other applications may require processing of the concatenated descriptions, as in various forms of numeric overlay. Consider first the application of a crude model for the prediction of drainage basin runoff based on an analysis of land use. First, an overlay of a land use

map with a coverage showing drainage basins could produce a breakdown of area by land use by basin. Application of a runoff model for each land use would then allow aggregation of runoff by basin. The essential ingredient of such a process is a matrix showing the area of each land use in each basin, and generated by polygon overlay.

A second example of this latter type is found in the process of areal interpolation from polygon data. Suppose that two images are available, one showing the outlines of census tracts, the other school districts. Population density data is available for the census tracts, but required for the school districts, which do not respect census tract boundaries. An estimate can be made, however, by computing a matrix showing the area of each school district in each census tract, and multiplying this matrix by the vector of census tract population densities to obtain a new vector of school district densities. Of course the quality of the estimate depends on the within-tract homogeneity of population density, and is better in some cases than others. (See for example Waters, 1977.)

ALGORITHMS

The two main classes of approach to the problem will be referred to here as the exact and the heuristic. In the exact technique, all intersections of boundary lines from the two images are found precisely, and all polygons, however small, are identified in the new map. In the heuristic approach, each map is first converted to a grid structure by applying one of the standard techniques. The grids are then overlaid in a very simple and rapid step. Finally the composite grid is revectorized. The result is only approximate, since any features, including polygons, smaller than the dimensions of a grid cell tend to be lost in the composite. This is especially true of linear features, such as roads, rivers, etc., whose importance may be out of proportion to their physical extent.

An interesting hybrid technique can be used when the images to be overlaid were created by vectorization of raster data. In this case the overlay grid can be chosen to correspond to the original raster, with no loss of resolution. The method is still approximate, however, when compared to the original documents used to create the raster.

Very little has been written on exact algorithms. It is possible to compare each polygon on one map with each polygon on the other, testing for intersections and creating new polygons when they occur. The inefficiency of such an approach is illustrated in Figure 1. There are three intersection points to be found, x_1, x_2 and x_3. x_1 would be identified twice, as it occurs in the intersections $A_1 \cap B_1$ and $A_2 \cap B_1$. x_3 would be found in two tests, and x_2 in all four, $A_1 \cap B_1$, $A_1 \cap B_2$, $A_2 \cap B_1$ and $A_2 \cap B_2$. An arc-based approach avoids such duplication. Each map is represented by a number of arc records, each one including an indication of the polygons on either side. For example u_2, u_5, A_2 left, A_1 right. x_2 is then identified only in the comparison of this arc with v_2, u_5, B_2 left, B_1 right. Furthermore, since the arc-based structure contains a record of the neighbour relationships of polygons, it is easy to find the full concatenated identifications of the new polygons. When x_2 is found, four new fully identified arcs are created as follows: u_2, x_2, A_2B_1 left, A_1B_1 right; x_2, u_5, A_2B_2 left, A_1B_2 right;

v_2, x_2, A_2B_2 left, A_2B_1 right; x_2 v_5, A_1B_2 left, A_1B_1 right.

Only those arcs which contain intersection points with the overlaid image will be fully identified in this manner. For those which do not, only one part of the concatenated identifier will be known in the new map. There are two examples in Figure 1, the arcs consisting of u_2, u_1, u_6 and u_5; and u_2, u_3, u_4 and u_5. Although their neighbouring polygons in A are known, because neither form intersections with B arcs their positions with respect to B polygons are unknown. However, the arcs u_2, u_1, u_6, u_5 is known to be anticlockwise to arc x_1, u_2 with respect to polygon A_1. Arc x_1, u_2 is fully identified as A_1B_0 left, A_2B_0 right, by virtue of its intersection at x_1 with v_2, v_1, v_6, v_5, B_1 left, B_0 right. It follows that u_2, u_1, u_5, u_6 can be identified as A_1B_0 left, A_0B_0 right. So provided every original arc record contains pointers to adjacent arcs around each polygon, it is possible to infer the full identification of all arcs as long as at least one intersection exists. Table 1 gives the full results of the Figure 1 overlay.

A complete algorithm requires one final step, since each unique combination of identifiers from A and B may be associated with more than one new polygon. Thus it is necessary to follow the arc pointers through the file so as to be able to assign a unique number or name to each new polygon.

The difficulties of programming this approach become clear when one considers that first, files may be extremely large, the order of 10^6 vertices being modest, and second, that very little general software is available to handle random access and update of files consisting of

TABLE 1

INPUT AND OUTPUT FOR FIGURE 1

Arc	Vertices	Left	Left Polygon Pointers: Previous Arc	Left Polygon Pointers: Next Arc	Right	Right Polygon Pointers: Previous Arc	Right Polygon Pointers: Next Arc
[illegible]	$u_2\ u_1\ u_6\ u_5$	A_1	2	2	A_0	3	3
2	$u_2\ u_5$	A_2	3	3	A_1	1	1
3	$u_2\ u_3\ u_4\ u_5$	A_0	1	1	A_2	2	2
1	$v_2\ v_1\ v_6\ v_5$	B_1	2	2	B_0	3	3
2	$v_2\ v_5$	B_2	3	3	B_1	1	1
3	$v_2\ v_3\ v_4\ v_5$	B_0	1	1	B_2	2	2
1	$u_2\ u_1\ u_6\ u_5$	A_1B_0	2	5	A_0B_0	6	6
2	$u_2\ x_1$	A_2B_0	6	7	A_1B_0	1	8
3	$x_1\ x_2$	A_2B_1	7	9	A_1B_1	8	10
[illegible]	$x_2\ x_3$	A_2B_2	9	11	A_1B_2	10	12
[illegible]	$x_3\ u_5$	A_2B_0	11	6	A_1B_0	12	1
[illegible]	$u_2\ u_3\ u_4\ u_5$	A_0B_0	1	1	A_2B_0	2	5
[illegible]	$v_2\ x_1$	A_2B_1	9	3	A_2B_0	11	2
[illegible]	$x_1\ v_1\ v_6\ v_5$	A_1B_1	3	10	A_1B_0	2	12
[illegible]	$v_2\ x_2$	A_2B_2	11	4	A_2B_1	7	3
[illegible]	$x_2\ v_5$	A_1B_2	4	12	A_1B_1	3	8
[illegible]	$v_2\ v_3\ v_4\ v_3$	A_2B_0	7	5	A_2B_2	9	4
[illegible]	$x_3\ v_5$	A_1B_0	5	8	A_1B_2	4	10

variable length records. The program used in the illustrations in this paper was written to operate entirely sequentially. As a result, it achieves somewhat less than the theoretical efficiency at the points identified by asterisks in the steps below.

1) Fill a central memory buffer with arcs from file B. Store the x and y limits of each arc. If file B is exhausted, go to 4.

2) Read each arc of file A. Eliminate trivial tests by comparing x and y limits of each arc with those for file B. For arcs which pass the test, search for intersections. If found (and if no intersections are found lower on the same arc)*, split the arc from file A and write out the pieces, and split the file B arc within the buffer. Else write out the file A arc.

3) At the end of file A, write out the buffer and return to step 1 to process a new buffer of file B, using the new version of file A.

4) Chain around all new polygons and assign unique numbers sequentiall[y]. When arcs terminate at intersections, they can be chained by following pointers. (When no intersections occur, arcs are chained by matching the x, y coordinates of end points.)*

Figure 2 shows a rather more elaborate example in which two polygons, each consisting of a single arc, have been overlaid. The composite contains 65 polygons and 127 arcs. Since only four types of polygon can exist on the new map, A.AND.B., A.AND.NOT.B., .NOT.A.AND.B., and .NOT.A.AND.NOT.B., it provides a rather graphic illustration of the validity of the four-colour theorem (though not unfortunately a proof). It also illustrates the large number of polygons that can be created in a finite overlay operation.

THE SPURIOUS POLYGON PROBLEM

The number of polygons created in an overlay depends not on the number of polygons being overlaid, but on the complexity of each, defined by the vertices. An overlay of two polygons, consisting of n_1 and n_2 vertices respectively, can produce any number of polygons from three to $n_1 n_2 + 2$ including .NOT.A.AND.NOT.B. Moderate numbers of polygons are produced when the overlaid maps show statistical independence, meaning that arcs on one map show no identifiable tendency to follow arcs on the other. Figure 3 shows an overlay of 478 polygons with an independent set of 150 which produced 1503 new polygons.

Serious problems arise, however, when arcs show a tendency to coincide. This occurs most often because a prominent linear feature, for example a road, may appear in the image of a number of different maps. It may form a parcel boundary on a land use map, or a census tract boundary, or may even be embedded on such an apparently unrelated coverage as a classification of soils. But the two versions of the line will not coincide, and an overlay will produce a number of so-called spurious polygons, small slices delimited by the two versions. The term 'coastline weave' is sometimes used, though more often in connection with polygon-based digitizing methods.

The problem has two somewhat paradoxical aspects. First, its severity increases with the accuracy of digitizing and overlay. The more accurately each arc is located, and the more vertices that are used to indicate its outline, the greater the number of spurious polygons produced. Relatively crude digitizing was used in Figure 4; here, an overlay of 49 polygons with 55 produced 388, of which approximately 70% are spurious. Had the digitizing been done more carefully, and the manuscripts been drawn on stable bases, and to the same projection, the percentage would have been much greater.

The second element of paradox arises in the subjective process of boundary drawing. Many boundaries are drawn with a deliberate aim of maximizing coincidence with lines and other maps. Census tracts deliberately respect parcel boundaries and such physical features as rivers and major streets. The purpose is in effect to make the process of overlay by manual means as easy as possible, given the historical lack of any automated means.

THE EXPECTED NUMBER OF SPURIOUS POLYGONS

Consider two digitized versions of the same arc, with n_1 and n_2 vertices respectively. We assume first that there are no points of inflection on the arc. Figure 5a illustrates the typical situation in which a spurious polygon is generated. It can be represented by the string --12121--, indicating the sequence in which vertices occur along the true arc; a point digitized in version 1, one in version 2, etc. In this notation, a spurious polygon is generated for every incidence of five adjacent runs of different types. Thus the sequence is responsible for the presence of one complete spurious polygon. Four more are possible, but not certain unless the sequence is extended.

The statistics of runs of binary symbols are well known (see for example Mendenhall, 1975, pp. 380-385) and only require a slight modification to allow for the condition of boundedness in this case.

The number of spurious polygons S generated by two versions using n_1 and n_2 vertices ranges from

$$S_{min} = 0$$

to

$$S_{max} = 2 \min(n_1,n_2) - 4$$

with a random expectation, if symbols occur independently in the sequence, of

$$E(S) = \frac{2n_1n_2}{n_1 + n_2} - 3$$

The minimum occurs when all symbols of one type occur together; the meximum, when symbols are maximally intermingled.

Points of inflection complicate the problem. Figure 5b shows a simple example, the sequence --212i121--, generating one definite spurious polygon and at least two possible. Points of inflection act as operators on the remainder of the sequence, exchanging symbols. Thus the sequence --212i121-- is equivalent to the reduced form --212212--, which clearly generates one definite polygon. Sequences with many points of inflection can be reduced by repeated operation of the symbol i, beginning with the right most. The sequence shown in Figure 5c reduces to --11111-- in four steps, indicating that it generates no spurious polygons.

Unfortunately there is an exception to this apparently simple relationship between S and the sequence. Figures 5d and 5e show two versions of the sequence 12i2i21. Reduction would suggest one spurious polygon, as in Figure 5d. But Figure 5e, with the identical sequence, generates none. The uncertainty is due to the presence of a single run enclosed in inflection points. The correct result is obtained for 5(e) by deleting the run and inflection points to obtain --1221--, S = 0. Although 5(c) also contains runs enclosed in inflection points, the deletion of any combination always yields S = 0.

Now that the possible range of S has been identified, the next section will consider likely values under various assumptions about the digitizing process.

SPURIOUS POLYGONS AND DIGITIZING

Since there a number of ways of generating the vertices used to represent an arc, there must necessarily be a number of models of the process involved. Perhaps the simplest to deal with is incremental digitizing, in which a point is encoded every time the cursor moves by a fixed interval. The number of points will thus be inversely proportional to the increment for a given length of line. Two versions of a line digitized in this way will clearly show a sequence with very close to the maximum number of runs, and thus generate close to S_{max} spurious polygons, because when viewed as a sequence along the true line, the vertices from the two versions will be highly intermingled.

Time-lapse digitizing produces similar results. Fewer points will be generated for a given length of line if the time interval is increased, or if the operator moves more rapidly, and presumably less carefully, over the line. In either situation we expect near-maximum intermingling, and close to S_{max} spurious polygons.

In point digitizing, the operator selects vertices according to some subjective criterion. The difference between crude and precise digitizing is therefore a matter of intensity, or the threshold level of the criterion measure. If the line being digitized has fairly homogeneous characteristics along its length, a change in threshold will produce a uniform proportional change in sampling intensity. The process which formed such a line can be said to be stationary in the statistical sense. On the other hand, it is possible to visualize lines for which a change in threshold would produce a non-uniform change in vertex sampling density. Lines with stationary characteristics tend to produce close to S_{max} spurious polygons, whereas non-stationary lines may have rather less, because of reduced intermingling of vertices.

When raster data is vectorized, vertices are produced whenever the line ends a run in one cardinal direction (usually in the Queen's sense). The frequency with which this occurs depends on the size of the raster cell as well as on the nature of the line. However it is clear that the notion of stationarity can be applied as in the previous paragraph, and that S will approach S_{max} under such an assumption.

Thus far we have assumed that all versions of a line will contain points which lie on a single, true line, or at least that any errors will be small in comparison with the distance between adjacent vertices. However, there are many sources of error which can lead to larger and more systematic deviations, as for example when an inaccurate coordinate transformation is made prior to overlay. The situation then resembles that illustrated in Figure 5c, and can lead paradoxically to far fewer spurious polygons than in the absense of systematic error.

Assuming, however, that systematic errors are absent, and that the

line shows

to the conclusion that S will approach its maximum value. The effect of inflection points will be to decrease S to an unknown degree. The next sections describe a number of experiments designed to test these ideas and to add empirical evidence.

DATA SETS

The data consists of three lines with widely varying characteristics (Figure 6). (a) is a meandering river from Louisiana, showing almost constantly changing direction, and therefore requiring very dense vertex sampling. More precisely, it shows very weak serial correlation of direction. Figure 7(a) is the direction autocorrelogram, showing the degree to which the line's direction can be predicted given the direction at a previous point on the line separated by the distance indicated on the lag scale. It drops off very rapidly to zero as lag increases. Since it is impossible to use the conventional Pearson correlation coefficient for directional data, correlation is defined here as

$$r(\ell) = \frac{\sum_{t=1+\ell}^{N} \underset{\sim}{x}_t \cdot \underset{\sim}{x}_{t-\ell}}{\sqrt{\sum_{t=1+\ell}^{N} \underset{\sim}{x}_t \cdot \underset{\sim}{x}_t \sum_{t=1+\ell}^{N} \underset{\sim}{x}_{t-\ell} \cdot \underset{\sim}{x}_{t-\ell}}}$$

where $\underset{\sim}{x}_t$ is a vector representing the line at step t

ℓ is the lag

N is the number of steps

A constant step size of 0.05 in. was used, yielding approximately 1000 steps over the length of the line.

Line (b) is the configuration of Skyline Drive, Virginia, within and to the south of Shenandoah National Park. Its autocorrelogram (Figure 7(b)) shows a less rapid decline with lag, indicating less uncertainty about direction, and a lower sampling density. Finally, line (c) is formed by Ontario Highway 7 from Elginfield to Madoc. Since it has lengthy straight sections, interrupted by sharp corners, it shows the greatest serial autocorrelation, even over substantial lags.

HYPOTHESES

Two hypotheses were tested using the data sets described in the previous section. First, there seems to have been very little research on the behavioral aspects of digitizing. Although operators find no difficulty in interpreting the basic idea of representing a line by a series of points connected by straight line segments, it is unclear precisely what criteria are chosen and implemented. On the other hand, a number of algorithms have been developed for an essentially similar process of reducing the number of points present in an arc, the so-called line culling problem. It is hypothesized that line culling is directly analogous to digitizing, and the next section describes a test.

The second hypothesis relates directly to the spurious polygon problem. The effect of points of inflection has been shown to be indeterminate. It is hypothesized that in real situations the net effect is to reduce the number of spurious polygons from the theoretical maximum, and therefore that the observed number generated by overlaying two versions of the same line will be a certain proportion of the theoretical maximum. The hypothesis is tested below using the three data sets.

DIGITIZING AND LINE CULLING

A number of algorithms have been proposed for line culling, and Douglas and Peucker (1973) have reviewed several. They concluded that the most generally useful criterion should be the maximum distance between the culled line and its parent, and presented heuristic algorithms for the selection of the minimum number of points so that a set criterion distance is not exceeded.

The three lines were first culled using a number of different criterion distances, according to the heuristic described by Douglas and Peucker as Method II. Relationships between the criterion and the number of points were obtained in Figure 8.

Five subjects were asked to digitize each of the three lines. The instructions were deliberately left vague, although each subject was encouraged not to use excessive care. The number of vertices was compared to Figure 8 to obtain an interpolated criterion distance, in other words the culling criterion which would have generated the same number of points. The actual maximum distance was then computed. It can be defined as the radius of the largest circle that can be drawn centred either on a vertex of the digitized line or on the true line, and tangential to the other line.

Table 2 shows the results. First, there are several anomalies. In case (a)3 the operator made a clear mistake in following the true line. The latter line is of course represented internally by a large number of vertices, which themselves have a criterion distance with respect to the original manuscript. Thus when a large number of vertices were used by

TABLE 2

Line	Case	Vertices	Interpolated Criterion Distance	Observed Distance	Ratio
(a)	1	158	0.058	0.124	2.0
	2	170	0.050	0.101	2.02
	3	174	0.048	0.470	9.79
	4	210	0.033	0.086	2.61
	5	263	0.016	0.114	7.13
(b)	1	85	0.045	0.077	1.71
	2	89	0.043	0.097	2.26
	3	98	0.038	0.098	2.58
	4	101	0.036	0.068	1.89
	5	174	0.003	0.054	18.0
(c)	1	60	0.040	0.082	2.05
	2	66	0.030	0.096	3.20
	3	69	0.027	0.072	2.67
	4	81	0.018	0.048	2.67
	5	99	0.005	0.039	7.8

the operator, as in cases (a)5, (b)5 and (c)5 it became impossible to correctly estimate a criterion distance, and hence the anomalous ratios.

With these exceptions, the ratios are remarkably consistent. They show that a digitizer operator will select a set of vertices such that the maximum observed distance between the digitized line and the true line is typically twice the minimum obtainable for the given number of vertices. The average appears to be rather higher in the case of line (c) than the others. This line shows the greatest variability, from long straight sections to sharp curves, although fewer vertices were selected in each case.

The autocorrelograms of the digitized lines were compared with two norms: the autocorrelogram of a line culled to give the observed number of vertices, and that of a line culled to the observed maximum distance. In all cases the observed autocorrelogram showed a greater resemblance to the former norm (an example is shown in Figure 9) indicating that despite the greater distance deviation, the digitized lines tended to preserve the autocorrelation structure of the culled line having the minimum distance deviation for the given number of points.

SIMULATED SPURIOUS POLYGONS

In order to evaluate the second hypothesis, each digitized line was overlaid on the true version, and spurious polygons counted. Table 3 shows the number of points in the digitized line in each case, the observed value of S, and two norms, S_{max} and the random expectation E(S). It is clear that points of inflection have a systematic effect,

TABLE 3

RESULTS OF OVERLAY SIMULATIONS

Case	n_2	Spurious Polygons	S_{max}	E(S)	% of S_{max}
(a) 1	158	199	312	206	64
2	170	239	336	216	71
3	174	195	344	220	57
4	210	185	416	247	45
5	263	179	522	281	34
(b) 1	85	114	166	113	69
2	89	84	174	117	49
3	98	96	192	125	51
4	101	131	198	127	67
5	174	135	344	176	40
(c) 1	60	79	116	78	69
2	66	59	128	83	47
3	69	51	134	86	39
4	81	73	158	95	47
5	99	71	194	107	37

since S never exceeds 71% of S_{max} and is in fact better estimated from E(S). On average, S is 52.4% of the theoretical maximum, and 82.7% of the random expectation. However, for all cases but three, a null hypothesis of randomness in the sequence can be rejected with 95% confidence.

CONCLUDING REMARKS

The presence of spurious polygons in an image resulting from overlay presents a major complication. It increases the complexity of the image, and can multiply by many times the volume of the data set describing polygon attributes. An overlay of five coverages from the Canada Geographic Information System described in a previous paper (Goodchild, 1974) produced 90599 polygons, of which 75.2% were estimated to be spurious. The presence of so many spurious polygons multiplies the volume of the image data set by a factor of as much as two, and the descriptive data set in this case by four, and theoretically without limit.

The aim of this paper has been to stress the severity of the problem, and to try to estimate its magnitude and the conditions under which it occurs. Removal of spurious polygons after overlay is another matter, and relies upon methods being developed for their identification. They can be isolated simply on the basis of size (Goodchild, 1974) but also tend to show serial correlation along arcs; if one polygon created by two arcs is found to be spurious, it is likely that other polygons formed by the same arcs will be also, as spurious polygons originate in properties of arcs. They also have the characteristic of being tructed from only two arcs. There is considerable potential for the lopment of sophisticated probabilistic decision rules.

Spurious polygons can be removed by a number of procedures. One simply dissolve one side on a random basis. Alternatively, the two sing intersections can be connected by a new arc. This could be a ple straight line, or a more elaborate procedure designed to give the t alignment based on both versions.

Traditionally, geographers have paid remarkably little attention to e accuracy of maps. The spurious polygon problem and the raster/vector ntroversy both point to a developing need for the explicit recognition such concepts as spatial resolution and generalization, since igital techniques, unlike analog ones, have no implicit imprecision.

BIBLIOGRAPHY

Douglas, D. H. and T. K. Peucker, 1973. "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature", Canadian Cartographer 10, 2, pp. 112-122.

Mendenhall, William, 1975. Introduction to Probability and Statistics. North Scituate, Mass.: Duxbury Press, 4th Edition.

Nordbeck, S. and B. Rystedt, 1969. Computer Cartography: Point in Polygon Programs. Lund, Sweden: University of Lund, Studies in Geography Series C, No. 7.

Waters, Nigel, 1977. "Methodology for Servicing the Geography of Urbai Fire". Unpublished Ph.D. thesis, Department of Geography, University of Western Ontario.

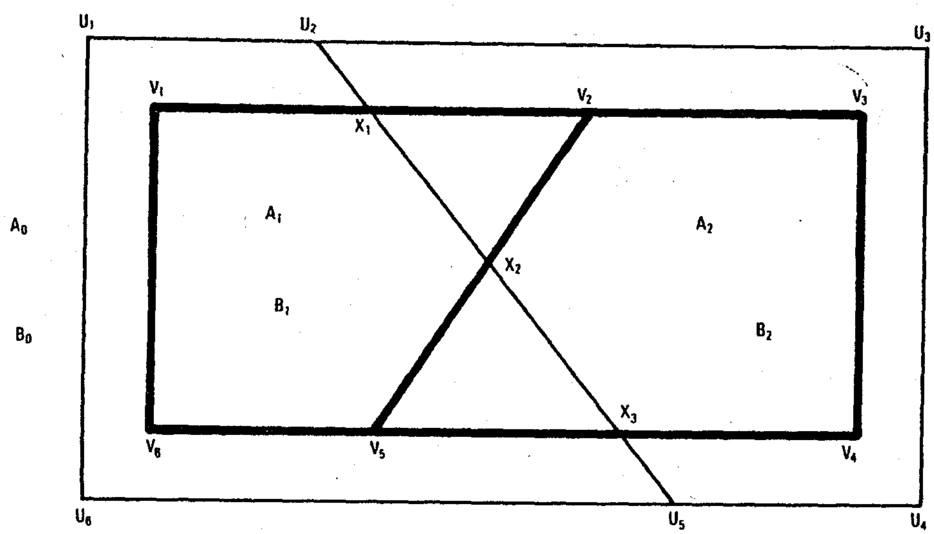

Figure 1. Example Overlay.

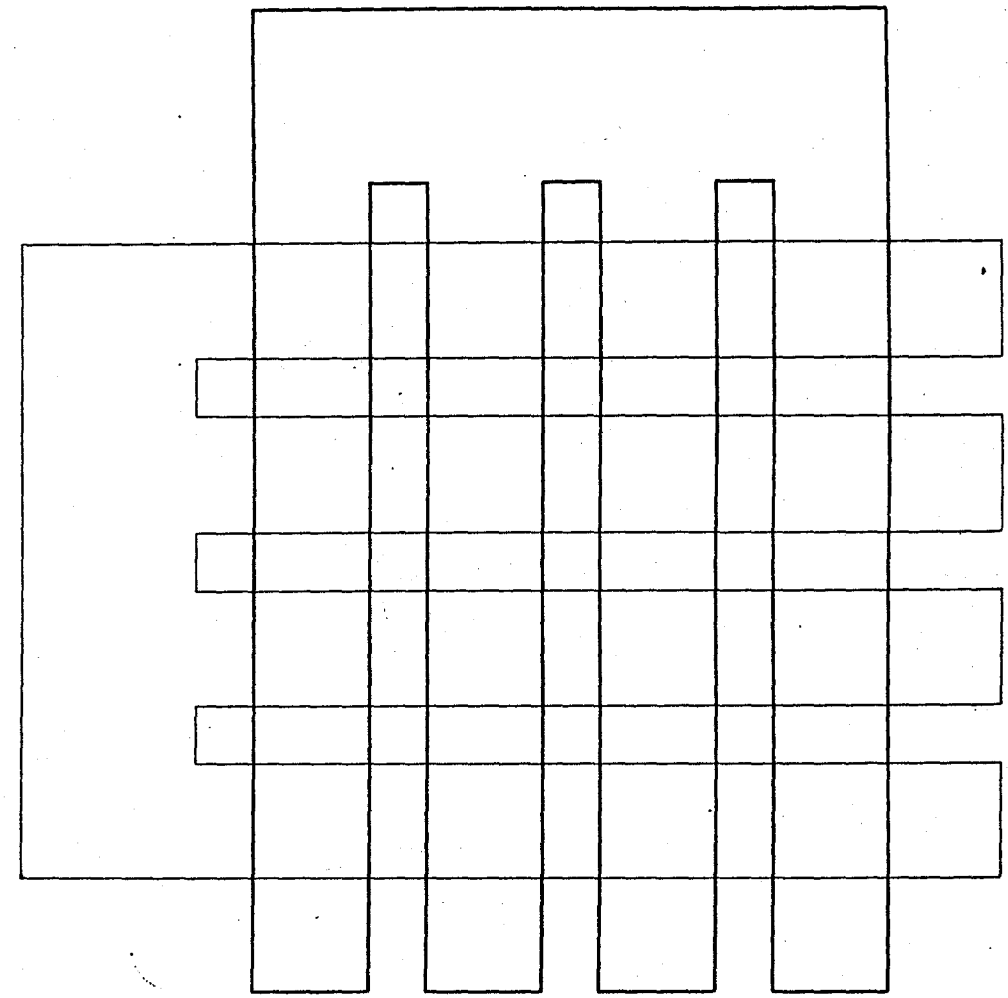

Figure 2. Overlay of Two Contorted Polygons.

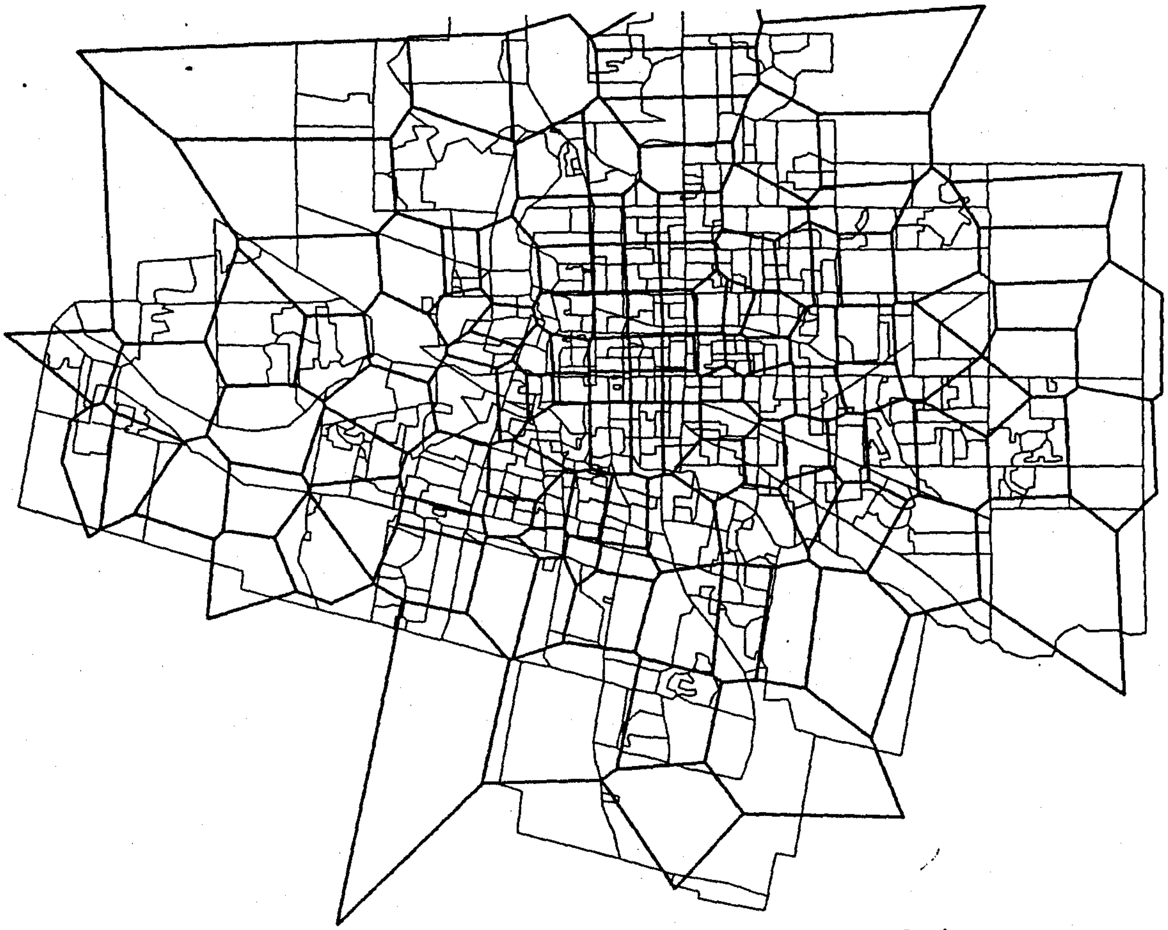

Figure 3. Overlay of a Set of Thiessen Polygons With Enumeration Areas For London, Ontario.

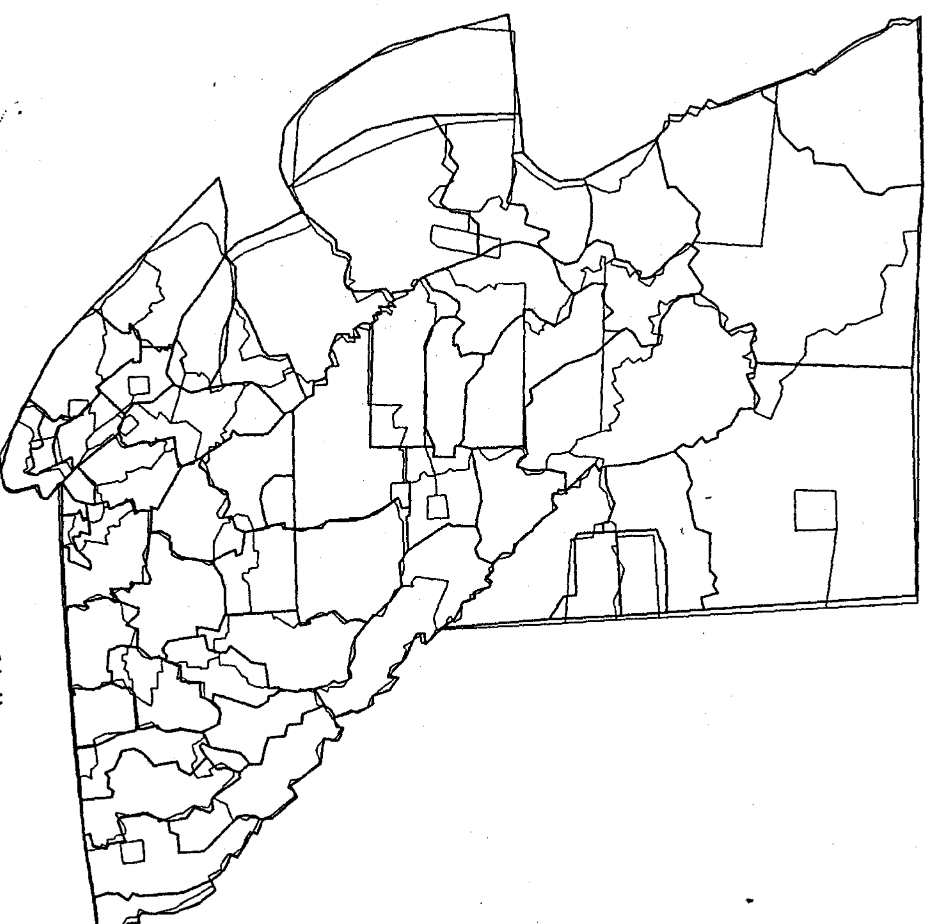

Figure 4. Example Overlay of Two Images of British Columbia.

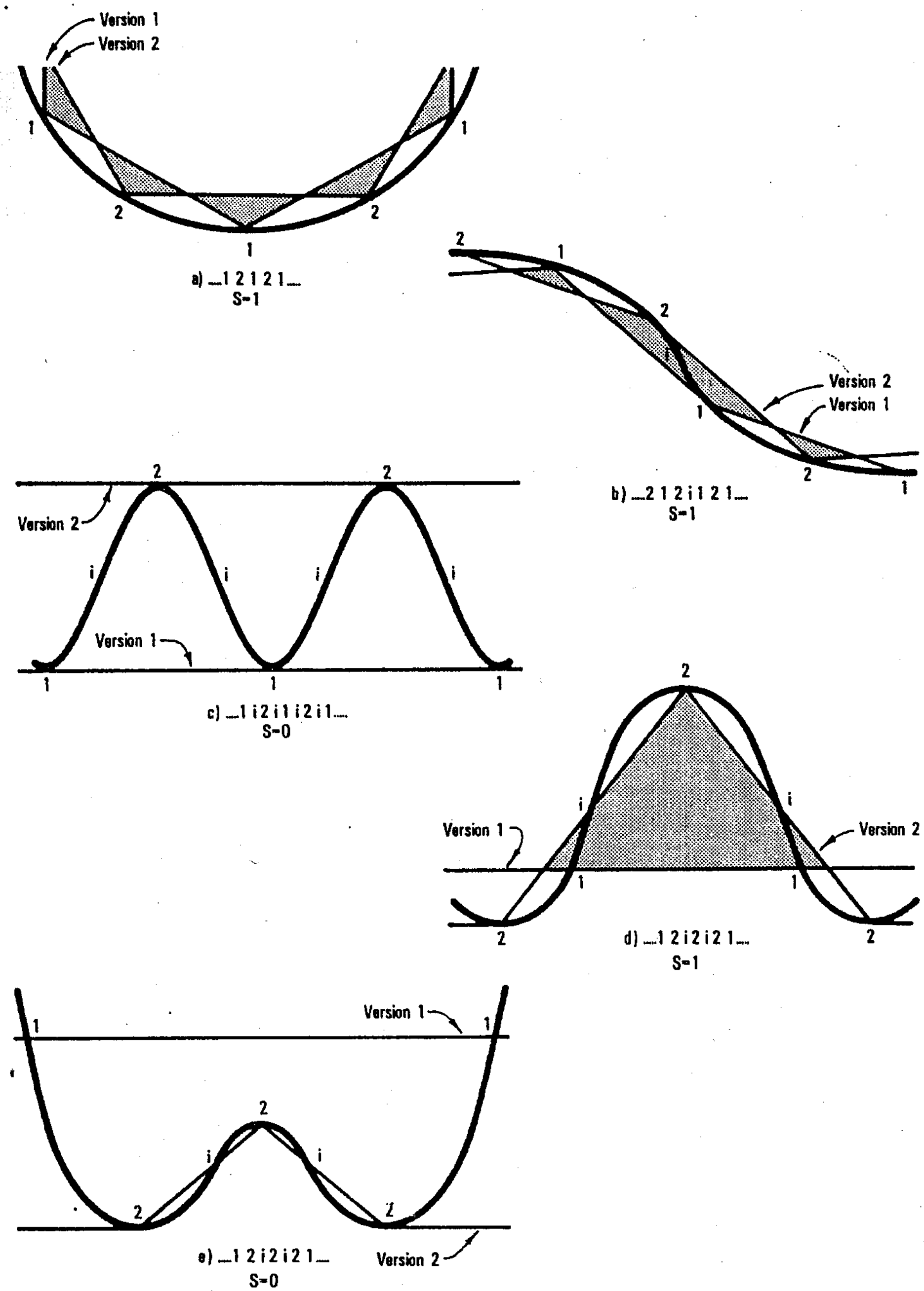

Figure 5. Situations Generating Spurious Polygons.

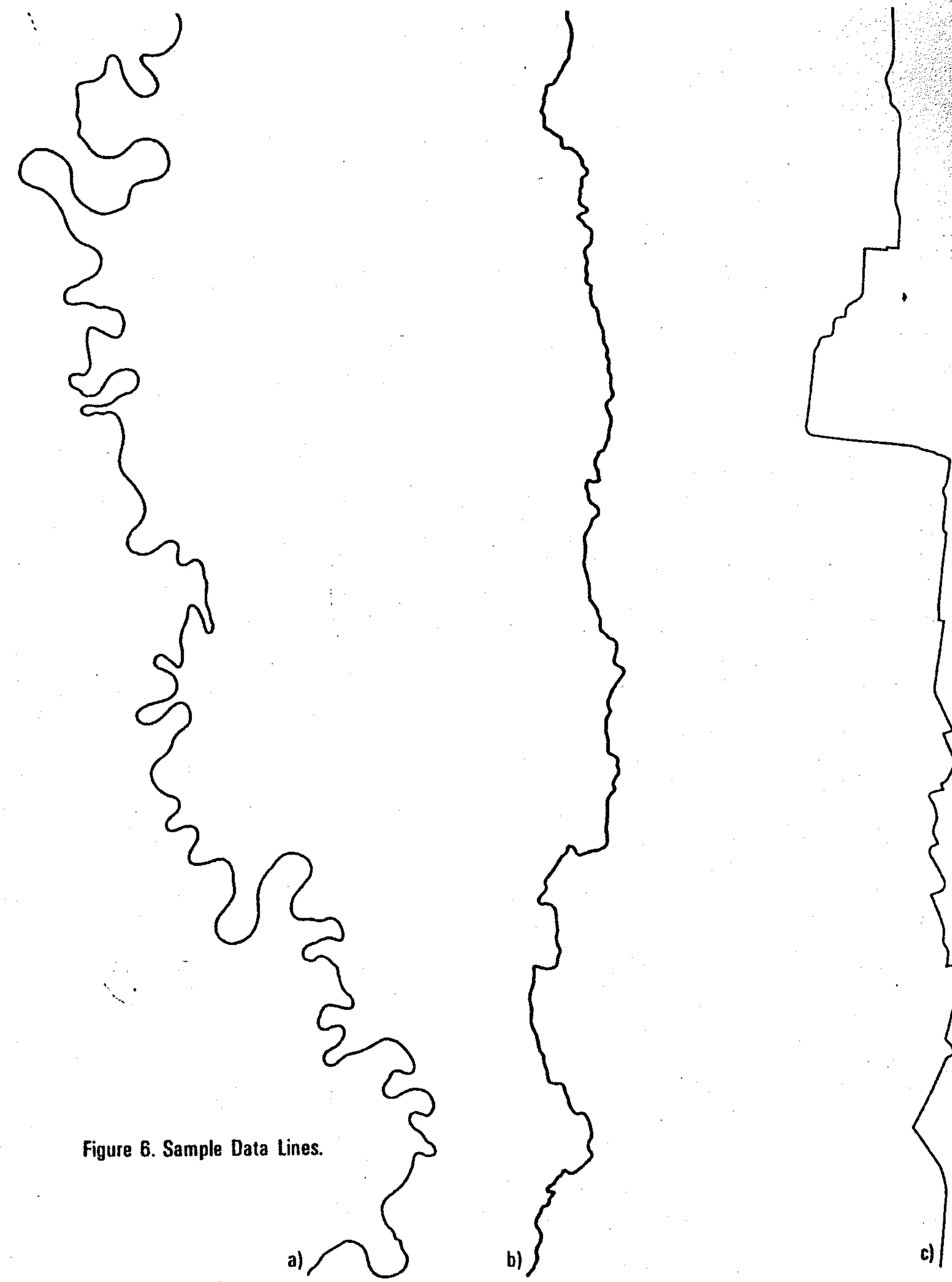

Figure 6. Sample Data Lines.

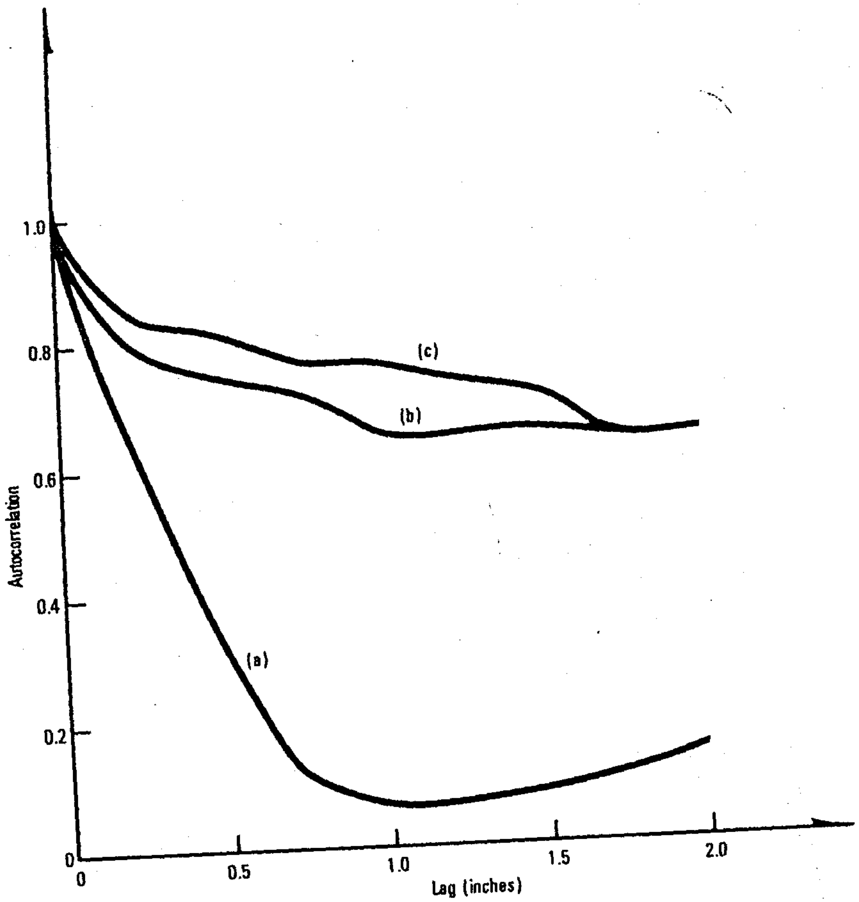

Figure 7. Autocorrelograms of Data Lines.

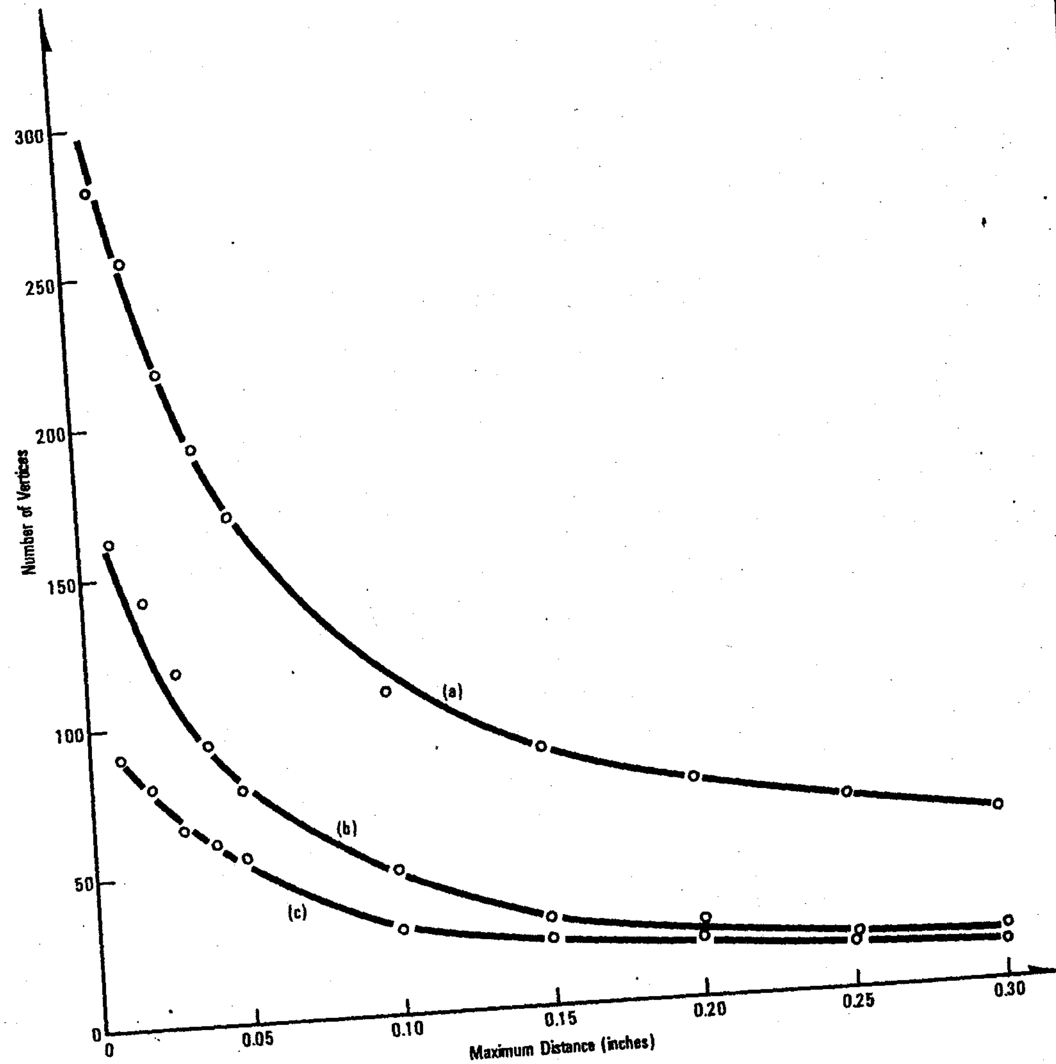

Figure 8. Culling Relationships.

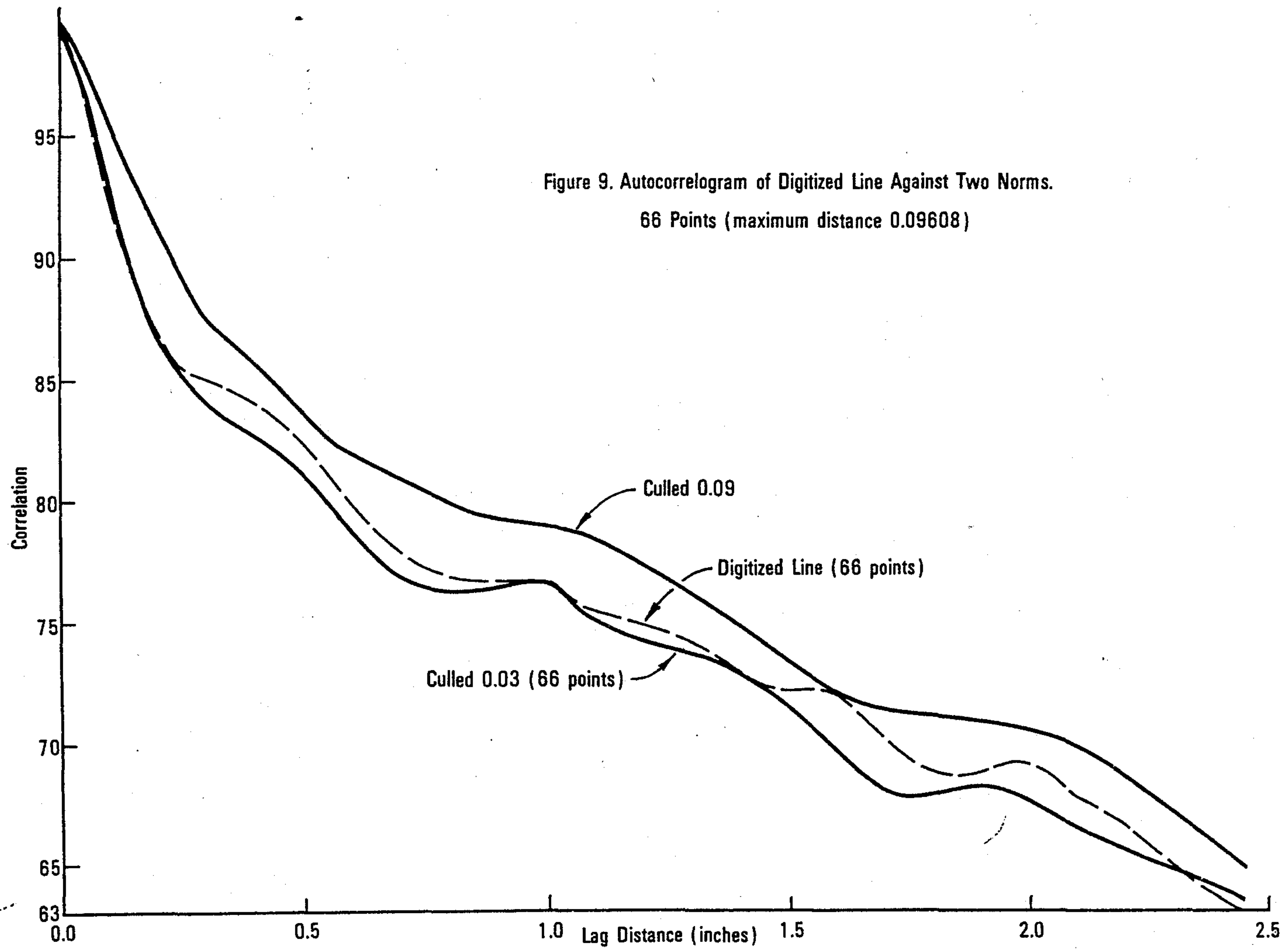

Figure 9. Autocorrelogram of Digitized Line Against Two Norms.
66 Points (maximum distance 0.09608)