
4 Algorithms for Transformations of Point Features

In Chapters 2 and 3 some foundations were laid down. From this chapter on, various types of algorithms for multi-scale representation will be presented. This chapter will describe algorithms for point features.

4.1 ALGORITHMS FOR POINT FEATURES: AN OVERVIEW

On a two-dimensional (2-D) spatial representation, spatial features can be categorized into three types according to the geometric characteristics of symbols: point, linear, and polygonal (or areal). From this chapter on, the algorithms of various transformations for each type of these features, as mentioned in Chapter 1, will be presented sequentially.

Two of the transformations for individual point features, that is, elimination and exaggeration, are very simple and thus will be omitted in this text. The displacement of point features will be discussed in Chapter 11, together with the displacement of line and area features. In this chapter, only the algorithms for simplifying a set of point features will be described. Usually, a set of point features is called a *point cluster*.

Two types of spatial features can be regarded as point set (or *point cluster*) on a spatial representation. The first type includes the features that are represented by point symbols on the original (source) representation (maps), for example, control points, wells, fountains, small buildings, and spot heights. The second type includes the features that are represented by areal symbols on the original (source) representation but will be represented (or will be analyzed) as point features on target representations because of scale reduction, for example, a cluster of islands, lakes, and ponds. [Figure 4.1](#) is a graphic illustration of the two types of point features. In this book we make no differentiation because our main concern is with the target map, that is, the representation of results after multi-scale transformations.

In summary, the algorithms to be discussed in the following sections are for aggregation, selective omission, structural simplification, and typification of a set of point features.

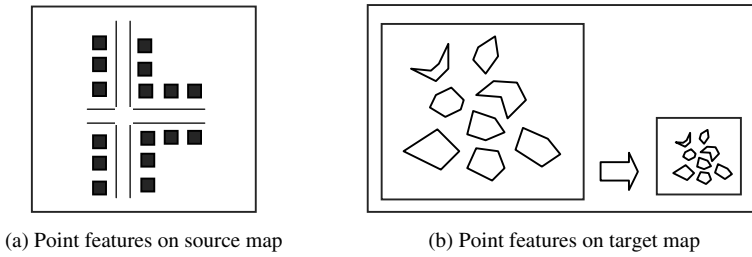


FIGURE 4.1 Two types of point features on a spatial representation.

4.2 ALGORITHMS FOR AGGREGATION OF A SET OF POINT FEATURES

As defined in Section 1.4.1, *aggregation of point features* means the categorization of a set of points into groups (classes or clusters) and then representation of each group by a single point. Therefore, this is a two-step process, and the critical part is the first step, which is often referred to as spatial clustering.

Spatial clustering is putting similar features into the same cluster (class or group). All *clustering algorithms* aim to minimize a measure of dispersion (in similarity) within the clusters. The measure of dispersion can be any of the following:

- Maximum distance to the cluster center (centroid) for any feature.
- Sum of the average distance to the center (centroid) over all clusters.
- Sum of the variance over all clusters.
- Total distance between all features and their center (centroid).

The similarity in nature could be measured by spatial, temporal, or socio-economic variables. In this context, distance is used as a general term to measure the similarity between two point features. That is, two close points will be classified in the same class.

Clustering techniques have been widely used for unsupervised classification of remote sensing images and spatial analysis. Clustering can be achieved sequentially or interactively. Clustering algorithms can be hierarchical or nonhierarchical. Therefore, there are many sequential hierarchical, sequential nonhierarchical, interactive hierarchical, and interactive nonhierarchical algorithms. In this chapter, only two commonly used algorithms, the K-means algorithm and the ISODATA (iterative self-organizing data analysis technique algorithm), are introduced.

4.2.1 K-MEANS CLUSTERING ALGORITHM

The basic idea of *K-means clustering* (MacQueen, 1967) is to partition (or classify) a set of N points into K clusters (groups or classes) that are mutually exclusive. The basic procedure is as follows:

1. Determine the number of clusters (K).
2. Designate a point as the cluster center for each of the K clusters.

3. Assign each of the N points to the nearest cluster (i.e., with shortest distance to the center, or centroid, of the cluster).
4. Compute the new center (centroid) for each of the K clusters after all points are assigned.
5. Repeat steps c and d until the centers (centroids) no longer move around significantly.

Figure 4.2 illustrates the process of K -means clustering. In this figure, 14 points are to be classified into 3 clusters. Points $K_{1,1}$, $K_{1,2}$, and $K_{1,3}$ are selected as the initial centers of these clusters (Figure 4.2a). After the first round of classification, six points are classified into cluster 1, five points into cluster 2, and three into cluster 3 (Figure 4.2b). The new centers are then computed and a second classification is carried out (Figure 4.2c). From the result of second classification, new centers are computed (Figure 4.2d). At this point the new centers do not move significantly, so

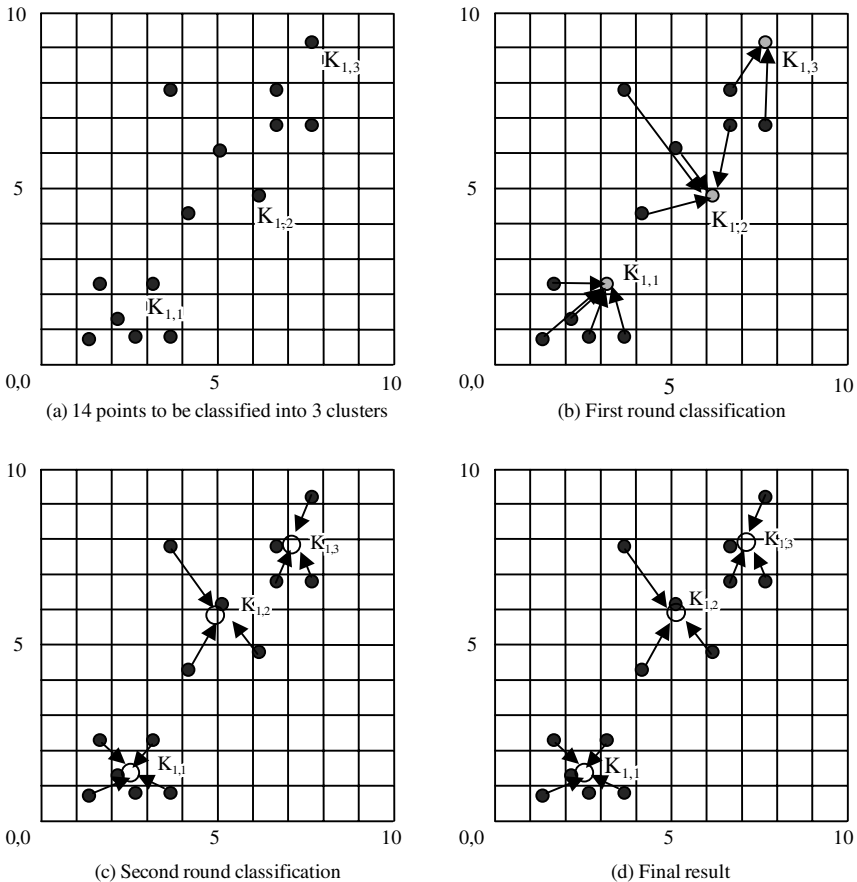


FIGURE 4.2 Clustering by the K -means algorithm.

the classification procedure stops. In the second round, the K -means algorithm ensures that

$$d(P_{i,j}, C_i) \leq d(P_{i,j}, C_l) \quad (i \neq l) \quad (4.1)$$

where P_{ij} is the j th point in cluster i ; C_i is the center (or mean) of cluster i ; C_l is the center (or mean) of cluster l ; $d(P_{ij}, C_l)$ is the distance from point P_{ij} to the center of cluster l .

4.2.2 ITERATIVE SELF-ORGANIZING DATA ANALYSIS TECHNIQUE ALGORITHM (ISODATA)

ISODATA (Tou and Gonzalez, 1974), as the name implies, is an interactive algorithm. It is not necessary to specify the number of clusters. It starts with a single cluster and applies a split-and-merge technique to progressively partition the points into more clusters through constantly assessing the similarity within a cluster (class or group). The similarity of points within a cluster is measured by the standard deviations of points in both the X and Y directions, that is, σ_x and σ_y . The procedure works as follows:

1. Determine the allowable values for the standard deviations, that is, $\sigma_{x,\max}$ and $\sigma_{y,\max}$.
2. Determine the number of clusters (K) and the number of iterations (n) (optional).
3. Treat all points as being in the same cluster to compute the means ($C_{old,X}$ and $C_{old,Y}$) and the standard deviations (σ_x and σ_y) in both X and Y.
4. Determine whether there is a need to split the cluster. If $\sigma_x < \sigma_{x,\max}$ and $\sigma_y < \sigma_{y,\max}$, then stop splitting. If the specified number of iterations or number of clusters is reached, stop splitting. Then, if $\sigma_x > \sigma_y$, consider the X direction, or else consider the Y direction.
5. Split the cluster into two in the X direction if $\sigma_x > \sigma_y$ and $\sigma_x > \sigma_{x,\max}$. The temporary new centers are $(C_{old,X} - \sigma_x)$ and $(C_{old,X} + \sigma_x)$. Classify the points in the old cluster into two new clusters based on distance criterion. However, if $\sigma_y > \sigma_x$ and $\sigma_y > \sigma_{y,\max}$, then the split will be in the Y direction.
6. For each of the new clusters, repeat steps 4 to 5.
7. Check each point to see whether the distance to its cluster centroid is the smallest among the distances to all centroids. If not, reclassify the point and recompute the corresponding centroids.

Figure 4.3 illustrates the process of ISODATA clustering. The 14 points shown in Figure 4.2 are used again. In this example, the number of clusters is defined as four, the number of iterations as three, and $\sigma_{x,\max}$ and $\sigma_{y,\max}$ as 1.4. Figure 4.3a shows the consideration of all points as being within the same cluster (class or group), with point K_1 as the center. The standard deviations of this cluster in X and Y are 2.1 and 3.0, respectively. These values are larger than the thresholds. The first split is carried out in the Y direction because $\sigma_y > \sigma_x$. $K_{2,1}$ and $K_{2,2}$ are the two new cluster centers. Clustering is then carried out based on distance, as shown in Figure 4.3b.

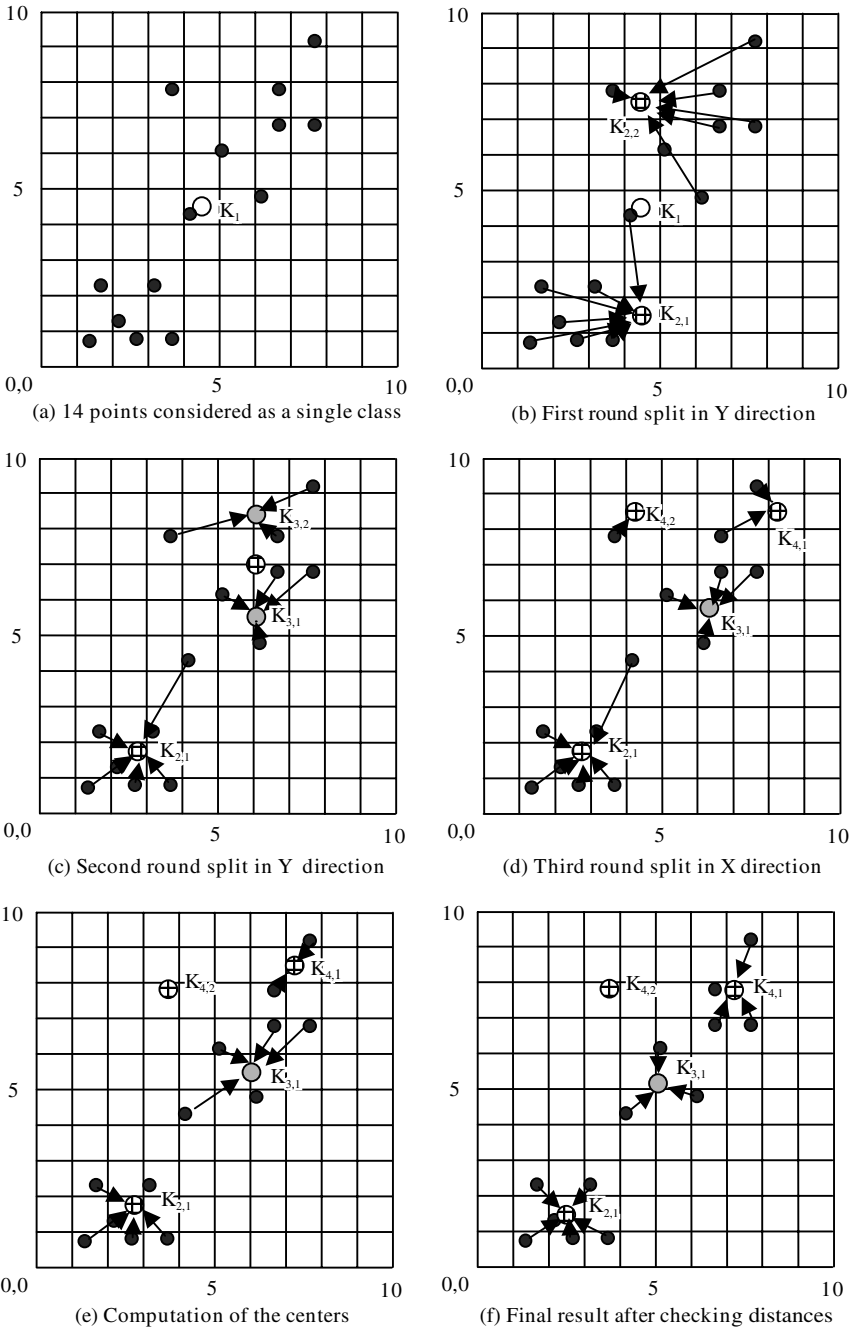


FIGURE 4.3 Clustering by ISODATA.

The centroids and the standard deviations of the two new clusters are then computed. It is found that the Y standard deviation of the cluster centered at $K_{2,2}$ is larger than $\sigma_{Y,\max}$, and then a further split is carried out. The process is shown in Figure 4.3c. The centroids and standard deviations of these two new clusters are computed. It has been found that the X standard deviation of the cluster centered at $K_{3,2}$ is larger than $\sigma_{X,\max}$, and then a further split is carried out, as shown in Figure 4.3d. The centroids of these two new clusters are then computed. In the end, each point is checked to see whether the distance to its cluster centroid is the smallest among the distances to all cluster centroids. It has been found that one point in the cluster centered at $K_{2,1}$ should be classified into the cluster centered at $K_{3,1}$. This point is then reclassified. As a result, the centroids of both clusters have been changed (Figure 4.3e). Consequently, a point in the cluster centered at $K_{3,1}$ is reclassified into $K_{4,1}$. Then the centroids of both clusters are moved, which causes a further point to be reclassified in the cluster centered at $K_{3,1}$ (Figure 4.3f).

4.2.3 DETERMINATION OF A REPRESENTATIVE POINT FOR A CLUSTER OF POINT FEATURES

After the first step of the aggregation operation, that is, clustering, all points are grouped into clusters. The next step is to represent the cluster by a single point. It can be imagined that the most representative point should be used. The question is, "What point is most representative?" From statistics, it can be found that the representatives are

- Mode: A point in an area with much higher distribution density compared with its surroundings (Figure 4.4a).
- Mean: The average coordinates in both X and Y coordinates (Figure 4.4b).
- Median: The point that partitions the planar space into two halves in X (i.e., left half and right half) and two halves in Y (lower and upper) to ensure equal numbers of points on each half in the same direction (Figure 4.4c).
- Nearest to mean: The point closest to the mean (Figure 4.4d).

It should be pointed out that the definition of *aggregation* given in this text may be different from others. Here, *aggregation* is referred to as the clustering followed by any of these four points (i.e., mode, mean, median, and nearest to the mean). However, the use of the mean to represent the cluster was termed as *typification* and the use of the point *closest to the mean as selection* by Jiang (2004).

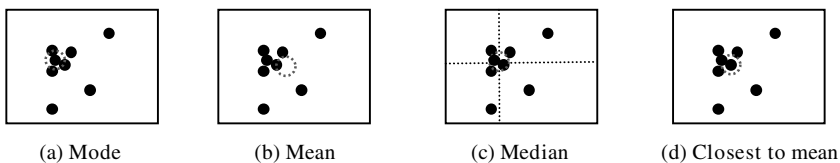


FIGURE 4.4 The most representative point of a point class cluster.

4.3 ALGORITHMS FOR SELECTIVE OMISSION OF A SET OF POINT FEATURES

Most *selective omission algorithms* are specially developed for selective omission of settlements. It is understandable that, when the scale of a map is reduced, not as many point symbols (e.g., settlements) can be represented. As a consequence, less important point features should be omitted. The importance of a point feature can be measured by one or more attributes. For example, a city may be signified by its population, gross domestic product, administrative status, or physical size. However, if some important point features are very close to an even more important feature, they may be omitted due to space problems while some less important points in sparse areas may be selected. For example, in eastern China large cities are located more closely together than in western China. When the map scale is reduced, some large cities in the east must be omitted because of a reduction in map space, while some medium cities in the west are retained. Figure 4.5 shows an example.

From the literature, it can be found that six algorithms have been proposed, five by Langran and Poiker (1986) (settlement-spacing ratio algorithm, distribution-coefficient algorithm, gravity-modeling algorithm, set-segmentation algorithm, and quadrant-reduction algorithm) and one by van Kreveld et al. (1995)



FIGURE 4.5 Selection and omission of cities as point features (<http://china-hotelguide.com/>).

(circle-growth algorithm). van Kreveld et al. pointed out that the set-segmentation and quadrat-reduction algorithms require a great deal of human intervention, so they are not suitable for automation. van Kreveld et al. also pointed out that the other three methods do not directly give a ranking of the base set, so changing the number of selected settlements involves recomputation. Indeed, the circle-growth algorithm is an improvement of the settlement-spacing ratio algorithm. Therefore, only the settlement-spacing ratio and circle-growth algorithms are described in this chapter.

4.3.1 SETTLEMENT-SPACING RATIO ALGORITHM

In the *settlement-spacing ratio algorithm*, a circle is used to indicate the significance of a point feature. The radius of the circle is inversely proportional to the importance of the point, that is,

$$R_i = \frac{C}{I_i} \quad (4.2)$$

where R_i is the radius of the circle for the i th point feature, I_i is the importance of the i th point feature, C is a constant, and $C > 0$.

In the selection process, points are tested in order of decreasing importance. That is, the most important point is tested first. A point will be selected only if its circle does not contain a previously selected point. In this way, important points close to a more important point may be omitted, while less important points with isolation may be selected. The critical step is the selection of an appropriate value for the constant C .

Figure 4.6 shows such a selection process. Figure 4.6a shows assets of five points with three levels of importance. Point A is the most important, point C is the least important, and points B, E, and D are the same level. Point A is selected first. A circle is then drawn from each point. It is found that the circle of point B contains point A, and thus point B is eliminated (Figure 4.6b). All the other points are retained, although point C is the least important (Figure 4.6c).

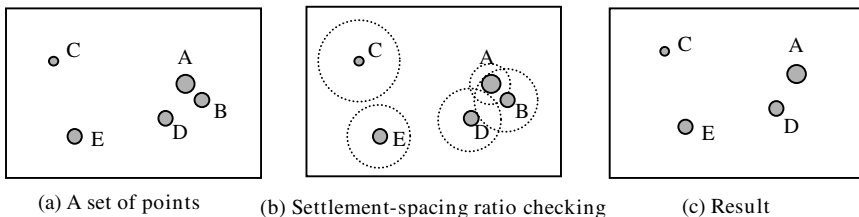


FIGURE 4.6 Selective omission of point features by the settlement-spacing ratio algorithm.

4.3.2 CIRCLE-GROWTH ALGORITHM

The idea of the *circle-growth algorithm* is the opposite of the settlement-spacing ratio. In the former, a larger circle is drawn for the more important point feature. The radius of the circle is directly proportional to the importance of the point, that is,

$$R_i = C \times I_i \tag{4.3}$$

where R_i is the radius of the circle for the i th point feature, I_i is the importance of the i th point feature, C is a constant, and $C > 0$.

In the selection process the critical step is to rank each of the point features. To produce such a ranking, a circle is drawn from each point, with radius proportional to its importance, according to Equation 4.3. The initial value of C in Equation 4.3 is set such that no circles will overlap. Then the value of C is increased so that one or more circles of less important points will be contained by the circle of a more important point. The one covered by a larger circle is given a lower ranking, while the one with a larger circle is assigned a higher ranking. This process is repeated until the most important circle remains. In the end, points with low rankings will be deleted.

Figure 4.7 shows such a selection process by circle-growth algorithm. Figure 4.7a shows a set of five points with three different levels of importance. Figure 4.7b shows the first round of circle growth so that the circle of point A first covers that of point B. Point B is then given the lower ranking and is the first removed. Figure 4.7c shows the second round of circle growth so that the circle of point

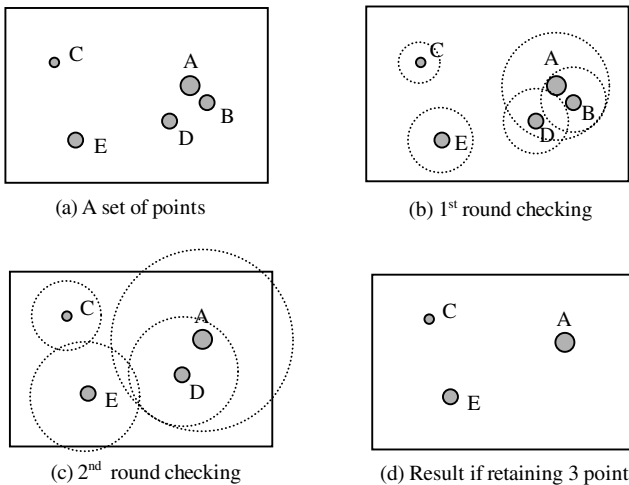


FIGURE 4.7 Selective omission of point features by the circle-growth algorithm.

A first covers that of point D. Point D is then given the lower ranking and is removed if appropriate. This process continues until the last point remains. Figure 4.7d shows the result if one decided to select three points from the set of data.

The drawback of the circle-growth algorithm is that the points with very high importance have too much influence on the selection, and this results in the opposite of preserving density locally (van Kreveld et al., 1995). This is also true for other algorithms mentioned previously in this section.

4.4 ALGORITHMS FOR STRUCTURAL SIMPLIFICATION OF A SET OF POINT FEATURES

To simplify the structure of a set of point features for representation at a smaller scale, one needs to have a set of parameters for the description of a set of points (or point cluster). From the literature (e.g., Ahuja, 1982; Flewelling and Egenhofer, 1993; Guo, 1997; Yukio, 1997; Ai and Liu, 2002), it can be found that the following parameters are possible:

- Point number: The number of the point features in the set.
- Importance value: A value assigned to a point as a measure of its importance among point features.
- Voronoi neighbors: Points whose Voronoi regions are adjacent to that of a given point.
- Distribution range: One or more regions enclosing all point features of interest.
- Distribution density: The number of points in a unit area or the average distance between point features.
- Distribution modes: One or more areas with much higher distribution density compared with their surroundings.
- Distribution axes: One or more axes extracted from the area of a point cluster whose extent is linear.

Other parameters, such as shape, size, color, and orientation, have also been used (Yukio, 1997), but they concern the symbolization of point features.

4.4.1 STRUCTURAL SIMPLIFICATION BASED ON METRIC INFORMATION

In the literature, one may notice that there are not many algorithms for the *structural simplification* of a set of point features. This section describes the algorithm by Ai and Liu (2002).

Ai and Liu (2002) developed an algorithm for preserving the distribution characteristics of point clusters after simplification. They try to retain the density, centers, axes, and range of the distribution. Through analysis, they claimed that as long as the range is preserved, the axes are automatically preserved, and as long as the

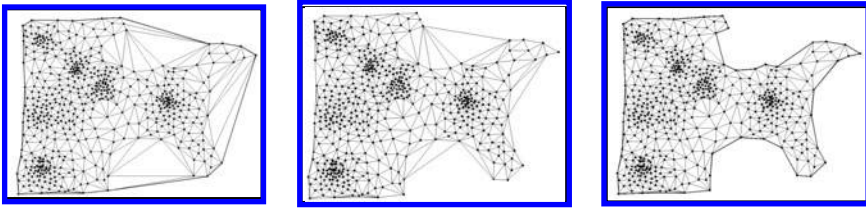


FIGURE 4.8 Defining the boundary of the point set by stripping the convex hull (Reprinted from Ai and Liu, 2002. With permission.).

relative density is preserved, the distribution centers are also preserved. Therefore, they tried to preserve the two more important parameters, that is, distribution range and relative density.

The working principle of this algorithm is as follows:

1. Determine the number of points to be retained based on the principle of selection discussed in Chapter 3 (Section 3.3.2).
2. Determine the range of the point distribution with two steps. The first step is to form a convex hull (see Chapter 8) of the points, and the second step is to strip the outer triangles, which is achieved by removing triangles whose outer edges are longer than a predefined value (Figure 4.8).
3. Simplify the boundary of the data points by a point-reduction algorithm (see Chapter 5) such as the Douglas–Peucker algorithm (Douglas and Peucker, 1973).
4. Compute a ranking for each point, taking value as $1/A_{VD,i}$, where $A_{VD,i}$ is the Voronoi region of the i th point.
5. Remove the point with the highest ranking from the data set. The procedure stops if the desired numbers are retained (or removed).
6. Update the new Voronoi diagram (and thus update ranking values) and consolidate the immediate neighbors (Voronoi neighbors); that is, suspend these points from the list.
7. Remove the point with the highest ranking from remaining point set. The procedure stops if the desired numbers are retained (or removed), or else repeat steps 6 and 7.
8. Release all the consolidated points for a second-round of selection. Repeat steps 5 and 7.

Figure 4.9 shows the structural simplification of a set of point features with the algorithm by Ai and Liu (2002). In this example 538 points were in the data set, and the distribution is shown in Figure 4.9a. The Voronoi diagram of this point set is shown in Figure 4.9b. After five rounds of selection, 265 points are retained, as shown in Figure 4.9c. The final result after graphic reduction is shown in Figure 4.9d.

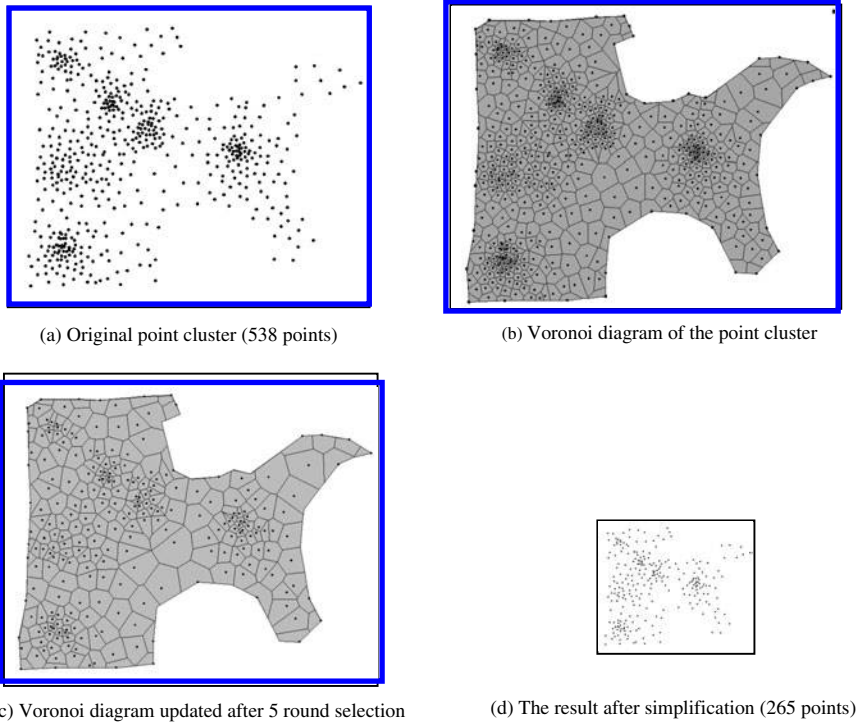


FIGURE 4.9 Structural simplification of a point set with the algorithm (Reprinted from Ai and Liu, 2002. With permission.).

4.4.2 STRUCTURAL SIMPLIFICATION CONCERNING METRIC AND THEMATIC INFORMATION

The algorithm by Ai and Liu (2002) concerns only the metric information of the points. However, as Li and Huang (2002) pointed out, four types of spatial information are contained in a cluster of spatial data (including point features):

- Statistical (positional).
- Metric.
- Thematic.
- Topologic.

That is to say, the algorithm by Ai and Liu (2002) is suitable for the structural simplification of a set of point features with the same thematic importance. However, in practice, some points are more important than others. In such a situation, Yan and Li (2004) suggest that the significance of each point be computed as follows:

$$S_i = I_i \times \frac{1}{A_{VD,i}} \quad (4.4)$$

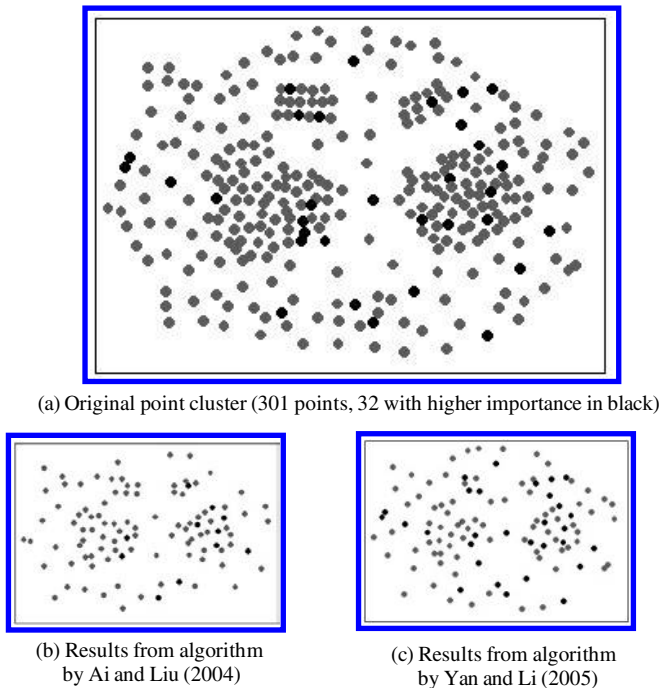


FIGURE 4.10 Comparison of results simplified by the algorithms with and without consideration of thematic information.

where S_i is the significance value of the i th point, I_i is the (thematic) importance value of the i th point, and $A_{VD,i}$ is the area of the Voronoi region of the i th point.

Figure 4.10 shows the simplification of a set of 301 points by the algorithms with and without the consideration of thematic importance. Figure 4.10a shows a dot map at 1:1:10,000 scale. Figure 4.10b shows the results simplified for representation at 1:50,000 scale without the consideration of thematic importance, and Figure 4.10c shows the result with the consideration of thematic importance. It is clear that more points with higher thematic importance are retained in Figure 4.10c.

4.5 ALGORITHMS FOR OUTLINING A SET OF POINT FEATURES: REGIONIZATION

Outlining a set of point features means grouping densely distributed point features into an area feature. McMaster and Shea (1992) referred to such an operation as *aggregation*. This is termed as a *regionization* operation in this text.

DeLucia and Black (1987) suggest a procedure as follows:

1. Generate the Voronoi diagram of the points.
2. Partition the Voronoi diagram into clusters with a given distance threshold.

3. Form Delaunay triangulation from the Voronoi regions.
4. Form the outline of each cluster by the boundary of the triangulation network.

However, the clustering algorithms described in [Section 4.2](#) seem to be the more appropriate means for forming clusters from a set of point features. Therefore, the following algorithm is suggested:

1. Form clusters from the point set with a clustering algorithm.
2. Form a Delaunay triangulation network for each cluster
3. Strip the outer triangles by removing triangles whose outer edges are longer than a predefined value.
4. Form the outline of each cluster with the boundary of the triangulation network.

The *alpha-shape* proposed by Edelsbrunner and his collaborators (Edelsbrunner and Muker, 1994) can be used to cut off the triangular sides whose length is larger than the diameter of the alpha circle, so as to make clustering effects.

REFERENCES

- Ahuja, N., Dot pattern processing using Voronoi neighbourhoods, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(3), 336–343, 1982.
- Ai, T. and Liu, Y., A method of point cluster simplification with spatial distribution properties preserved, *Acta Geodaetica et Cartographica Sinica*, 25(1), 35–41, 2002. (in Chinese)
- DeLucia, A. A. and Black, R. B., A comprehensive approach to automatic feature generalisation, *Proceedings of 13th International Cartographic Conference*, Morelia, Mexico, October 1987, 4, 169–192, 1987.
- Douglas, D. H. and Peucker, T. K., Algorithms for the reduction of the number of points required to represent a digitised line or its caricature, *Canadian Cartographer*, 10(2), 112–122, 1973.
- Edelsbrunner, H. and Muker, E. P., Three-dimensional alpha shapes, *ACM Transaction in Graphics*, 13, 43–72, 1994.
- Flewelling, D. M. and Egenhofer, M. J., Formalizing importance: parameters for settlement selection from a geographic database, in *Proceedings of Auto-Carto 11 (11th International Conference on Automated Cartography)*, Minneapolis, 1993, pp. 167–175.
- Guo, R. Z., *Spatial Analysis*, Wuhan Technical University of Surveying and Mapping Press, Wuhan, China, 1997. (in Chinese)
- Jiang, B., Spatial clustering for mining knowledge in support of generalization processes in GIS, in *ICA Workshop on Generalization and Multiple Representation*, 20–21/08/2004, Leicester, 2004.
- Langran, C. E. and Poiker, T. K., Integration of name selection and name placement, in *Proceedings of 2nd International Symposium on Spatial Data Handling*, 1986, pp. 50–64.
- Li, Z. L. and Huang, P., Quantitative measures for spatial information of maps, *International Journal of Geographical Information Systems*, 16(7), 699–709, 2002.
- MacQueen, J. B., Some methods for classification and analysis of multivariate observations, *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1, University of California Press, Berkeley, 1967 pp. 281–297.

- McMaster, R. B. and Shea, K. S., *Generalization in Digital Cartography*, Association of American Geographers, Washington, DC, 1992.
- van Kreveld, M., van Oostrum, R., and Snoeyink, J., Efficient settlement selection for interactive display, in *Proceedings of Auto Carto 12*, Bethesda, MD, 1995, pp. 287–296.
- Tou, J. T. and González, R. C., *Pattern Recognition Principles*, Addison–Wesley Publishing Co., Reading, MA, 1974.
- Yan, H. W and Li, Z. L., A Voronoi-based algorithm for point cluster generalization, in *Proceedings of 11th International Conference on Geometry and Graphics*, August 2004, Guangzhou, 2004. (CD-ROM)
- Yukio, S., Cluster perception in the distribution of point objects, *Cartographica*, 34(1), 49–61, 1997.