Research Article

A raster-based method for computing Voronoi diagrams of spatial objects using dynamic distance transformation

C. LI†§, J. CHEN†‡ and Z. LI§

†National Key Lab for Geomatic Engineering (Liesmars), Wuhan Technical University of Surveying and Mapping, 39 Luoyu Road, Wuhan, China
‡National Geomatics Center, 1 Baishengcun, Zhizhuyuan, Beijing, China
email: jchen@gps.ceic.gov.cn
§Department of Land Surveying and Geo-Informatics, Hong Kong Polytechnic University, Kowloon, Hong Kong
email: Lszlli@polyu.edu.hk

(Received 29 January 1996; accepted 13 August 1998)

Abstract. The Voronoi diagram has been suggested as an appropriate model for the description of relations between spatial objects (Gold 1992) and it is the only possible solution (which is currently available) to dynamic (measurement-based) GIS (Wright and Goodchild 1997). A Voronoi diagram can be computed either in vector mode or in raster mode. Most existing methods are vector-based. However, vector-based methods are complex for line sets and area sets. To overcome this serious deficiency, attempts have also been made to use raster-based methods. This paper describes a raster-based method for computing Voronoi diagrams of spatial objects (including points, line and areas) using dynamic distance transformation achieved by the dilation operator in mathematical morphology. Furthermore, an extension is presented to accommodate complex spatial objects.

1. Introduction

The Voronoi diagram is also known as a Thiessen diagram, Wigner-Seitz cells or Dirichlet tessellation. The actual term used seems to vary among different scientific disciplines although the basic idea is common to them all, i.e. the description of boundaries of the so-called 'region of influence' or 'spatial proximity' for each of a set of spatial data points, as originally used by the climatologist A. H. Thiessen in 1911 (Brassel and Reif 1979).

It has been suggested that the neighbour relations defined by Voronoi diagrams are more closely related to human perception than other data models (Gold 1992). In fact, the Voronoi diagram might be considered as a hybrid of both a feature-based tessellation (vector) and a space-based tessellation (raster). In other words, it can be argued that the Voronoi diagram is a more natural data model for spatial analysis than other coordinate-based data models. With the Voronoi data model, spatial operations can be simplified whether they are straightforward (e.g. closest objects) to more complex queries (e.g. adjacency). With such useful properties, increas-

International Journal of Geographical Information Science
ISSN 1365-8816 print/ISSN 1362-3087 online © 1999 Taylor & Francis Ltd
http://www.tandf.co.uk/JNLS/gis.htm
http://www.taylorandfrancis.com/JNLS/gis.htm

ing attentions have been paid on the use of Voronoi diagram for various applications (Klein 1988, Gold 1991, 1994, Okabe *et al.* 1992, 1994, Mizutani *et al.* 1993, Gold *et al.* 1996, Yang and Gold 1996, Chen and Cui 1997, Li and Chen 1997).

Another property of interest for Voronoi diagrams is their dynamic component. Voronoi diagrams allow users to 'add or delete points without destroying the "bubble" structure of the cells' (Gold and Condal 1995). Gold and Condal (1995) present a more advanced data structure for Voronoi diagrams to 'allow points to be moved about the map in sequence while their spatial relations (i.e. neighbours) are still preserved all the time'. Such a data model is, as commented by Wright and Goodchild (1997), the only possible answer, which is currently known, to a GIS for the marine environment which has been considered as the candidate to 'captivate the public and to serve the pragmatic interests as vital as the military', while creating a vast opportunity for jobs and investments (Wright and Goodchild 1997).

Purely from the viewpoint of computational geometry, a Voronoi diagram is essentially 'a partition of the plane into N polygonal regions, each of which is associated with a given point. The region associated with a point is the locus of points closer to that point than to any other given point' (Lee and Drysdale 1981). The development of efficient and robust methods for the computation of Voronoi diagrams has been considered a challenging topic and it has attracted attentions from researchers in various communities (Green and Sibson 1977, Brassel and Reif 1979, Bowyer 1981, Lee and Drysdale 1981, Miles and Maillardet 1982, Ohya *et al.* 1984a, 1984b, Klein 1988, Masser 1988, Sugihara 1992, Okabe *et al.* 1992). Efforts have also been spent on the development of dynamic Voronoi diagrams (Zaninetti 1990, Gold and Condal 1995).

It should be clear by now that Voronoi diagram is a type of spatial data model which has become increasingly important and a lot of efforts have been spent on the development of the methods. However, most of them are vector-based. On the other hand, it has been realized (Gold 1992, Gold and Condal 1995) that vector-based methods are good only for point sets and are complicated for line and area sets although they can be approximated (e.g. Okabe *et al.* 1992). This is a serious deficiency. On the other hand, in raster mode, the spatial objects (or features) can be treated as entities and a Voronoi diagram for entities can be formed easily. Therefore, an exploration of raster-based methods seems to be the only feasible solution if a Voronoi diagram of spatial objects (including points, lines and areas) needs to be computed. Indeed, this paper aims to describe a raster-based method for the computation of a Voronoi diagram in raster mode using dynamic distance transformation via the dilation operator developed in mathematical morphology. Another reason behind the development of raster-based methods is that more and more raster data (e.g. from high resolution satellite images) is becoming available, and thus raster-based methods will definitely become increasingly attractive with further advances in computer power.

The next section introduces some basic concepts related to Voronoi diagrams and analyzes the advantages and disadvantages of vector-based methods. Section 3 will discuss the advantages of raster data handling and the solutions for computing the Voronoi diagram of point sets based upon traditional distance transformation. In §4, the methods based upon dynamic distance transformation via morphological operators will be considered. In §5, these raster-based methods are extended for the computation of Voronoi diagrams for sets from only point data to all possible data including complex objects. A comparative analysis of these raster-based methods is represented in §6.

2. Methods for Voronoi diagram computation: a review and an overview

Suppose there are N distinct points P_1, P_2, ..., P_n in the plane. Each point will have a Thiessen polygon. All of these Thiessen polygons (or Voronoi regions) together will form a pattern of packed convex polygons covering the whole plane (gaps or overlaps). This pattern is known as the Voronoi diagram of the point set (figure 1(*b*)).

From figure 1(*a*), it can be found that the Thiessen polygon of a point is formed by perpendicular bisectors of the edges of its surrounding triangles. Therefore, one natural approach is to build a triangular network first and then to derive the Voronoi diagram from the triangulated network if the relationship between Voronoi diagram and a triangular network is unique. Indeed, it has been found that the triangular network formed by the Delauney methods and the Voronoi diagram have a dual relationship (figure 2(*a*)) and this has been followed by some researchers (Rogers 1964, Watson 1981, Katajainen and Koppinen 1988, Ohya *et al.* 1984a).

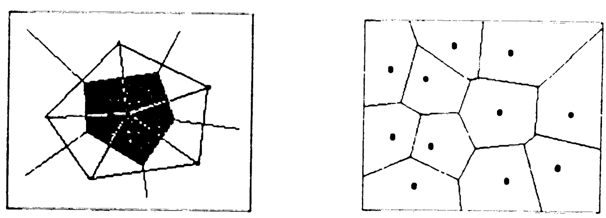
Other researchers prefer to generate the Voronoi diagram directly from data points and to derive the triangular network from the Voronoi diagram, if required. A number of such direct methods have been developed and a comparative analysis of these methods has been made (Ohya *et al.* 1984b). The incremental and divide-and-conquer methods are two typical methods, which are briefly described.

The basic idea of the incremental method is, as the name implies, to expand the Voronoi diagram incrementally, i.e. to add one point each time. First, the Voronoi diagram of three points is computed, and then a fourth point is considered and the Voronoi diagram of the four points computed. The process continues until the Voronoi diagram of N points is completed through the Voronoi diagram of N-1 points by adding the last point (Fortune 1975, Green and Sibson 1977, Bowyer 1981, Lee and Drysdale 1981, Ohya *et al.* 1984a, 1984b). The process is shown in figure 2(*b*).

With the divide-and-conquer method, the original point set is divided into disjoint subsets. A Voronoi diagram for each of these subsets is first computed. Then all the Voronoi diagrams of these subsets are combined to form a large Voronoi diagram for the whole dataset (Lee and Drysdale 1981, Ohya *et al.* 1984a, 1984b, Okabe *et al.* 1992). The process is illustrated in figure 2(*c*).

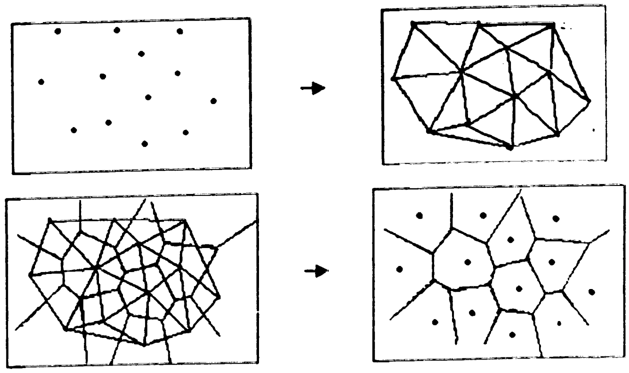
It is obvious that vector-based methods are good for point sets. For line and area sets, it seems that no efficient rigorous algorithms have been developed. Indeed, only some approximate methods, such as the so-called area sweep method of Okabe *et al.* (1992), are known to the authors. This is a serious deficiency of vector-based methods.

By contrast, in raster mode, lines and areas can be dealt with easily. In raster mode, the Voronoi diagrams for line and area sets can also be computed as easily

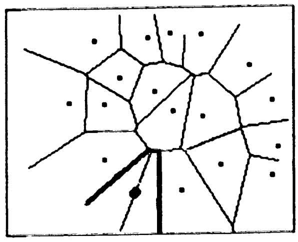


(*a*) Thiessen polygon. (*b*) Voronoi diagram.

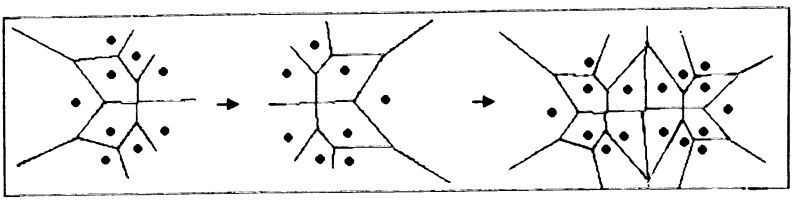
Figure 1. Thiessen Polygon (Voronoi region) and Voronoi diagram.



(*a*) Indirect generating method.



(*b*) Incremental method.



(*c*) Divide and conquer.

Figure 2. Vector-based methods for Voronoi diagram computation.

as for point sets, as shown in figure 10 (see §5.2). Another advantage of raster-based methods is that they can easily be extended to the 3-D Voronoi diagram. For these reasons, some raster-based methods have been developed and are described in the next two sections.

So far, there is very little literature purely on the computation of Voronoi diagrams in raster. Closely related to this topic is a body of work for distance transformation (Borgefors 1986) and a body of work for the computation of Delauney Triangulations (the dual of Voronoi diagrams in raster mode) (Tang 1989). Indeed, Borgefors (1986) has the potential Voronoi diagram the 'Pseudo-Dirichlet tessellation' derived from distance transformation of point sets. Tang (1989) has derived the Delauney Triangulation from these 'Pseudo-Dirichlet tessellations' ('quasi-Voronoi diagrams').

3. Methods based upon traditional distance transformation

Although the potential of computing Voronoi diagrams via distance transformation has been recognised (Borgefors 1986, Tang 1989, Okabe *et al.* 1992) and thus the idea is not original, a detailed description of methods will be given here for the following reasons: (*a*) Such information is scattered in the literature of other disciplines (but not purely on Voronoi diagrams), and so it may not be familiar to a GIS reader; (*b*) The concept of distance transformation forms a foundation for the dynamic distance transformation described in the next section; and (*c*) Existing ideas of the Voronoi diagram computation via distance transformation are all confined to point sets, but these ideas will form a basis for computing the Voronoi diagram of complex spatial objects (including point, line and areas) developed later in this paper.

3.1. *The definitions of raster distance*

As has been discussed previously, the Voronoi diagram is formed by a series of contiguous Thiessen polygons and a Thiessen polygon is computed according to the distances between points. This approach to the computation of the Voronoi diagram in vector mode is also valid in raster mode. The critical problem arising is 'how to determine the distances between points in raster mode'.

In vector mode, 'distance' means the Euclidean distance. The distance between two points $P_1(X_1, Y_1)$ and $P_2(X_2, Y_2)$ is defined as follows:

$$D(P_1, P_2)=f(X_1, X_2, Y_1, Y_2)=\sqrt{(X_1-X_2)^2+(Y_1-Y_2)^2} \qquad (1)$$

In raster mode, the coordinates are defined by integer numbers of row and column of raster pixels. Suppose there are two points at $P_1(i, j)$ and $P_2(m, n)$, then the Euclidean distance between them is defined as follows:

$$D(P_1, P_2)=f(i, j, m, n)=\sqrt{(i-m)^2+(j-n)^2} \qquad (2)$$

The unit is number of pixels. For example, if the two points are at (2,2) and (3,3), then the result is $\sqrt{2}$ ($=1.414$) pixels. This result in decimal form is inconvenient to use in raster mode and a distance in integer number is more desirable and thus normally employed. The problem arising now is 'how to find an integer number for every possible distance between two points, which is the best approximation of the Euclidean distance'. In the example given above, either 1 or 2 could be the best candidate to be used as the raster distance to approximate the Euclidean distance of $\sqrt{2}$. However, other integer numbers (e.g. 3 in the case of Chamfer 2–3 function) could also be used, depending on the definition given.

As far as the definition of raster distance is concerned, many have been suggested (Rosenfeld and Pfaltz 1968, Borgefors 1986, 1994, Melter 1987, Breu *et al.* 1995, Embrechts and Roose 1996). Initially, the concept of raster distance is directly related to the 'number of neighbours' or the 'directions of connection'. Suppose that one is travelling from the point at (2,2) to the point at (3,3), there is only one step if one is allowed to travel along the diagonals. In this case, the most appropriate raster distance between these two points is 1. This is the case with eight (8) directions of connection (i.e. left, right, up, low, upper/right, upper/left, lower/left and lower/right) or eight neighbours. A diagrammatic representation of this type of distance is given in figure 3(*b*). The shape of this diagram is like a chessboard, and so is called the 'chessboard distance'. However, if one is only allowed to travel in four directions (i.e. not along diagonals), then from (2,2), one needs to travel via either point (2,3) or (3,2) to point (3,3). There are two steps involved, and so the distance between

4	3	2	3	4
3	2	1	2	3
2	1	0	1	2
3	2	1	2	3
4	3	2	3	4

(*a*) City block.

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

(*b*) Chess board.

2	1	1	1	2
1	1	1	1	1
1	1	0	1	1
1	1	1	1	1
2	1	1	1	2

(*c*) Octagon.

8	7	6	7	8
7	4	3	4	7
6	3	0	3	6
7	4	3	4	7
8	7	6	7	8

(*d*) Chamfer 3–4.

6	5	4	5	6
5	3	2	3	5
4	2	0	2	4
5	3	2	3	5
6	5	4	5	6

(*e*) Chamfer 2–3.

Figure 3. Various definitions of raster distance.

(2,2) and (3,3) is two. The resulting distance diagram is shown in figure 3(*a*). This is what happens when one is travelling in a city and one has to travel along streets but not to go through the city blocks. Hence this is called the 'city block distance'.

It is obvious that the approximation of these two types to the Euclidean distance becomes poorer and poorer when the distance becomes larger and larger. To make such approximation better, other distances are proposed such as orthogonal, Chamfer 3–4 and Chamfer 2–3 distance (see Borgefors 1986), (figure 3(*c*–*e*)).

Recently, the efficiency of methods has been considered and parallel methods for computing the Voronoi diagram have been developed (Embrechts and Roose 1996). This method provides for any pixel a vector pointing to the closest foreground pixel instead of the distance to the closest foreground pixel. That is, it makes use of Euclidean distance instead of the concept of raster distance. The transformation is based on the propagation of distances over 4-connected neighbours.

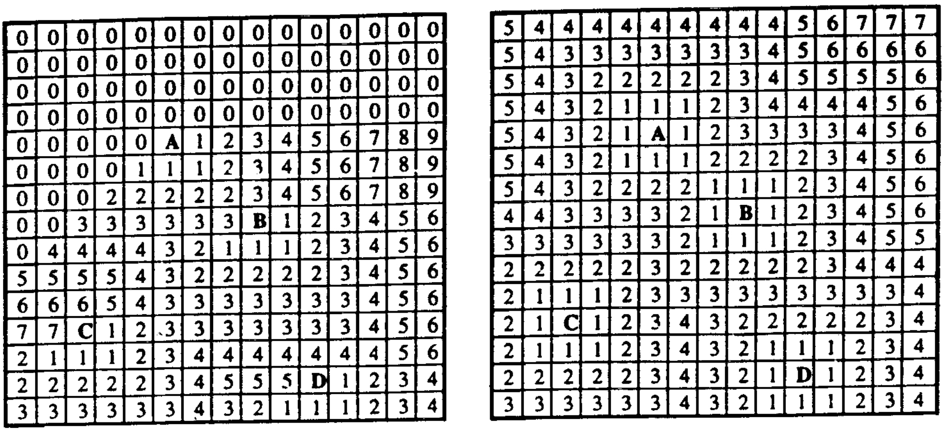
3.2. *Voronoi diagram formation from distance contours*

After a definition of raster distance is selected, one can then compute distances of every pixel to all feature points–distance transformation. As one can imagine, there could be 20 different distance values for every single pixel if there are 20 features in the point set. However, in any case, only the distance with smallest value is taken for any pixel. In practice, distance transformation is carried in two sweeps. The first sweep is from left to right and from top to bottom, and the second sweep is from right to left and from bottom to top. The process is illustrated in figure 4.

Figure 5(*a*) is an example of distance transformation of a point set. This diagram is also called the distance contour because contour lines are formed if all points with the same distance from a feature point are joined. After the distance contours of a point set is obtained, one can then easily obtain the Voronoi diagram of the point set as shown in figure 5(*b*) by joining the largest distances (highest numbers) in the distance diagram.

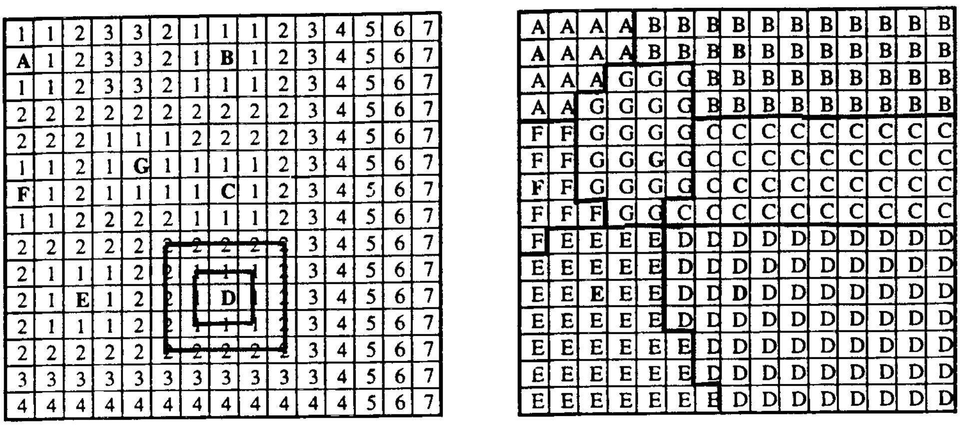
4. Methods based upon dynamic distance transformation

From figure 5 it can be found that the distance diagram consists of a number of distance contours (rings) radiated from each data point. The most distant contours of data points forms the boundaries of Thiessen Polygons, thus the Voronoi diagram.



(*a*) From left to right and from top to bottom. (*b*) From right to left and from bottom to top.

Figure 4. The process of distance transformation.



(*a*) Distance contours. (*b*) Voronoi diagram.

Figure 5. Voronoi diagram computation from Distance contours.

Therefore, the most important element to be considered in the formation of the Voronoi diagram based upon the distance transformation is the relative position of the most distant distance contour of each point, rather than the absolute distance. In fact, these relative positions of most distant contours can also be obtained through alternative approaches, i.e. systematic expansion from each data point. For such expansion, the dilation operator developed in mathematical morphology is a good tool and will be employed in this study.

4.1. *Basic morphological operators*

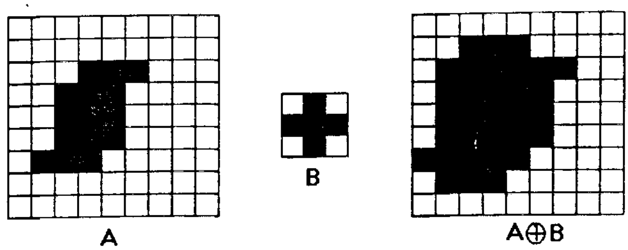
Dilation is one of the basic operators in mathematical morphology, which was developed by G. Matheron and J. Serra in the 1960s (Serra 1982), is a science of form and structure based upon set theory, topology and geometric concepts, and has found wide application in digital image processing as well as geographical information science (Su *et al.* 1997). As well as dilation another basic operator in mathematical morphology is erosion. These two basic operators are defined as follows (Serra 1982, Haralick *et al.* 1987):

$$\text{Dilation:} \quad A \oplus B = \cup_{b \in B} A_b \tag{3a}$$

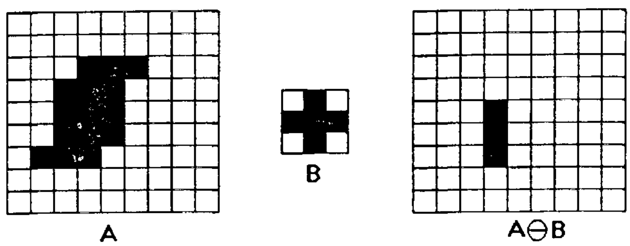
$$\text{Erosion:} \quad A \ominus B = \cap_{b \in B} A_b \tag{3b}$$

Where A is the original image with features, and B is the structuring element. Examples of dilation and erosion are given in figure 6.

In fact, structuring elements play an important role in a morphology operation. It has three basic parameters, i.e. size (e.g. 3×3), shape (e.g. square) and origin (e.g. at the geometric centre). Some of the 3×3 structure elements which are frequently used in this paper are given in figure 7.

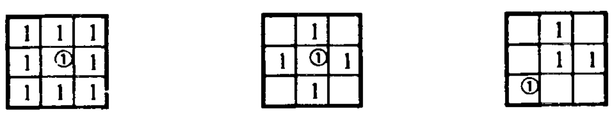


(*a*) Dilation.

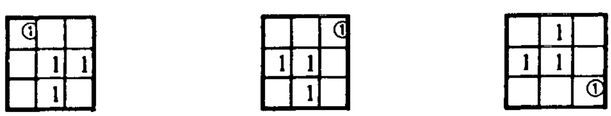


(*b*) Erosion.

Figure 6. Two basic morphological operators: dilation and erosion.



(*a*) B_0 (*b*) B_1 (*c*) ©B_2



(*d*) B_3 (*e*) B_4 (*f*) B_5

Figure 7. Some structuring elements used in this study. '①' represents the central pixel.

4.2. *Distance contour computation using dilation operator*

With the basic concepts and operators in mathematical morphology introduced, the next step is to employ an appropriate structuring element so that distance contours could be obtained using the dilation operator repeatedly. The structuring elements B_1 and B_0 as shown in figure 7 are the appropriate choices for city block distance and chess board distance, respectively. These two type of distance contours can be expressed mathematically as follows:

$$\text{City block:} \quad Y_n = Y_{n-1} \oplus B_1 \qquad (4)$$
$$Y_0 = X \oplus B_1$$

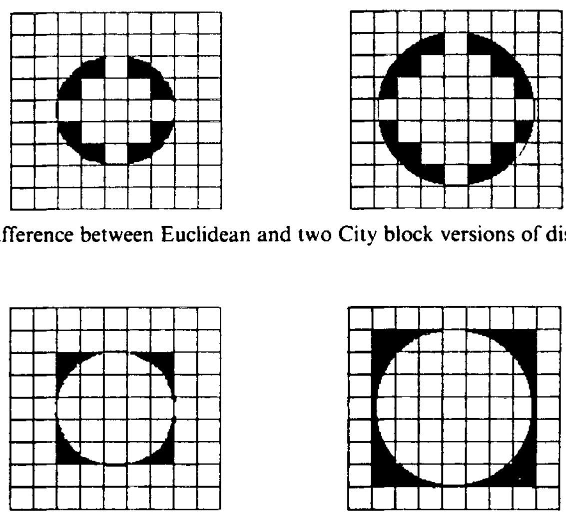
$$\text{Chess board:} \quad Y_n = Y_{n-1} \oplus B_0 \qquad (5)$$
$$Y_0 = X \oplus B_0$$

The results of the distance contours for 'chess board' and 'city block' distances based upon the dilation operator are shown in figure 8. However, as has been discussed previously, for these two types of raster distance, the approximation to Eulidean distances will become very poor if the distance becomes large. It is possible, however, to use more than one type of structuring element iteratively so that the distance contours are good approximations to the Eulidean distances.

4.3. *Dynamic distance transformation*

In general, there are three key factors during the process of computing Voronoi diagram by operators in mathematical morphology. The first is a set of appropriate structuring elements for the dilation of objects to generate distance contours, the second is a definition of the end condition, and the last is to obtain adjacency relations.

The structuring elements to be used in this study have been illustrated in figure 7. The first one is for dilation in all eight directions. The second is for dilation in the four neighbour directions. The third to the sixth are for dilation along and near the



(*a*) Difference between Euclidean and two City block versions of distance.

(*b*) Difference between Euclidean and Chess board distance.

Figure 8. Differences between Euclidean distance and various measures of distance.

diagonal directions. If these structuring elements are used properly, the combined effect of dilation will be an area very close to a circle. To achieve this goal, a comparison of the characteristics of both raster distance and a circle is needed. Many parameters might be used as conditions, including perimeters of raster size, size of raster area, etc. However, tests reveal that these parameters are only good for dilation from a single pixel (point). When a feature is an irregular area, however, the desirable shape after each dilation is not circular. Furthermore, when there are more than two features, the areas formed by dilation from these features will form common boundaries, thus it is impossible for all of them to have circular shapes. Therefore, in this study, another parameter, the characteristics of boundary lengths between two directions, of a circle will be considered.

Figure 9 shows the difference between a circle in vector space and its approximation in raster space. In figure 9(*b*) points P_1, P_2, P_3, are the intersection points between the circle and directions OP_1, OP_2, OP_3, respectively. Their coordinates are $(r,0)$, $(\sqrt{3}/2r, \frac{1}{2}r)$, and $(\sqrt{2}/2r, \sqrt{2}/2r)$, respectively. The boundary length of OP_1 is $NS = 2 \times NP_1$, MN for OP_2 and $2MP_3(=MT)$ for OP_3. The actual values are:

$$2P_1N = 2(\sqrt{5}-2)r \tag{6a}$$

$$2P_3M = 2(\sqrt{10}-3)r \tag{6b}$$

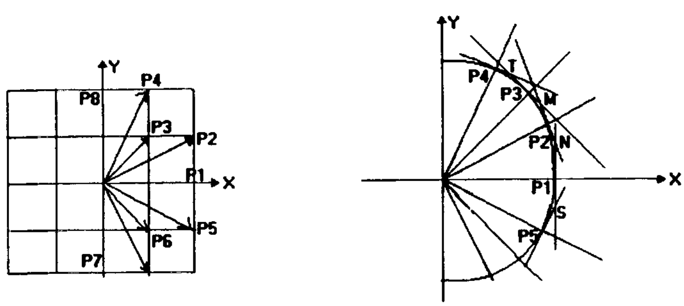
$$MN = (\sqrt{10}+\sqrt{5}-5)r \tag{6c}$$

Equation (6) is re-arranged into equation (7):

$$\frac{2P_1N}{2(\sqrt{5}-2)} = \frac{2P_3M}{2(\sqrt{10}-3)} = \frac{MN}{\sqrt{10}+\sqrt{5}-5} = r \tag{7}$$

Equation (7) describes vector space. If raster distances approximate Euclidean distances this equation should also hold for raster space. The next step therefore is to apply this condition to the raster distances.

Suppose a pixel, e.g. O in figure 9a, is dilated by B_0K_1 times, by B_1K_2 times and by $(B_2 \oplus B_3 \oplus B_4 \oplus B_5)\ K_3$ times. The boundary length in OP_1, OP_2, OP_3, directions, translated into Euclidean distances, are $2K_1$, $\sqrt{5}K_3$, $\sqrt{2}(K_2+K_3)$ because (*a*) the unit length of three directions in terms of Euclidean distance are 1, $\sqrt{5}$, $\sqrt{2}$ respectively; (*b*) every dilation by B_0 will produce two unit boundary length for OP_1 direction, every dilation by B_1 will produce one unit boundary length for OP_3



(*a*) Raster space. (*b*) Vector space.

Figure 9. Boundary length in both vector and raster space.

direction and every dilation by $(B_2 \oplus B_3 \oplus B_4 \oplus B_5)$ will produce one unit boundary length for both OP_3 and OP_2 directions. Therefore, if the approximation of Euclidean distances using raster distance were perfect, then the following equation would hold:

$$\frac{2K_1}{2(\sqrt{5}-2)} = \frac{\sqrt{2}(K_2+K_3)}{2(\sqrt{10}-3)} = \frac{\sqrt{5}\,K_3}{\sqrt{10}+\sqrt{5}-5} \tag{8}$$

Let $a = \frac{1}{\sqrt{5}-2}$, $b = \frac{1}{\sqrt{2}(\sqrt{10}-3)}$, $c = \frac{1}{\sqrt{2}+1-\sqrt{5}}$, then equation (8) will be rewritten as follows:

$$aK_1 = b(K_2+K_3) = cK_3 \tag{9}$$

However, in practice, such an equation will never hold because the approximation of Euclidean distances using raster distance will never be perfect. On the other hand, as one can imagine, very good approximation has been achieved if the differences between these two terms are very small. Therefore, the sum of the differences between any two of the three terms in Equation (9) should be kept to a minimum if the best approximation of Euclidean distances is to be achieved. In other words, the following equation can be used as a condition for the selection of the next structuring element in the dynamic distance transformation using dilation operators:

$$f(K_1, K_2, K_3) = |aK_1 - b(K_2+K_3)| + |aK_1 - cK_3| + |b(K_2+K_3) - cK_3| = \min \tag{10}$$

4.4. *Voronoi diagram formation*

Voronoi diagram formation via dynamic distance transformation means (*a*) use of equation (10) as a condition to select the most appropriate structuring element for the next dilation; (*b*) to record the boundary pixels which have the same distance to two or more objects; (*c*) record the adjacent relationship between objects. The procedure is as follows (where X_0 is the original image and X_i results after the ith dilation):

Step 1: Initials: $B_0 = \{0\}$, $i = 0$, $K_1 = 0$, $K_2 = 0$, $K_3 = 0$

Step 2: Compute three possible difference values for equation (10)
$d_1 = f(K_1 + 1, K_2, K_3)$
$d_2 = f(K_1, K_2 + 1, K_3)$
$d_3 = f(K_1, K_2, K_3 + 1)$
$d_{min} = Min\{d_1, d_2, d_3\}$

Step 3: Selection of most appropriate structuring element
If $d_{min} = d_1$,
then $K_1 = K_1 + 1$, $i = i + 1$, $X_i = X_{i-1} \oplus B_0$. If one pixel is dilated by two or more objects, record both the boundary pixel and the adjacent relations of them.
If $d_{min} = d_2$,
then $K_2 = K_2 + 1$, $i = i + 1$, $X_i = X_{i-1} \oplus B_1$. If one pixel is dilated by two or more objects, record both the boundary pixel and the adjacent relations of them.
If $d_{min} = d_3$,
then $K_3 = K_3 + 1$, $i = i + 1$, $X_i = X_{i-1} \oplus B_2 \oplus B_3 \oplus B_4 \oplus B_5$. If one pixel is

dilated by two or more objects, record both the boundary pixel and the adjacent relations of them.

Step 4: Check whether non-feature pixels have all been reached by the dilation process. If 'yes', then END. Otherwise go to step 2.

5. Voronoi diagram of spatial objects

So far, two types of raster-based methods for the computation of Voronoi diagrams for point sets have been described. This section will discuss how to extend these methods to other types of spatial objects, i.e. lines and areas, so that the Voronoi diagrams of spatial objects (including points, lines and areas) can be computed easily. To achieve this goal, two tasks need to be carried out. The first task is to extend related concepts and the second is to modify the method.

5.1. *Extension of concepts*

First of all, the definition of the Voronoi diagram itself needs to be extended and a discussion of that extension has been given by Okabe *et al.* (1994). For the point sets, if the plane was divided into polygonal areas (Thiessen polygons) such that each area contains only one point and the part of the plane is nearer to that point than others, then such a division of the plane is called a Voronoi diagram of points. Similarly, by replacing 'point' with 'object', the definition of the Voronoi diagram can be extended to spatial objects. Therefore, a Voronoi diagram of spatial objects is a partition of the plane into N polygonal regions, each of which is associated with a given object and the region associated with an object is the locus of points closer to that object than to any other given object.

With this new definition of the Voronoi diagram, the problem has been transformed to the definition of 'locus of points closer to that object'. In fact, this 'locus' can be obtained by connecting the perpendicular bi-sectors of all distances between objects. Now the problem has been transformed to the definition of 'distance between objects'. Indeed, this is the toughest part which almost prevents the Voronoi diagram of spatial object sets (including points, line and areas) from being computed by vector-based methods. However, this can be easily achieved in raster mode.

Another extension which needs to be considered is that complex objects should also be added to the set of spatial objects in addition to points, lines and areas.

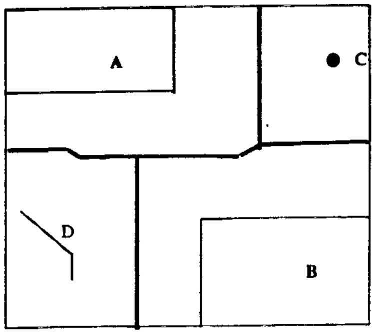
5.2. *Modification of implementation*

To accommodate area features, some modifications of the implementation are necessary. The key modification is to add a pre-processing step to convert area objects which are described by boundaries into solid area entities. Figure 10(*a*) shows the distance contours of points (point C), lines (line D) and areas (areas A and B), and figure 10(*b*) shows the Voronoi diagram of spatial objects, formed from figure 10(*a*).

If spatial objects overlap, the situation is more complicated. In order to satisfy the definition of the Voronoi diagram, the area features that have overlapping areas may be represented by their boundaries, instead of area entities. An example of a Voronoi diagram for objects with overlapping areas is given in figure 11, where two rectangular areas A and B are shown. The Voronoi region of object B is the composite of the shaded area and the overlapping part. It means that the influence area of B also covers part of the influence area of A and vice versa.

A	A	A	A	A	A	A	1	2	3	2	2	2	2	2
A	A	A	A	A	A	A	1	2	3	2	1	1	1	2
A	A	A	A	A	A	A	1	2	3	2	1	C	1	2
A	A	A	A	A	A	A	1	2	3	2	1	1	1	2
1	1	1	1	1	1	1	1	2	3	2	2	2	2	2
2	2	2	2	2	2	2	2	2	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	4	4	4	4	4
2	2	2	3	3	3	3	3	3	3	3	3	3	3	3
1	1	2	2	3	2	2	2	2	2	2	2	2	2	2
D	1	1	2	2	2	1	1	1	1	1	1	1	1	1
1	D	1	1	2	2	1	B	B	B	B	B	B	B	B
1	1	D	1	2	2	1	B	B	B	B	B	B	B	B
2	1	D	1	2	2	1	B	B	B	B	B	B	B	B
2	1	1	1	2	2	1	B	B	B	B	B	B	B	B
2	2	2	2	2	2	1	B	B	B	B	B	B	B	B

(*a*) Distance contours of spatial objects.



(*b*) Voronoi diagram of spatial objects.

Figure 10. Voronoi diagram of spatial objects.

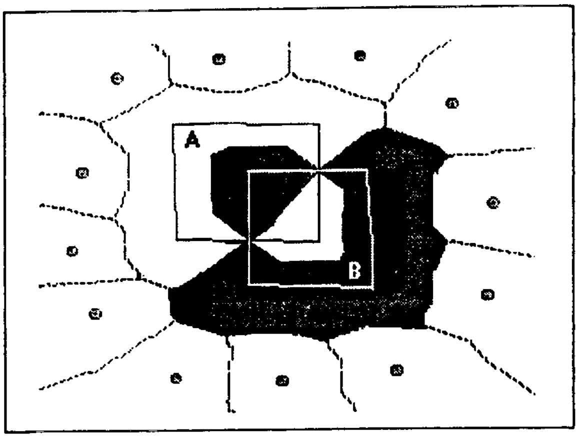


Figure 11. Voronoi diagram of overlapping objects.

6. A comparative analysis

As has been discussed previously, a raster distance is an integer (in pixel units) used to approximate the Euclidean distance. The distortion (or difference) in distance between raster distance and Euclidean distance, which is a measure of the quality of the distance function, will be analysed in this section, and efficiency will also be discussed.

6.1. *Comparison of distance distortion*

According to Borgefors (1986, 1994), the maximum difference between Euclidean distance and raster distance is a linear function of M, the raster distance in horizontal direction and are listed in table 1. A diagrammatic representation is given in figure 12.

From table 1 and figure 12, it can be noted that the distance distortion of all traditional functions is a linear function of the distance itself. It means that distortion will become larger and larger when a pixel is farther away from the data point. The City block and chess board distance functions will always have greatest distortion. On the other hand, the distortion resulting from the dynamic distance transformation is more consistent with the maximum distortion about 1 pixel. Only when the

Table 1. Distance distortion between Euclidean distance and various raster distances.

M	City block	Chess board	Chamfer 2–3	Chamfer 3–4	Dynamic function
1	−0.414	0.414	0.134	0.081	0.414
3	−1.242	1.242	0.402	0.243	0.764
5	−2.070	2.070	0.670	0.405	0.877
7	−2.898	2.898	0.938	0.567	0.810
9	−3.726	3.726	1.206	0.729	0.938
11	−4.554	4.554	1.474	0.891	0.704
13	−5.382	5.382	1.742	1.053	0.601
15	−6.210	6.210	2.01	1.215	1.108
17	−7.038	7.038	2.278	1.377	0.875
19	−7.866	7.866	2.546	1.539	1.000
21	−8.694	8.694	2.814	1.701	0.875
23	−9.522	9.522	3.082	1.863	0.770
25	−10.350	10.350	3.350	2.025	0.962
27	−11.178	11.178	3.618	2.187	1.058
29	−12.006	12.006	3.886	2.349	0.833
31	−12.834	12.834	4.154	2.511	1.033

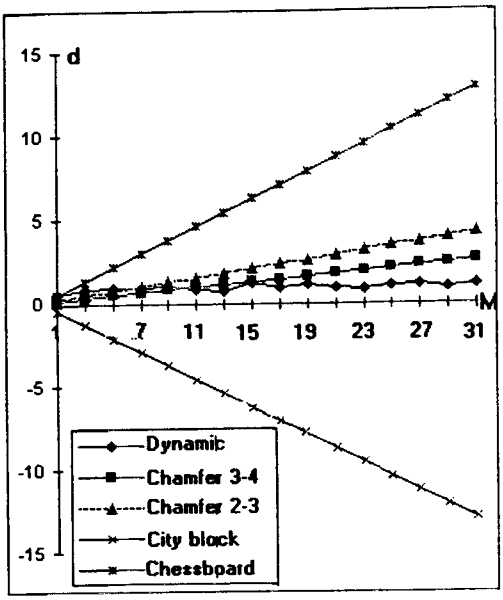


Figure 12. Distance distortion by various distance transformation functions.

distances are small do the Chamfer 3–4 or Chamfer 2–3 functions perform better than the dynamic distance transformation.

6.2. *Comparison of efficiency*

Generally speaking, the computing time has little to do with the number of spatial objects, but largely depends on the size of grid. Table 2 lists the time in seconds for different sizes of grid with different methods. The computer used here is a 96 Hz 586 PC with 8 m RAM. The dynamic distance function takes a little more time than the others.

Table 2. Efficiency of various raster distance transformations (in second).

The size of grid	Number of objects	City-block	Chess board	Octagon	Chamfer 2–3	Chamfer 3–4	Dynamic function
30 × 30	50	8	12	15	19	19	23
	300	8	12	15	19	19	23
50 × 50	50	17	25	29	31	32	38
	300	17	25	29	31	32	38
80 × 80	50	21	31	37	39	39	42
	300	21	31	37	39	39	42
100 × 100	50	25	37	49	55	57	59
	300	25	37	49	55	57	59
300 × 300	50	87	114	134	163	167	198
	300	87	114	134	163	167	198

7. Conclusions

Voronoi diagrams have found wide application in various fields and so much attention has been paid to its computation. However, most of the existing methods are vector-based for point sets. Although, theoretically speaking, it is also possible to compute Voronoi diagrams in vector mode for composite objects (i.e. with points, lines and areas), although it is difficult in practice. Indeed, only some approximate methods have been designed. On the other hand, spatial objects include lines and areas, in addition to point features. This has motivated the authors to develop a new method, which can easily handle spatial objects (include points, lines and areas).

In this paper, two types of raster-based methods have been described: one based upon the traditional distance transformation and the other upon dynamic distance transformation. As has been noted previously, the fact that the Voronoi diagram of point sets can be computed via distance transformation has been recognized by researchers (Borgefors 1986, Tang 1989) although they didn't discuss actual implementation. The second method described in this paper is based upon the so-called dynamic distance transformation, which is an original development by the authors. Another important contribution in this paper is the extension of these methods so that they can be used for the computation of Voronoi diagrams for spatial object sets with lines, areas and even complex objects, instead of only point sets.

A comparative analysis of some of the distance transformation functions is made using the following parameters as criteria: distance distortion and efficiency. It is found that the distance distortion of all traditional functions is a linear function of the distance itself and the maximum distortion resulting from the dynamic distance transformation is about 1 pixel. Therefore, the dynamic distance transformation seems to be the most robust method. As far as efficiency is concerned, the dynamic distance transformation takes only a little more time. For efficiency, no comparison with vector-based methods is made because, as has been pointed out previously, there is still a difficulty for vector-based methods, dealing with complex objects (e.g. overlapping objects).

With the methods described in this paper, Voronoi diagrams for spatial object sets (including point, lines, areas, complex objects and overlapping objects) can be computed almost as easily as for points only. The next step in this research is to develop a methodology for computing the 3-D Voronoi diagram, and to use the

Voronoi diagram for the description of spatial relations, for the management of dynamic data, for data updating, for spatial query, etc.

Acknowledgment

J. Chen and C. Li would like to thank the National Science Foundation of China for its financial support of this work under research grant No. 49471059. Z. Li and C. Li would like to thank the Hong Kong Polytechnic University for its support (A/C Code 351/489).

References

BORGEFORS, G., 1986, Distance transformations in digital images. *Computer Vision, Graphics and Image Processing*, **34**, 344–371.

BORGEFORS, G., 1994, Applications using distance transformations. In *Aspects of Visual Form Processing*, edited by C. L. Arcelli, P. Corella, and G. S. di Boja, (Singapore; River Edge, NJ: World Scientific).

BOWYER, A., 1981, Computing Dirichlet tessellations. *The Computer Journal*, **24**, 162–166.

BRASSEL, K., and REIF, D., 1979, Procedure to generate Thiessen polygon. *Geographical Analysis*, **11**, 289–303.

BREU, H., GIL, J., and KIRKPATRICK, D., and WERMAN, M., 1995, Linear time Euclidean distance transformation methods. *IEEE Transactions Pattern Analysis and Machine Intelligence*, **17**, 529–533.

CHEN, J., and CUI, B., 1997, Using Voronoi approach of developing topological functions in MapInfo. *Journal of Wuhan Technical University of Surveying and Mapping*, **22**, 195–200. (in Chinese).

EMBRECHTS, H., and ROOSE, D., 1996, A parallel Euclidean distance transformation method. *Computer Vision and Image Understanding*, **63**, 15–26.

FORTUNE, S., 1975, A sweep line method for Voronoi diagram. *Methodica*, **2**, 153–174.

GOLD, C. M., 1991, Problems with handling spatial data—the Voronoi approach. *CISM Journal*, **45**, 65–80.

GOLD, C. M., 1992, The meaning of 'Neighbour'. In *Theories and Methods of Spatio-Temporal reasoning in Geographic Space*, Lecture Notes in Computing Science, No. 39, edited by A. Frank, I. Campari, and U. Formentini (Berlin: Springer-Verlag), pp. 220–235.

GOLD, C. M., 1994, A review of potential applications of Voronoi methods in geomatics. *Proceedings of the Canadian Conference on GIS, 4–10 June, Ottawa*, pp. 1647–1656.

GOLD, C. M., and CONDAL, A. R., 1995, A spatial data structure integrating GIS and simulation in a marine environment. *Marine Geodesy*, **18**, 213–228.

GOLD, C. M., NANTEL, J., and YANG, W., 1996, Outside-in: an alternative approach to forest map digitising. *International Journal of Geographical Information Systems*, **10**, 291–310.

GOODCHILD, M., 1987, A spatial analytical perspective on geographical information system. *International Journal of Geographical Information System*, **1**, 327–334.

GREEN, P. J., and SIBSON, R., 1977, Computing Dirichlet tessellations in the plane. *The Computer Journal*, **21**, 168–173.

HARALICK, R., STERNBERG, S., and ZHUANG, X., 1987, Image analysis using mathematical morphology. *IEEE Transactions of Pattern Analysis and Machine Intelligence*. **9**, 532–550.

HILDITCH, J., and RUTOVITZ, D., 1969, Chromosome recognition. *Annals of New York Academy of Sciences*, **157**, 339–364.

KATAJAINEN, J., and KOPPINEN, M., 1988, Computing Delaunay triangulation by merging buckets in quadtree order. *Fundamental Informatica*, **XI**, 275–288.

KLEIN, R., 1988, *Abstract Voronoi Diagrams and Their Applications*. Lecture Notes in Computer Science 333 (Berlin: Springer-Verlag).

LEE, D. T., and DRYSDALE, R. L., 1981, Generalization of Voronoi diagram in the plane. *SIAM Journal of Computing*, **10**, 73–87.

LI, C., and CHEN, J., 1997, The nine-intersection model for describing spatial relations. *Journal of Wuhan Technical University Surveying and Mapping*, **22**, 207–211, (in Chinese).

MASSER, I., 1988, The regional research laboratory initiative: a progress report, *International Journal of Geographical Information System*, **2**, 11–22.

MELTER, R. A., 1987, Some characterisations of city block distance. *Pattern Recognition Letters*, **6**, 235–240.

MILES, R. E., and MAILLARDET, R. J., 1982, The basic structures of Voronöi and generalized Voronoi polygons. *Journal of Applied Probability*, **19A**, 97–111.

MIZUTANI, N., WATANABE, T., YOSHIDA, Y., and OKABE, N., 1993, Extraction of contour lines by identification of neighbour relationships on a Voronoi line graph. *Systems and Computers in Japan*, **24**, 57–68.

MONTANARI, U., 1968, A method for obtaining skeletons using a quasi-Euclidean distance. *Journal of the Association of Computing Machinery*, **15**, 600–624.

OHYA, T., IRI, M., and MUROTA, K., 1984a, A fast Voronoi-diagram method with quaternary tree bucketing. *Information Processing Letters*, **18**, 227–231.

OHYA, T., IRI, M., and MUROTA, K., 1984b, Improvement of the incremental method for the Voronoi diagram with computational comparison of various methods. *Journal of the Operations Research Society of Japan*, **27**, 306–336.

OKABE, A., BOOTS, B., and SUGIHARA, K., 1992, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams* (Chichester: John Wiley).

OKABE, A., BOOTS, B., and SUGIHARA, K., 1994, Nearest neighbourhood operations with generalized Voronoi diagrams. *International Journal of Geographical Information System*, **8**, 43–71.

OPENSHAW, S., 1987, An automated geographical analysis system. *Environment and Planning A*, **19**, 431–436.

ROGERS, C. A., 1964, *Packing and Covering* (Cambridge: Cambridge University Press).

ROSENFELD, A., and PFALTZ, J., 1968, Distance functions on digital pictures. *Pattern Recongition*, **1**, 33–61.

SERRA, J., 1982, *Image Processing and mathematical Morphology* (New York: Academic Press).

SHAMOS, M. I., and HOEY, D., 1975, Closet-point problem. In *Proceedings of the 16th Annual IEEE Symposium on Foundation of Computer Science*, Washington, pp. 151–162.

SU, B., LI, Z., LODWICK, G., and MÜLLER, J.-C., 1997, Algebraic models for the aggregation of area features based upon morphological operators. *International Journal of Geographical Information System*, **11**, 233–246.

SUGIHARA, K., 1992, A simple method for avoiding numerical errors and degeneracy in Voronoi diagram computation. *IEICE Trans Fundamentals*, **E75-A**, 468–477.

TANG, L., 1989, Surface modelling and visualization based upon digital image processing techniques. In *Optical 3-D Measurement Techniques*, edited by A. Grun, and H. K. Kahmen (Karlsruhe: Wichmann Verlag), pp. 317–325.

THIESSEN, A. H., 1911, Precipitation averages for large areas. *Monthly Weather Review*, **39**, 1082–1084.

WATSON, D. F., 1981, Computing the N-dimensional Delaunay tessellation with applications to Voronoi diagram. *The Computer Journal*, **24**, 167–172.

WRIGHT, D. J., and GOODCHILD, M. F., 1997, Data from the deep: implications for GIS community. *International Journal of Geographical Information Science*, **11**, 523–528.

YANG, W. P., and GOLD, C. M., 1996, The VMO-tree: a dynamic spatial data model for maps. In *Proceedings of International Conference on Geographical Information System in Urban, Regional and Environmental Planning, Pythagorion, Island of Samos, Greece, 19–21 April*, Pychagorion, Island of Samos, Greece, pp. 260–270.

ZANINETTI, L., 1990, Dynamic Voronoi tessellation. *Astronomy and Astrophysics*, **233**, 293–300.