The Relative Neighbourhood Graph of a Finite Planar Set

Godfried T. Toussaint

Published in *Pattern Recognition*, Vol. 12, 1980, pp. 261-268.
Winner of an *Outstanding Paper Award* given by the *Pattern Recognition Society.*

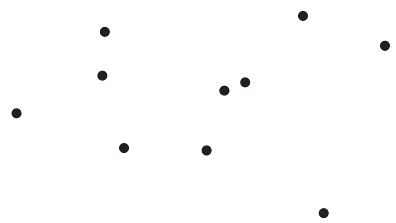
Abstract

The relative neighbourhood graph (RNG) of a set of n points on the plane is defined. The ability of the RNG to extract a perceptually meaningful structure from the set of points is briefly discussed and compared to that of two other graph structures: the minimal spanning tree (MST) and the Delaunay (Voronoi) triangulation (DT). It is shown that the RNG is a superset of the MST and a subset of the DT. Two algorithms for obtaining the RNG of n points on the plane are presented. One algorithm runs in $O(n^2)$ time and the other runs in $O(n^3)$ time but works also for the d-dimensional case. Finally, several open problems concerning the RNG in several areas such as geometric complexity, computational perception, and geometric probability, are outlined.

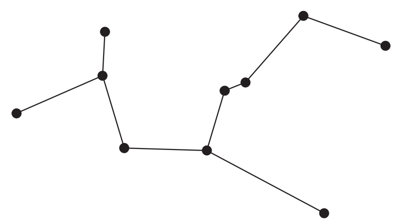
1 Introduction

In many problems in pattern recognition, such as clustering [18] and computational approaches to perception [13], one is given a set of points on the plane and it is desired to find some structure among the points in the form of edges connecting a subset of the pairs of points. In the clustering problem we would like to have an algorithm that joins two points if the two points belong to the same cluster of points [18]. In computational perception we would like an algorithm to join pairs of points such that the final graph obtained is perceptually meaningful in some sense [13].

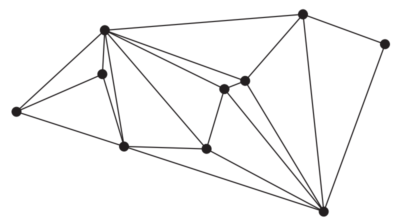
Two graphs that we will be concerned with in this paper are the minimal spanning tree (MST) and the Delaunay triangulation (DT). In the minimal spanning tree of a set of points P edges are formed by joining pairs of points to form a tree that spans P such that the sum of the Euclidean edge lengths is less than the sum for any other spanning tree. As an example consider the set of points on the plane in Fig. 1(a). The MST of that set is given in Fig. 1(b). Observation of Fig. 1(b) suggests that the MST is a form of 'skeleton' of the data points. In addition, there is evidence that it has perceptual significance [18, 13], and it has been used extensively as a tool for cluster analysis [18, 7].



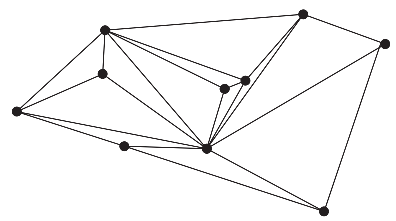
(a) A set of points on the plane.



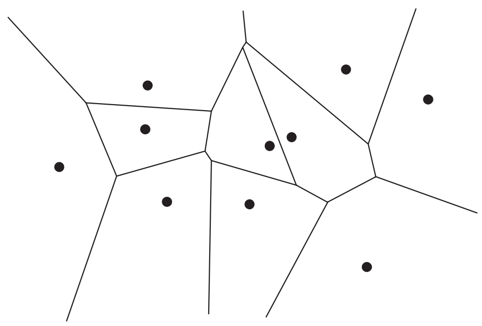
(b) Minimal spanning tree of points in (a).



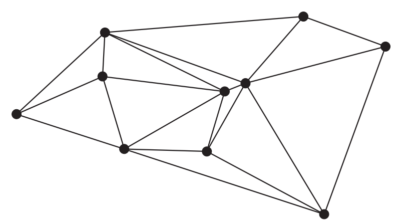
(c)



(d)



(e) Voronoi diagram of points in (a).



(f) Delaunay triangulation of points in (a).

Figure 1:

A triangulation is more difficult to define formally. A set P, of points in the plane, is triangulated by a subset, T, of the straight line segments whose endpoints are in P, if T is a maximal subset such that the line segments in T intersect only at their end points. Intuitively, edges are formed such that as many triangles as possible are created without crossing lines. A set of points can be triangulated in many differnent ways. Two triangulations of the set of points given in Fig. 1(a) are illustrated in Figs 1(c) and (d). Triangulations have received a lot of attention in the past and have many areas of application [9]. The triangulation we are concerned with here is the Delaunay triangulation, or alternatively the locally equiangular triangulation [15]. To define the Delaunay triangulation we first introduce a structure known as the Voronoi diagram [5] (also Dirichlet tessellation).

Let $P = \{p_1, p_2, \ldots, p_n\}$ denote n distinct points on the plane. The Voronoi diagram partitions the plane into disjoint regions or tiles (also Thiessen polygons) such that the tile of p_i is the set T_i defined by

$$T_i = \{x : d(x, p_i) < d(x, p_j \ \forall j \neq i\},$$

where d denotes Euclidean distance. The Voronoi diagram finds wide application in various fields such as geography [11] and has been used to find elegant and efficient algorithms for many problems in computational geometry by Shamos and Hoey [14]. The Voronoi diagram for the set of points of Fig. 1(a) is given in Fig. 1(e). The Delaunay triangulation is now easily explained. Two points p_i and p_j are joined by an edge if, and only if, their corresponding tiles T_i and T_j share a side. If this operation is carried out on the Voronoi diagram of Fig. 1(e) one obtains the Delaunay triangulation illustrated in Fig. 1(f).

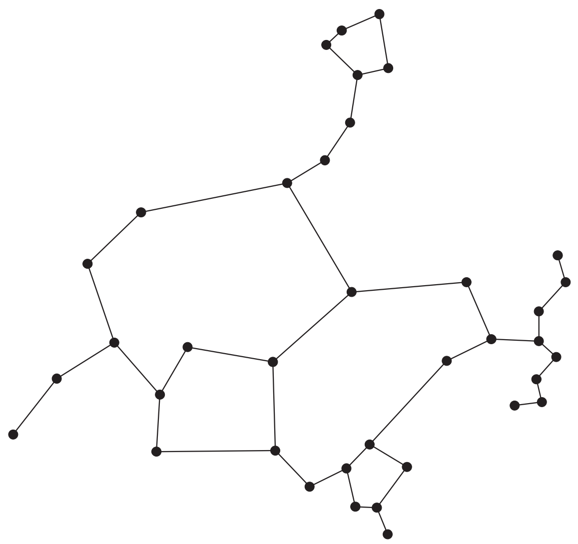
In this paper we investigate a graph which we call the relative neighbourhood graph (RNG) of a finite planar set. It is based on the notion of "relatively close" neighbours defined by Lankford [8] and has not received any attention in the literature on computational geometry or pattern recognition. The RNG is defined in section 2 and its ability to extract perceptually relevant structures from sets of points is compared to that of the MST and the DT. In section 3 the RNG is related to both the MST and the DT. Two algorithms for finding the RNG, of n points on the plane, are given in section 4. One algorithm runs in $O(n^3)$ time while the other requires $O(n^2)$ time. Finally, some open problems are discussed in section 5.

2 The Relative Neighbourhood Graph

Consider again a set P of n distinct points on the plane: $P = \{p_1, p_2, \ldots, p_n\}$. There are many possible ways of defining whether or not two points p_i and p_j are neighbours of each other. Several definitions are considered [18, 13, 8, 10]. Lankford [8] defines two points p_i and p_j as being "relatively close" if $d(p_i, p_j) \leq \max[d(p_i, p_k), d(p_j, p_k)] \forall k = 1, \ldots, n, \ k \neq i, j$. Actually, Lankford uses '$<$' rather than '$\leq$' in his definition. The difference is essentially that with this minor modification, in a degenerate situation such as three points lying equidistant from each other, all three points are considered relative neighbours of each other, whereas with only '$<$' in the definition none of the three points are relative neighbours of each other. Intuitively, the definition states that two points are relative neighbours if they are at least as close to each other as they are to any other point. The relative neighbourhood graph is obtained by connecting an edge between points p_i and p_j for all $i, j = 1, \ldots, n, \ i \neq j$ if, and only if, p_i and p_j are relative neighbours. Figure 2 illustrates a set of points and its RNG.

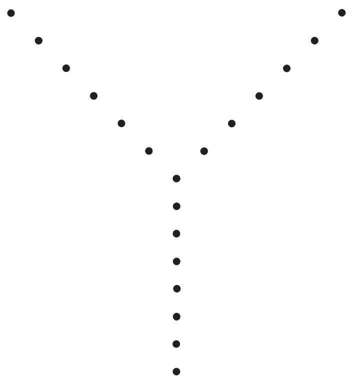
It is interesting to compare the perceptual relevance of the RNG to that of the MST and the DT. To this end consider a set of points for which the MST extracts a perceptually relevant structure. Such a set and its MST are illustrated in Fig. 3. Clearly the DT of the set of points of Fig. 3(a) fails as a perceptually meaningful structure. However, we can find sets of points where the reverse is true. Figure 4(a) shows a set of points with its DT in

(a) A set of points.

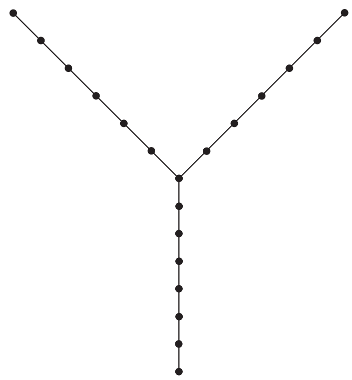


(b) Relative neighbourhood graph of points in (a).

Figure 2:



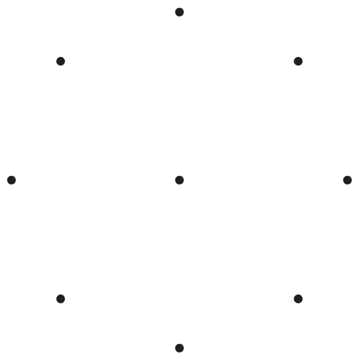
(a) A set of points on the plane.



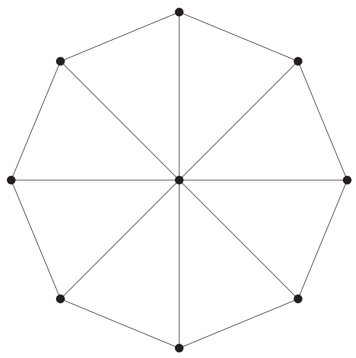
(b) Minimal spanning tree of points in (a).

Figure 3:

Fig. 4(b). In this case the DT appears to be more perceptually relevant than the MST. However, the RNG works well in both cases: in Fig. 3 the RNG agrees with the MST, and in Fig. 4 the RNG agrees with the DT. It is further possible to find very simple and basic sets of points where both the MST and the DT fail to extract the perceptually relevant structure. A strong case, in fact, for the RNG is a set of four points arranged in a square or rectangular fashion as in Fig. 5(a). Perceptually, the set of points immediately suggests a rectangle [1]. Here both the MST [Fig. 5(b)] and the DT [Fig. 5(c)] fail and only the RNG

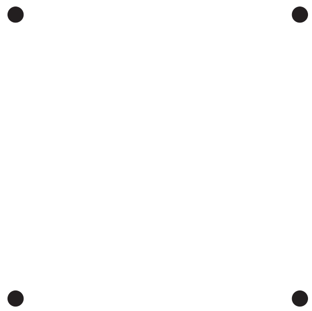


(a) Set of points on the plane.

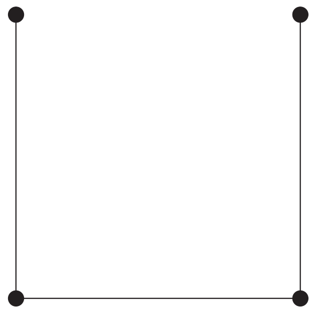


(b) Delaunay triangulation of points in (a).

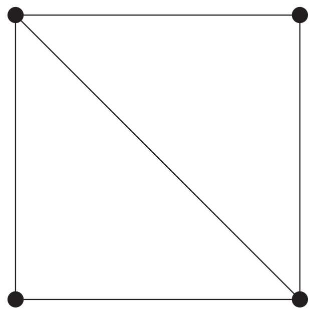
Figure 4:



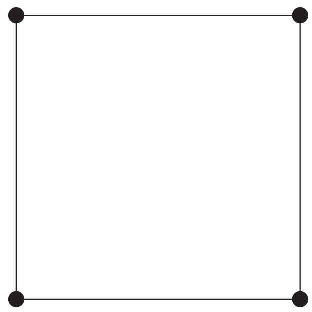
(a) A set of four points on the plane.



(b) Minimal spanning tree of points in (a).



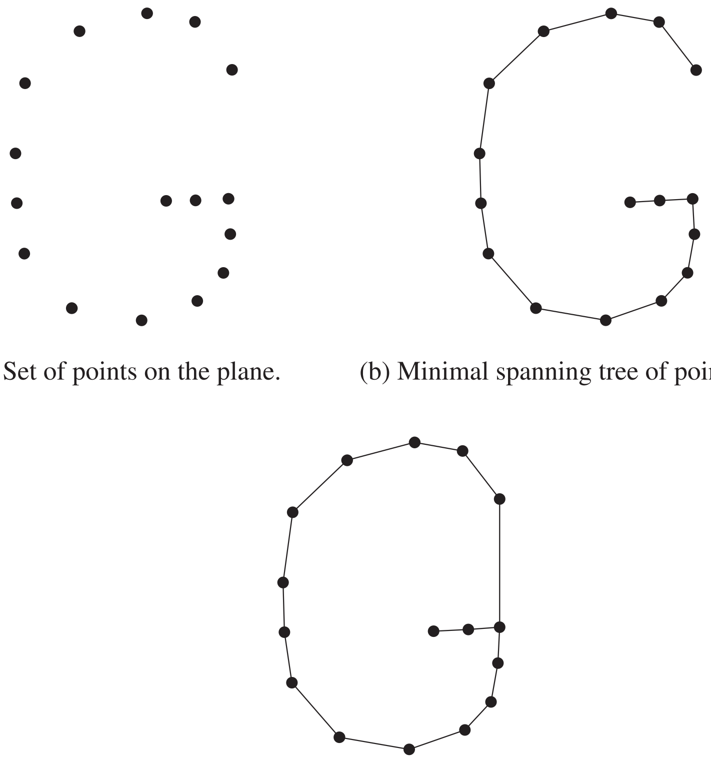
(c) Delaunay triangulation of points in (a).



(d) Relative neighbourhood graph of points in (a).

Figure 5:

yields the rectangle. This suggests that the RNG may have more perceptual significance than the MST considered in [18], at least for arbitrary sets of points. That this is not always the case can be shown by constructing a set of points that have important *symbolic* meaning such as those in Fig. 6(a) which clearly indicate the letter “G”. The MST [Fig. 6(b)] yields



(a) Set of points on the plane. (b) Minimal spanning tree of points in (a).

(c) Relative neighbourhood graph of points in (a).

Figure 6:

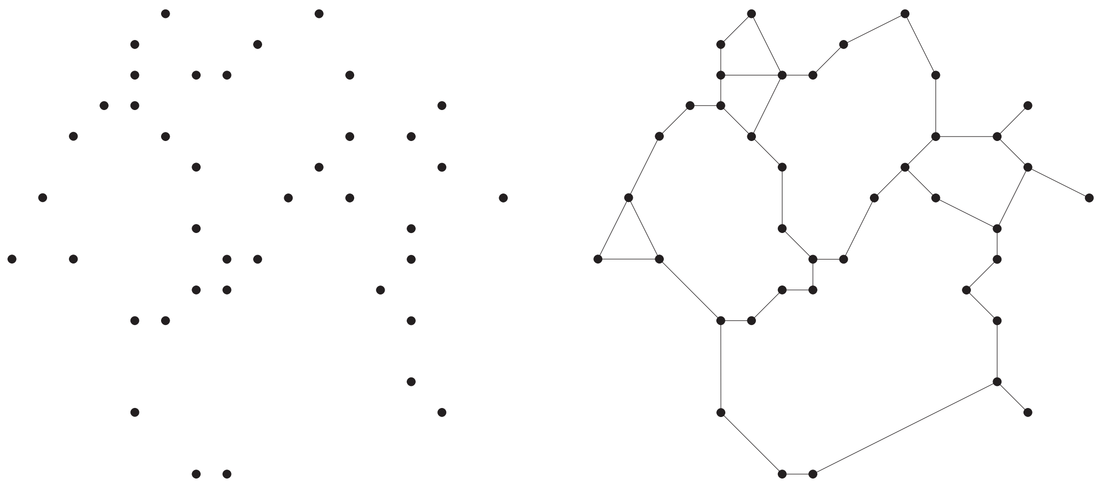
a “G” but the RNG [Fig. 6(c)] does not. In short, the RNG is more adaptive to the data in the sense that it imposes less structure than either the MST or the DT.

Figures 7, 8 and 9 illustrate further sets of points and their RNGs. Unlike the set of points in Fig. 2 which consists of points in a somewhat arbitrary position, the points in Figs 7, 8, and 9 are more symmetrically arranged. As a result many more triangles and squares appear. Recall from Figs 3 and 4 that the RNG sometimes behaves like an MST and at other times like a DT. These figures, in particular Fig. 2, suggest that the RNG is a structure “lying somewhere in between” the MST and the DT. In the next section we show, in fact, that the RNG is a superset of the MST and a subset of the DT.

3 Two Theorems Concerning The RNG

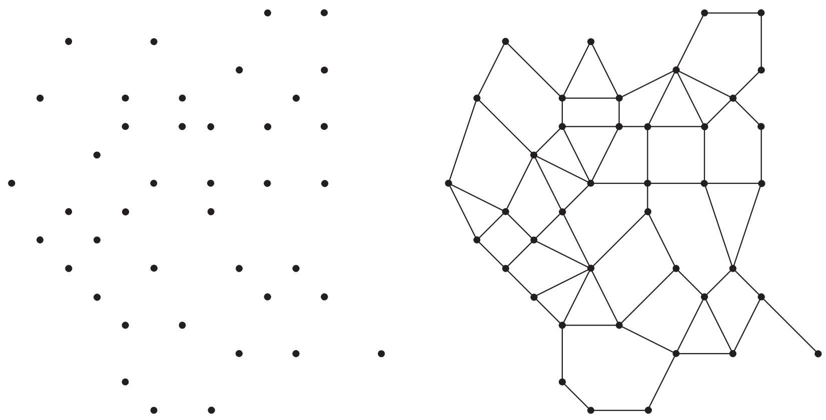
Theorem 3.1 *The relative neighbourhood graph is a superset of the minimal spanning tree.*

Proof: Consider two points a and b as in Fig. 10. Let R denote the interior of the intersec-



(a) (b)

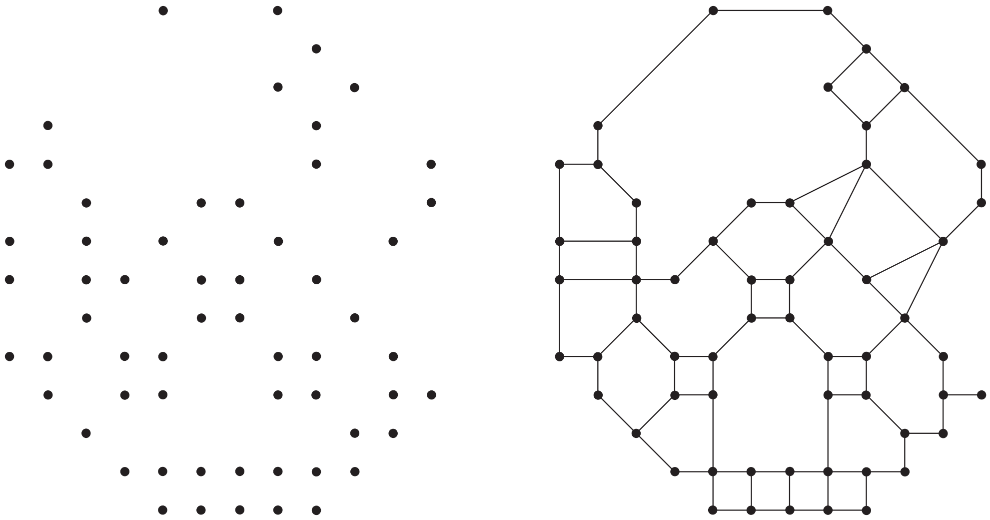
Figure 7:



(a) (b)

Figure 8:

tion of the two circles with centres at a and b and let B denote the boundary of R. Let c be a third point in a non-degenerate position, i.e. $c \notin B$. (If c lies in B, $d(a,b) = d(c,b)$ or $d(c,a)$ and the MST in non-unique). Thus c must lie either in R or in $\overline{R}$, the complement of $R \cup B$. If $c \in R$ then $d(a,c) < d(a,b)$ and $d(b,c) < d(a,b)$, and it follows that $ab \notin$ MST. Thus a necessary but not a sufficient condition for $ab \in$ MST is that all other points lie in $\overline{R}$. However, this is a necessary and sufficient condition for $ab \in$ RNG. Therefore, MST $\in$ RNG. ■



(a) (b)

Figure 9:

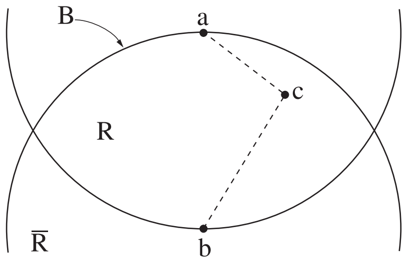


Figure 10:

Theorem 3.2 *The relative neighbourhood graph is a subset of the Delaunay triangulation.*

Proof: Consider again two arbitrary points a and b from a finite set of points as in Fig. 10. We must show that if $ab \in$ RNG then $ab \in$ DT. Assume that $ab \in$ RNG. It follows from the definition that no other points lie in R. We now ask under what conditions the tiles T_a and T_b of the Voronoi diagram do not share a side. Consider two additional points, say c and d, that lie somewhere on B as in Fig. 11, where the Voronoi diagram is indicated with dashed lines. Observe that the Voronoi lines associated with ca and ad are the perpendicular bisectors of the lines joining ca and ad, respectively. Furthermore, since a, c, and d all lie on a circle with its center at b, it follows that no matter how close c and d are to a, T_a and T_b will always share a side of non-zero length. Furthermore, if either c or d lie in $\overline{R}$, their

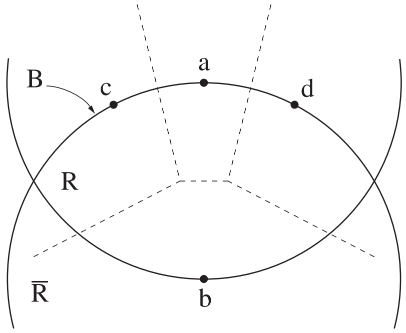


Figure 11:

length of the shared side of T_a and T_b will tend to increase. Therefore, when no other points lie in R the tiles T_a and T_b always share a side. Therefore, $ab \in$ DT. ■

Let N_e denote the number of edges contained in the RNG of n points on the plane.

Corollary 3.3 *The number of edges in the RNG of n points on the plane is bounded by $n - 1 \leq N_e \leq 3n - 6$, and is thus $O(n)$.*

Proof: By definition, the MST contains $n - 1$ edges. Since the MST is a subset of the RNG (Theorem 3.1) it follows that $N_e \geq n-1$, where the equality holds when RNG = MST. It has been shown that a graph without crossings can contain at most $3n - 6$ edges and that this bound is achieved when all faces of the graph are triangles [6]. Therefore the DT contains at most $3n - 6$ edges. Since the RNG is a subset of the DT (Theorem 3.2) it follows that $N_e \leq 3n - 6$ where the equality holds when RNG = DT. Thus $O(n) \leq N_e \leq O(n)$ and we have $N_e = O(n)$. ■

4 Two algorithms for finding the RNG

Let n points be given by their Cartesian coordinates $p_1(x_1, y_1), p_2(x_2, y_2), \ldots, p_n(x_n, y_n)$. It is required to find their RNG. In this section we give two algorithms for finding the RNG. The first, RNG-1, is the obvious and naïve algorithm which has complexity $O(n^3)$. The second, RNG-2, uses the knowledge from Theorem 3.2 and has complexity $O(n^2)$.

Algorithm RNG-1

Step 1. Compute the distance between all pairs of points $d(p_i, p_j)$ $i, j = 1, \ldots, n,\ i \neq j$.

Step 2. For each pair of points (p_i, p_j) compute $d^k_{\max} = \max\{d(p_k, p_i), d(p_k, p_j)\}$ for $k = 1, \ldots, n,\ k \neq i,\ k \neq j$.

Step 3. For each pair of points (p_i, p_j) search for a value of $d^k_{\max}$ that is smaller than $d(p_i, p_j)$. If such a point is not found, an edge is formed between p_i and p_j.

Step 1 of this algorithm requires $O(n^2)$ operations to yield $O(n^2)$ pairs. Furthermore, for each of these pairs steps 2 and 3 each require $O(n)$ operations. Hence the overall complexity of this algorithm is $O(n^3)$.

Algorithm RNG-2

Step 1. Compute the Voronoi diagram of the set of points.

Step 2. Obtain the Delaunay triangulation from the Voronoi diagram.

Step 3. For each pair of points (p_i, p_j), associated with an edge of the DT, compute $d^k_{\max} = \max\{d(p_k, p_i), d(p_k, p_j)\}$ for $k = 1, \ldots, n,\ k \neq i,\ k \neq j$.

Step 4. Same as Step 3 of algorithm RNG-1, with edges of the DT only.

The Voronoi diagram (Step 1) can be computed in $O(n \log n)$ worst-case running time [14]. Since there are $O(n)$ edges in the Voronoi diagram [14] and each yields an edge in its dual, the DT, Step 2 requires $O(n)$ operations. In Step 3, for each pair of points considered, $O(n)$ operations are required, and since only $O(n)$ pairs need be investigated this step runs in $O(n^2)$ time. Since Step 4 is $O(n^2)$ the algorithm's complexity is dominated by Steps 3 and 4 and is thus of $O(n^2)$.

It is clear that the reason why algorithm RNG-2 requires $O(n^2)$ running time is that for each of the $O(n)$ edges of the DT, $O(n)$ points are tested for proximity. It follows that if for each edge of the DT only a small fraction of the points were tested for proximity the algorithm should run faster. Such a modification of algorithm RNG-2 is possible using the 'cell' or 'bin' approach [17]. First, the region in which the n points lie is divided into a $\sqrt{n} \times \sqrt{n}$ grid of n bins and each point is 'placed' in a bin using the floor function. Thus each bin has a pointer to a list of its points. Each edge of the DT now specifies a small group of cells, as a function of the edge length, which in turn identifies the points to be tested. It is conjectured that for some 'nice' distributions of the points this modification of algorithm RNG-2 will run in $O(n)$ expected time.

An idea that also comes to mind for restricting the number of points tested for each edge of the DT is to test only those points that are Delaunay neighbours of the two vertices of the edge being considered. It is also conjectured that for some 'nice' distributions this algorithm will run in $O(n)$ expected time. Unfortunately the algorithm does not always work. It is easy for the reader to construct an example with five points to convince himself that this algorithm does not yield the RNG. Nevertheless, it can be considered as an approximate algorithm for the RNG.

5 Open problems and concluding remarks

One of the main goals of this paper is to introduce the RNG to the literature on pattern recognition, perception, and computational geometry, and to point out that the RNG opens a host of problems in several areas of research. In this section we will outline some of the more interesting open problems concerning the RNG.

(a) *Non-Euclidean case.* In this paper we assumed that the weight, associated with an edge of the complete graph of the set of points, was equal to the Euclidean distance between the vertex points of the edge. In general the weights could take on arbitrary values and we would then have a non-Euclidean RNG. Algorithm 1 would still work in this case and thus we have an $O(n^3)$ algorithm for the non-Euclidean RNG. It remains an open problem whether we can do better than this.

(b) *d-Dimensional case.* In this paper only the two-dimensional problem was considered. It is clear that algorithm 1 generalizes to the d-dimensional case. Thus we have an $O(n^3)$ algorithm for arbitrary fixed dimensions. It is an open problem whether we can do better than this in three and higher dimensions.

(c) *Is there an $O(n \log n)$ algorithm?* Alternatively, find a lower bound on the problem.

(d) Several techniques have been found recently for making algorithms run in linear expected time [2, 16, 3]. Can such algorithms be found for the RNG problem?

(e) In this paper the RNG was related to two other graph structures: the MST and the DT. It would be interesting to find relationships between the RNG and other graph structures such as the minimum weight triangulation and those considered in [13] and [8].

(f) Another class of problems relates to the statistical properties of the RNG. Such problems include: what is the expected number of circuits in the RNG of n points when the n points are randomly chosen from a given distribution? What is the expected length of the RNG? Theorem 3.1 partially answers the last question with the help of existing results for the MST. Let $E\{\mathrm{MST}\}$ and $E\{\mathrm{RNG}\}$ denote the expected length of the MST and RNG, respectively. Gilbert [4] and Roberts [12] showed that for n points distributed according to a Poisson process in the unit square, in the limit as $n \to \infty$, $E\{\mathrm{MST}\} = k\sqrt{n}$ where $0.5 \leq k \leq 0.707$. From Theorem 3.1 it follows that $E\{\mathrm{RNG}\} \geq E\{\mathrm{MST}\}$ and therefore for the Poisson process above $E\{\mathrm{RNG}\} \geq k\sqrt{n}$.

(g) The RNG also opens new problems in geometrical probability. In general, if n points are generated randomly according to a given distribution, we can ask what is the probability that a certain event occurs in the RNG of the points. For example, let $n = 4$. In this case the RNG can assume only a few easy recognizable classes of forms illustrated in Fig. 12: the 'toad', the 'starfish', the 'wiggly snake', the 'convex snake', and last and least degenerate 'kite' (degenerate because this form only occurs when the sides joining the tail have equal length which happens with probability zero). Are there more 'toads' than 'starfish'?

(h) A final open problem has to do with the relevance of the RNG to the perceptual process. Section 2 suggests that the RNG is a powerful model of low-level visual processes involved in the perception of certain dot patterns. It would be interesting to compare the RNG to other structures such as those used in [18] and [13] and to determine its limits as a

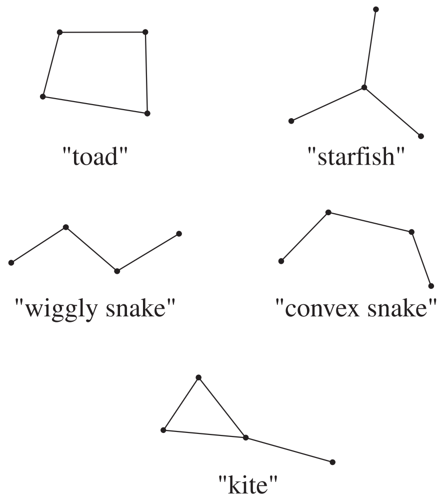


Figure 12:

model of human perceptual mechanisms.

Summary

The problem of extracting the perceptually meaningful resources from a dot pattern or finite set of points on the plane is considered. Two relevant graph-theoretical structures are reviewed: the minimal spanning tree (MST) and the Delaunay (Voronoi) triangulation (DT), and their limitations for solving the above problem are illustrated with examples. A graph, termed the relative neighbourhood graph (RNG), based on the notion of relative neighbours proposed by Lankford in a geographical context, is proposed as an improvement to the MST in solving the problem of extracting the perceptually relevant structure from a dot pattern. It is shown that the RNG is a supergraph of the MST and a subgraph of the DT. Two algorithms for obtaining the RNG of n points on the plane are presented. One algorithm runs in $O(n^2)$ time and the other in $O(n^3)$ time but works for the d-dimensional case. Furthermore, a modification is proposed that should make the algorithms much faster on average, and it is conjectured that for n points drawn randomly and independently from a uniform distribution in the unit square the latter algorithm will run in $O(n)$ time. Finally, several open problems concerning the RNG in several areas such as geometric complexity, computational perception, and geometric probability, are outlined.

Acknowledgments

The author is indebted to David Avis and Duncan McCallum for several helpful discussions during the course of this research. Ernesto Correa obtained some of the figures using a

CalComp plotter. I am also grateful to Luc Devroye for reading a preliminary version of this paper and for his helpful comments.

References

[1] R. Arnheim. *Art and Visual Perception.* University of California Press, 1974. p. 54.

[2] J.L. Bentley and M. Shamos. Divide and conquer for linear expected time. *Inf. Process. Lett.*, 7:87–91, 1978.

[3] L. Devroye. A note on finding convex hulls via maximal vectors. Manuscript, 1979. McGill University.

[4] E.N. Gilbert. Random minimal trees. *SIAM J. appl. Math.*, 13:376–387, 1965.

[5] P.J. Green and R. Sibson. Computing Dirichlet tesselations in the plane. *Comput. J.*, 21:168–173, 1978.

[6] F. Harary. *Graph Theory.* Addison Wesley, Reading, MA, 1972.

[7] R.A. Jarvis. Shared near neighbour maximal spanning trees for cluster analysis. In *Proc. 4th. Intl. Joint Conf. on Pattern Recognition*, pages 308–313, Kyoto, Japan, 1978.

[8] P.M. Lankford. Regionalization: theory and alternative algorithms. *Geogrl Anal.*, 1:196–212, 1969.

[9] B.A. Lewis and J.S. Robinson. Triangulation of planar regions with applications. *Comput. J.*, 21:324–332, 1978.

[10] J.F. O'Callaghan. An alternative definition for neighbourhood of a point. *IEEE Trans. Comput.*, C-24:1121–1125, 1975.

[11] D. Rhynsburger. Analytic delineation of Thiessen polygons. *Geogrl Anal.*, 5:133–144, 1973.

[12] F.D.K. Roberts. Random minimal trees. *Biometrika*, 55:255–258, 1968.

[13] B. Rosenberg and D.J. Langridge. A computational view of perception. *Perception*, 2:415–424, 1973.

[14] M.I. Shamos and D. Hoey. Closest point problems. In *Proc. 16th Ann. Symp. on the Foundations of Computer Science*, pages 151–162. IEEE, 1975.

[15] R. Sibon. Locally equiangular triangulations. *Comput. J.*, 21:243–245, 1978.

[16] G.T. Toussaint, S.G. Akl, and L.P. Devroye. Efficient convex hull algorithms for points in two and more dimensions. Technical Report SOCS 78.5, McGill University, 1978.

[17] B.W. Weide. *Statistical methods in algorithm design and analysis*. PhD thesis, Department of Computer Science, Carnegie-Mellon University, August 1978.

[18] C.T. Zahn. Graph-theoretical methods for detecting and describing gestalt clusters. *IEEE Trans. Comput.*, C-20:68–86, 1971.