Background

November 25, 2010

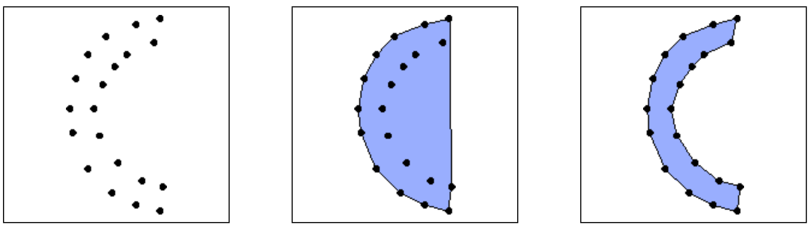
0.1 Footprints

There is a fairly large body of work about the generation of footprints, publications from as early as 1973 ([10]) presenting a variety of different algorithms to create representational shapes from dot patterns. Amongst this there are surprisingly few that examine the footprints created in a comparative fashion. Also conspicuous by its absence is a systematic approach to determining the quality of the produced footprint, Galton [7] makes significant inroads in to both determining how 'good' a footprint is and why this is difficult to judge.

The rest of this section consists of analysis of some of the existing literature in chronological order.

Jarvis [10] presents an algorithm, since called the 'Jarvis March' to generate the convex hull of a dot pattern. The convex hull is almost a base level of footprint algorithm, it is easily computable and has distinct mathematical properties. Importantly the convex hull is unique for any particular dot pattern. This paper was amongst the first to give an efficient algorithm for its computation and is such amongs the first to attempt to provide a representational shape for a dot pattern.

The convex hull is not without its problems as a representation.



(a) Point set (b) Convex Hull (c) Non-Convex Footprint

Figure 1: When a convex hull is inappropriate

As can be seen in Fig.1(b) the convex hull can potentially lose information about the pattern, whereas Fig.1(c) may be a better approximation of the underlying data. **COMMENT:** [*From here to cosit paper reference is basically plaigarising myself, need to rewrite*]An algorithm capable of reaching a better fit representation is a non-trivial problem and one of the earliest, and much-referenced, papers on the subject is by Edelsbrunner et al. [6]. The method produces straight-line graphs called α-shapes, obtained from a generalisation of the convex hull. For a set S the convex hull can be considered to be the intersection of all closed half-planes that contain all the points of S. The α-*hull* is obtained by using closed discs of radius $1/\alpha$ instead of half-planes; the α-shape is derived from this in a straightforward way. The authors do not discuss any principled way to choose the appropriate α for the type of shape required.

Chaudhuri et al. [3] present two methods for generating a footprint, called the *external shape*, from a dot pattern. Although they use the term 'dot pattern' they make no distinction between points and dots. For the first method, a grid of squares of side-length s is drawn on the plane, and the union of all grid-squares containing at least one of the dots is returned as the footprint, called the *s-shape*. For the r-shape they inscribe a disc of radius r round each dot, and draw an edge connecting any pair of dots whose discs intersect in a point not contained in any of the other discs. These edges provide an outline which, in our terms, may be regarded as the boundary of the footprint. As with the α-shape, no principles are given for selecting appropriate values of r or s.

Garai and Chaudhuri [9] propose a 'split and merge' method for generating footprints. This method starts from the convex hull and attempts to refine it to a shape more closely resembling what they refer to as the *underlying shape*. The method consists of three separate algorithms (four if the convex hull algorithm is included): *splitting*, *isolation*, and *merging*. This is one of the few algorithms that provides a way of *aiming* for a particular shape without having to re-run the algorithm with different parameters, so long as the user is able to identify a desired maximum area or number of sides just from a cursory examination of the dot pattern. Again the authors say little about the quality or type of footprint they generate.

Alani et al. [1] developed the *Dynamic Spatial Approximation Method* (DSAM). This system takes in both the dot pattern of the region to be found and the dot pattern of the area known to exist outside the region. It builds a Voronoi diagram based on these coordinates and takes the union of all the cells which contain an 'interior' point as its footprint. This work pays more attention than many in the area to the quality of footprint produced; this can be assessed in terms of how closely the region found fits the expected region. The existence of a contextually determined target shape differentiates this paper from others in the field.

Arampatzis et al. [2] follow on from Alani et al. [1]. However, they adapt DSAM to use Delaunay triangulations in conjunction with a system for finding point locations using web queries. They call this adaptation *the recolouring algorithm* and use it to generate boundaries for imprecise regions. Much like the DSAM this system has a target shape and, as such, this paper has more analysis of the footprint found than much of the field.

Galton and Duckham [8] propose two methods for finding footprints. The first method is a generalisation of the Jarvis March ('gift-wrapping') algorithm for convex hulls. The idea behind the Jarvis March is simple. From an origin point outside the dot set a radial half-line is swung in an arbitrary direction until it meets one of the dots. This dot is made the new origin point from which a radius is swung in the same direction as before until it meets another dot. This is repeated until the first dot is encountered again; the sequence of dots encountered in this way form the vertices of the convex hull. Dots are removed from consideration if they have already been marked as being on the convex hull or if they lie within the area enclosed by the dots encountered so far. The 'Swinging Arm' algorithm is similar except that it uses a line-segment of some predetermined length instead of a half-line. The second method starts with the Delaunay triangulation and successively removes the longest external edge, subject to constraints of maintaining connectedness and regularity, until either some predetermined minimum length is reached, or no more edges can be removed. The authors note that there can be no uniquely 'optimal' footprint when the application context is considered to be general. The paper proposes nine criteria which may be used for evaluating footprint algorithms with respect to different application contexts, although little is said about any actual applications.

Moreira and Santos [11] present a 'Concave Hull' algorithm. Like the Swinging Arm, Concave Hull is also derived from the Jarvis March algorithm, its difference being that it always selects the next vertex from the k nearest neighbours of the current vertex. This is the crux of the algorithm's effectiveness: by having a non-contextual integer as the variable that restrains the hull algorithm, they have a default base value from which they can run the algorithm (i.e. $k = 3$); if this fails to produce a footprint that satisfies the criteria (having no intersecting lines and containing all the points) then the algorithm is run with increasing values of k till such a footprint is created. Like most of the other authors they pay little attention to the quality of the footprint in relation to any application type, though they do mention the criteria given in [8]. Like the split and merge method [9], the Concave Hull algorithm requires some pre-processing of dots, using the Shared Nearest Neighbour (SNN) algorithm to determine any separable groupings in the dot pattern prior to running the algorithm. Like Garai and Chaudhuri they do not take account of this pre-processing algorithm in determining the computational complexity of their own.

Duckham et al. [4] provide a fuller account of the Delaunay-based method introduced in [8], now called the χ-algorithm. This paper includes a discussion of the footprint's properties, and how these are directly tied to the method by which it is created. More attention is paid to the choice of the length parameter l. There are practical limits on l for any triangulation (if it is too large then no lines will be removed, if it is too small too many will be removed) and consequently l can be normalised. Duckham *et al.* propose using this normalised parameter (λp) to find a starting value which should achieve what they call a *characteristic shape* for many, if not all, dot patterns. While they conclude that there is no λp that always produces a "good" characterization, the fact that they spend time considering this is unusual within the field. Unlike Moreira and Santos [11] and Garai and Chaudhuri [9], Duckham *et al.* do not discount the pre-processing (in this case computing the Delaunay triangulation and sorting the edges) when determining the complexity of the algorithm.

Galton [7], instead of proposing an algorithm, searches for objective criteria for evaluating the acceptability of any proposed footprint in relation to the 'perceived' shape of a dot pattern. The paper notes that in most of the published work, "while lip-service is generally paid to the fact that there is no objective definition of such a 'perceived shape', little is said about how to verify this, or indeed, about exactly what it means". Restricting attention to footprints in the form of *polygonal hulls*, simple polygons having vertices selected from the dot pattern, all the other dots being within the interior, the paper presents evidence that while a dot pattern may have several equally acceptable perceived shapes, they all represent optimal or near-optimal compromises between the conflicting goals of simultaneously minimising both the area and the perimeter of the hull.

Dupenois and Galton [5], suggests a method for classifying the footprints. Unlike Galton [7] it does not look at their 'fitness' but approaches the subject from a desire to be able to describe algorithms by the types of footprints they can create. The paper notes that the context in which the algorithm is being used determines the type of footprint that is satisfactory. With this in mind it proposes a method of using the application specific knowledge to limit the choice of algorithms for any particular user requirement. The classification bears some similarity to the set of criteria proposed by Galton and Duckham [8] for evaluating the footprints produced by different algorithms.

0.2 Dot Patterns

Examining dot patterns has generally been within the field of geospatial information. However, if we move away from real-world phenomena, we can imagine that any data that can be represented on a 2-dimensional plane (e.g., classification data, multi-objective optimisation) can be viewed as a dot-pattern. This leads to a daunting amount of possible literature to examine so the analysis given is by no mean exhaustive but should serve to give a general overview.

O'Sullivan and Unwin [12] gives a good description of the treatement of dot patterns (called point patterns) from a geographic standpoint. The chapter begins with noting that point patterns frequently occur in GIS and gives the examples of crime or death hot-spot analysis.

Within GIS events have a set of criteria that must be satisfied for them to be considered point patterns [Quote: 0.2].

Quote 0.1 Point Pattern Requirements

O'Sullivan and Unwin [12]

1. The pattern should be *mapped on the plane*.
2. The study area should be *determined objectively*.
3. The pattern should be an enumeration or *census* of the entities of interest, not a sample.
4. There should be *one-to-one correspondence* between objects in the study area and events in the pattern.
5. Event locations must be *proper*. They should not be, for example the centroids of areal units chosen as representative ... They really should represent the point locations of entities that can be sensibly be considered points at the scale of the study.

While we do not need to be so strict when considering dot patterns, it is important to keep this in mind when assessing any GIS specific literature on the subject.

It is impossible to discuss the possibilities for change present in a dot pattern without first describing the pattern's properties in an analytical manner.

COMMENT: [*Things to reference:*]

- Geographic Information Analysis book, check for further references
- Density measures
- Probability distributions
- Existing work on things like variance and mean
- Worboys – Geographic Information Systems: A computing perspective. for some data structures
- see if there is any literature on describing dot patterns within classification or optimisation

0.3 Change

COMMENT: [*List types of dot pattern change, note work on convex hull updating using data structures*]

Bibliography

[1] H. Alani, C. B. Jones, and D. Tudhope. Voronoi-based region approximation for geographical information retrieval with gazetteers. *International Journal of Geographical Information Science*, 15(4):287–306, 2001.

[2] Avi Arampatzis, Marc van Kreveld, Iris Reinacher, Christopher B. Jones, Subodh Vaid, Paul Clough, Hideo Joho, and Mark Sanderson. Web-based delineation of imprecise regions. In *Computers, Environment and Urban Systems*, volume 30, pages 436–459. Elsevier, 2006.

[3] A. Ray Chaudhuri, B. B. Chaudhuri, and S. K. Parui. A novel approach to computation of the shape of a dot pattern and extraction of its perceptual border. In *Computer Vision and Image Understanding*, volume 68, pages 257–275. Academic Press, 1997.

[4] Matt Duckham, Lars Kulik, Mike Worboys, and Antony Galton. Efficient generation of simple polygons for characterizing the shape of a set of points in the plane. In *Pattern Recognition*, volume 41, pages 3224–3236. Elsevier, 2008.

[5] Max Dupenois and Antony Galton. Assigning footprints to dot sets: An analytical survey. In K. S. Hornsby, C. Claramunt, M. Denis, and G. Ligozat, editors, *Spatial Information Theory: Proceedings of the 9th International Conference COSIT 2009*, pages 227–244, Berlin, 2009. Springer.

[6] Herbert Edelsbrunner, David G. Kirkpatrick, and Raimund Seidel. On the shape of a set of points in the plane. In *Computer Vision and Image Understanding*, volume IT-29, pages 551–559. IEEE, 1983.

[7] Antony Galton. Pareto-optimality of cognitively preferred polygonal hulls for dot patterns. In *Spatial Cognition*, 2008.

[8] Antony Galton and Matt Duckham. What is the region occupied by a set of points? In *GIScience*, 2006.

[9] Gautam Garai and B. B. Chaudhuri. A split and merge procedure for polygonal border detection of dot pattern. In *Image and Vision Computing*, volume 17, pages 75–82. Elsevier, 1999.

[10] R. A. Jarvis. On the identification of the convex hull of a finite set of points in the plane. In *Information Processing Letters*, volume 2, pages 18–21. North-Holland Publishing Company, 1973.

[11] Adriano Moreira and Maribel Yasmina Santos. Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points. In *International Conference on Computer Graphics Theory and Applications GRAPP*, 2007.

[12] David O'Sullivan and David J. Unwin. *Geographic Information Analysis*, chapter 4, pages 77–114. Wiley, November 2002. ISBN 0471211761.