A review of the the field for assigning the footprints to a point set.

Max Dupenois

February 4, 2009

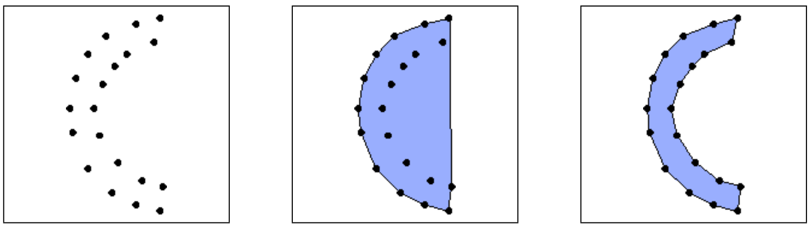
1 Introduction

This review concerns itself with a field that focuses on attempting to achieve representational shape for a set of dots in a plane. Specifically it centres on algorithms that can display a *perceived* shape from the set.

2 Background

2.1 Perceived Shape

The need for a method to find the *perceived* shape is best explained by first looking at the most commonly used form of shape generation. Commonly the shape is found by finding the convex hull [1]. One of the first examples of this was in R.A. Jarvis' paper [10] in which the 'Jarvis March' or 'Gift-Wrapping Algorithm' was proposed. There are a variety of other convex hull algorithms but one of them and the 'Jarvis March' will be examined in greater detail later in the review. For now it suffices to point out that the convex hull can be achieved with no more input than the point set and in $O(n \log n)$ time. The issues inherent in the use of the convex hull relate to its inability to achieve the *perceived* shape.



(a) Point set (b) Convex Hull (c) Likely Perceived Shape

Figure 1: When a convex hull is inappropriate

Fig.1 illustrates an example where the general shape of the point set is not accurately represented by the convex hull of the set, as can be seen Fig.1(c) is closer to the shape that we would perceive when looking at Fig.1(a).

2.1.1 Used Shape Terms

The shapes can be split into three parts, the points representing the data, the elements that make up the outline and the shape itself. Joins appears to be perfectly acceptable and uncontested term for the connections between the extrema of the shape. Boundary also seems a consistent used term for the union of all the joins. The word mostly used to describe the elements of the data set is points, however Galton [6], Edelsbrunner et al. [4] and Garai and Chaudhuri [8] all use the phrase dot pattern. While it is not explicitly mentioned in his paper [6], discussions with Antony Galton about the term have revealed that the use of

[1] *"A subset S of the plane is called convex if and only if for any points $p, q \epsilon S$ the line segment $\overline{pq}$ is completely contained in S. The convex hull $\mathcal{CH}(S)$ of a set S is the smallest convex set that contains S. To be more precise it is the intersection of all the convex sets that contain S."*[1]

'point' is woefully inconsistent with the objects the data represents. Rarely do the points represent something which has no other data than the co-ordinates associated with it. As such 'dot' provides a stronger linking to the real world application. It allows for the to be data associated with itself more than its co-ordinates, for example shape, velocity and region of influence. The shape itself has had many associated terms, below are a couple that have been used with a short review of each.

α-shape - Edelsbrunner et al. [4]
: This is different to the other terms as the α-shape is tied directly to algorithm that formed it. The reason it is mentioned here is that Edelsbrunner et al. [4] was one of the first papers to present an algorithm for attaining a shape other than the convex hull from the data set.

Outline of a Set Of Points - Gofman [9]
: This a very clear statement and would appear to be everything that is needed from such a term. Unfortunately there doesn't seem to be much discussion on what is meant by outline or what the requirements on it being valid are. The only criteria is the rather nebulous *'The algorithm is tuned to extract outlines similar to those extracted by a human observer.'*

External Shape - Chaudhuri et al. [2]
: This paper says that *'Another problem of interest to find the border, formally known as the external shape, of a point set'*, although I have found no other reference to external shape as a formal term for the shape of a dot set. The term itself is interesting, *external* implying that we have at least two planes (interior and exterior). This matches the reasoning followed in Edelsbrunner et al. [4] in which they point out that discussing the shape is largely pointless without differentiating between the faces.

Non-Convex Hull - Galton [5]
: Seeing as for any point set you can make the convex hull, there being many algorithms to do this, many of the methods for extracting the shape use an altered version of one of the convex hull algorithms. This paper has one such algorithm called the Swinging Arm algorithm (an extension of the Jarvis March Jarvis [10] that uses a fixed line length). From this it is easy to see how the term non-convex hull could arise. However the term implies that the convex hull is not allowed, if the dot pattern is arranged such that the obvious shape is in fact the convex hull this becomes problematic.

Footprint - Galton and Duckham [7]
: Footprint is an interesting term, there are some very good reasons why it works, but the best one is that it fits. There is something satisfying about describing possible shapes on a dot pattern, when looking for a possibly perceived one, using a term that one perceives to fit without the need for further rationalisation. Footprint is generally used in various fields to denote of the impression of an object[2], this may be one of the reasons it fits so well, as we're already prepared to accept it as a term for describing many possible types of shape. Importantly footprint does not have any common pre-existing computational geometry meaning, because of this it does not disallow any particular type of shape.

Concave Hull - Moreira and Santos [12]
: Concave hull follows the same basic principle as the non-convex hull term. This means it falls prey to the same issues, specifically it suggests that that the shape cannot be the convex hull. Going by the fact that all the angles in the convex hull are convex it implies that all the angles are concave. This is obviously not the case the paper is trying to describe, but as it isn't instantly clear what is meant it is not ideal as a term.

Characteristic Shape / χ-shape - Duckham et al. [3]
: This is another term similar to the α-shape in that it is linked to the algorithm that creates it, the χ-shape algorithm. However before using this term the paper uses Characteristic Shape. This is defined as *'a possibly non-convex, simple polygon that characterizes the shape of a set of input points in the plane, termed a characteristic shape'*. Characterise is an excellent term in that it implies we're looking for the perceived shape, and by the above definition still allows us to have the convex hull.

[2] e.g. the memory space a piece of software uses, the actual footprint of an animal, the carbon footprint and so on.

Polygonal Hull - Galton [6]

This paper uses the term as it's discussing all the possible shapes[3] that could be created for the dot pattern. It covers a wide range but does not allow dots outside the boundary, a shape made up of many parts or a non-polygonal shape all of which could occur when looking for the perceived shape.

The terms that would appear to be a best fit, in relation to describing the variety of possible percieved shapes, are *external shape*, *characteristic shape* and *footprint*. For the purposes of the rest of the literature review footprint will be used. This is more of a personal choice than anything else, but it could be argued that it is a better fitting term because of its terseness and natural aptness.

A few of the above terms include the word hull, this can be problematic as many of them do not conform to the mathematically defined hull; Klette and Rosenfeld [11]. They lay down three properties which a function has to have if it is consideredto be a hull operator.

Let $\mathbf{S}$ be a class of subsets of a set S. A function H that takes sets in $\mathbf{S}$ into sets in $\mathbf{S}$ is called a *hull* operator iff it has the following properties:

H1: $M \subseteq H(M)$ for all $M \in \mathbf{S}$.

H2: $M_1 \subseteq M_2$ implies $H(M_1) \subseteq H(M_2)$ for all $M_1, M_2 \in \mathbf{S}$.

H3: $H(H(M)) \subseteq H(M)$ for all $M \in \mathbf{S}$.

They also supply a fourth possible property.

H4: $M_1 \subseteq M_2$ implies $\mathcal{A}(H(M_1)) \leqslant \mathcal{A}(H(M_2))$ for all sets $M_1, M_2 \in \mathbf{S}$ where $\mathcal{A}(S)$ is the area of S.

Together these provide requirements for three versions of a hull. If the first three are satisfied then you have a hull, H1 and H2 gives a pseudohull and H1, H3, and H4 gives a near-hull. The possible range of footprints covered by the algorithms for the perceived shape do not conform to these quite stringent requirements. As such whenever it is used in the above mentioned papers it is done with a modifier (e.g. 'non-convex' in non-convex hull [5] and 'concave' in concave hull[12]). The Polygonal Hull, however, [6] does in fact meet the requirements to be classified as a hull.

3 Previous Work Reviewed

Having reviewed the terms used in the papers, we can look closer at the actual work some of the papers set out to accomplish.

3.1 Research

Nearly all the papers present algorithms for the generation of the footprints with little reference to any real research into what makes a perceived shape perceivable. The only paper that focuses on this is Galton [6] in which he aptly points out:

> *The evaluation of this behaviour* [4] *is typically very informal, often amounting to little more than observing that the shape produced by the algorithm is a 'good approximation' to the perceived shape of the dots. 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.*

[3] assuming the shape is polygonal and *1. The outline is a polygon whose vertices are members of the dot pattern. 2. Any member of the dot pattern which is not a vertex of the polygon lies in the interior of the polygon. 3. The boundary of the polygon forms a Jordan curve (so in particular no point is encountered more than once in a full traversal of the boundary).*

[4] The behaviour of the algorithm when used on various dot patterns.

The paper is an attempt to rectify this. To this effect he comes up with a strict definition of his polygonal hull: The vertices of the hull coincide with dots of the dot pattern, all dots which aren't vertices of the hull exist within the boundary and the polygon forms a jordan curve. Using this, the conclusion can be drawn that all the vertices of the convex hull for a particular dot pattern are vertices of any polygonal hull for that dot pattern. As such there is an upperbound for the number of polygonal hulls that can be created for a dot pattern based on the number of vertices in its convex hull and the total number of dots.

$$\sum_{r=0}^{n-k} {}^{n-k}C_r \, {}^{k+r-1}C_r \, r!$$

See the paper [6] for a full derviation of the equation.

With a definite upperbound it is possible to create a program that generated all the possible polygonal hulls for a dot pattern. With these generated Galton proposed a hypothesis:

> **Hypothesis:** *The points in area-perimeter space corresponding to polygonal hulls which best capture a perceived shape of a dot pattern lie on or close to the Pareto front.*

Essentially the hulls which would be chosen as perceived shapes try to minimise both area and perimeter. To test this a pilot study was performed; eight dot patterns presented to 13 adults. They were asked to draw the *'polygonal outline'* that best captured the shape of the dot pattern. Even though the study was very small there was no major dissention from the hulls chosen. All existed on the pareto front, the furthest away having a *'relative domination'*[5] of 0.008578 (0 would be the Pareto-optimal hull and 1 the furthest from the Pareto-optimal). The paper suggests that there is still work to be done, aside from obviously performing a larger study. These include looking into factors like the *sinuosity* of the hull, symmetry and number of vertices. Also suggested is more work into properly evaluating the performance of the algorithms and the application context they perform in.

3.2 Algorithms

Many of the methods for finding the footprints for a point set use the convex hull as a base, as such it makes sense to cover a couple of the alglorithms used in their creation.

1. **Gift Wrapping Algorithm/Jarvis March [10]**
   Created by R. A. Jarvis this algorithm is conceptually quite simple. For the set P where if p is a point in the plane $p\epsilon P$ start at point p_0 (usually the leftmost point). Then, assuming $i = 0$, select the point p_{i+1} such that all the other points lie to the right of the line $p_i p_{i+1}$. Let $i = i + 1$ and continue till the original point is reached again. Essentially like wrapping paper around the point set. The major issue the algorithm has is its time complexity which is $O(nh)$ where n is the number of points and h is the number of vertices on the convex hull, this gives a worst case complexity of $O(n^2)$.

2. **Divide and Conquer**
   Divide and conquer is a strange algorithm in that it in itself doesn't have a method for creating the convex hull. What it does is to recursively divide the points into two equally sized sets. The splitting is done on the x-axis such that all the points in a set are to the left of the other set and vice versa. When the set is suitably small (generally 3 or 4 points) the convex hull of that set is created, it is then joined to its set pair's convex hull using the upper and lower common tangents. This is repeated till all the pairs have been joined and the total convex hull is acheived. This algorithm can be completed in $O(n \log n)$ time.

Having specified some of the base methods used for shape creation we have a solid base which we can use to discuss the creation of footprints. One of the earliest examples of these is α-shapes[4] and nearly all of the others reference the paper when they are being described.

[5] *"that is, the ratio of number of hulls which dominate it to the maximum number of hulls that dominate any one hull for that dot pattern"*

This paper presents an algorithm to produce a set of α-shapes from a set of points. The central premise of this method is that if a disk of radius $1/\alpha$ can be placed with two points from the set on its circumference and all the other points reside within the disk then those two points become an edge. If α is negative then, as opposed to requiring the points to be in the disk, we're interested in all the points existing within the complement of said disk i.e. not in it. Finally if α is 0, as the radius is infinite, we have a half-plane instead of a disk, all the points having to be on one side of it, essentially we end up with a convex-hull. In Edelsbrunner et al.'s words: *"we define a generalized disc of radius $1/\alpha$ as a disc of radius $1/\alpha$ if $\alpha > 0$, the complement of a disc of radius $-1/\alpha$ if $a < 0$, and a halfplane if $\alpha = 0$"*. This leads to the definition: *"for an arbitrary real α and a set S of points in the plane, the α-hull of S is the intersection of all closed generalized discs of radius $1/\alpha$ that contain all the points of S"*. The alpha shape produces a wide range of possible footprints and, while this would appear to be a benefit, this leads to difficulty in selecting an appropriate value for α. Particularly as merely looking at a dot pattern does not immediately suggest any relevant α, unlike, for example, the avearge distance between dots.

Next it is worth looking at one of the first papers produced with the primary intent of attaining the perceived shape Chaudhuri et al. [2]. In this paper Chaudhuri et al. provide two algorithms for obtaing what they call the perceptual border (or external shape) of a dot pattern. The first is the s-shape, this is described as taking *"the union of all grids containing points od the DP"* for a grid of length s. This leads to an obviously very blocky outline. They spend some time using the *dispersion matrix* of the grid, essentially how many dots fall into each grid section, and use this to refine the outline; removing unecessary blocks. This leaves us with a still blocky footprint. To combat this they move on to a second algorithm called the r-shape. This *"is the graph constructed by connecting the respective centers of each apir of intersecting disks which are partially exposed to the background."* Here we have a much better outline shape in terms of removing excess internal space. However it is still set up such that the dots never coincide with the boundary of the shape and we have a very bumpy looking shape. The r-shape is proposed in such away as to make it seem an extension of the the s-shape, but this is not really the case, as r-shape could have been produced independantly with no need for the s-shape to have even existed. On of the issues the a re-occurs with these algorithms is the parameter required for the algorithm to produce one of the many possible footprints is how nebulous they are. It is not immediately apparent when viewing the shape what radius of disk or length of grid will be the most appropriate for a perceived shape.

4 Conclusion

There are many algorithms using a variety of different methods to produce these footprints. However there has been little work done on either the classification or evaluation of the methods or the shapes produced. Only one paper [6] makes a real concerted effort to satisfy some of these lacking concepts. Much work still needs to be done into defining what is mathematically meant by 'perceived shape', when this is done there can be some real thought into evaluation. Even with this done none of the algorithms pay much attention to application context, despite the fact that it greatly affects what is required from the footprint. A cartographer maybe looking for a different perceived shape than an individual in the letter recognition field.

The papers presenting these algorithms also avoid going into great detail about how one goes about choosing the appropriate parameter for their method. This is an issue because, as mentioned above, often the parameter does not directly relate to what we see when we look at a dot pattern. For example it is easier to see what level of sinuosity you want for your shape than it is to say what line length is required to acheive that level.

In general the work that has been done is good and with solid mathematical grounding. It lacks in its broadness, focusing solely on the production of algorithms without paying much attention to why or in what manner. There is however much more than is presented here, due to limitations it was impossible to go into greater detail on the algorithms than was really fair to the papers which presented them. As such this review suggests that, while this hopefully provides a wide perspective on the type of work out there, the papers listed in the bibliography are worth reading in detail to get a clearer picture.

References

[1] Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry, Algorithms and Applications.* Springer, third edition, 2008.

[2] 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. 1997.

[3] 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. 2008.

[4] 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. 1983.

[5] Antony Galton. Families of non-convex hulls for aggregate shape and generalised location. unpublished, 2005.

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

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

[8] 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. 1999.

[9] Yossi Gofman. Outline of a set of points. In *Pattern Recognition Letters*, volume 14, pages 31–38. 1993.

[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. 1973.

[11] R Klette and A Rosenfeld. *Digital Geometry: Geometric Methods for Digital Picture Analysis.* Morgan Kaufmann, 2004.

[12] 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.