Chapter 4

Algorithm that Mimics Human perceptual Grouping of Dot Patterns

Based on the content of the paper

Abstract

We propose an algorithm that groups points similarly to how human observers do. It is simple, totally unsupervised and able to find clusters of complex and not necessarily convex shape. Groups are identified as the connected components of a Reduced Delaunay Graph (RDG) that we define in this chapter. Our method can be seen as an algorithmic equivalent of the Gestalt law of perceptual grouping according to proximity. We introduce a measure of dissimilarity between two different groupings of a point set and use this measure to compare our algorithm with human visual perception and the k-means clustering algorithm. Our algorithm mimics human perceptual grouping and outperforms the k-means algorithm in all cases that we studied. We also sketch a potential application in the segmentation of structural textures.

4.1 Introduction

One of the remarkable properties of the human visual system is its ability to group together bits and pieces of visual information in order to recognize objects in complex scenes. In psychology, there is a long tradition of research devoted to perceptual grouping. This tradition has been largely formed and is still very much influenced by Gestalt laws that name factors, such as proximity or similarity in orientation, color, shape, or speed and direction of movement which play a role in perceptual grouping (see e.g. [1-4]). The sole naming of these factors, however, is not sufficient for a quantitative analysis of an image aimed at grouping features together automatically, i.e. without the involvement of a human observer. Such automatic grouping is essential for the computerized recognition of objects in digital images.

Several approaches have been proposed in the past for identifying groups in a given dot pattern. Examples include algorithms based on a mathematical model of the local point density [5], neural gas networks [6], clustering algorithms based on fuzzy logic[7] [8], mean shift [9], graph theoretical methods[10], and algorithms for analyzing high dimensional data[11]. Other work concerns criteria for the evaluation of a given clustering algorithm[12].
An overview of these techniques is presented in [13]. Most of the aforementioned approaches are designed to work with points in a feature space, while ignoring important aspects of the way how humans perceive that point pattern.

In the current chapter we describe an algorithm for the grouping of points (Section 4.2) and demonstrate that this algorithm delivers results that are in agreement with human visual perception (Section 4.3). To quantify the degree of agreement we introduce a measure of dissimilarity between two possible groupings of the points of one set. We also compare the performance of our algorithm in mimicking human visual perception with the performance of the $k$-means clustering algorithm, commonly used in computer vision (Section 4.4). Our algorithm is closely related to the Gestalt law of proximity but it goes beyond that law in that it has predictive power. In Section 4.5 we refer to some related previous work, draw conclusions and outline future work on possible applications in computer vision.

4.2.  Grouping Algorithm

4.2.1  Voronoi Tessellation and Delaunay Graph.

Given a set $S = \{p_1, \ldots, p_N\}$ of $N$ points in the plane, we partition the plane in cells $C_1, \ldots, C_N$ (bound or unbound) such that the points which belong to cell $C_j$, associated with point $p_j \in S$, are closer to $p_j$ than to any other point $p_k \in S, k \neq j$:

$$q \in C_j \Rightarrow d(q, p_j) \leq d(q, p_k), \quad \forall q, \quad \forall p_j, p_k \in S. \quad (4.2.1)$$

Such a partition of the plane is called the **Voronoi Tessellation** (VT) or **Voronoi diagram** related to $S$. The dual of the Voronoi tessellation, called the **Delaunay Graph** (DG), is obtained by connecting all pairs of points of $S$ whose Voronoi diagram cells share a boundary (Fig. 4.1). More details about Voronoi tessellation and Delaunay graph can be found in the standard literature (see for example [14]).

![Fig. 4.1. Voronoi tessellation and Delaunay graph of a set of points.](image)

4.2.2  Reduced Delaunay Graph

Perceptually, a group of points in $S$ is a subset of points, which are much closer to each other than to the other points of $S$. However, the concept of “much closer” is subjective and it is not well defined mathematically. We propose an algorithm which partitions $S$ in disjoint subsets, and show that this partitioning corresponds to human visual perception of groups.
4.1. Grouping Algorithm

compute the Delaunay Graph \( DG \) of the set \( S \) and eliminate some edges from it to obtain a new graph that we call the Reduced Delaunay Graph; then, we regard the connected components of that graph as groups. This is illustrated by Fig. 4.2. In order to choose which edges must be removed, for each edge \( pq \) of the \( DG \) we first compute the distance \( d(p, q) \) (Fig. 4.2a); then we normalize it with the distances of \( p \) and \( q \) to their respective nearest neighbours:

\[
\xi(p, q) = \frac{d(p, q)}{\min_{x \in N(p)} \{d(p, x)\}}, \quad \xi(q, p) = \frac{d(q, p)}{\min_{x \in N(q)} \{d(q, x)\}},
\]

(4.2.2)

where \( N(p) \) is the set of those points which are directly connected to \( p \) by one arc of the delaunay graph. Note that in general \( \xi(p, q) \neq \xi(q, p) \). In this way, we assign two ratios, \( r_1(e) = \xi(p, q) \) and \( r_2(e) = \xi(q, p) \), to each edge \( e \) of the \( DG \) (Fig. 4.2b).

Next, we reduce the two above mentioned numbers to a single quantity, computed as their geometric average:

\[
r(e) = \sqrt{r_1(e) \cdot r_2(e)}.
\]

(4.2.3)

and remove from the \( DG \) every edge \( e \) for which \( r(e) \) is larger than a fixed threshold \( r_T \). We call the remaining graph \textit{Reduced Delaunay Graph} (RDG).

More precisely, \( RDG = (V_{RDG}, E_{RDG}) \) is a graph whose vertex set \( V_{RDG} \) is the same as the vertex set of the \( DG \), and whose edge set \( E_{RDG} \) contains those edges \( e \in E_{DG} \) for which \( r(e) \) is less than or equal to a given threshold value \( r_T \):

\[
V_{RDG} = V_{DG}; \quad E_{RDG} = \{e \in E_{DG} | r(e) \leq r_T \}.
\]

(4.2.4)

Finally, we regard the connected components of the RDG as \textit{groups}. We observe that quantity \( r(e) \) defined in (4.2.3) is not affected by an isotropic scaling of the point set, thus making the proposed grouping algorithms scale-independent.

This is illustrated in Fig. 4.2, where the value chosen for \( r_T \) is 1.65. The two normalized distances assigned to edge \( p_1p_2 \), for instance, are \( \xi(p_1, p_2) = 4.95 \) and \( \xi(p_2, p_1) = 2.48 \). Their geometrical average is 3.5, larger than \( r_T \), so \( p_1p_2 \) must be removed. The other edges that must...

\footnote{Other possible ways to collapse \( r_1(e) \) and \( r_2(e) \) into a single number could have been \( r_{max}(e) = \max\{r_1(e), r_2(e)\} \), \( r_{min}(e) = \min\{r_1(e), r_2(e)\} \), or, more in general, \( r_{q}(e) = \left[ \frac{r_1(e)^q + r_2(e)^q}{2} \right]^{\frac{1}{q}} \). The choices \( r_{max}(e) \) and \( r_{min}(e) \) are limit cases of \( r_q(e) \), which are obtained for \( q \to \infty \) and \( q \to -\infty \) respectively, while the geometric average deployed in (4.2.3) is reached for \( q \to 0 \). In our experiments, we found that the choice of \( q \) is not very relevant, though in some critical cases, in which the distance between points that humans would judge to belong to the same group is very close to the distance between points that humans would judge to belong to different groups, the choice \( r_{max}(e) \) would lead to over-fragmented RDG, while \( r_{min}(e) \) would give rise to too many connections. As an intermediate case, the geometric average deployed in (4.2.3) has been chosen for reasons of elegance, as it stay perfectly in between \( r_{max}(e) \) and \( r_{min}(e) \). Experimental results show that, in general the choice \( q = 0 \) gives the best similarity between the outcome of human observers.}
be removed are shown in dashed line in Fig. 4.2c. The remaining edges (shown in solid line) produce two connected components, \( \{p_1, p_3, p_4\} \) and \( \{p_2, p_5, p_6\} \), which we regard as groups.

Fig. 4.2. (a) A set of points and its Delaunay graph. The numbers assigned to the edges are equal to the distances between the corresponding vertices. (b) A pair of normalized distances is assigned to each edge of the DG. (c) The pair of normalized distances assigned to an edge is replaced by a single number, computed as their geometric mean. The dash lines represent the edges removed from the DG; these are the lines that are assigned numbers larger than the threshold (1.65). The vertices and the remaining edges define the reduced Delaunay graph; its connected components are regarded as groups.

4.3 Results

In this section we apply our algorithm to several sets of points (see Figs. 4.3-4.5); in all these examples we take a threshold value \( r_T = 1.65 \), which has been chosen empirically in order to maximize the performance in terms of the metric defined in Section 4.4.

In the cases shown in Fig. 4.3a-b the RDG is connected and all points are grouped together in one single group; this corresponds to the human visual perception of these dot patterns; the examples of Fig. 4.3c-f are point configurations in which distinct groups, arranged both regularly and randomly, are formed. In the case shown in Fig. 4.4a the points arranged in a circle and the points inside that circle form separate groups. Fig. 4.4b-c present examples of sets in which there are isolated points. Fig. 4.4d-f illustrate the ability of our algorithm to find clusters of complex shape that are not necessarily convex.

Fig. 4.5a-b show examples of point sets in which groups of points are immersed in a group that covers the whole image. In Fig. 4.5a, for instance, we perceive a set of crosses immersed in a square grid, and it is exactly what the RDG reveals. Similarly, in Fig. 4.5b we perceive small squares surrounded by a grid of octagons and big squares. In Fig. 4.5c the RDG reveals the lines traced by the points. This example has a certain relation to the Gestalt law of good continuation. The arcs of the RDG not only allow identifying groups but also reveal the order of continuation.
Fig. 4.3. Six point sets (first and third rows) and their corresponding RDGs (second and fourth rows). In cases (a-b) the RDGs are fully connected and all points belong to the same group. In cases (c-f) the points are clustered in groups, both regularly and randomly arranged. The connected components of the respective RDGs are in agreement with the way how human observers would act if required to group points according to proximity.
Fig. 4.4. Other six point sets (first and third rows) and their corresponding RDGs (second and fourth rows). In case (a) the points that form a circle belong to a different group than the point inside that circle. Cases (b-c) are examples of RDGs that have isolated points; in examples (d-f) the shape of the clusters is complex and not necessarily convex.
Fig. 4.5. Point sets (above) and their corresponding RDGs (below). (a-b) A system of groups is immersed in a group that covers the whole image. (c) A case related to the Gestalt principle of good continuation. The edges of the RDG reveal the connectivity of that continuation.

4.4 Quantitative comparison to human observers

We claim that our algorithm mimics the grouping properties of the human visual system. To quantify this claim, we now compare the grouping results obtained with this algorithm with the groupings defined by human observers. For this purpose we first introduce a measure of dissimilarity between two possible groupings of the points of one set.

4.4.1 Dissimilarity between two partitions of a point set

Let $C_1 = \{U_1, \ldots, U_N\}$ and $C_2 = \{W_1, \ldots, W_M\}$ be two different partitions of $S$. We introduce the following quantities:

$$\alpha_{i,k} = \frac{\text{card}(U_i \cap W_k)}{\text{card}(S)}; \quad u_i = \frac{\text{card}(U_i)}{\text{card}(S)}; \quad w_k = \frac{\text{card}(W_k)}{\text{card}(S)}.$$ (4.4.1)

In analogy with probability theory, we call $\alpha_{i,k}$ joint probability and $u_i$ and $w_k$ marginal probabilities; we also say that $C_1$ and $C_2$ are independent if $\alpha_{i,k} = u_i w_k$. We also define the following formal entropies [15]:

$$H(C_1) = \sum_{i=1}^{N} u_i \log u_i,$$

$$H(C_2) = \sum_{k=1}^{M} w_k \log w_k,$$

$$H(C_1, C_2) = \sum_{i=1}^{N} \sum_{k=1}^{M} \alpha_{i,k} \log \frac{\alpha_{i,k}}{u_i w_k}.$$
We now define the dissimilarity coefficient \( \rho(C_1, C_2) \) between the two partitions \( C_1 \) and \( C_2 \) as follows:

\[
\rho(C_1, C_2) = \frac{H(C_1) - \min \{ H(C_1), H(C_2) \}}{\max \{ H(C_1), H(C_2) \}}. \tag{4.4.3}
\]

It can easily be shown that \( \rho \) is symmetric, positive, equal to zero iff \( C_1 = C_2 \), and that it satisfies the triangular inequality; therefore, it defines a metrics in the space of the groupings. \( \rho \) has also the property to be always between 0 and 1, with \( \rho = 1 \) if and only if the two concerned groupings are independent. Fig. 4.6 illustrates that the concept of dissimilarity between two groupings is related to their correlation.

### 4.4.2 Results

We used the dissimilarity coefficient defined above to compare the results of our grouping algorithm with the perceptual grouping as done by human observers. For each of the images named in the first column of Table 1, we asked eight human observers (male, age varying between 25 and 48) to group the points of the corresponding set. Originally, we used a larger number of point sets but then we excluded sets for which the observers produced different groupings; the set shown in Fig. 4.4d is one such case that is included here for illustration; the maximum dissimilarity coefficient of two groupings produced by two observers for this set was 0.026. For all other point sets included here the groupings produced by different observers are identical.

**Fig. 4.6.** (a) The groupings/partitions \( C_1 = \{ U_1, U_2, U_3 \} \) and \( C_2 = \{ U_1, U_2, U_3 \} \) defined by the continuous and dashed lines, respectively, are strongly correlated and consequently their dissimilarity \( \rho(C_1, C_2) \) is small. (b) The partitions \( C_1 = \{ U_1, U_2, U_3 \} \) are totally uncorrelated and their dissimilarity coefficient is maximal. In all the cases but the ones presented in Fig. 4.4d and Fig. 4.5c our algorithm produces...
groupings that are identical with the groupings defined by the human observers. Consequently, the dissimilarity coefficients of the groupings obtained by the algorithm and by perceptual grouping are 0 in these cases, Table 1. One case in which a difference arises is shown in Fig. 4.7. As can be seen, this difference is small which is well reflected in the small value of the corresponding dissimilarity coefficient. For the case shown in Fig. 4.4d the dissimilarity coefficient is computed as the average of the dissimilarity coefficients between the algorithmic grouping on one hand and each of the human perceptual groupings on the other hand. The average value obtained in this way is smaller than the maximum value of dissimilarity between the perceptual groupings by two different observers. Table 1 shows also the results achieved by another algorithm - the $k$-mean clustering [16]. Since this algorithm requires a number of clusters to be specified, this number has been selected, for each point set, to be equal to the number of groups drawn by the human observers, in order to compare our results to the best results that $k$-means could produce. In all cases, the RDG algorithm outperforms the $k$-means algorithm in its ability to mimic human perception. The performance difference is especially large for point sets in which the groups (as defined by human visual perception) are not convex and have complex form, Fig. 4.7.

4.5 Summary and Conclusion

Attempts to identify perceptually meaningful structure in a set of points have been made in previous works. Voronoi neighbourhood is used in [17]. A discussion in [18] motivates a graph-approach to detect Gestalt groups. In [19] the following concept is introduced: two points $p$ and $q$ of a set $S$ are considered relatively close if $d(p,q) < \max\{d(p,x), d(q,x)\}, \forall x \in S$; linking all points of $S$ which are relatively close, we obtain what is called the Relative Neighbourhood Graph [20]. In [20] it is shown by some examples that points are linked in a way that is perceptually natural for human observers. However, this algorithm cannot be used to find groups of points, which is the goal of the current chapter. A nice algorithm, presented in [21], repeatedly splits the convex hull of the point set, until clusters are found; but it is unable to find clusters which are one inside the other.

We introduced the concept of a RDG that we use to group points. We demonstrated that this algorithm mimics human visual perception. For this purpose we introduced a quantitative measure of dissimilarity between two possible groupings of the points of a set. The dissimilarities between groupings produced by our algorithm and perceptual groupings done by human observers are zero or very small for a number of point sets that we studied. In contrast, the popular $k$-means clustering algorithm groups points in a way that is quite different from visual perception. Our algorithm is very simple, totally unsupervised, and is able to find groups of complex shape. At the present, we are unaware of any mechanism in the human brain which justifies the choice of the Delaunay graph over other graphs, like the spanning tree or the $k$-th neighbour graph. This issue will be the object of further research.
In future work we will also apply this algorithm to computer vision problems, such as segmentation of structural textures. Such textures cannot be treated adequately with traditional filter based approaches (see e.g. [22]). Our idea is to first reduce a texture (Fig. 4.8a) to one or more sets of points that indicate the positions of structural elements, using morphological filters, and to group them using the RDG approach, Fig. 8b–c.

Table 1. Dissimilarity coefficient values between perceptual grouping by humans and two grouping algorithms, RDG and k-means.

<table>
<thead>
<tr>
<th>Point set</th>
<th>ρ (RDG to human)</th>
<th>ρ (k-mean to human)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig. 3c</td>
<td>0</td>
<td>0.08</td>
</tr>
<tr>
<td>Fig. 3d</td>
<td>0</td>
<td>0.04</td>
</tr>
<tr>
<td>Fig. 3e</td>
<td>0</td>
<td>0.05</td>
</tr>
<tr>
<td>Fig. 3f</td>
<td>0</td>
<td>0.05</td>
</tr>
<tr>
<td>Fig. 4a</td>
<td>0</td>
<td>0.24</td>
</tr>
<tr>
<td>Fig. 4d</td>
<td>0.007</td>
<td>0.48</td>
</tr>
<tr>
<td>Fig. 4e</td>
<td>0.21</td>
<td>0.85</td>
</tr>
<tr>
<td>Fig. 4f</td>
<td>0</td>
<td>0.68</td>
</tr>
<tr>
<td>Fig. 5c</td>
<td>0.17</td>
<td>0.30</td>
</tr>
</tbody>
</table>

Fig. 4.7. (left) The RDG grouping algorithm mimics human perception while (right) the k-means algorithm (k = 7) clusters points in a very different way.

Fig. 4.8 (a) A texture in which the regions are defined by different structural elements. (b) The positions of the + elements are identified by a morphological filter selective for the + shape, and the points are connected in a RDG to identify regions of + elements. (c) The regions that contain L elements are identified in a similar way using another morphological filter.
Bibliography


4. Algorithm that Mimics human Perceptual Grouping of Dot Patterns

Analysis, 1 : 196-212, 1969

