Published in Image Processing On Line on 2017–04–24.
Submitted on 2016–04–21, accepted on 2017–01–18.
ISSN 2105–1232 © 2017 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol.2017.176

An Unsupervised Algorithm for Detecting Good Continuation in Dot Patterns

José Lezama[1], Gregory Randall[1], Jean-Michel Morel[2], Rafael Grompone von Gioi[2]

[1] IIE, Universidad de la República, Uruguay ({jlezama, randall}@fing.edu.uy)
[2] CMLA, ENS Cachan, France ({morel, grompone}@cmla.ens-cachan.fr)

Communicated by Julie Digne *Demo edited by* José Lezama

Abstract

In this article we describe an algorithm for the automatic detection of perceptually relevant configurations of 'good continuation' of points in 2D point patterns. The algorithm is based on the 'a contrario' detection theory and on the assumption that 'good continuation' of points are locally quasi-symmetric. The algorithm has only one critical parameter, which controls the number of false detections.

Source Code

The MATLAB reviewed source code for this algorithm and a online demo are available from the web page of this article[1]. Compilation and usage instruction are included in the `README.txt` file of the archive.

Keywords: Gestalt; good continuation; dots; non-accidentalness; local symmetry

1 Introduction

The Gestalt school of psychology [16, 11, 5, 14, 15] proposed the existence of a short list of grouping laws governing the early stages of visual perception. Among them, the law of *good continuation* can be stated as "All else being equal, elements that can be seen as smooth continuations of each other tend to be grouped together" [12, p. 259]. Figure 1 exemplifies this law; a perceptual organization of this image would result in a three part configuration: a line, an arc of circle, and a zigzag, all formed by dots. Unfortunately, this law was enunciated only qualitatively, without a formalization into a predictive framework. This was the case for all Gestalt laws.

In this article we describe an algorithm for the grouping of unoriented elements by good continuation. Our simple model favors local symmetries and includes a detection control based on the non-accidentalness principle. This allows the method to be general in the sense that it can capture smooth curves of any shape and scale, and is robust to the presence of outliers and noise. It is also

[1] https://doi.org/10.5201/ipol.2017.176

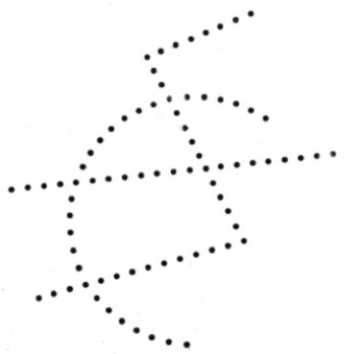

Figure 1: *Good Continuation* law: human perception tends to group elements on a smooth, continuous order. Image extracted from [6].

unsupervised because detections are given by their statistical significance which requires only a single parameter, namely the number of false detections that would be allowed in a pattern of random points. A preliminary version of this work was presented in [7] and the version implemented in this article was introduced in [8]. The algorithm consists of two main steps: building candidate chains of points and validating them. Candidate chains of points are built by considering triplets of points formed by joining nearest neighbors. Once valid triplets have been obtained, a graph representation is produced where each node corresponds to a triplet. A classic path finding algorithm is run on this graph to obtain paths between all pairs of triplets. Finally, the paths found are validated as non-accidental or rejected using thresholds obtained with the *a contrario* approach [2].

This article is accompanied by an online demo where the user can upload a set of points or draw them in the browser. The purpose of the online demo is to show that the number of false alarms (NFA) defined by the *a contrario* theory is an effective measure of the meaningfulness of a good continuation configuration and that the algorithm described in this article is an effective heuristic for finding such structures.

This article is organized as follows: Section 2 presents the mathematical model of good continuation chains. Section 3 describes the algorithm for detecting good continuation configurations in dot patterns. The mathematical model and the algorithm are then evaluated in Section 4. Finally, Section 5 concludes the paper and discusses some perspectives for future work.

2 Mathematical Model

Let us consider a set of N planar points. The aim is to find a mathematical model that can predict when an ordered subset of points lie on a smooth curve that is salient with respect to the background of the other points, see Figure 2(a). Each ordered subset of points (a sequence of points) will be called a *chain*; each set of three consecutive points in a chain will be called a *triplet*. The proposed model is based on the simple idea that the better the symmetry of the triplets, the better the saliency of the sequence. Figure 2(b) shows an example chain with an almost symmetric triplet marked in red (corresponding to a smooth part) and a non-symmetric triplet marked in blue (corresponding to a kink in the chain). Ideally, the third point in a triplet should be symmetric to the first relative to the middle point (the position marked with an X on Figure 2(c)). Symmetric triplets also enforce a second Gestalt grouping law: *proximity* [14]. A regular spacing of the points along the curve favors their perceptual grouping; inversely, an irregular spacing [16, 4] would tend to split the curve at larger gaps.

To obtain a perceptually plausible model, the evaluation of a chain of points is based on the *non-accidentalness principle*, proposed as the rationale underlying perceptual thresholds [17, 13, 10, 1].

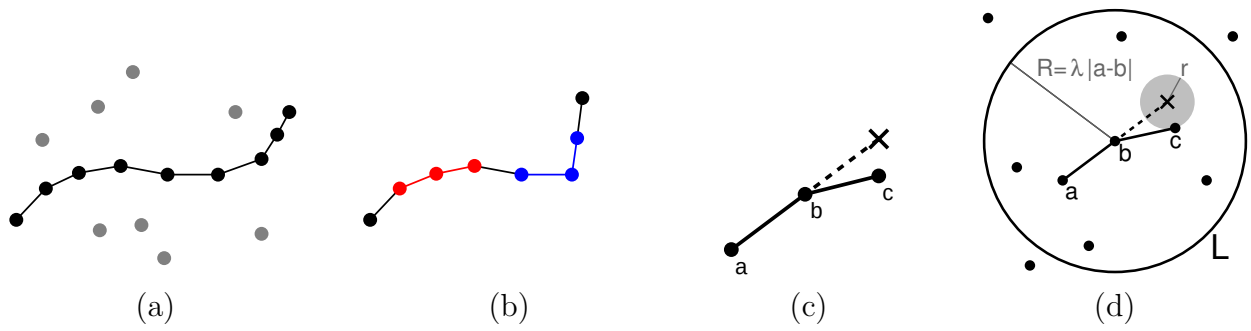

Figure 2: Definition of the *good continuation* event. **(a)** A candidate *chain* is defined by an ordered sequence of points. **(b)** A candidate *chain* with an almost symmetric triplet marked in red and a non-symmetric triplet marked in blue. The more symmetric the triplet, the more smooth the corresponding part of the chain. **(c)** Three consecutive points in a chain define a *triplet*, (a, b, c) in this case. Ideally, the triplet should be symmetric. That is, the third point c should be symmetric to the first point a, relative to the middle point b. The ideal third point is represented by X. **(d)** The symmetry precision of a triplet is measured by the distance r from the third point c to the ideal point X. The scale-invariant error is expressed as the probability that, among the n points in the local window of radius R, the nearest point to X be at most at a distance r.

In a nutshell, an observed structure is relevant if it would rarely occur by chance. Quoting D. Lowe, "we need to determine the probability that each relation in the image could have arisen by accident, $P(a)$. Naturally, the smaller that this value is, the more likely the relation is to have a causal interpretation" [9, p. 39]. The *a contrario* framework [2], a formalization of this principle, is used to provide automatic detection thresholds compatible with perception and robust to the presence of noise points. Given a random model for the data, the *a contrario* methodology consists in evaluating the expectation of the occurrence of an error as small as the one observed, relative to an ideal structure. If this expectation is small, the event is termed non accidental and thus perceptually meaningful.

Consider the probability of observing a chain of points where all of its triplets have a given degree of symmetry. This probability is evaluated in a background model of randomly distributed points. The lack-of-symmetry of a triplet translates into the distance r between the observed third point and its ideal symmetric position X, giving the local context. To provide scale-invariance, this error is evaluated relatively to the context contained in a circular local window L with radius R, where $R = \lambda \cdot |a - b|$ is proportional to the triplet size, see Figure 2(d).

Given that n points are observed in the local window L (not counting the two points which define L which are necessarily inside), the random or *a contrario* model H_0 used to evaluate accidentalness is that these points are independent and uniformly distributed in L. In other words, the *a contrario* model assumes that the n points result from a spatial uniform Poisson process in L. This *a contrario* model H_0 is not intended to model the statistics of the sought structure; quite the opposite, it models random data where the sought structure is not present and is used to calibrate the rejection thresholds.

Under these assumptions, the error of each triplet is translated into probabilistic terms. Let us call ρ the distance between the ideal point X and its nearest point in L under H_0. Given an observed radius r, the algorithm evaluates the precision of a triplet by the probability $\mathbb{P}(\rho \leq r)$. To simplify the computation, we will start by evaluating the complementary probability $\mathbb{P}(\rho > r)$ which implies that all the n points in L fall outside the disk of radius r. Given that L is a disk of radius R, $\mathbb{P}(\rho > r) = \left(1 - \frac{\pi r^2}{\pi R^2}\right)^n$. Finally, the error associated to a triplet is

$$e = \mathbb{P}(\rho \leq r) = 1 - \left(1 - \frac{r^2}{R^2}\right)^n. \tag{1}$$

This quantity would be zero only for the ideal symmetric triplet ($r = 0$) and a small value corresponds to a near symmetric triplet. If only one point is observed in the local window L (not counting the points a and b that define the triplet), this error measures the deviation of the triplet from the ideal symmetric configuration. When more points are present the error grows, reflecting the lower relevance of the triplet due to crowding. Note that r only appears in the ratio $\frac{r}{R}$, thus making the measure independent of its absolute value and ensuring the scale-invariance of the criterion.

Consider a chain $\mathcal{C}$ of k points $a_1, a_2, \ldots, a_k$. The error e_i of each of the $k-2$ triplets (a_i, a_{i+1}, a_{i+2}) can be evaluated by Equation (1) and the worst case value $e_{\max} = \max\{e_1, e_2, \ldots, e_{k-2}\}$ is associated to the whole chain. The event considered is a chain $\mathcal{C}$ of $k-2$ triplets, each one with error $e_{\max}$ or less relative to the ideal symmetric triplet. The probability of this event is evaluated under H_0.

Consider a random triplet under H_0, with the third point selected as the nearest one to the ideal symmetric point. By construction of the error measure, when Equation (1) is evaluated on points following the same model H_0, the error defined by Equation (1) becomes a random variable E with uniform distribution in $[0, 1]$[2]. Thus, $\mathbb{P}(E \leq \alpha) = \alpha$ for any $\alpha \in [0, 1]$. We are now in a position to evaluate a random chain of $k-2$ triplets under H_0. The corresponding errors are $E_1, E_2, \ldots, E_{k-2}$ and the worst error is $E_{\max} = \max\{E_1, E_2, \ldots, E_{k-2}\}$. Now, under the *a contrario* Poisson assumption, the probability of each triplet is independent from the previous ones, so the probability of all errors being lower than $e_{\max}$ is

$$\begin{aligned} \mathbb{P}(E_{\max} \leq e_{\max}) &= \mathbb{P}\big(E_1 \leq e_{\max} \cap E_2 \leq e_{\max} \cap \cdots \cap E_{k-2} \leq e_{\max}\big) \\ &= \mathbb{P}(E_1 \leq e_{\max}) \times \mathbb{P}(E_2 \leq e_{\max}) \times \cdots \times \mathbb{P}(E_{k-2} \leq e_{\max}) \\ &= e_{\max}^{k-2}. \end{aligned} \tag{2}$$

Notice that this is not the probability of observing the exact chain $\mathcal{C}$ but the probability (under H_0) of observing chains whose triplets have all an error with respect to an ideal symmetric triplet which is less or equal to $e_{\max}$. This term can take the value zero only for straight chains with equally-spaced points, where each one of the triplets would be perfectly symmetric. Inversely, irregular curves necessarily have values close to one.

The fundamental quantity in the *a contrario* methodology is the number of false alarms (NFA) of an event, defined as the number of tests times the probability of the event [2]. The NFA for a chain of points in *good continuation* is computed as

$$\mathrm{NFA}(\mathcal{C}) = N_{tests} \cdot \mathbb{P}(E_{\max} \leq e_{\max}). \tag{3}$$

The NFA is an upper bound on the expected number of chains with the same error as $\mathcal{C}$ or smaller to be observed by chance in the *a contrario* model H_0. A large NFA means that such an event is to be expected under the *a contrario* model and is therefore too frequent to be of interest. On the other hand, a small NFA corresponds to a rare event and therefore arguably a meaningful one. Geometrically, a perfectly aligned and equally-spaced curve would have NFA $= 0$ and small values correspond to regular curves. On the other hand, irregular curves necessarily have large NFAs, with values close to N_{tests}.

The number of tests N_{tests} counts the chains considered as potential good continuations. The proposed method will generate candidates starting at each of the N points and every one of its b nearest neighbors will be tried as a second point. Using these two points, the ideal point X is constructed (see Figure 2(c)) and the closest point to it is selected as the third point of the triplet. Iteratively, the last two points of the previous triplet are used to construct the ideal point X for the next one. This process is repeated until a maximal chain length of $\sqrt{N}$ is reached. The assumption

[2]This is due to the fact that, given a continuous random variable X with CDF F_X, the random variable $Y = F_X(X)$ is uniform in $[0, 1]$.

here is that a smooth 1D subset of a 2D set of N points would be typically limited to $\sqrt{N}$ points. Each of the intermediate chains is also tested as a potential meaningful chain. Thus, the number of tests is $bN\sqrt{N}$. Finally, the NFA of the event "having k points in *good continuation* configuration with error $e_{\max}$ or less" is

$$\text{NFA} = bN\sqrt{N} \cdot e_{\max}^{k-2}. \tag{4}$$

Given an observed set of N points and a candidate chain of k points, the algorithm considers the latter event as an ε-meaningful *good continuation* when the corresponding NFA is lower than ε. It can be shown [2] that the expected number of events with NFA $< \varepsilon$ is bounded by ε in the *a contrario* model H_0. This justifies the definition and name of the NFA as it controls the average number of accidental (thus false) detections. Following [2], ε is fixed to $\varepsilon = 1$ to ensure an unsupervised detection as having less than one false detection on average is tolerable in this case. (See the evaluation section for a confirmation of this fact.)

Using Equation (4) one can compute the maximum error that a triplet could have so that a chain of length k that contains the triplet is meaningful. The maximum error E_{max} for a chain of length k is

$$E_{max}(k) = \left(\frac{\varepsilon}{bN\sqrt{N}}\right)^{\frac{1}{k-2}}. \tag{5}$$

3 Algorithm

This section describes the algorithm for searching candidate chains of dots and validating them as meaningful using the model presented in the previous section. Given an input of N planar points, the candidate chains are obtained by first computing possible triplets of points and then grouping them.

For each of the N points, its b nearest neighbors are explored to form a pair. For each of the $N \cdot b$ pairs, the ideal symmetric point is computed. The two points closest to the ideal symmetric point are used to form candidate triplets. Note that in the theoretical model for the number of tests, only the closest point to the ideal point is used. In practice, most triplets in a chain will be of that type, but to give more flexibility to the algorithm, the two closest points are explored. This small deviation from the theoretical number of tests yields better results in practice, but does not harm the discriminative power of the NFA as will be demonstrated in the experimental section.

To group the triplets into chains a graph representation of the triplets is constructed and the Floyd-Warshall algorithm is used to find the shortest path joining every pair of triplets. The Floyd-Warshall algorithm is a classic graph analysis algorithm that finds the shortest paths between all pairs of nodes in a graph. Each path found is a candidate chain that is finally evaluated using the NFA, Equation (4). Chains with an NFA lower than a meaningfulness threshold $\varepsilon = 1$ are kept as detections. Finally, only the most meaningful chains are kept by using a redundancy reduction step described below. The main algorithm is described in Algorithm 1.

3.1 Algorithm Description

The algorithm requires three parameters. The first two are the number of nearest neighbors b used for exploration and the ratio λ of the local window size to a triplet's size[3] (see Section 2). The third parameter Δ, is the minimum precision in the discrete domain of the image. The use of this parameter is to avoid considering the probability of the exact symmetric triplet occurring (the probability would

[3]All the results shown in this article use $b = 5$ and $\lambda = 4$.

Algorithm 1: *Good Continuation* detection

Input: $\mathbf{X}$: a list of N planar points. b: the number of nearest neighbors used for exploration. λ: the proportion of the local window radius w.r.t. the triplet size. Δ: the minimum discrete precision of the domain.

Output: A list of *good continuation* chains $\mathcal{GC}$

1 $\bar{\mathbf{X}} \leftarrow$ Extension($\mathbf{X}$) `// Create a symmetric extension of` $\mathbf{X}$
2 $\mathcal{T} \leftarrow [\,]$
3 **for** *point* $i = 1$ *to* N **do**
4 **for** *point* $j \in \text{NearestNeighbors}_b(i)$ **do**
5 **if** $||i - j|| < \Delta$ **then**
6 Continue
7 Count n, number of points in the local window L centered at j with radius $R \leftarrow \lambda \cdot dist(i, j)$
8 Compute point s_j^i, symmetric to i w.r.t. j
9 **for** *point* $k \in \text{NearestNeighbors}_2(s_j^i)$ **do**
10 $e \leftarrow \mathbb{P}\left(\rho < dist(s_j^i, k) \,\middle|\, R, n\right)$ `// Equation (1)`
11 **if** $e < E_{max}(N)$ **then** `// Equation (5)`
12 $\mathcal{T} \leftarrow (i, j, k;\, e)$
13 $\mathcal{D} \leftarrow$ a $|\mathcal{T}| \times |\mathcal{T}|$ matrix with values ∞
14 **for** *triplet* $s \in \mathcal{T}$ **do**
15 **for** *triplet* $t \in \mathcal{T}$ **do**
16 **if** s *is adjacent to* t **then**
17 $\mathcal{D}_{s,t} \leftarrow e_s + e_t$
18 $\mathcal{P} \leftarrow$ Floyd-Warshall($\mathcal{D}$)
19 **for** *path* $p \in \mathcal{P}$ **do**
20 **if** NFA(p) < 1 **then** `// Equation (4)`
21 $\mathcal{GC} \leftarrow p$
22 $\mathcal{GC} \leftarrow$ `masking(`$\mathcal{GC}$`)` `// Algorithm 2`

be zero, which produces some numerical complications and has no sense from a practical viewpoint). In this article we used $\Delta = W/256$ where W is the width of the domain (assumed square).

Lines 1 to 12 of Algorithm 1 build the list $\mathcal{T}$ of triplets to be considered. Note that each triplet is stored with its error e, Equation (1). The triplets whose error is higher than E_{max}, Equation (5), are discarded as they could never be part of a meaningful chain (line 11). A symmetric extension of the point set across the domain boundary is used to estimate the point density in windows meeting the outside of the domain (line 1). The candidates are still selected among the original points.

In the graph representation of the triplets, a pair of triplets is considered *adjacent* when they share two points in such a way that they can form a chain of four points. (Triplets that share two points but form a "Y" shape are not adjacent.) Each vertex in the graph is a triplet and adjacent triplets s and t share an edge with value $e_s + e_t$, the sum of their errors (lines 13 to 17). Note that the shortest paths found by Floyd-Warshall (line 18) will be shortest in terms of the sum of the errors. This does not correspond to the NFA formula but is only a heuristic step, without any probabilistic interpretation, justified in what follows. All the candidate chains provided by Floyd-Warshall are

Algorithm 2: Masking

Input: A list $\mathcal{GC}$ of chains of *good continuation* of points and their NFAs.
Output: A non-redundant list of *good continuation* chains $\mathcal{F}$

1 $\mathcal{F} \leftarrow \emptyset$
2 **while** $\mathcal{GC}$ *not empty* **do**
3 $\quad c \leftarrow \arg\min_{\zeta \in \mathcal{GC}} \text{NFA}(\zeta)$ `// most meaningful chain in GC`
4 $\quad$ **if** *#repeated_points(c)*≤ 1 **and** $\forall c' \in \mathcal{F}, |c \cap c'| \leq 1$ **then**
5 $\quad\quad$ **add** c to $\mathcal{F}$ `// not masked, add chain to output`
6 $\quad$ **remove** c from $\mathcal{GC}$ `// chain processed, removed it`

evaluated for significance using Equation (4) and the ones with NFA $< \varepsilon$ are kept (lines 19 to 21). The heuristic is necessary to be able to benefit from the efficiency of the Floyd-Warshall algorithm, which works with additive distances.

The paths found by the Floyd-Warshall algorithm are the best in the sense of the smallest sum $\sum_i e_i$ along the path. Notice that the "distance" used is not Euclidean, but the sum of the triplet errors (thus the common bias toward small curves in shortest path methods is not present here). There is no theoretical guarantee that paths with minimal NFA (smallest $e_{\max}^{k-2}$) are all contained in minimal paths for the Floyd-Warshall algorithm. However, our simulations show that this approximation is acceptable.

Once all the *good continuation* events are found, we are interested in keeping only non-redundant detections. Note that a *good continuation* event might mask another smaller event contained in itself (e.g. a subset of the points in a meaningful chain can be also meaningful). We shall say that an event A masks an event B if $\text{NFA}_A < \text{NFA}_B$ and the chains share at least two points. The latter is just a simple criterion to allow crossing chains, which share one point. Obtaining a list of only the most meaningful events, defined as those that are not masked by any other event, is done by the following simple steps, see Algorithm 2. The general idea is to evaluate the chains in order, from the most meaningful (smallest NFA value) to the least meaningful one (largest NFA value). If a chain is not masked by any previously found chain, it is added to the output list $\mathcal{F}$; otherwise it is rejected.

The output list is initialized as empty (line 1). Then the chain c with the smallest NFA value in $\mathcal{GC}$ is selected (line 3). The chain c must satisfy two conditions to be added to the output list $\mathcal{F}$. First, no more than one of its points can be repeated. This prevents the case of a chain where the same points appear multiple times but in different triplets. Second, it cannot share more than one point with any chain already in the output list $\mathcal{F}$. Because the chains are processed by NFA value order, with smaller NFA values first, any chain in $\mathcal{F}$ would have a smaller NFA value that the one being analyzed; if they share two or more points, the current chain must be masked. Chains that are not masked are added to the output list $\mathcal{F}$ (line 5). Finally, the chain is removed from $\mathcal{GC}$ (line 6). When there is no chain left in $\mathcal{GC}$, the output list $\mathcal{F}$ gives the non-redundant *good continuation* chains.

3.2 Computational Complexity

The computational complexity of the Floyd-Warshall algorithm is $O(T^3)$, where T is the number of vertices in the graph, i.e., the number of triplets. In the extreme case where all triplets are considered valid, $T = 2bN$, since the triplets are built by considering every one of the N points, then its b nearest neighbors and then the 2 points closest to the ideal symmetric point. In practice however, many triplets will have an error higher than $E_{max}(N)$ (Algorithm 1, line 11).

For large datasets, the number of valid triplets can be restricted even further by setting a lower error threshold. For example, using $E_{max}(\sqrt{N})$ (the maximum error so that a chain of length $\sqrt{N}$ would no longer be meaningful, Equation (5)), instead of $E_{max}(N)$. We added a conditional statement to the algorithm so that $E_{max}(\sqrt{N})$ is used as the threshold if $N > 200$. Otherwise the threshold is set to $E_{max}(N)$. In practice this implies that if the input dataset contains a single very long curve, it will be detected as multiple shorter curves.

The second stage of the algorithm is to compute the NFA of each path found by the Floyd-Warshall algorithm (Algorithm 1, lines 19 to 21). To do this, the path between each of the $T(T-1)/2$ pairs of triplets needs to be evaluated. In the extreme case that every path is of length T (meaning every path goes through every triplet), the complexity of this stage would also be $O(T^3)$. In practice this is always much lower.

The final stage of the algorithm is the computation of the masking principle (Algorithm 2). Again, let us consider a theoretically extreme case. Suppose all the $T(T-1)/2$ paths found by Floyd-Warshall are meaningful. Because of the masking principle, each triplet can participate in only one meaningful, non-redundant chain. Thus, the final list of meaningful chains can be at most of size T. In this extreme case, comparing the paths in the two lists would have a complexity of $O(T^3)$. In practice this would rarely be the case.

All in all, the total computational complexity of the algorithm is $O(T^3) = O(N^3)$. In practice, the total computation time will depend on the amount of structure present in the dataset. In a set of random points, no meaningful chains will be found and the second and third stages of the algorithm will require no computations.

To evaluate the dependence of the computation time on the amount of structure present in the dataset, we carried out the following experiment. We created six different datasets, all of which have 500 points. The first dataset contains only random points. The second dataset contains a perfectly spaced circle of 100 points, and the other 400 points are randomly distributed. In the third dataset, the circle has 200 points, and the background has 300 random points. This pattern is continued until in the last dataset the circle has 500 points and there are no background points. The datasets and the running times, as well as the algorithm result are shown in Figure 3.

The computation times were obtained with a 2.27Ghz Intel Xeon X7560 CPU with 32GB of RAM. The algorithm is implemented in Matlab and no parallelization is used.

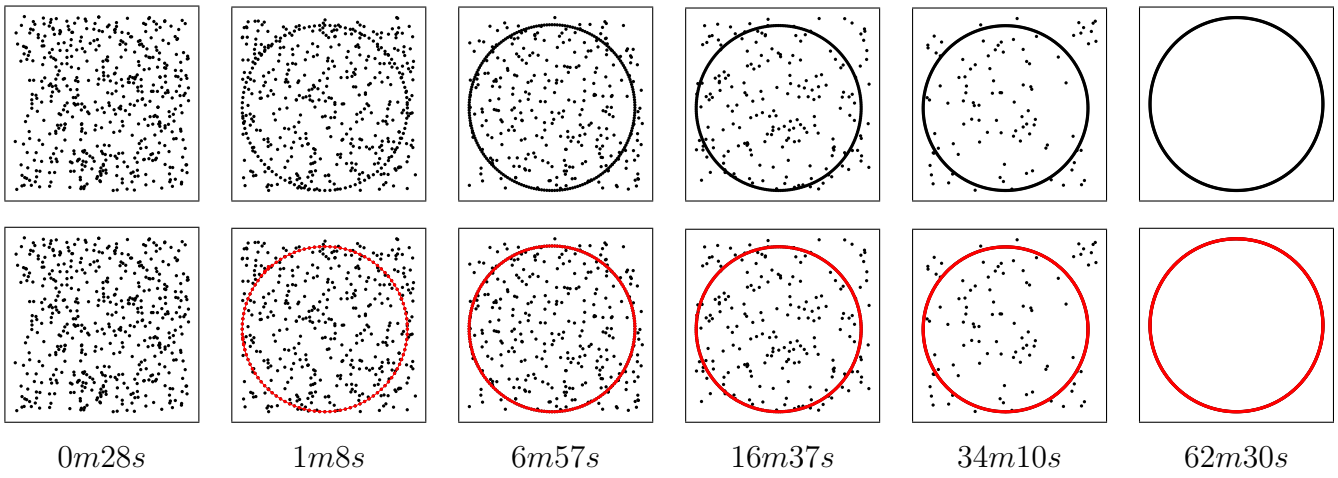

Figure 3: Evaluation of the computation time of the algorithm related to the amount of structure in the dataset. From left to right, the circle contains 0, 100, 200, 300, 400 and 500 equally spaced points, with respectively 500, 400, 300, 200, 100 and 0 random points in the background. The computational complexity of the algorithm is $O(T^3)$, where T is the number of triplets. The more qualifying triplets in the dataset, the longer the algorithm will take.

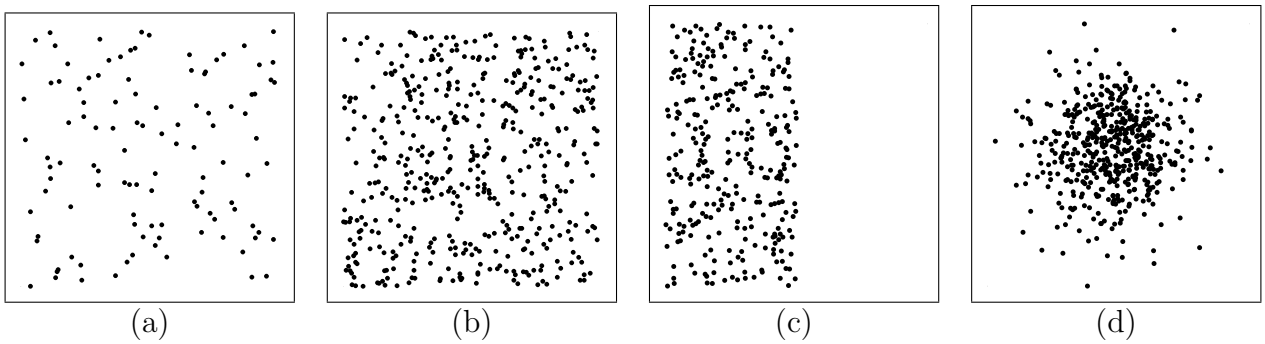

Figure 4: Experiments with figures of random points. The algorithm finds no detections in any of the images. **(a)** A global Poisson process with 100 points. **(b)** A global Poisson process with 500 points. **(c)** A local Poisson process with 300 points. **(d)** A Gaussian point process with 500 points. Running times from (a) to (d): 5.4s, 23.9s, 6.0s, 26.9s.

4 Experiments

This section is divided into three groups of experiments. The first group consists of sets of random points, and is used to control that no detections are made when the points follow the *a contrario* model or null hypothesis. The second group of experiments presents figures where different curvilinear structures are present and the algorithm is able to detect them all. Finally, to illustrate the limitations of the algorithm, a third group of experiments is presented with cases where the algorithm does not give the best interpretation.

A first experiment is to verify that under the *a contrario* hypothesis the detector finds no meaningful structure. Figure 4 shows the result of applying the detector to sets of randomly distributed points. Figures 4(a) and 4 (b) show sets of 100 and 500 uniformly and independently distributed random points. Both sets of points follow exactly the *a contrario* model and no detection is made by the algorithm. Figure 4(c) shows an example where the background model is only satisfied locally. The algorithm is able to correctly handle this situation thanks to the local estimation of points density. Finally, Figure 4(d) presents points with a different random distribution: a Gaussian distribution centered in the image center. This experiment suggests that the *a contrario* hypothesis of a local Poisson process is general enough to model points that are unstructured at a local scale.

Figure 5 shows examples of point sets where curvilinear structures are present and correctly detected. Note how the algorithm automatically determines the number of structures in each figure, and handles the different scales, even when changes of scale occur inside a structure as in Figure 5 3rd row.

The third group of experiments, shown in Figure 6, is presented to illustrate the limitations of the algorithm. The main shortcoming is that the symmetry model penalizes heavily the curvature of a chain of points. This type of problem is shown in Figure 6(a), where the algorithm prefers a shorter but inaccurate curve instead of the longer, more perceptually plausible curve following the perimeter of the figure, but which contains a highly curved section. Figure 6(b) shows an example where short, irregularly spaced segments are not detected as meaningful. The symmetry model, which was chosen for its simplicity, might be too constraining. In this example, it is also possible that the global interpretation of the figure helps the human observer to perceive such segments regardless of the uneven spacing of their points. The penalizing effect of the symmetric model is more evident when the scale of the points spacing is very small, such as in Figure 6(c). Because of the very small distance between the dots, a human observer might not be as sensitive to the irregularities as the algorithm is.

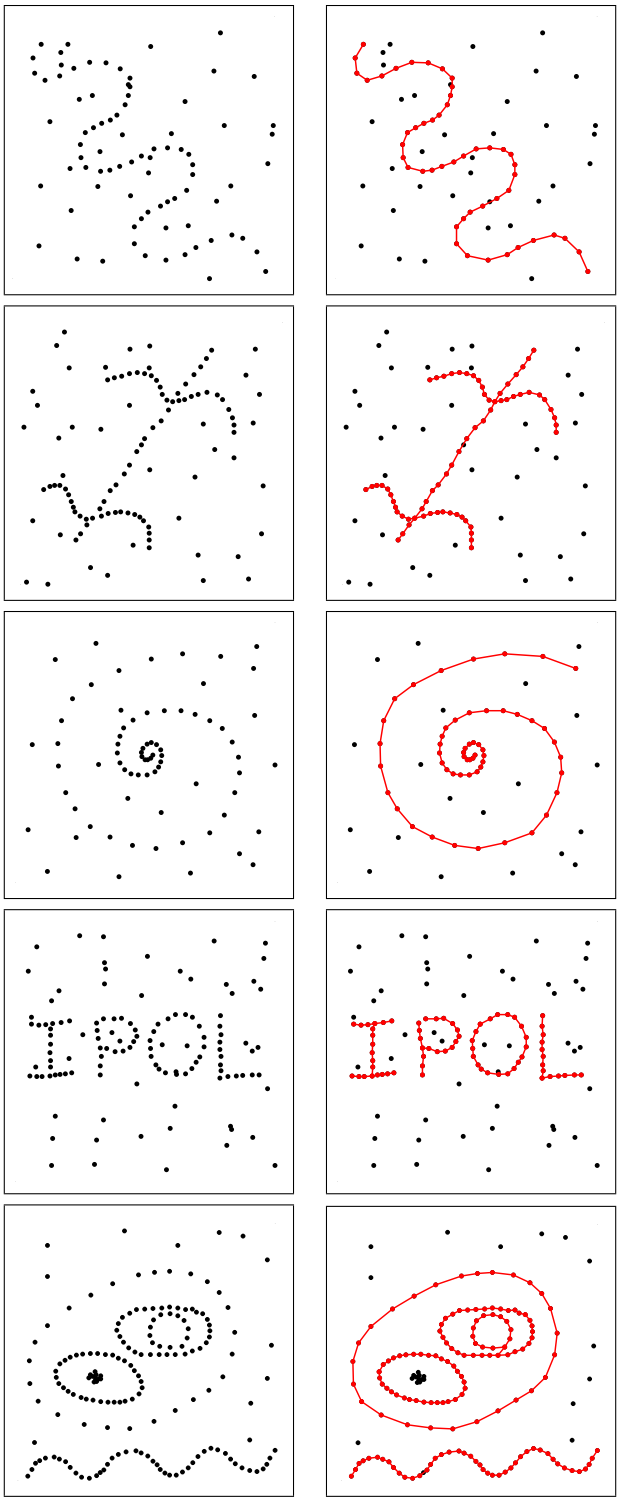

Figure 5: Examples of a successful result given by the algorithm. Left: input points. Right: algorithm result. Running times from top to bottom: 6.0s, 13.0s, 4.5s, 8.9s, 47.5s.

5 Conclusion

We described an algorithm for the automatic detection of configurations of *good continuation* of points in 2D point patterns. While our experiments suggest grouping results compatible with human perception, a more systematic study is needed. From an algorithmic viewpoint, the main limitation

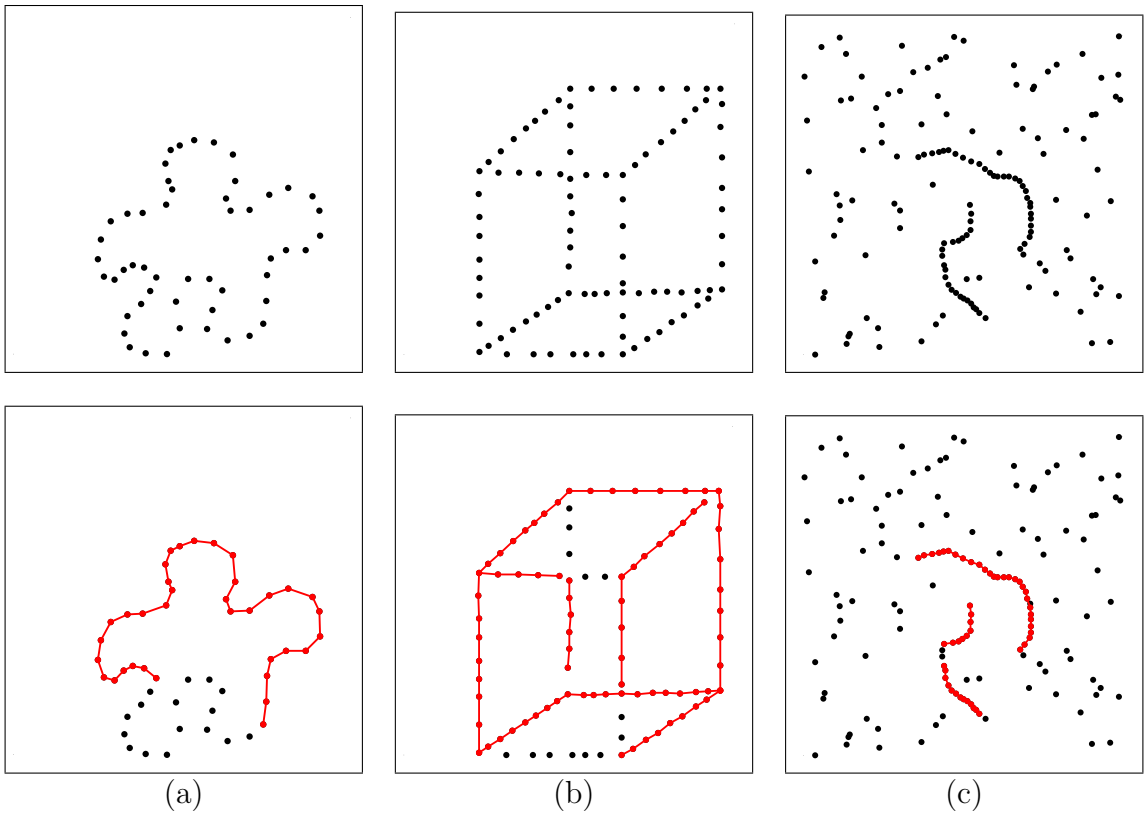

Figure 6: Partially successful examples that illustrate the limitations of the algorithm. **(a)** The model of a circle around the ideal symmetric point makes the algorithm too penalizing on strong curvatures. In this case, the algorithm is not able to follow the perimeter of this shape, which could be the most perceptually plausible interpretation. **(b)** Short, irregularly spaced segments may not reach the threshold of meaningfulness. Yet, perceptually they contribute to the global interpretation of this figure. **(c)** At a very small scale, human perception seems to be more tolerant to deviations from perfect symmetry. However, the algorithm is equally sensitive to strong curvatures, regardless of the scale. Running times from (a) to (c): 1.8s, 21.0s, 20.2s.

is the cubic computational complexity on the number of points, which forbids its utilization for large datasets.

Image Credits

All images by the authors.

References

[1] M. K. Albert and D. D. Hoffman, *Genericity in spatial vision*, in Geometric Representations of Perceptual Phenomena: Articles in Honor of Tarow Indow's 70th Birthday, D. Luce, K. Romney, D. Hoffman, and D'Zmura M., eds., Erlbaum, 1995, pp. 95–112.

[2] A. Desolneux, L. Moisan, and J.M. Morel, *From Gestalt Theory to Image Analysis, a Probabilistic Approach*, Springer, 2008. http://dx.doi.org/10.1007/978-0-387-74378-3.

[3] W.D. Ellis, ed., *A Source Book of Gestalt Psychology*, Humanities Press, 1967 (originally 1938). ISBN 9780415209571.

[4] S. Gori and L. Spillmann, *Detection vs. grouping thresholds for elements differing in spacing, size and luminance. An alternative approach towards the psychophysics of gestalten*, Vision research, 50 (2010), pp. 1194–1202. http://dx.doi.org/10.1016/j.visres.2010.03.022.

[5] G. Kanizsa, *Organization in vision: Essays on Gestalt perception*, Praeger New York:, 1979. ISBN 978-0275903732.

[6] ——, *Grammatica del vedere*, Il Mulino, 1980. ISBN 978-8815060907.

[7] J. Lezama, R. Grompone von Gioi, G. Randall, and J.-M. Morel, *A contrario detection of good continuation of points*, in IEEE International Conference on Image Processing, 2014. http://dx.doi.org/10.1109/ICIP.2014.7025964.

[8] J. Lezama, G. Randall, J.-M. Morel, and R. Grompone von Gioi, *Good continuation in dot patterns: A quantitative approach based on local symmetry and non-accidentalness*, Vision Research, 126 (2016), pp. 183–191. http://dx.doi.org/10.1016/j.visres.2015.09.004.

[9] D. Lowe, *Perceptual Organization and Visual Recognition*, Kluwer Academic Publishers, 1985. http://dx.doi.org/10.1007/978-1-4613-2551-2.

[10] D. Lowe and T. O. Binford, *The interpretation of three-dimensional structure from image curves*, Proceedings of IJCAI-7, (1981), pp. 613–618.

[11] W. Metzger, *Gesetze des Sehens*, Verlag Waldemar Kramer, Frankfurt am Main, third ed., 1975. ISBN 978-3880744929.

[12] S. E. Palmer, *Vision Science: Photons to Phenomenology*, The MIT Press, 1999. ISBN 978-0262161831.

[13] I. Rock, *The Logic of Perception*, The MIT Press, 1983. ISBN 978-0262680455.

[14] J. Wagemans, J. H. Elder, M. Kubovy, S. E. Palmer, M. A. Peterson, M. Singh, and R. von der Heydt, *A century of gestalt psychology in visual perception: I. perceptual grouping and figureground organization.*, Psychological Bulletin, 138 (2012), pp. 1172–1217. http://dx.doi.org/10.1037/a0029333.

[15] J. Wagemans, J. Feldman, S. Gepshtein, R. Kimchi, J. R. Pomerantz, P. A. van der Helm, and C. van Leeuwen, *A century of gestalt psychology in visual perception: II. conceptual and theoretical foundations.*, Psychological Bulletin, 138 (2012), pp. 1218–1252. http://dx.doi.org/10.1037/a0029334.

[16] M. Wertheimer, *Untersuchungen zur Lehre von der Gestalt. II*, Psychologische Forschung, 4 (1923), pp. 301–350. An abridged translation to English is included in [3]. http://dx.doi.org/10.1007/BF00410640.

[17] A. P. Witkin and J. M. Tenenbaum, *On the role of structure in vision*, in Human and Machine Vision, J. Beck, B. Hope, and A. Rosenfeld, eds., Academic Press, 1983, pp. 481–543. http://dx.doi.org/10.1016/B978-0-12-084320-6.50022-0.