Dominant point detection based on discrete curve structure and applications

Hayat Nasser, Phuc Ngo, Isabelle Debled-Rennesson

▶ To cite this version:

Dominant point detection based on discrete curve structure and applications

Hayat Nasser[a,b], Phuc Ngo[a,b], Isabelle Debled-Rennesson[a,b]

[a] *Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France*
[b] *CNRS, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France*

Abstract

In this paper, we investigate the problem of dominant point detection on digital curves which consists in finding points with local maximum curvature. Thanks to previous studies of the decomposition of curves into sequence of discrete structures [1, 2, 3], namely maximal blurred segments of width ν [4], an initial algorithm has been proposed in [5] to detect dominant points. However, an heuristic strategy is used to identify the dominant points. We now propose a modified algorithm without heuristics but a simple measure of angle. In addition, two applications are as well presented: (1) polygonal simplification to reduce the number of detected dominant points by associating a weight to each of them, and (2) classification using the polygon issued from the reduced dominant points. The experimental results demonstrate the efficiency and robustness of the proposed method.

Keywords:
Dominant point, polygonal simplification, discrete structure, maximal blurred segments, classification

1. Introduction

Dominant points (DP) of discrete curves are identified by Attneave [6] as the local maximum curvature points on a curve. Such points represent a rich information which is sufficient to characterize this curve. The dominant points are considered as representative features for the object contours, because they preserve the significant features of the digitized curve of images. They play a critical role in curve approximation, image regognition and matching and in other domains of computer vision [6]. Dominant point detection methods choose points from a digital contour that can be used to represent the curve effectively using less number of points. The digital curve is then represented as a polygon with dominant points as vertices and the line segments connecting adjacent dominant points as edges. So dominant points can be used to segment the contour. Many works have been conducted regarding the dominant point detection [5, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19], in which several problems have

been identified: time computation, number of parameters, selection of starting point, bad results with noisy curves,...

Recent dominant point detection methods can be classified as follows (see [20] for more details): some used dynamic programming [21, 22, 23, 24], while others used splitting [25, 26], merging [27], digital straight segments [5, 9], suppression of breakpoints [11, 13, 14, 28], curvature and convexity [17, 29, 30].

A large number of algorithms for dominant points detection uses a notion of region of support (ROS) [31, 17, 29, 11, 28] for each point on the curve contour. A region of support of a point of a curve C contains its neighboring points on C. For each point of C, a particular measure is computed in relation with its ROS. The dominant points are determined by suppression of the points with the lowest measure values. On the other hand, there exists a lot of works that use an approximation of curvature value [17, 29, 30] to determine DPs. Other methods use the principle of suppression of breakpoints [11, 13, 24, 25, 26, 28, 32] to obtain DPs.

Nguyen *et. al* proposed in [5] a sequential method deduced from theoretical results of discrete geometry, it only requires to set one parameter, it is invariant to the choice of the starting point and it works with discrete curves possibly being noisy or disconnected. It relies on the geometrical structure of the studied curve, in particular the decomposition of the curve into maximal blurred segments for a given width [4, 2]. However at the end of the proposed method, the choice of dominant points is done by an heuristic strategy.

In this paper, we imprive Nguyen's method [5] to be non-heuristic. We use the mono-width value to decompose the curve into maximal blurred segments. An improvement of this dominant point detection algorithm is proposed with an efficient choice of dominant points by considering a simple angle measure.

Furthermore, an algorithm of polygonal simplification is also proposed to reduce the number of detected dominant points while preserving the principal angular deviations in the discrete curve of the shape border. We propose two strategies to polygonal simplification. The first one is to fix the number of dominant points to obtain. The second one is to apply an automatic selection process on the detected dominant points according to an error criteria. We also show the effectiveness of our dominant point detection method in the field of classification.

The rest of paper is presented as follows: in section 2, we recall results of discrete geometry used in this paper to analyze a curve. Then, in section 3, we propose improvements of the previous dominant point detection algorithm [5]. In section 4, we present experimental results, comparisons with previous method. Section 5 presents two strategies for polygonal simplification and comparisons between our proposed method and some other algorithms of polygonal simplification. In section 6, we propose an application to classification, and the main conclusion is summarised in section 7.

2. Geometrical tools for discrete curves analysis

We recall hereafter several notions concerning discrete lines [33], blurred segments [1] and maximal blurred segments [4] which are used throughout the paper.

A ***discrete line*** $\mathcal{D}(a, b, \mu, \omega)$ (with a, b, μ and ω being integer such that $gcd(a, b) = 1$) is the set of integer points (x, y) verifying $\mu \leq ax - by < \mu + \omega$.

Let us consider $\mathcal{S}_f$ as a sequence of integer points. A discrete line $\mathcal{D}(a, b, \mu, \omega)$ is said ***bounding for*** $\mathcal{S}_f$ if all points of $\mathcal{S}_f$ belong to $\mathcal{D}$. Finally, a bounding line is said ***optimal*** if its vertical (or horizontal) distance $\frac{\omega - 1}{max(|a|, |b|)}$ is minimal, i.e. if its vertical (or horizontal) distance is equal to the vertical (or horizontal) distance of the convex hull of $\mathcal{S}_f$. This notion is illustrated in Fig. 1 and leads to the blurred segment.
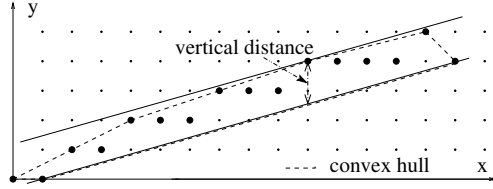

Figure 1: $\mathcal{D}(2, 7, -8, 11)$, the optimal bounding line of the set of points (vertical distance = $\frac{10}{7} = 1.42$).

$\mathcal{S}_f$ is a ***blurred segment of width*** ν if its optimal bounding line has a vertical or horizontal distance less than or equal to ν i.e. if $\frac{\omega - 1}{max(|a|, |b|)} \leq \nu$.

The notion of maximal blurred segment on a given discrete curve was introduced in [4]. Let C be a discrete curve and $C_{i,j}$ a sequence of points of C indexed from i to j. Let us suppose that the predicate "$C_{i,j}$ is a blurred segment of width ν" is denoted by $BS(i, j, \nu)$.

Definition 1. $C_{i,j}$ *is called a **maximal blurred segment of width** ν and noted* $MBS(i, j, \nu)$ *iff* $BS(i, j, \nu)$, $\neg BS(i, j + 1, \nu)$ *and* $\neg BS(i - 1, j, \nu)$.

An incremental algorithm was proposed in [4] to determine the set of all maximal blurred segments of width ν of a discrete curve C. The main idea is to maintain a blurred segment when a point is added (or removed) to (from) it. The following important property was proved.

Property 1. *Let* $MBS_\nu(C)$ *be the set of width ν maximal blurred segments of the curve C. Then,*
$MBS_\nu(C) = \{MBS(B_0, E_0, \nu), MBS(B_1, E_1, \nu), \ldots, MBS(B_{m-1}, E_{m-1}, \nu)\}$
and satisfies $B_0 < B_1 < \ldots < B_{m-1}$. *So we have:* $E_0 < E_1 < \ldots < E_{m-1}$ *(see Fig. 2).*
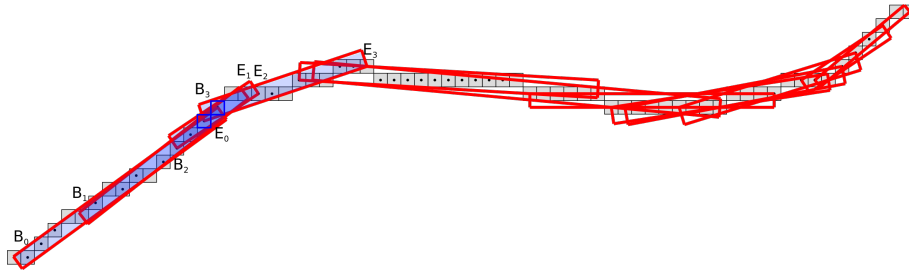
3

Figure 2: The set of width 1.4 maximal blurred segments of the curve with grey pixels (each red box bounds the pixels of a maximal blurred segment of width 1.4). The indexes of the begin (B_i) and end (E_i) points of the four first maximal blurred segments are indicated $(B_0 = 0$ and $E_0 = 15$, $B_1 = 5$ and $E_1 = 17$, ...). The points with blue frame are points in the common zone – intersecting zone – defined by these four first maximal blurred segments.

For a given width ν, the set of the maximal blurred segments of a curve C entirely determines the structure of C. It can be considered as an extension to thick curves of the notion of **tangential cover** introduced by Feschet and Tougne in [3] and used to obtain curvature profile or fast polygon approximation [10].

In [5], a method for dominant point detection, constructed from the set of maximal blurred segments, was given. In the next section, we propose an improvement of this method which allows new applications.

3. Dominant point detection

The set of maximal blurred segments of a curve provides important information about the geometrical structure of the studied curve. The width of the maximal blurred segments permits to work in different scales and to consider the noise possibly present in a curve C.

In [6], Attneave defined a dominant point (corner point) on a curve as a point of local maximum curvature. In a similar way, the method proposed in [5] uses a notion of region of support (ROS) of a point in a discrete curve C, defined with a sequence of discrete points, $C = (C_i)_{i=1..n}$.

Definition 2. *Width ν maximal left and right blurred segments of a point constitute its region of support (ROS). The angle between them is called the ROS angle of this point.*

The method first chooses a width ν and computes $MBS_\nu(C)$, the set of maximal blurred segments of C. The main idea of the method is that dominant points of C are localised in the common zones of successive maximal blurred segments of C(see Fig. 2). Hereafter, we recall the properties and the heuristic strategy used in [5]. We show that problems can occur and propose a solution.

4

3.1. Heuristic strategy of method [5]

Let us now consider the common zone of more than 2 successive maximal blurred segments.

Proposition 1. *The smallest common zone of successive width ν maximal blurred segments, of which slopes are either increasing or decreasing, contains a candidate as dominant point.*

Proposition 2. *A maximal blurred segment contains at most 2 dominant points.*

Based on the previous propositions, the algorithm proposed by Nguyen *et al.* in [5] consists in finding the smallest common zone induced by successive maximal blurred segments of width ν which have either increasing or decreasing slopes. It is stated that such a zone contains the candidates of dominant point, since the points in this zone have the smallest ROS angle. By an heuristic strategy, the dominant point is identified as ***the middle point*** of this zone.

The choice of the middle point is an heuristic and, in the next section, we propose a new strategy, related to curvature, to identify the dominant point in the common zone.

3.2. New dominant point detection algorithm

In this section, we present a modified algorithm of [5] for detecting the dominant points of discrete curves with high accuracy.

Since the purpose of dominant point detection is to detect significant points with curvature extrema, we propose a new decision-making strategy of dominant point in common zones. For this, we use a simple measure, as a pseudo-curvature, estimated for each point located in the interest regions. Inspired by the k-cosine measure [31], we use an angle measure adapted to the decomposition into width ν blurred segments of the input curve. More precisely, it is the angle between the considered point and the two left and right extremities of the left and right maximal blurred segments, involved in the localisation of the studied common zone and passing through the point. It should be mentioned that this angle measure is generally not symmetric in ROS, contrary to the k-cosine measure. Obviously, the smaller angle, the greater curvature and thus the higher dominant character of the point. Accordingly, the dominant point is identified as the point, in the common zone, having a local minimum measure of angle.

The modified algorithm is given in Algorithm 1. It is similar to the one in [5] except for the selection of dominant points as described above. Moreover we simplify the process because we do not need to decompose the slopes of the maximal blurred segments located in the common zones into monotone sequences. Indeed, the angle computation permits to well localize the dominant point in each common zone.

Particularly, in this algorithm, each common zone is computed by Line 6, where q is the index of the first maximal blurred segment involved in the common
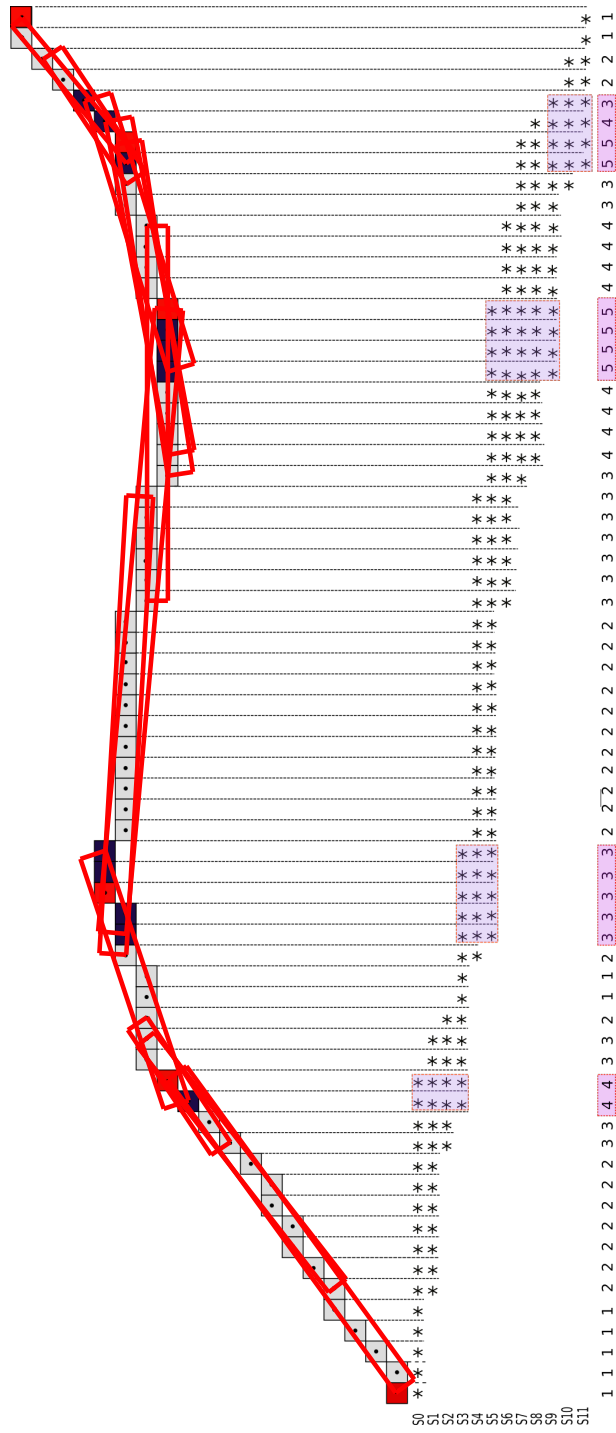
Figure 3: Interest regions (common zones) obtained with the set of width 1.4 maximal blurred segments of the curve with grey pixels.

Algorithm 1: Dominant point detection.

Input : C discrete curve of n points, ν width of the segmentation
Output: D set of dominant points

---
1 **begin**
2   Build $MBS_\nu = \{MBS(B_i, E_i, \nu)\}_{i=0}^{m-1}$;
3      $n = |C|$; $m = |MBS_\nu|$;
4      $q = 0$; $p = 1$; $D = \emptyset$;
5   **while** $p < m$ **do**
6   **while** $E_q > B_p$ **do** $p{+}{+}$;
7          $D = D \cup \min\{Angle(C_{B_q}, C_i, C_{E_{p-1}}) \mid i \in [\![B_{p-1}, E_q]\!]\}$;
8          $q = p - 1$;

---

zone and $p-1$ is the index of the last. The points in the common zone are points C_i for $i \in [\![B_{p-1}, E_q]\!]$.

Moreover, at Line 7, the function *Angle* calculates, for each point in the common zone, the angle between the point and its two extremities. Then, in a common zone, the dominant point is the point with local minimum angle. Afterwards, the next common zone is computed by considering the last segment (index $p-1$) of the previous zone as the first one (index q) of the next common zone. This process permits to detect the points of the curve through which a maximum number of maximal blurred segments is passing (see Fig. 3).

If the curve C is not closed, the first and the last points of C are dominant points and they must be added to D.

The algorithm is illustrated in Fig. 3; a curve C is analysed by the algorithm, the set $MBS_{1.4}(C)$ is computed and 12 maximal blurred segments (named $s0$ to $s11$ in the figure) are obtained. Their localisation in the curve is indicated by one star ($*$) for each point belonging to the considered maximal blurred segment. Four common zones (dark pixels on the curve) are detected by the algorithm: C_{14} to C_{15}, C_{22} to C_{26}, C_{49} to C_{52} and C_{59} to C_{62}. The pink boxes in the figure permit to count the number of maximal blurred segments passing through each point of common zones. The angles are calculated by considering the starting point of the first maximal blurred segment involved in the common zone and the end point of the last involved maximal blurred segment. The table 1 shows the angles at each point of common zones and permits to deduce the dominant points (minimal values are in bold in the table and the corresponding points are in red in Fig. 3). In the common zone 4, the selected dominant point is not located at the middle of the common zone.

Complexity analysis. The complexity of the proposed algorithm is the same as the one in [5], which depends on the complexity of decomposing a curve of n points into maximal blurred segments. We can use the technique proposed in [2] to obtain the tangential cover (corresponding to the set of maximal blurred segments for a given width) and the complexity of this method is in $O(n \log n)$.

7

Common Zone	Points	Angle
Common Zone 1	C_{14}	162.9
B_{S0} and E_{S3}	C_{15}	**159**
Common Zone 2	C_{22}	155.6
B_{S3} and E_{S5}	C_{23}	157.6
	C_{24}	**152**
	C_{25}	153.6
	C_{26}	154.9
Common Zone 3	C_{49}	158.6
B_{S5} and E_{S9}	C_{50}	157.4
	C_{51}	156
	C_{52}	**154.3**
Common Zone 4	C_{59}	155.7
B_{S9} and E_{S11}	C_{60}	**150.5**
	C_{61}	155.3
	C_{62}	160.2

Table 1: Angles at each point of common zones (see Fig. 3).

Parameter selection. The only parameter of the proposed approach is the width ν of maximal blurred segment computation of the input curve. This parameter can be selected by the user according to the desired polygonal simplification. More precisely, the smaller ν captures better details –including noise– of the curve and the bigger ν allows a smoother dominant point detection and reduces the influence of noise along the curve. When the noise is regularly distributed, it is proposed in [34] to use the notion of **meaningful thickness** [35, 36, 37] to fix this value of ν. In particular, ν equals to the average of meaningful thickness values detected along the curve.

4. Evaluation results

We now present some experimental results of dominant point detection using the proposed method, and evaluate by widely used error criteria [16, 15] described in Sec 4.1. Using these criteria, in Sec 4.2, we show the improvements of the modified algorithm over the original method of Nguyen [5].

4.1. Evaluation criteria

In order to assess the effectiveness of the proposed method, we consider the following five evaluation criteria:

1. Number of dominant points (#DP).
2. Compression ratio (CR) is defined as the ratio between number of curve points and number of detected dominant points. The larger CR, the more effective data simplification.

$$CR = \frac{n}{\#DP}$$

3. Integral sum of square errors (ISSE) is the sum of squared distances of the curve points from approximating polygon. The smaller ISSE, the better description of the shape by the approximating polygon.

$$ISSE = \sum_{i=1}^{n} d_i^2$$

where d_i is the distance from the ith curve point to the approximating polygon.

4. Maximum error (L_∞) is the maximum distance of the curve points from approximating polygon. The smaller L_∞, the better fitness of polygonal approximation.

$$L_\infty = max\{d_i\}_{i=1}^{n}$$

5. Figure of merit (FOM) is estimated as ratio between CR and ISSE. FOM is a compromise between the low approximation error and the benefit of high data reduction.

$$FOM = \frac{CR}{ISSE}$$

4.2. Effectiveness compared to Nguyen's algorithm

In this section, we present the experimental results of the modified algorithm (see Sec 3.2) and we compare them with Nguyen's algorithm [5]. The experiments are carried out on the data with and without noise. For each input curve, a fixed width $\nu = 1.5$ of maximal blurred segments is used for both algorithms, the results are shown in Fig. 4 and Table 2. It can be seen that selecting the middle point of common zone is not always a relevant strategy, in particular in the high-pass zones.

5. Applications to polygonal simplification

One goal of finding the dominant points is to have an approximate description of the input curve, called **polygonal simplification** or **approximating polygon**. On the one hand, due to the nature of the maximal blurred segment sequence defined on a discrete curve, we observe that the common zones − contain candidates of dominant points − may be close to each others. On the other hand, using the algorithm proposed in Sec 3.2, the dominant point is selected according to its angle with the extremities of maximal segments. As a consequence, detected dominant points are sometimes redundant or stay very
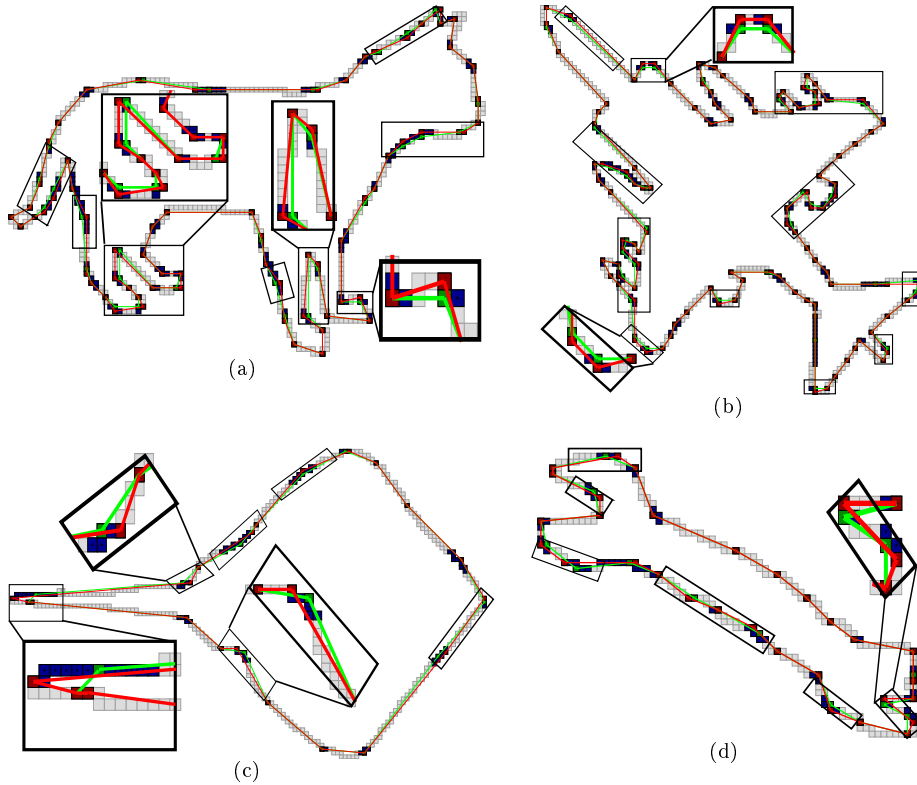
9

Figure 4: Improved results with respect to Nguyen [5] (see also Table 2). The effectiveness of the modified algorithm is more significant in the *highly derivative zones* of the shape as highlighted by the black boxes. Green (resp. Red) points are dominant points detected by Nguyen's (resp. modified) algorithm. Lines in green (resp. red) denote the polygonal approximation from detected dominant points. Blue points are candidates as dominant points in the interest regions.

near, which is presumably undesirable in particular for polygonal simplification. More precisely, this leads to an overmuch dominant points and thus the polygon induced by such points is not always a very good and suitable solution of curve simplification.

In order to achieve a high compression of approximating polygon of the input curve, certain dominant points need to be eliminated. To do so, we associate to each detected dominant point a weight indicating its importance with respect to the approximating polygon of the curve. Such a weight must: (1) be related to some error criteria of approximating polygon, and (2) not induce costly computation.

From Sec 4.1, the criterion ISSE describes the distortions caused by the approximated polygon of a curve. More precisely, ISSE allows to evaluate how much the approximated polygon is similar to the curve, thus smaller ISSE means

10

Curve	Method	#DP	ISSE	L_∞	FOM
(a)	Nguyen	68	149.15	1	0.0523
$n = 536$	Algo 1	68	**125.763**	1	**0.0627**
(b)	Nguyen	105	202.809	1	0.0339
$n = 722$	Algo 1	105	**166.74**	1	**0.0412**
(c)	Nguyen	20	236.806	3.536	0.0853
$n = 404$	Ours	20	**150.314**	**1.539**	**0.1344**
(d)	Nguyen	43	68.896	1	0.0851
$n = 252$	Ours	43	**57.582**	1	**0.1018**

Table 2: Comparison with Nguyen's method [5] on Fig. 4 using significant measures described in Sec 4.1.

better description of the curve by the polygon. This error evaluation is suitable to our criterion of weight associated to dominant points since: (1) we would like to eliminate points that less affect the possible error (i.e. less ISSE), and (2) in particular the ISSE can be calculated *locally* using its two neighbouring dominant points. More precisely, let D_i be the considering dominant point, v_{i-1} and v_{i+1} be respectively the indexes in C of the two neighbouring dominant points D_{i-1} and D_{i+1} of D_i. Then, we have

$$ISSE_i = ISSE(D_i) = \sum_{j=v_{i-1}}^{v_{i+1}} d^2(C_j, \overline{D_{i-1}D_{i+1}}),$$

where $d(P, \overline{AB})$ is distance from the point P to the line segment $\overline{AB}$.

Indeed, the value of ISSE between the approximated polygon and the curve before and after deleting a dominant point differs only at the part related to the point, in particular its two neighbours as illustrated in Fig.5. The ISSE induced by the two neighbours of a dominant point thus characterizes the importance of the point; the greater ISSE, the more important the point.

Still in Fig.5, we remark that dominant point may have a small ISSE, however regarding its neighbours, in particular the angular relationship, it is more important than the others. In other words, this angle with the neighbours plays an important role in the decision of suppressing a dominant point. This leads to a consideration of weight associated to each dominant point detected that not only involves ISSE but also the angle to its two neighbours. In particular, this **weight** is determined by the ratio of ISSE and angle, *i.e.*, $weight_i = ISSE_i/angle_i$.

A question raised is how many dominant points should be removed. To handle this, there are **two strategies**:

- The first one consists in **fixing the desired number m of dominant points** on the approximated polygon, roughly speaking, the process of

polygonal simplification is performed by removing one by one the dominant points of small weight until reaching m.

- The second approach is **_to apply an automatic simplification process_** on the detected dominant points according to an error criteria.

5.1. Automatic polygonal simplification algorithm

Since, the goal is (1) to keep the less dominant point possible, and (2) to have a good description of the curve by the approximated polygon. The first requirement can be measured by CR criterion, while the second one by FOM criterion. In other words, our objective function of the automatic process is to minimize the multiplication of CR and FOM; i.e., $CR^2/ISSE$ (namely, FOM_2). Roughly speaking, the process of polygonal simplification is performed by removing one by one dominant point having the smallest weight until the objective function FOM_2 is still decreased (see Algorithm 2).

Algorithm 2: Automatic polygonal simplification from detected dominant points

---

    Input   **:** D set of dominant points obtained with Algorithm 1
    Output: $\widehat{D}$ new set of dominant points that minimizes FOM_2 criterion

1 **begin**
2      $\widehat{D} = D$;
3      $F = FOM_2(D)$;
4   **repeat**
5          $\widehat{F} = F$;
6          $\widehat{D} = \widehat{D} \setminus \{\min_{p \in \widehat{D}} weight(p)\}$;
7          $F = FOM_2(\widehat{D})$;
8   **until** $F < \widehat{F}$;

Algorithm 2 uses two functions:

- $FOM_2(D)$ calculates $FOM_2 = CR^2/ISSE$ of the set D

- $weight(p)$ calculates the $weight = ISSE/angle$ associated to the point p.

Note that one point is removed at each iteration thus the values of FOM_2 and $weigth$ are changed only around the removed point. In other words, FOM_2 and $weigth$ is calculated for all points once at the beginning. Then at each iteration, we need to recalculate only for two neighbours of the removed point.

It should be mentioned that Algorithm 2 can be adapted to the strategy in which we fix the number m of dominant points on the output approximated polygon. This is done by modifying Line 8 with the following condition about the number of dominant points in $\widehat{D}$: $|\widehat{D}| \leq m$.

In the next section, we test the proposed method and compare it to the other polygonal simplification techniques based on dominant points extraction.

B

(ISSE=2.59,angle=3.02)
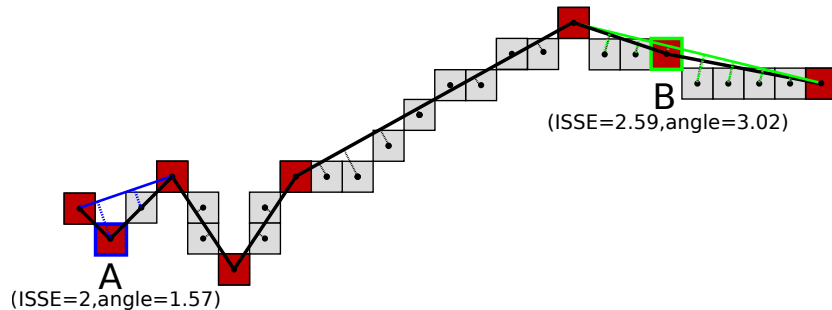
A

(ISSE=2,angle=1.57)

Figure 5: Illustration of calculating the weights associated to detected dominant points. In grey (resp. red) are points (resp. dominant points) of the curves. The continuous black line connecting the dominant points denotes the approximated polygon, while dashed lines are perpendicular distances from points on the curve to the polygon, the sum of these distances defines the ISSE. Considering the point A in blue (resp. B in green) frame, deleting this point leads to a *local* modification in the approximated polygon, and thus ISSE induced by blue (resp. green) dashed lines. One remarks that ISSE of the point in blue frame is smaller than the green, however according to the neighbours – in angle – it is more important than the green one. Therefore, the weight associated to each dominant point involves both ISSE and angle criteria, in particular $\frac{ISSE}{angle}$. Then, the weight of A and B are 1.27 and 0.86, respectively.

In the experiments, the automatic polygonal simplication algorithm is called **Algorithm 2**$_{APS}$ and the variant algorithm, fixing the number of dominant points to obtain, is called **Algorithm 2**$_{FDP}$.

5.2. Tests and comparisons with other methods

5.2.1. Tests of the two simplification strategies

Fig.6 shows polygonal simplification results on noisy digital curves. The results demonstrate the effectiveness and robustness of the simplication method on noisy data. The red curves are the polygonal representations directly deduced from the dominant points obtained by Algorithm 1. On the right hand side of Fig.6, the green curves are the polygonal simplifications obtained with Algorithm 2$_{FDP}$: for example a reduction of 20% from the dominant point number initially obtained with Algorithm 1. On the left hand side, the green curves are obtained with Algorithm 2$_{APS}$.

Table 3, shows the different evaluation criteria of the two approaches for polygonal simplification w.r.t Algorithm 1. In all of experiments, the two approaches present an improvement on FOM and FOM_2 regarding to Algorithm 1. In particular, the results of Algorithm 2$_{APS}$ have the highest value of FOM_2 comparing to the others. It should be recalled that Algorithm 2$_{APS}$ is an automatic polygonal simplification by finding a local maximum of FOM_2.
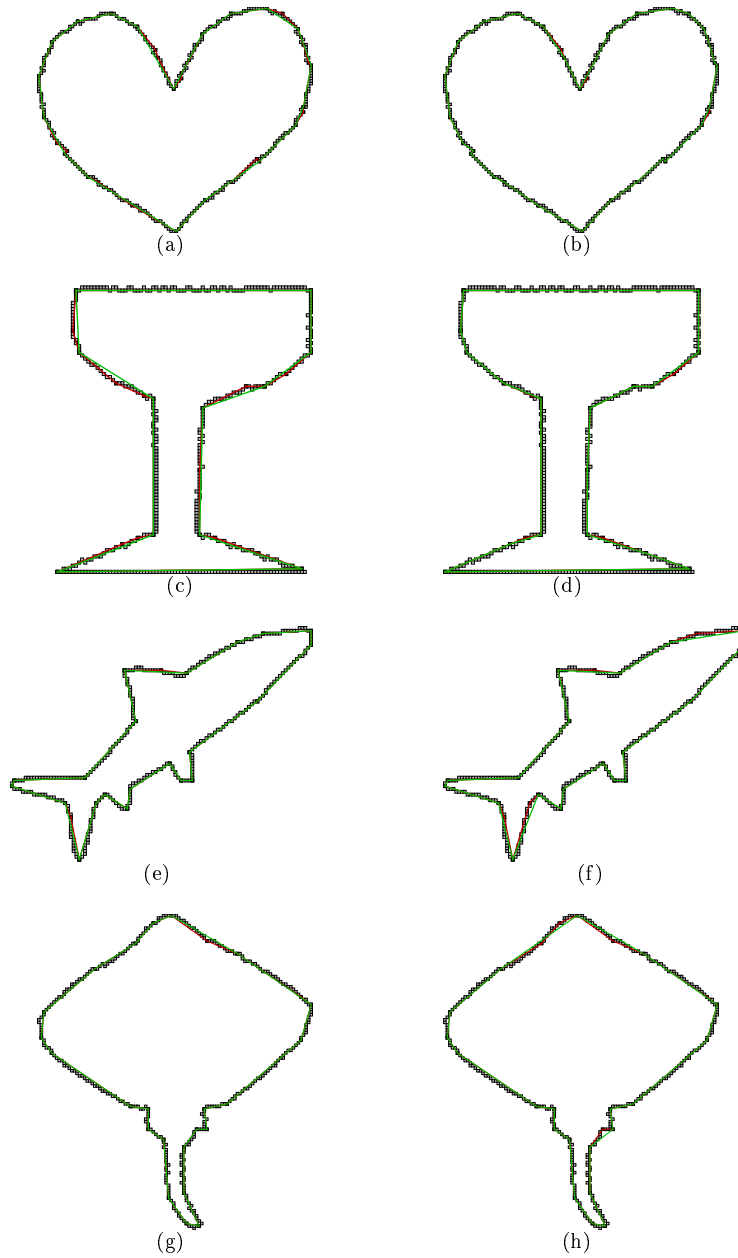
Figure 6: Polygonal representation (in red) on noisy data using the dominant points detected by Algorithm 1. Polygonal simplification results (in green) using mean value of meaningful thickness as width parameter : on the right we have a reduction of 20% of the initial dominant point number using Algorithm 2_{FDP} and on the left we have the automatic simplification using Algorithm 2_{APS}.

Curve	Method	#DP	CR	ISSE	L_∞	FOM	FOM$_2$
(a) and (b)	Algo 1	37	8.162	71.205	1.29	0.114	0.935
$n = 302$	Algo 2$_{APS}$	17	17.764	85.669	1.66	**0.185**	**2.949**
	Algo 2$_{FDP}$	30	10.066	**68.009**	**1.29**	0.148	1.490
(c) and (d)	Algo 1	21	19.285	160.701	1.299	0.120	2.314
$n = 405$	Algo 2$_{APS}$	11	36.818	238.449	2.12	**0.154**	**5.684**
	Algo 2$_{FDP}$	17	23.823	**159.557**	**1.245**	0.149	3.557
(e) and (f)	Algo 1	23	12.739	76.675	1.435	0.166	2.116
$n = 293$	Algo 2$_{APS}$	21	13.952	**72.701**	**1.340**	**0.191**	**2.677**
	Algo 2$_{FDP}$	19	15.421	105.003	1.902	0.146	2.264
(g) and (h)	Algo 1	25	13.12	**118.661**	**1.691**	0.110	1.450
$n = 328$	Algo 2$_{APS}$	23	14.260	121.376	**1.691**	**0.117**	**1.675**
	Algo 2$_{FDP}$	20	16.4	165.749	2.455	0.098	1.622

Table 3: Comparisons with the results of Algorithm 1 and the results of the two strategies of polygonal simplification (fixed DP number, algo 2$_{FDP}$ and automatic polygonal simplification, algo 2$_{APS}$) using significant measures described in Sec 4.1.

5.2.2. *Comparisons with other methods*

We propose two types of experiments[1]:

- On the one hand, on curves with few points, tested by the well-known methods of the literature [11, 13, 17]. We compared the values provided by their authors with those computed with our method.

- On the other hand, we have tested three other methods [7, 38, 39], using the source code made available by the authors. The tests were carried out on curves containing numerous points with irregularities. The obtained results are compared with those obtained with our method.

The experiments are first carried out on three benchmarks: chromosome, leaf and semicircle as shown in Fig.7, 8 and 9 respectively. Table 4 compares the proposed method (Algorithm 2$_{FDP}$) with other popular algorithms such as: Masood [13], Marji [11] and Teh [17], on the benchmarks by the evaluation criteria described in Sec 4.1. In all of experiments, the Algorithm 2$_{FDP}$ presents an improvement on FOM, and has a better approximation error in ISSE and L_∞. However, the tested curves contain few points and are not noisy.

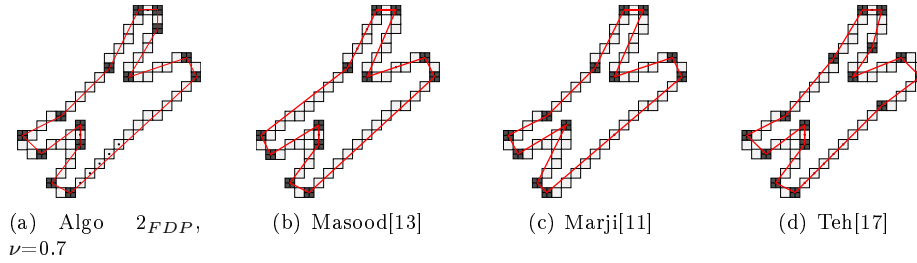[1]The files of the tested curves are available at `https://members.loria.fr/IDebledRennesson/files/DataJCSS.zip`

(a) Algo 2_{FDP}, ν=0.7

(b) Masood[13]

(c) Marji[11]

(d) Teh[17]

Figure 7: Dominant points of the chromosome curve.

(a) Algo 2_{FDP}, ν=0.7

(b) Masood[13]

(c) Marji[11]

(d) Teh[17]
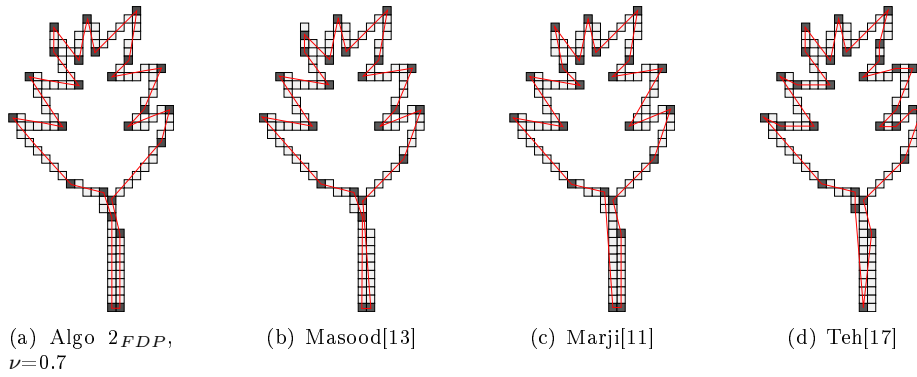
Figure 8: Dominant points of the leaf curve.

(a) Algo 2_{FDP}, ν=0.7

(b) Masood[13]

(c) Marji[11]

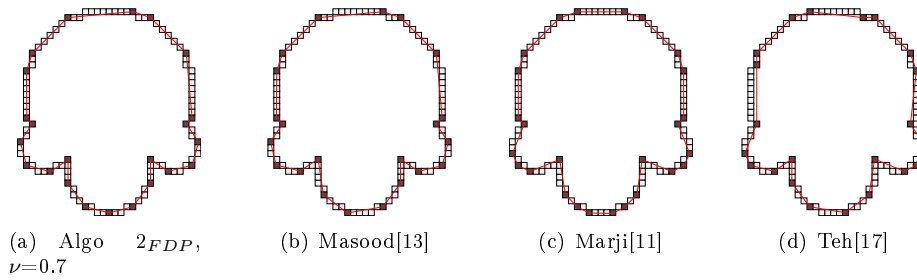(d) Teh[17]

Figure 9: Dominant points of the semicircle curve.

Therefore, other complementary comparisons were also performed with three recent methods. The first one (Liu [38]) is the polygonal reconstruction from the visual curvature which uses a parameter associated to the scale s of the contour analysis. An application of this method is the detection of corners by varying the scale value.

Curve	Method	#DP	CR	ISSE	L_∞	FOM	FOM$_2$
Chromosome n=60	Algo 2_{FDP}, ν=0.7	14	4.28	**4.97**	**0.70**	**0.86**	**3.68**
	Masood [13]	12	5	7.76	0.88	0.65	3.25
	Marji [11]	12	5	8.03	0.89	0.62	3.1
	Teh [17]	15	4	7.20	0.74	0.55	2.2
Leaf n=120	Algo 2_{FDP}, $\nu = 0.7$	24	4.95	**9.96**	**0.74**	**0.49**	2.43
	Masood [13]	23	5.21	10.61	**0.74**	**0.49**	**2.55**
	Marji [11]	22	5.45	13.21	0.78	0.41	2.23
	Teh [17]	29	4.14	14.96	0.99	0.27	1.11
Semicircle n=102	Algo 2_{FDP}, ν=0.7	23	4.43	**7.63**	**0.72**	**0.58**	**2.56**
	Masood [13]	22	4.64	8.61	**0.72**	0.54	2.50
	Marji [11]	26	3.92	9.01	0.74	0.43	1.68
	Teh [17]	22	4.64	20.61	1	0.22	1.02

Table 4: Results of the proposed method and of the other methods.

(a) Algo 2_{APS}, $\nu = 1.9$, #DP=63

(b) Liu[38], s=0.063, #DP=63

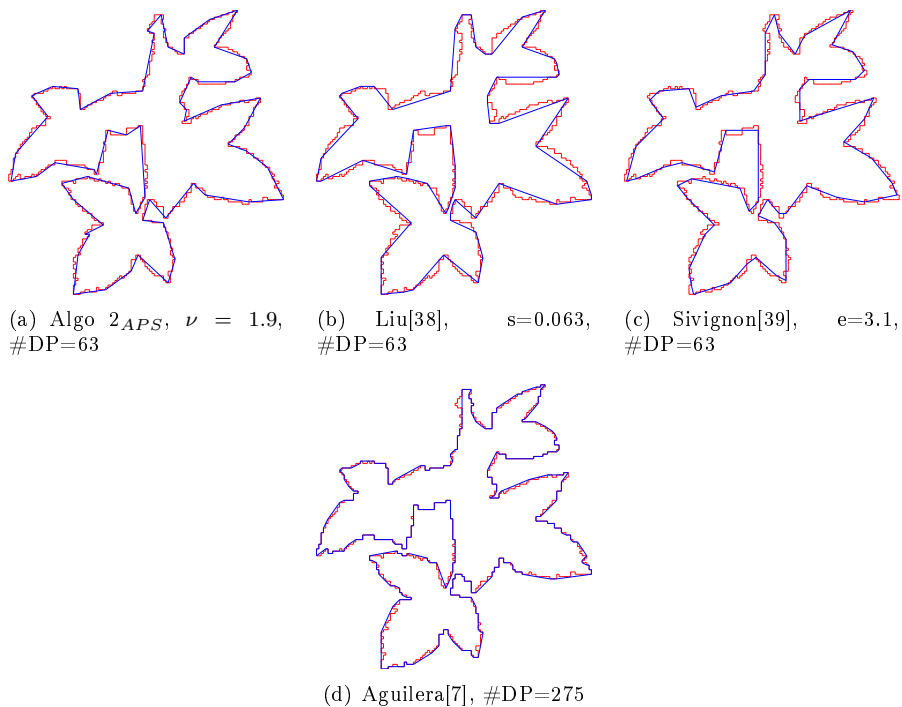(c) Sivignon[39], e=3.1, #DP=63

(d) Aguilera[7], #DP=275

Figure 10: Red lines depict the original contour and blue lines depict the polygonal simplification.

The second one (Sivignon [39]) computes a simplification of the curve such that the Fréchet distance between the original and the simplified curve is less than the error e. The third one (Aiguilera [7]), without parameter, is based on the concavity tree to compute the polygonal approximation of the curve.

Fig.10 and Fig.11 illustrate the different methods for two examples. To be objective in our comparison, we choose the value of the error e and scale s in order to obtain the same number of dominant points $\#DP$ which has been detected by Algorithm 2_{ASP}.

(a) Algo 2_{APS}, $\nu = 1.7$, #DP=45

(b) Liu[38], s=0.035, #DP=45

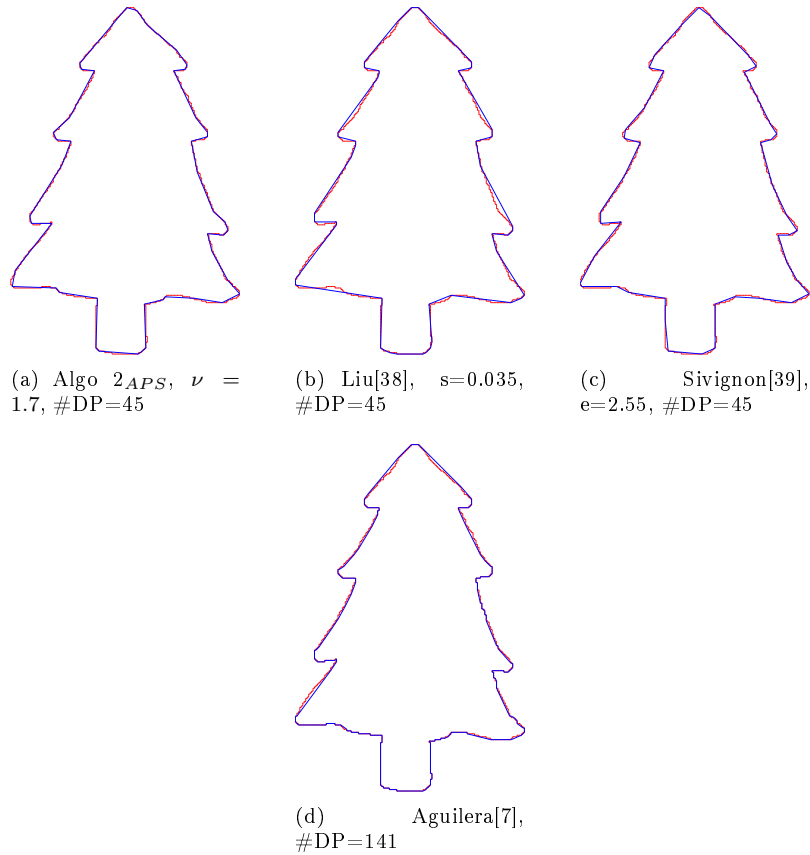(c) Sivignon[39], e=2.55, #DP=45

(d) Aguilera[7], #DP=141

Figure 11: Red lines depict the original contour and blue lines depict the polygonal simplification.

Table 5 shows different evaluation criteria of the polygonal representations deduced from dominant points computed by the different methods.

The Algorithm 2_{ASP} obtains, as expected, the best values of FOM and FOM_2 but also good values of ISSE. Aiguilera's method [7] has a high score for ISSE value, induced by the high number of computed dominant points.

18

The Sivignon's method [39] has the best L_∞ values, that seems logical because this method minimizes the distance between the curve and its polygonal representation. However the L_∞ values obtained by the Algorithm 2_{ASP} are very close to those of Sivignon's method. For example, in Fig.11(a) the value of L_∞ is 1.8392 against a value of 1.8023 in Fig.11(c).

Curve	Method	#DP	CR	ISSE	L_∞	FOM	FOM$_2$
Fig.10 $n = 1082$	Algo 2_{APS}, ν=1.9	63	17.1746	664.941	2.5399	**0.0255**	**0.4338**
	Liu[38], s=0.063	63	17.1746	2172.32	4.4141	0.0078	0.1327
	Sivignon[39], e=3.1	63	17.1746	1200.75	**2.1828**	0.0141	0.2402
	Aguilera[7]	275	3.9345	**407.97.6**	3	0.0095	0.0371
Fig.11 $n = 950$	Algo 2_{APS}, ν=1.7	45	21.1111	**391.191**	1.8392	**0.0539**	**1.1392**
	Liu[38], s=0.035	45	21.1111	1197.74	4.0834	0.0176	0.3720
	Sivignon[39], e=2.55	45	21.1111	623.846	**1.8023**	0.0338	0.7144
	Aguilera[7]	141	6.7375	543.887	2.5607	0.0123	0.0834

Table 5: Results of the proposed method and of the other methods for Fig.10 and Fig.11.
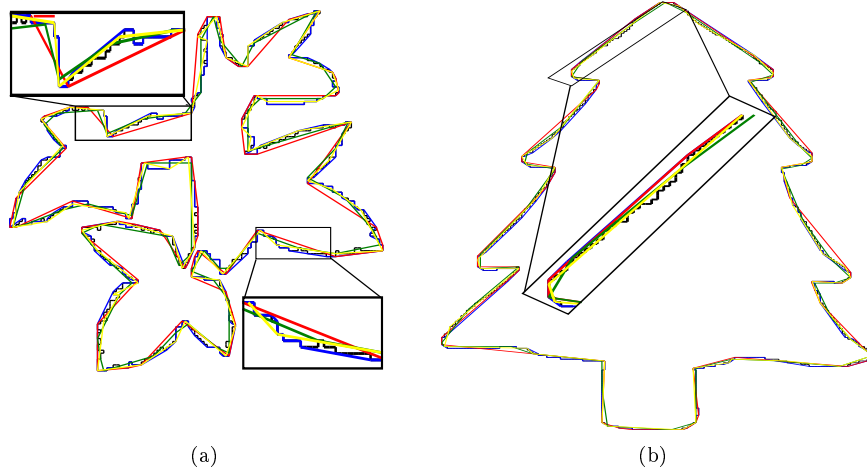

(a)   (b)

Figure 12: (a) superimposition of Fig.10, (b) superimposition of Fig.11. Black curve represents the original curve. Each color indicates a method, for example blue (resp. red, green) color represents the method described by Aguilera [7] (resp. Liu [38], Sivignon [39]). Yellow curve represents the result of Algorithm 2_{APS}.

To better evaluate the experiments, we show in Fig.12 the superimposition of the polygonal simplifications computed by the different methods.

19

As the Aiguilera's method [7] has detected many more dominant points than other methods, the blue polygon (Aguilera [7]) is very close to the initial curve. The red polygon (Liu [38]) is the furthest from the initial contour. This is justified by the fact that when we choose a higher value of scale s, the polygonal representation becomes simpler until it converges to the boundary curve of its convex hull.

The yellow (Algorithm 2_{APS}) and green (Sivignon [39]) polygons seem to well represent the initial curves. However, the corners of the contour are better detected by the Algorithm 2_{APS}, corresponding to the yellow polygons (see for example the zoom part of Fig.12). Hence the polygonal representations are closer to the initial curve.

6. Application to classification

In this section, we present several **shape geometrical descriptors** based on the dominant points obtained with the Algorithm 2_{APS} and we test these descriptors through a classification process.

Shape geometrical descriptors. We propose eight descriptors from the selected dominant points (DP) detected by the Algorithm 2_{APS} applied on the contour of a shape S.

The polygon obtained with the DP is called $DP(S)$. Let $per(DP(S))$ and $area(DP(S))$ be respectively the perimeter and the area of $DP(S)$. Let $ch(DP(S))$ be the convex hull of $DP(S)$. The following descriptors give indications about **compacity** and **convexity** of the shape S:

+ $\frac{per(DP(S))^2}{area(DP(S))}$

+ $\frac{per(ch(DP(S)))^2}{area(ch(DP(S)))}$

+ $\frac{area(ch(DP(S)))}{area(DP(S))}$

We also compute descriptors that indicate if the contour of the shape S is regular or contains big irregularities :

+ Variance value of angle between two successive segments formed by three successive dominant points of $DP(S)$

+ Minimun and maximun distances of the centroid of S to dominant points of S as well as the difference between minimum and maximum distances

+ Variance of segment lengths in $DP(S)$

Classification method. We use the **K-means** method [40]. It is one of the simplest unsupervised learning algorithms that solve a clustering problem. Clustering is the process of partitioning a group of data points into a small number of clusters. In general, we have n data points X_i, $i = 1...n$ that have to be partitioned in k clusters. The goal is to assign a cluster to each input data point. K-means is a clustering method that aims to find the positions of centroid c_j, $j = 1...k$ of the clusters that minimize the distance from the data points to centroid of each cluster.
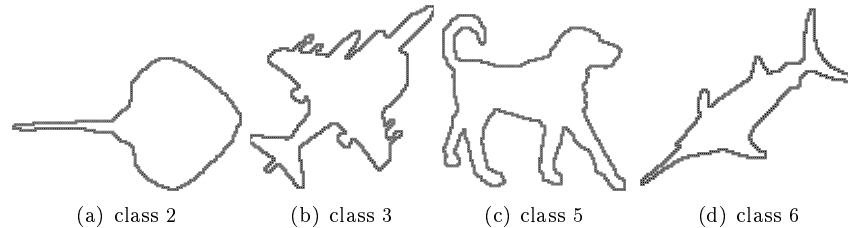
| (a) class 2 | (b) class 3 | (c) class 5 | (d) class 6 |

Figure 13: Discrete contours of images from Sharvit database.

| (a) class 1 | (b) class 4 | (c) class 7 |
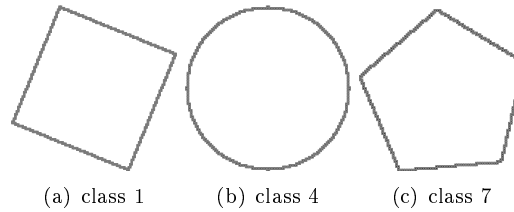
Figure 14: Discrete contours of some geometric shapes.

Classes	class 1	class 2	class 3	class 4	class 5	class 6	class 7
Precision	1	0.89	0.71	1	1	0.7	1
Recall	1	1	0.62	1	0.75	0.87	1
F-measure	1	0.94	0.67	1	0.86	0.78	1

Table 6: Evaluation criteria of the classification of Fig.13 and Fig.14.

A recent study [41] demonstrated that the city block distance in the k-means classification is better than other types of distance, that is why we use this distance in our experiments.

Experiments and result evaluation. We classify with the **K-means** method (with Matlab software) some images of Sharvit's database (cf. Fig. 13) and geometric shapes as polygon and circle (cf. Fig. 14). We consider 8 examples

21

from each class and for each example we compute the geometrical descriptors described previously.

A confusion matrix is a specific table layout that allows visualization of the performance of an algorithm, Fig. 15 shows the confusion matrix of the classification of images in Fig. 13 and Fig. 14. Each column of the matrix represents the instances in a predicted class while each row represents the instances in an actual class. For example in Fig. 15 we have 8 actual images in class 3, the system predicted that 5 belongs to class 3, and 3 belongs to class 6, but the system predicted that all images from class 4 belong to class 4. For the geometric shapes the classification is 100%.

All true values are located in the diagonal of the table, so it is easy to visually inspect the table for errors, as they will be represented by values outside the diagonal. From Fig. 15, we show that the accuracy of the system is equal to 89%.

Table 6 shows the evaluation criteria *(precision, recall and F-measure)* with:

$$+ \ precision_i = \frac{\text{number of objects correctly assigned to the class i}}{\text{number of objects allocated to the class i}}$$

$$+ \ recall_i = \frac{\text{number of objects correctly assigned to the class i}}{\text{number of objects belonging to the class i}}$$

$$+ \ F\text{-}measure_i = 2 * \frac{precision_i * recall_i}{precision_i + recall_i}$$
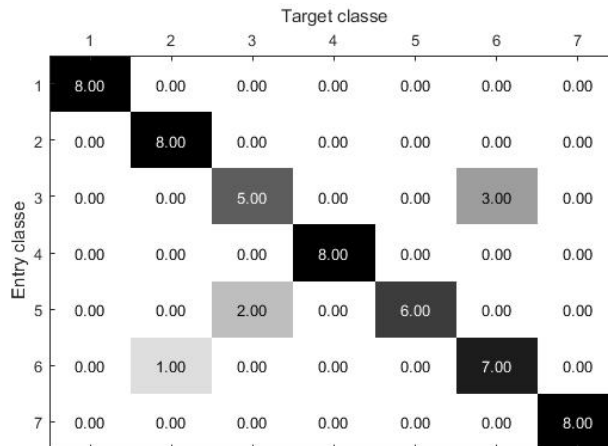

Figure 15: Confusion matrix for K-means classification with K=7.

If there are several classes, precision and recall of all the system are the mean value of all $precision_i$ and $recall_i$. For our system we have a precision equal to 0,9 and a recall equal to 0,89.

7. Conclusion

We presented in this paper a dominant point detection algorithm, improvement of a previous algorithm which uses an heuristic strategy. This algorithm relies on the structure of the discrete curve and the width, parameter of the algorithm, permits to work on data with or without noise. For a given width, our method associates to each detected dominant point a weight which permits to evaluate the importance of the dominant point in the structure of the studied curve. We then deduce a method to reduce the number of dominant points and to obtain a smaller error. In particular, two strategies are considered for the reduction: either the number of dominant points is fixed, or an automatic process is applied to have a polygonal simplification according to FOM_2 criterion. Furthermore, an application for classification is as well presented using the polygon obtained by the set of dominant points.

The proposed method opens numerous perspective of future work, for example the study of the method behaviour in a multiscale approach to automatically detect the more appropriate width for dominant point selection.

References

[1] I. Debled-Rennesson, F. Feschet, J. Rouyer-Degli, Optimal blurred segments decomposition of noisy shapes in linear time, Computers & Graphics 30 (1) (2006) 30–36.

[2] A. Faure, L. Buzer, F. Feschet, Tangential cover for thick digital curves, Pattern Recognition 42 (10) (2009) 2279–2287.

[3] F. Feschet, L. Tougne, Optimal time computation of the tangent of a discrete curve: Application to the curvature., in: DGCI, Vol. 1568 of LNCS, 1999, pp. 31–40.

[4] T. P. Nguyen, I. Debled-Rennesson, Curvature estimation in noisy curves, in: CAIP, Vol. 4673 of LNCS, 2007, pp. 474–481.

[5] T. P. Nguyen, I. Debled-Rennesson, A discrete geometry approach for dominant point detection, Pattern Recognition 44 (1) (2011) 32–44.

[6] E. Attneave, Some informational aspects of visual perception, Psychological Review 61 (3) (1954) 183–193.

[7] E. J. Aguilera-Aguilera, Á. C. Poyato, F. J. Madrid-Cuevas, R. M. Carnicer, The computation of polygonal approximations for 2d contours based on a concavity tree, Visual Communication and Image Representation 25 (8) (2014) 1905–1917.
URL https://github.com/eusebioaguilera/cdp

[8] A. R. Backes, O. M. Bruno, Polygonal approximation of digital planar curves through vertex betweenness, Information Sciences 222 (2013) 795–804.

[9] P. Bhowmick, B. B. Bhattacharya, Fast polygonal approximation of digital curves using relaxed straightness properties, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (9) (2007) 1590–1602.

[10] F. Feschet, Fast guaranteed polygonal approximations of closed digital curves, in: SCIA, Vol. 3540 of LNCS, 2005, pp. 910–919.

[11] M. Marji, P. Siy, A new algorithm for dominant points detection and polygonization of digital curves, Pattern Recognition 36 (10) (2003) 2239–2251.

[12] M. Marji, P. Siy, Polygonal representation of digital planar curves through dominant point detection - a nonparametric algorithm, Pattern Recognition 37 (11) (2004) 2113–2130.

[13] A. Masood, Dominant point detection by reverse polygonization of digital curves, Image and Vision Computing 26 (5) (2008) 702–715.

[14] A. Masood, Optimized polygonal approximation by dominant point deletion, Pattern Recognition 41 (1) (2008) 227–239.

[15] P. L. Rosin, Techniques for assessing polygonal approximations of curves, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (6) (1997) 659–666.

[16] D. Sarkar, A simple algorithm for detection of significant vertices for polygonal approximation of chain-coded curves, Pattern Recognition Letters 14 (12) (1993) 959–964.

[17] C.H.Teh, R.T.Chin, On the detection of dominant points on the digital curves, IEEE Transactions on Pattern Analysis and Machine Intelligence 2 (1989) 859–872.

[18] B. Wang, D. Brown, X. Zhang, H. Li, Y. Gao, J. Cao, Polygonal approximation using integer particle swarm optimization, Information Sciences 278 (2014) 311–326.

[19] E. J. Aguilera-Aguilera, Á. C. Poyato, F. J. Madrid-Cuevas, R. Muñoz-Salinas, Novel method to obtain the optimal polygonal approximation of digital planar curves based on mixed integer programming, Visual Communication and Image Representation 30 (2015) 106–116.

[20] D. K. Prasad, M. K. H. Leung, C. Quek, S. Cho, A novel framework for making dominant point detection methods non-parametric, Image Vision Computing 30 (11) (2012) 843–859.

[21] A. Kolesnikov, P. Fränti, Reduced-search dynamic programming for approximation of polygonal curves, Pattern Recognition Letters 24 (14) (2003) 2243–2254.

[22] A. Kolesnikov, P. Fränti, Polygonal approximation of closed discrete curves, Pattern Recognition 40 (4) (2007) 1282–1293.

[23] J.-C. Perez, E. Vidal, Optimum polygonal approximation of digitized curves, Pattern Recognition Letters 15 (8) (1994) 743–750.

[24] N. L. F. García, L. D. Martínez, Á. C. Poyato, F. J. Madrid-Cuevas, R. M. Carnicer, A new thresholding approach for automatic generation of polygonal approximations, Visual Communication and Image Representation 35 (2016) 155–168.

[25] U.Ramer, An iterative procedure for the polygonal approximation of planar curves, Computer, Graphic and Image processing 1 (1972) 224–256.

[26] D. H. Douglas, T. K. Peucker, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature, Cartographica: The International Journal for Geographic Information and Geovisualization 10 (2) (1973) 112–122.

[27] L. J. Latecki, R. Lakämper, Convexity rule for shape decomposition based on discrete contour evolution, Computer Vision and Image Understanding 73 (3) (1999) 441–454.

[28] Á. C. Poyato, N. L. F. García, R. M. Carnicer, F. J. Madrid-Cuevas, Dominant point detection: A new proposal, Image Vision Computing 23 (13) (2005) 1226–1236.

[29] B. K. Ray, K. S. Ray, Detection of significant points and polygonal approximation of digitized curves, Pattern Recognition Letters 13 (6) (1992) 443–452.

[30] W. Wu, Dominant point detection using adaptive bending value, Image Vision Computing 21 (6) (2003) 517–525.

[31] A. Rosenfeld, E. Johnston., Angle detection on digital curves, IEEE Transactions on Computers 22 (9) (1973) 875–878.

[32] M. T. Parvez, S. A. Mahmoud, Polygonal approximation of digital planar curves through adaptive optimizations, Pattern Recognition Letters 31 (13) (2010) 1997–2005.

[33] J.-P. Reveillès, Géométrie discrète, calculs en nombre entiers et algorithmique, thèse d'état. Université Louis Pasteur, Strasbourg (1991).

[34] T. P. Nguyen, B. Kerautret, I. Debled-Rennesson, J. O. Lachaud, Unsupervised, Fast and Precise Recognition of Digital Arcs in Noisy Images, in: ICCVG, Vol. 6374, LNCS, 2010, pp. 59–68.

[35] B. Kerautret, J.-O. Lachaud, M. Said, Meaningful Thickness Detection on Polygonal Curve, in: ICPRAM, SciTePress, 2012, pp. 372–379.

[36] B. Kerautret, J.-O. Lachaud, Meaningful Scales Detection: an Unsupervised Noise Detection Algorithm for Digital Contours, Image Processing On Line 4 (2014) 18.

[37] B. Kerautret, J.-O. Lachaud, Meaningful Scales Detection along Digital Contours for Unsupervised Local Noise Estimation, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (12) (2012) 2379–2392.

[38] H. Liu, L. J. Latecki, W. Liu, A unified curvature definition for regular, polygonal, and digital planar curves, International Journal Comput. Vision 80 (1) (2008) 104–124.

[39] I. Sivignon, A Near-Linear Time Guaranteed Algorithm for Digital Curve Simplification Under the Fréchet Distance, Image Processing On Line 4 (2014) 116–127.

[40] J. Macqueen, Some methods for classification and analysis of multivariate observations, in: BSMSP, Vol. 1, 1967, pp. 281–297.

[41] D. J. Bora, A. K. Gupta, Effect of different distance measures on the performance of k-means algorithm: An experimental study in matlab, International Journal of Computer Science and Information Technologies 5 (2014) 2501–2506.