The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

Prasad, Dilip K. and Leung, Maylor K.H. and Quek, Chai and Cho, Siu-Yeung (2012) A novel framework for making dominant point detection methods non-parametric. Image and Vision Computing, 30 (11). pp. 843-859. ISSN 0262-8856

Access from the University of Nottingham repository:
http://eprints.nottingham.ac.uk/47521/1/A%20novel%20framework%20for%20making%20dominant%20point%20detection%20methods%20non-parametric.pdf

Copyright and reuse:

The Nottingham ePrints service makes this work by researchers of the University of Nottingham available open access under the following conditions.

This article is made available under the Creative Commons Attribution Non-commercial No Derivatives licence and may be reused according to the conditions of the licence. For more details see: http://creativecommons.org/licenses/by-nc-nd/2.5/

A note on versions:

The version presented here may differ from the published version or from the version of record. If you wish to cite this item you are advised to consult the publisher's version. Please see the repository url above for details on accessing the published version and note that access may require a subscription.

For more information, please contact eprints@nottingham.ac.uk

A novel framework for making dominant point detection methods non-parametric

Dilip K. Prasad [a,*], Maylor K.H. Leung [b], Chai Quek [a], Siu-Yeung Cho [c]

[a] School of Computer Engineering, Nanyang Technological University, 639798, Singapore
[b] Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman (Kampar), Malaysia
[c] University of Nottingham Ningbo, China

Keywords:
Non-parametric
Line fitting
Polygonal approximation
Dominant points
Digital curves

A B S T R A C T

Most dominant point detection methods require heuristically chosen control parameters. One of the commonly used control parameter is maximum deviation. This paper uses a theoretical bound of the maximum deviation of pixels obtained by digitization of a line segment for constructing a general framework to make most dominant point detection methods non-parametric. The derived analytical bound of the maximum deviation can be used as a natural bench mark for the line fitting algorithms and thus dominant point detection methods can be made parameter-independent and non-heuristic. Most methods can easily incorporate the bound. This is demonstrated using three categorically different dominant point detection methods. Such non-parametric approach retains the characteristics of the digital curve while providing good fitting performance and compression ratio for all the three methods using a variety of digital, non-digital, and noisy curves.

1. Introduction

In several image processing applications [1–8], it is desired to express the boundaries of shapes (edges) using polygons made of a few representative pixels (called the dominant points) from the boundary itself. Through polygonal approximation, it is sought to represent a digital curve using fewer points such that:

1. The representation is insensitive to the digitization noise of the digital curve.
2. The properties of the curvature of the digital curve are retained, so that geometrical properties like inflexion points or concavities can be subsequently assessed.
3. The time efficiency of higher level processing can be improved since the digital curve is represented by fewer points.

This problem is popularly known as the dominant point detection problem. Dominant point detection methods choose points from a digital curve 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 the edges. An example is presented in Fig. 1. In Fig. 1, a digital shape of a maple leaf is illustrated. The boundary of the shape is made of 244 pixels. A polygonal approximation of this shape is shown in Fig. 1. The maple leaf is represented using only 27 dominant points in this approximation and the concavities associated with the maple leaf are preserved (labeled A–F).

The problem of finding the dominant points on the boundary for polygonal approximation has often been cast in either a min-# problem or a min-ε problem [9]. While both problems are essentially minimization problems, the former's aim is to find the minimum number of points such that the value of a particular error function is below a certain threshold, and the latter's aim is to find a fixed number of dominant points such that the error function has minimum value. In both the cases, heuristics are involved in choosing the threshold (for min-#

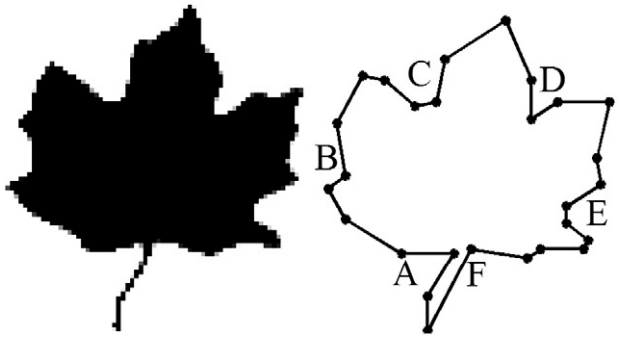

Fig. 1. An example of a digital shape and its polygonal approximation. The dominant points are denoted using the dots. The boundary of the maple leaf shape consists of 244 pixels. The polygonal approximation uses 27 dominant points.

* Corresponding author.
E-mail address: dilipprasad@gmail.com (D.K. Prasad).

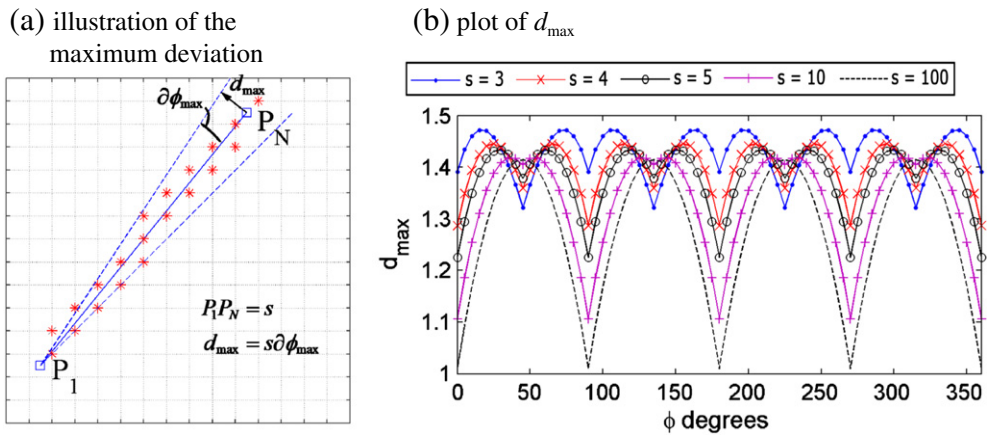

Fig. 2. The maximum deviation $d_{\max}$ of a continuous line segment and the digital line segment obtained from the continuous line segment.

problem) or the fixed number of points (in min-ε problem). In the recent times, several methods have been proposed to obtain the polygonal approximation of digital curves in the framework of min-# problem. This is because it is difficult to determine the fixed number of points in min-ε problem suitably for many shapes, while if the error function in min-# is related to the quality of fit, it is easier to use heuristics to determine a generally acceptable threshold for the error function.

Some of the recent dominant point detection methods are proposed by Masood [10,11], Carmona-Poyato [12], Ngyuen [13], Wu [14], Kolesnikov [3,15], Bhowmick [16] and Marji [17] while few older ones are found in [18–29]. These algorithms can be generally classified based upon the approach taken by them. For example, some used dynamic programming [3,15,19], while others used splitting [20–22], merging [23], digitally straight segments [13,16], suppression of break points [10–12,17], curvature and convexity [14,18,24,27]. The control parameters used in most dominant point detection methods are often related to the maximum deviation of the pixels on the digital curve segment between adjacent dominant points from the line segment connecting the dominant points. When the control parameters are related to the maximum deviation, the allowable or tolerable maximum deviation is chosen heuristically as a threshold value. Although the threshold in generally chosen to be a constant, a suitable value of the threshold varies from one digital curve to another and even within the digital curve. However, no specific rules are available for choosing either the constant threshold value or an adaptive threshold value depending upon the digital curve.

This paper concentrates on the min-# problem and considers the maximum deviation of the digital curve from the fitted polygon as an error function related to the quality of fit. Under this premise, a non-parametric framework is proposed in this paper for the automatic and adaptive determination of the threshold for the min-# problem. In this paper, a theoretical bound for the maximum deviation of a set of pixels by digitizing a line segment is first derived. This explicit and analytically defined bound is related to the length and the slope of the

(a) Pseudocode for RDP (original)

Function DP=RDP_original ($\{P_1, P_2, \ldots, P_N\}$, d_{tol})
{ DP=NULL; % DP contains the dominant points
%step 1: line
Fit a line l using P_1 and P_N .
% step 2: maximum deviation
Find deviation $\{d_1, \ldots, d_N\}$ of pixels $\{P_1, \ldots, P_N\}$ from the line l .
Find $d_{\max} = \max\{d_1, \ldots, d_N\}$ and point $P_{\max}$ corresponding to $d_{\max}$.
%step 3: termination/recursion condition
If $d_{\max} \le d_{tol}$
DP=$\{DP, \quad P_1, P_N\}$
Else
{ DP=$\{DP, \quad RDP\_max\ (P_1, P_{\max})\}$.
DP=$\{DP, \quad RDP\_max\ (P_{\max}, P_N)\}$.
}
End
Remove redundant points in DP.
Return(DP).
}

(b) Pseudocode for RDP (modified)

Function DP=RDP_modified ($\{P_1, P_2, \ldots, P_N\}$)
{ DP=NULL; % DP contains the dominant points
%step 1: line and its parameters
Fit a line l using P_1 and P_N .
For the line, find distance $s = |P_1P_N|$ and slope m .
Compute $\partial\phi_{\max}$ and $d_{tol} = s\partial\phi_{\max}$ using eqns. (1) and (2) .
% step 2: maximum deviation
Find deviation $\{d_1, \ldots, d_N\}$ of pixels $\{P_1, \ldots, P_N\}$ from the line l .
Find $d_{\max} = \max\{d_1, \ldots, d_N\}$ and point $P_{\max}$ corresponding to $d_{\max}$.
%step 3: termination/recursion condition
If $d_{\max} \le d_{tol}$
DP=$\{DP, \quad P_1, P_N\}$
Else
{ DP=$\{DP, \quad RDP\_max\ (P_1, P_{\max})\}$.
DP=$\{DP, \quad RDP\_max\ (P_{\max}, P_N)\}$.
}
End
Remove redundant points in DP.
Return(DP).
}

Fig. 3. Pseudocodes for algorithms in Sections 3.1.1 and 3.1.2, respectively.

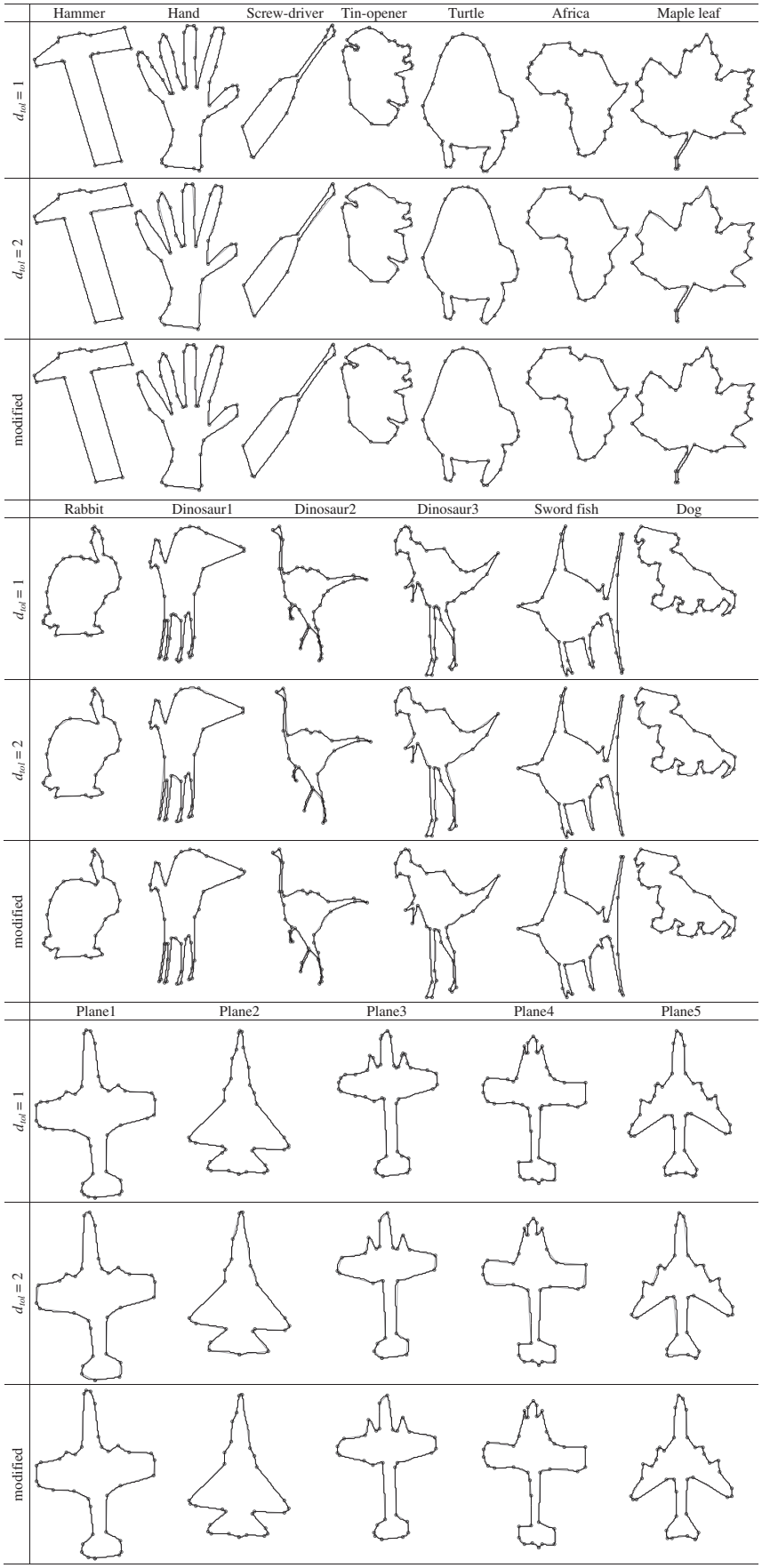

Fig. 4. Comparison of results of Ramer–Douglas–Peucker (original and modified) for several digital curves.

Table 1
Quantitative comparison of the original and modified methods of Ramer–Douglas–Peucker.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Hammer						
RDP(1.0)	388	16	0.98	55.17	0.44	24.25
RDP(2.0)	388	13	1.65	92.94	0.32	29.85
RDP(mod)	388	14	1.30	67.90	0.41	27.71
Hand						
RDP(1.0)	642	53	1.00	94.80	0.13	12.11
RDP(2.0)	642	32	1.99	314.65	0.06	20.06
RDP(mod)	642	39	1.40	178.80	0.09	16.46
Screw Driver						
RDP(1.0)	253	16	0.95	34.52	0.46	15.81
RDP(2.0)	253	9	2.00	120.59	0.23	28.11
RDP(mod)	253	11	1.44	55.34	0.42	23.00
Tin Opener						
RDP(1.0)	278	42	0.95	33.99	0.19	6.62
RDP(2.0)	278	24	1.83	127.99	0.09	11.58
RDP(mod)	278	31	1.48	77.84	0.12	8.97
Turtle						
RDP(1.0)	354	35	1.00	49.99	0.20	10.11
RDP(2.0)	354	24	1.87	120.92	0.12	14.75
RDP(mod)	354	26	1.49	91.53	0.15	13.62
Africa						
RDP(1.0)	291	38	1.00	37.02	0.21	7.66
RDP(2.0)	291	25	1.81	102.93	0.11	11.64
RDP(mod)	291	29	1.27	63.21	0.16	10.03
Maple leaf						
RDP(1.0)	424	58	1.00	53.11	0.14	7.31
RDP(2.0)	424	35	1.98	202.30	0.06	12.11
RDP(mod)	424	47	1.41	93.97	0.10	9.02
Rabbit						
RDP(1.0)	293	40	0.97	35.98	0.20	7.33
RDP(2.0)	293	26	1.74	103.46	0.11	11.27
RDP(mod)	293	30	1.41	73.04	0.13	9.77
Dinosaur1						
RDP(1.0)	587	46	1.00	71.49	0.18	12.76
RDP(2.0)	587	31	1.96	289.83	0.07	18.94
RDP(mod)	587	41	1.46	107.80	0.13	14.32
Dinosaur2						
RDP(1.0)	409	46	0.99	56.34	0.16	8.89
RDP(2.0)	409	27	2.00	219.75	0.07	15.15
RDP(mod)	409	38	1.41	91.64	0.12	10.76
Dinosaur3						
RDP(1.0)	528	57	0.99	76.40	0.12	9.26
RDP(2.0)	528	36	1.97	234.95	0.06	14.67
RDP(mod)	528	44	1.46	143.87	0.08	12.00
Sword Fish						
RDP(1.0)	627	46	1.00	80.21	0.17	13.63
RDP(2.0)	627	33	1.94	220.35	0.09	19.00
RDP(mod)	627	38	1.41	138.68	0.12	16.50
Dog						
RDP(1.0)	343	52	1.00	44.81	0.15	6.60
RDP(2.0)	343	36	2.00	149.19	0.06	9.53
RDP(mod)	343	41	1.39	82.69	0.10	8.37
Plane1						
RDP(1.0)	462	40	1.00	67.57	0.17	11.55
RDP(2.0)	462	31	1.79	148.71	0.10	14.90
RDP(mod)	462	36	1.20	87.51	0.15	12.83
Plane2						
RDP(1.0)	365	35	0.95	40.48	0.26	10.43
RDP(2.0)	365	21	1.85	165.20	0.11	17.38
RDP(mod)	365	30	1.53	61.44	0.20	12.17
Plane3						
RDP(1.0)	431	45	0.99	56.46	0.17	9.58
RDP(2.0)	431	26	1.91	232.08	0.07	16.58
RDP(mod)	431	32	1.46	117.18	0.11	13.47
Plane4						
RDP(1.0)	450	46	1.00	68.62	0.14	9.78
RDP(2.0)	450	33	1.80	228.60	0.06	13.64
RDP(mod)	450	39	1.41	101.99	0.11	11.54
Plane5						
RDP(1.0)	431	44	1.00	57.52	0.17	9.80
RDP(2.0)	431	33	2.00	194.95	0.07	13.06
RDP(mod)	431	39	1.34	75.63	0.15	11.05

line segment. Through this derivation, since the maximum possible deviation due to digitization can be computed for any line segment, it can be easily used as a threshold for each individual edge of the polygon obtained by a dominant point detection method. In this sense, it serves as natural benchmark for any dominant point detection method.

In order to demonstrate the application of the proposed non-parametric framework in the dominant point detection methods, three popular dominant point detection methods by Ramer, Douglas, and Peucker [21,22] (referred to as RDP), Masood [10] (referred to as Masood), and Carmona-Poyato [12] (referred to as Carmona for brevity) are considered, and adapted in the proposed non-parametric framework. The adapted versions of the methods are control parameter independent and do not require user specified inputs (thus making these algorithms free from heuristics). The modified methods show balanced performance over their original version despite being control parameter independent. The bound based non-parametric framework can be easily integrated in other methods as well and can be used to make them non-heuristic (or less heuristic) and self-adaptive. We highlight that in the proposed framework, the basic construct and the nature of the algorithms remain unchanged, while only optimization or termination condition is altered in order to make the method non-parametric.

The outline of the paper is as follows. Section 2 introduces the proposed non-parametric framework. Section 3 presents the adaptation of three methods in the proposed framework. In particular, Section 3.1 presents the original method of RDP, its modification under the proposed framework, and numerical comparison of the original and modified methods. Analogous to Section 3.1, Section 3.2 presents the method of Masood and Section 3.3 presents the method of Carmona. Section 4 provides brief discussion about various aspects like the nature of fitting, effect of scaling, fitting for non-digitized curves, and noisy digital curves. Section 5 concludes the paper. Further the detailed derivation of the error bound used in Section 2 is included in the Appendix A.

2. Proposed non-parametric framework

The foundation of the proposed non-parametric framework is based upon the concept of digitization, which is ubiquitous in digital images. It was shown in [30] and [31], that if a line segment in the continuous 2-dimensional space is digitized, the maximum difference between the angles made by the digital line segment and the continuous line segment with the $x-$axis is given by:

$$\partial\phi_{\max} \approx \max\left(\tan^{-1}\left\{\frac{1}{s}(|\sin\phi \pm \cos\phi|)\left(1-t_{\max}+t_{\max}^2\right)\right\}\right) \quad (1)$$

where,

$$\phi = \tan^{-1}(m) \tag{2}$$

$$t_{\max} = \left(\frac{1}{s}\right)(|\cos\phi| + |\sin\phi|) \tag{3}$$

The detailed derivation is reproduced in the Appendix A for the sake of completeness. Now, let us consider a line segment joining two points P_1 and P_N and its corresponding digital line segment given by pixels $\{P'_1, \ldots, P'_N\}$ (an illustration is shown in Fig. 2(a)). The distances of the pixels $\{P'_1, \ldots, P'_N\}$ from the line segment P_1P_N be denoted by d_i; $i = 1$ to N. For convenience, we refer to these distances as the deviations.

(a) Pseudocode for Masood (original)

Global $\{P_1, P_2, \ldots, P_M\}$.

Function **I**=Masood_original (d_{tol})

{ **%step 1: break points**

Compute the break points. Find their indices $\left\{I_n^{\text{break}}\right\}$;

Assign $i = 0$; $\mathbf{I}_i = \left\{I_n^{\text{break}}\right\}$; flag_stop=FALSE

%step 2: iterations

do

{ Assign N = number of dominant points

%step 2.1: Compute AEV

For $n = 2$ to $(N-1)$: Call (AEV (n), $\mathbf{I}^{\text{opt}}(n)$) =

compute_AEV ($\mathbf{I}_i$,n);

%step 2.2: Update or terminate

Compute d_m (defined immediately after eqn. (10));

If $\max\left((d_m)^2\right) > d_{\text{tol}}$, Then flag_stop=TRUE, Else { $i = i+1$;

$\mathbf{I}_i = \mathbf{I}^{\text{opt}}(n')$ };

} while(flag_stop=FALSE)

Return($\mathbf{I}_i$).

}

Function (AEV, $\mathbf{I}^{\text{opt}}$) = compute_AEV(**I** ,n)

{ **%Optimization step 1**

Assign $\mathbf{I}^{\text{up}} = \{I_1, I_2, \ldots, I_{n-1}\}$; Assign $k = 1$;

Do

{ Assign $I_{\min} = I_{n-k-1}, I_{\max} = I_{n-k+1}$.

Find $I_{\text{opt}} \in (I_{\min}, I_{\max})$ such that $\sum_{m=I_{\min}}^{I_{\max}} (d_m)^2$ (i.e. ISE) is minimized.

If $I_{\text{opt}} = I_{n-k}$. Then Flag_terminate_opt=TRUE, Else

{ $I_{n-k} = I_{\text{opt}}$; $k = k+1$;Flag_terminate_opt=FALSE};

}

While(Flag_terminate_opt=FALSE)

%Optimization step 2

Assign $\mathbf{I}^{\text{down}} = \{I_{n+1}, I_{n+2}, \ldots, I_N\}$; Assign $k = 1$;

Do

{ Assign $I_{\min} = I_{n+k-1}, I_{\max} = I_{n+k+1}$.

Find $I_{\text{opt}} \in (I_{\min}, I_{\max})$ such that $\sum_{m=I_{\min}}^{I_{\max}} (d_m)^2$ (i.e. ISE) is minimized.

If $I_{\text{opt}} = I_{n+k}$, Then Flag_terminate_opt=TRUE, Else

{ $I_{n+k} = I_{\text{opt}}$; $k = k+1$;Flag_terminate_opt=FALSE};

}

While(Flag_terminate_opt=FALSE)

%Computation of AEV

Compute AEV using eqn. (9);

Assign $\mathbf{I}^{\text{opt}} = \mathbf{I}^{\text{up}} \cup \mathbf{I}^{\text{down}} = \{I_1, I_2, \ldots, I_{n-1}, I_{n+1}, \ldots, I_N\}$;

Return(AEV, $\mathbf{I}^{\text{opt}}$)

}

(b) Pseudocode for Masood (modified)

Global $\{P_1, P_2, \ldots, P_M\}$.

Function **I**=Masood_modified

{ **%step 1: break points**

Compute the break points. Find their indices $\left\{I_n^{\text{break}}\right\}$;

Assign $i = 0$; $\mathbf{I}_i = \left\{I_n^{\text{break}}\right\}$; flag_stop=FALSE

%step 2: iterations

do

{ Assign N = number of dominant points

%step 2.1: Compute AEV

For $n = 2$ to $(N-1)$: Call (AEV (n), $\mathbf{I}^{\text{opt}}(n)$) =

compute_AEV ($\mathbf{I}_i$,n);

%step 2.2: Update or terminate

Find the index $n' = \arg\left(\min\left\{\text{AEV}_n; n = 2 \text{ to } N\right\}\right)$;

Compute $d^{\max}_{n'-1,n'+1}$ (using eqn. (14));

If AEV(n') > $d^{\max}_{n'-1,n'+1}$, Then flag_stop=TRUE, Else

{ $i = i+1$; $\mathbf{I}_i = \mathbf{I}^{\text{opt}}(n')$ };

} while(flag_stop=FALSE)

Return($\mathbf{I}_i$).

}

Function (AEV, $\mathbf{I}^{\text{opt}}$) = compute_AEV(**I** ,n)

{ **%Optimization step 1**

Assign $\mathbf{I}^{\text{up}} = \{I_1, I_2, \ldots, I_{n-1}\}$; Assign $k = 1$;

Do

{ Assign $I_{\min} = I_{n-k-1}, I_{\max} = I_{n-k+1}$.

Find $I_{\text{opt}} \in (I_{\min}, I_{\max})$ such that d_{n-k} computed using eqn. (11) is minimized.

If $I_{\text{opt}} = I_{n-k}$. Then Flag_terminate_opt=TRUE, Else

{ $I_{n-k} = I_{\text{opt}}$; $k = k+1$;Flag_terminate_opt=FALSE};

}

While(Flag_terminate_opt=FALSE)

%Optimization step 2

Assign $\mathbf{I}^{\text{down}} = \{I_{n+1}, I_{n+2}, \ldots, I_N\}$; Assign $k = 1$;

Do

{ Assign $I_{\min} = I_{n+k-1}, I_{\max} = I_{n+k+1}$.

Find $I_{\text{opt}} \in (I_{\min}, I_{\max})$ such that d_{n+k-1} computed using eqn. (11) is minimized.

If $I_{\text{opt}} = I_{n+k}$, Then Flag_terminate_opt=TRUE, Else

{ $I_{n+k} = I_{\text{opt}}$; $k = k+1$;Flag_terminate_opt=FALSE};

}

While(Flag_terminate_opt=FALSE)

%Computation of AEV

Compute AEV using eqn. (12);

Assign $\mathbf{I}^{\text{opt}} = \mathbf{I}^{\text{up}} \cup \mathbf{I}^{\text{down}} = \{I_1, I_2, \ldots, I_{n-1}, I_{n+1}, \ldots, I_N\}$;

Return(AEV, $\mathbf{I}^{\text{opt}}$)

}

Fig. 5. Pseudocodes for the original (Section 3.2.1) and modified (Section 3.2.2) methods of Masood.

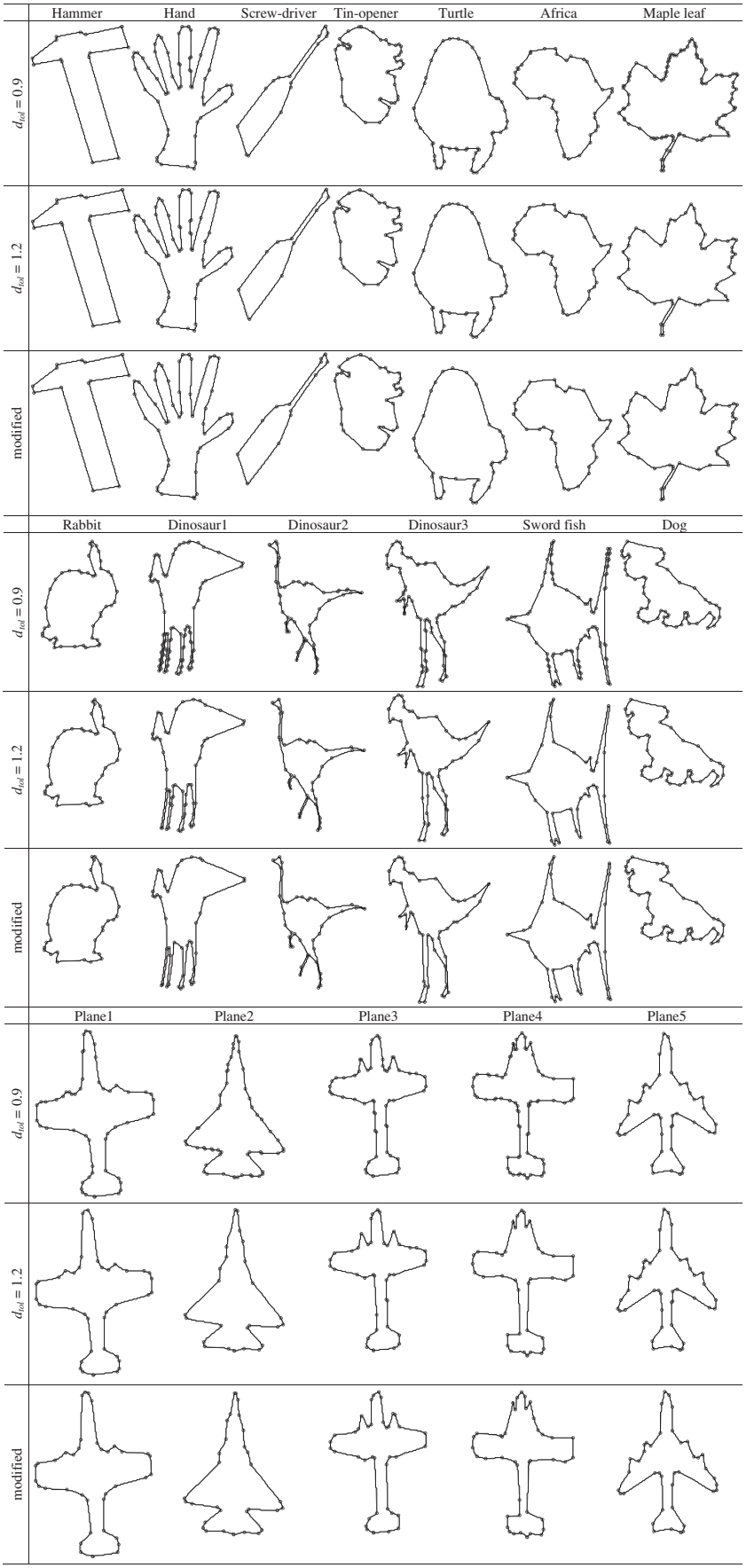

Fig. 6. Comparison of results of Masood (original and modified) for several digital curves.

Then, using Eq. (1) and assuming that only digitization is present, the distances should lie within [0, d_{max}], where d_{max} is given as:

$$d_{max} = s\partial\phi_{max} \approx \max\left(s\tan^{-1}\left\{\frac{1}{s}(|\sin\phi \pm \cos\phi|)\left(1-t_{max}+t_{max}^2\right)\right\}\right) \quad (4)$$

where ϕ corresponds to the slope m of the continuous line segment P_1P_N. For a digital curve, since the distance s between any two points and the slope m of the line passing through them can be computed, the upper bound of the deviations due to digitization alone d_{max} can be computed using Eq. (4). Sample plots of d_{max} for various values of s and ϕ are shown in Fig. 2(b). This bound is the underlying concept in the proposed non-parametric framework for dominant point detection methods.

In the proposed framework, we intend to compare the deviations of the pixels from a line segment with the bound in Eq. (4) and use it either as an optimization goal or a termination condition in the dominant point detection methods. This concept is used as a framework in which most dominant point detection methods can be adapted and made parameter-free.

3. Dominant point detection methods in the proposed framework

In this section we use the proposed framework for automatically computing the tolerable maximum deviations for various methods. For this, we consider three categorically different dominant point detection methods and demonstrate the applicability of the error bound of the deviation. The first method is proposed by Ramer, Douglas, and Peucker [21,22], the second method is by Masood [10], and the third method is by Carmona-Poyato et al. [12].

3.1. Ramer–Douglas–Peucker method for recursive determination of dominant points

3.1.1. Original method

Ramer, Douglas, and Peucker [21,22] (referred to as RDP) proposed a fast recursive method for computing the dominant points on digital curves. The method is described as follows. Let us consider a digital curve $e = \{P_1 \quad P_2 \quad \ldots \quad P_N\}$, where P_i is the ith edge pixel in the digital curve e. The line passing through a pair of pixels $P_a(x_a, y_a)$ and $P_b(x_b, y_b)$ is given by:

$$x(y_a - y_b) + y(x_b - x_a) + y_b x_a - y_a x_b = 0. \quad (5)$$

Then the deviation d_i of a pixel $P_i(x_i, y_i) \in e$ from the line passing through the pair $\{P_1, P_N\}$ is given as:

$$d_i = \frac{|x_i(y_1 - y_N) + y_i(x_N - x_1) + y_N x_1 - y_1 x_N|}{\sqrt{(x_N - x_1)^2 + (y_1 - y_N)^2}}. \quad (6)$$

Accordingly, the pixel with maximum deviation can be found. Let it be denoted as P_{max}. Then considering the pairs $\{P_1, P_{max}\}$ and $\{P_{max}, P_N\}$, we find two new pixels from e using the concept in Eqs. (5) and (6). It is evident that the maximum deviation goes on decreasing as we choose newer pixels of maximum deviation between a pair. This process can be repeated till a certain condition is satisfied by all the

(a) Pseudocode for Carmona (original)

```
Function DP=Carmona_original ( {P_1, P_2, ..., P_N} , r_tol )
{   DP=NULL; % DP contains the dominant points
    %step 1: choosing initial dominant point
    If P_1 = P_N Then
    {   Compute {d_n^C}; Find n' = arg(max(d_n^C)).
        Assign DP={P_n', P_n'+1, ..., P_N, P_2, P_3, ..., P_n'}
    }
    %step 2: termination/updation
    Assign d_tol = 0 ; i = 0 ; Flag_terminate = FALSE;
    Do
    {   Assign i = i+1; d_tol = d_tol + 0.5 ; n = 1 ; l_n^deleted =NULL;
        While (n is not the last point in DP)
        {
            Compute d_n^C and l_n using eqns. (15) and (16);
            If d_n^C < d_tol Then
            {   Append l_n^deleted with l_n ;
                Delete P_n from DP;
            }
            Else Assign n = n+1
            End;
        }
        Compute {d_n} and r_i using eqns. (11) and (17).
        If r_i < r_tol Then Flag_terminate = TRUE
    }
    While (Flag_terminate = FALSE);
    Return(DP).
}
```

(b) Pseudocode for Carmona (modified)

```
Function DP=Carmona_modified ( {P_1, P_2, ..., P_N} )
{   DP=NULL; % DP contains the dominant points
    %step 1: choosing initial dominant point
    If P_1 = P_N Then
    {   Compute {d_n^C}; Find n' = arg(max(d_n^C)).
        Assign DP={P_n', P_n'+1, ..., P_N, P_2, P_3, ..., P_n'}
    }
    %step 2: termination/updation
    Assign d_tol = 0 ; i = 0 ; Flag_terminate = FALSE;
    Do
    {   Assign i = i+1; d_tol = d_tol + 0.5 ; n = 1 ;
        While (n is not the last point in DP)
        {
            Compute d_n^C using eqn. (15);
            If d_n^C < d_tol Then Delete P_n from DP Else Assign
            n = n+1
        }
        Compute {d_n} and {d_n^max} using eqns. (11) and (18).
        If (for any n , d_n > d_n^max ) Then Flag_terminate = TRUE
    }
    While (Flag_terminate = FALSE);
    Return(DP).
}
```

Fig. 7. Pseudocodes for the original (Section 3.3.1) and modified (Section 3.3.2) methods of Carmona.

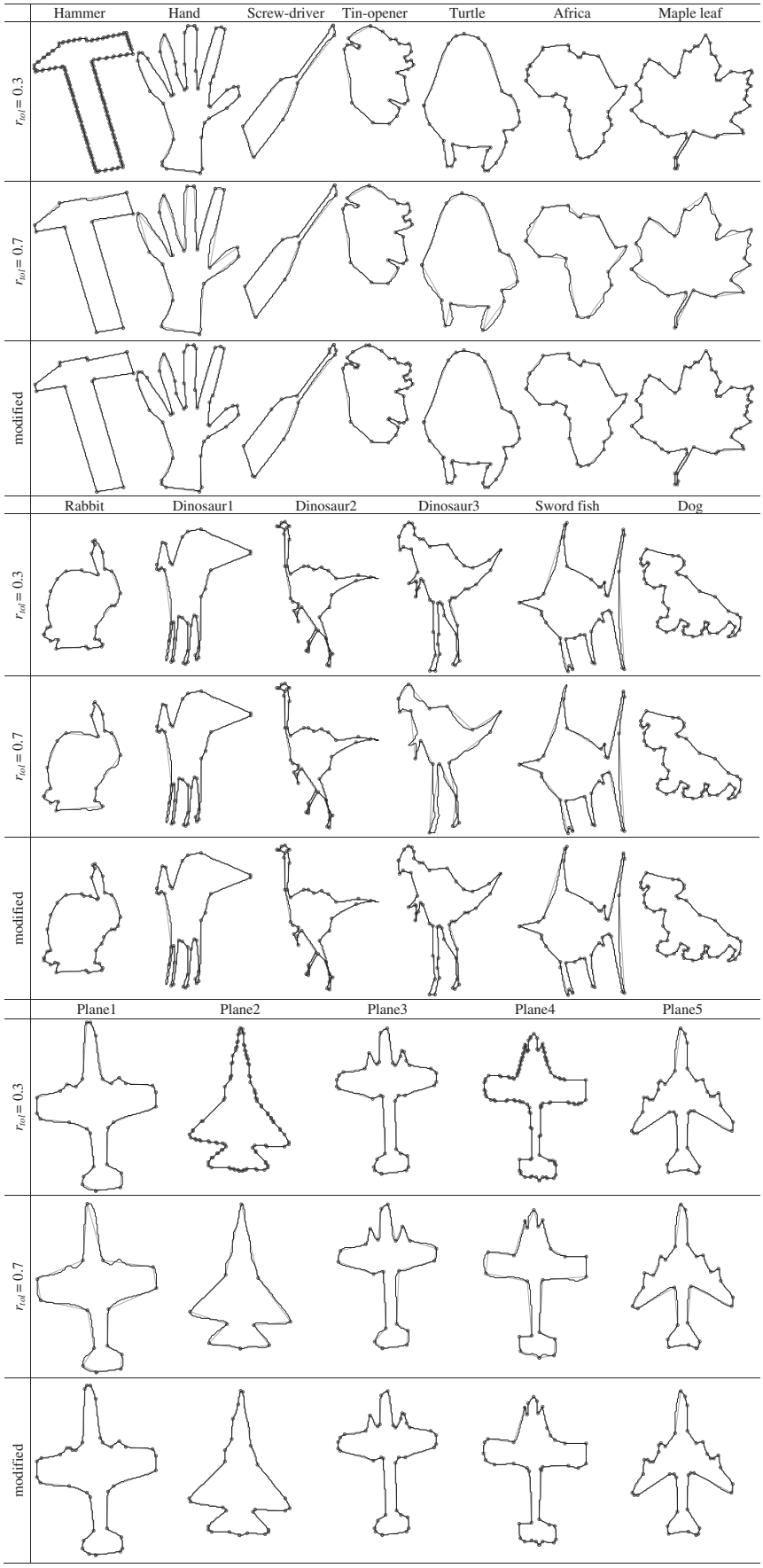

Fig. 8. Comparison of results of Carmona (original and modified) for several digital curves.

Table 2
Quantitative comparison of results of the original and modified methods of Masood.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Hammer						
Masood (0.9)	388	15	1.46	38.35	0.67	25.87
Masood (1.2)	388	15	1.46	38.35	0.67	25.87
Masood (mod)	388	14	0.79	48.86	0.57	27.71
Hand						
Masood (0.9)	642	58	1.32	47.75	0.23	11.07
Masood (1.2)	642	58	1.32	47.75	0.23	11.07
Masood (mod)	642	48	1.14	97.50	0.14	13.38
Screw driver						
Masood (0.9)	253	16	1.11	24.68	0.64	15.81
Masood (1.2)	253	11	1.53	46.62	0.49	23.00
Masood (mod)	253	14	1.13	42.59	0.42	18.07
Tin opener						
Masood (0.9)	278	45	1.00	20.69	0.30	6.18
Masood (1.2)	278	34	1.40	40.37	0.20	8.18
Masood (mod)	278	35	1.02	51.71	0.15	7.94
Turtle						
Masood (0.9)	354	48	1.00	23.48	0.31	7.38
Masood (1.2)	354	38	1.31	33.11	0.28	9.32
Masood (mod)	354	33	1.14	51.65	0.21	10.73
Africa						
Masood (0.9)	291	39	0.97	25.89	0.29	7.46
Masood (1.2)	291	35	1.21	31.26	0.27	8.31
Masood (mod)	291	28	1.05	56.74	0.18	10.39
Maple leaf						
Masood (0.9)	424	105	1.00	15.29	0.26	4.04
Masood (1.2)	424	50	1.23	46.34	0.18	8.48
Masood (mod)	424	50	1.20	64.45	0.13	8.48
Rabbit						
Masood (0.9)	293	45	0.93	21.67	0.30	6.51
Masood (1.2)	293	36	1.21	35.91	0.23	8.14
Masood (mod)	293	37	1.00	42.82	0.18	7.92
Dinosaur1						
Masood (0.9)	587	96	1.00	24.09	0.25	6.11
Masood (1.2)	587	55	1.26	44.38	0.24	10.67
Masood (mod)	587	42	0.96	91.04	0.15	13.98
Dinosaur2						
Masood (0.9)	409	56	1.00	30.36	0.24	7.30
Masood (1.2)	409	38	1.20	62.98	0.17	10.76
Masood (mod)	409	42	0.95	64.38	0.15	9.74
Dinosaur3						
Masood (0.9)	528	84	1.00	22.63	0.28	6.29
Masood (1.2)	528	56	1.37	50.69	0.19	9.43
Masood (mod)	528	48	1.27	89.38	0.12	11.00
Sword fish						
Masood (0.9)	627	91	1.00	33.14	0.21	6.89
Masood (1.2)	627	38	1.46	90.27	0.18	16.50
Masood (mod)	627	40	1.11	97.94	0.16	15.68
Dog						
Masood (0.9)	343	54	0.91	32.51	0.20	6.35
Masood (1.2)	343	49	1.26	43.19	0.16	7.00
Masood (mod)	343	52	1.00	43.45	0.15	6.60
Plane1						
Masood (0.9)	462	52	1.04	31.49	0.28	8.88
Masood (1.2)	462	40	1.54	50.59	0.23	11.55
Masood (mod)	462	38	1.11	64.81	0.19	12.16
Plane2						
Masood (0.9)	365	54	0.95	18.54	0.36	6.76
Masood (1.2)	365	26	1.31	58.20	0.24	14.04
Masood (mod)	365	32	1.00	44.67	0.26	11.41
Plane3						
Masood (0.9)	431	54	0.90	32.06	0.25	7.98
Masood (1.2)	431	41	1.36	47.91	0.22	10.51
Masood (mod)	431	39	1.00	71.76	0.15	11.05
Plane4						
Masood (0.9)	450	68	1.00	16.92	0.39	6.62
Masood (1.2)	450	39	1.26	59.66	0.19	11.54
Masood (mod)	450	42	0.96	59.54	0.18	10.71
Plane5						
Masood (0.9)	431	45	1.21	36.86	0.26	9.58
Masood (1.2)	431	45	1.21	36.86	0.26	9.58
Masood (mod)	431	41	0.96	49.41	0.21	10.51

line segments. This condition shall be referred to as the optimization goal for the ease of reference.

The condition used by RDP [21,22] is that for each line segment, the maximum deviation of the pixels contained in its corresponding edge segment is less than a certain tolerance value:

$$\max(d_i) < d_{\text{tol}}. \tag{7}$$

where d_{tol} is the chosen threshold and is typically a few pixels.

3.1.2. Non-parametric adaptation of RDP

In the above method, at each step in the recursion, if the length of the line segment that is fit most recently on the curve (or sub-curve) is s and the slope of the line segment is m, then using Eq. (4), we compute $d_{\max}$ and use it in Eq. (7) as $d_{tol} = d_{\max}$. The pseudocodes of the original and the modified methods are given in Fig. 3 and the changes are highlighted for the ease of comparison. As a consequence of the proposed modification, the original method does not require any control parameter and adaptively computes the suitable value of d_{tol} automatically.

3.1.3. Comparison of the RDP original and RDP modified methods

18 digital curves used in recent publications [10,12] are considered. For comparison, two values of the control parameter d_{tol} of the original method, $d_{tol} = 1$ and $d_{tol} = 2$, are used, and compared against the proposed modification which does not require user specified control parameter. The results are plotted in Fig. 4 and quantitative comparisons are provided in Table 1. Fig. 4 shows that the proposed modification provides good approximation to all the digital curves. In Table 1, the number of pixels M in the digital curves, number of dominant points N found by a method, the maximum deviation $\max(d_m)$ of the polygon from the digital curve, the integral square error (ISE), the figure of merit (FOM), and the compression ratio $CR = M/N$ are listed. In general it is desired that $\max(d_m)$ and ISE are as less as possible and FOM and CR are as large as possible [32].

The value of the maximum deviation $\max(d_m)$ for the modified method is between 1.20 and 1.53 while it varies from 0.95 to 2.00 for the original RDP with $d_{tol} = 1$ and $d_{tol} = 2$. The values of ISE, FOM, and CR for the modified RDP method are also between the values of these parameters for the original RDP. Thus, it can be concluded that the modified RDP gives cruder fit in comparison to the original RDP with $d_{tol} = 1$ and finer fit in comparison with the original RDP with $d_{tol} = 2$.

3.2. Masood's method of dominant point suppression

3.2.1. Original method

As opposed to the method proposed by Ramer, Douglas, and Peuker [21,22] (Section 3.1), Masood [10] begins with the break points as the

Table 3
Quantitative comparison of the original and modified versions of Carmona.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Hammer						
Carmona (0.3)	388	194	0.49	1.32	1.52	2.00
Carmona (0.7)	388	10	2.32	177.40	0.22	38.80
Carmona (mod)	388	14	1.29	57.96	0.48	27.71
Hand						
Carmona (0.3)	642	43	2.06	215.14	0.07	14.93
Carmona (0.7)	642	20	3.89	1111.60	0.03	32.10
Carmona (mod)	642	43	2.06	215.14	0.07	14.93
Screw driver						
Carmona (0.3)	253	9	2.08	179.47	0.16	28.11
Carmona (0.7)	253	9	2.08	179.47	0.16	28.11
Carmona (mod)	253	14	2.08	138.59	0.13	18.07
Tin opener						
Carmona (0.3)	278	29	2.28	145.12	0.07	9.59
Carmona (0.7)	278	22	2.28	214.60	0.06	12.64
Carmona (mod)	278	39	2.19	73.08	0.10	7.13
Turtle						
Carmona (0.3)	354	27	1.68	151.65	0.09	13.11
Carmona (0.7)	354	15	5.34	1134.83	0.02	23.60
Carmona (mod)	354	30	1.68	130.32	0.09	11.80
Africa						
Carmona (0.3)	291	35	1.24	52.62	0.16	8.31
Carmona (0.7)	291	14	3.81	547.22	0.04	20.79
Carmona (mod)	291	26	1.80	93.62	0.12	11.19
Maple leaf						
Carmona (0.3)	424	53	1.56	91.78	0.09	8.00
Carmona (0.7)	424	20	4.06	847.89	0.03	21.20
Carmona (mod)	424	53	1.56	91.78	0.09	8.00
Rabbit						
Carmona (0.3)	293	30	1.86	101.53	0.10	9.77
Carmona (0.7)	293	19	3.12	309.72	0.05	15.42
Carmona (mod)	293	38	1.86	66.64	0.12	7.71
Dinosaur1						
Carmona (0.3)	587	38	2.73	302.83	0.05	15.45
Carmona (0.7)	587	38	2.73	302.83	0.05	15.45
Carmona (mod)	587	38	2.73	302.83	0.05	15.45
Dinosaur2						
Carmona (0.3)	446	47	2.55	161.59	0.06	9.49
Carmona (0.7)	446	47	2.55	161.59	0.06	9.49
Carmona (mod)	446	47	2.55	161.59	0.06	9.49
Dinosaur3						
Carmona (0.3)	528	54	1.78	104.35	0.09	9.78
Carmona (0.7)	528	19	6.75	2115.16	0.01	27.79
Carmona (mod)	528	54	1.78	104.35	0.09	9.78
Sword fish						
Carmona (0.3)	627	39	3.19	573.52	0.03	16.08
Carmona (0.7)	627	30	3.00	646.28	0.03	20.90
Carmona (mod)	627	39	3.19	573.52	0.03	16.08
Dog						
Carmona (0.3)	343	54	1.39	54.26	0.12	6.35
Carmona (0.7)	343	54	1.39	54.26	0.12	6.35
Carmona (mod)	343	54	1.39	54.26	0.12	6.35
Plane1						
Carmona (0.3)	462	30	1.79	213.57	0.07	15.40
Carmona (0.7)	462	16	4.63	1105.78	0.03	28.88
Carmona (mod)	462	37	1.60	104.82	0.12	12.49
Plane2						
Carmona (0.3)	365	104	0.50	4.14	0.85	3.51
Carmona (0.7)	365	11	2.85	400.76	0.08	33.18
Carmona (mod)	365	28	1.44	80.54	0.16	13.04
Plane3						
Carmona (0.3)	431	39	1.53	92.99	0.12	11.05
Carmona (0.7)	431	22	2.79	397.56	0.05	19.59
Carmona (mod)	431	39	1.53	92.99	0.12	11.05
Plane4						
Carmona (0.3)	450	114	0.71	4.88	0.81	3.95
Carmona (0.7)	450	22	3.45	632.35	0.03	20.45
Carmona (mod)	450	41	2.00	127.93	0.09	10.98
Plane5						
Carmona (0.3)	431	41	2.53	168.71	0.06	10.51
Carmona (0.7)	431	28	2.53	349.77	0.04	15.39
Carmona (mod)	431	41	2.53	168.71	0.06	10.51

first list of dominant points and then iteratively removes one dominant point at a time till a termination condition is satisfied. In every iteration, when a dominant point is deleted, the remaining dominant points are re-optimized such that the dominant points after the optimization are such that the integral square error (ISE) is minimum. For convenience, we denote the iteration number with i and the number of dominant points in the ith iteration with $n = 1$ to N. The points in the digital curves are indexed from $m = 1$ to M. The list of dominant points in an iteration can then be specified using:

$$\mathbf{I}_i = \{I_n; n = 1 \text{ to } N\}_i; I_{\forall n} \in \{1, 2, \ldots, M\} \text{ and } I_n < I_{n+1} \tag{8}$$

where I denotes the index of the point on the digital curve. Before beginning the optimization, the break points are taken as the initial set of dominant points, i.e., $\mathbf{I}_{i=0} = \{I_n^{\text{break}}\}$. In an iteration, for each dominant point specified by I_n, an associated error value (AEV) is computed. This associated error value is calculated as follows. Considering the hypothesis that dominant point I_n shall be deleted, an optimization of the remaining dominant points, i.e., $(\mathbf{I}_i - \{I_n\})$, is performed to minimize the integral square error (ISE). This is done in two independent steps. In the first step, the indices of the dominant points before I_n, i.e., $\{I_1, \ldots, I_{n-1}\}$, are optimized. For this, first it is checked that does changing I_{n-1} within the range (I_{n-2}, I_n) yields to a lower value of ISE. If this happens, the value of I_{n-1} is changed to the optimal value within (I_{n-2}, I_n) that yields the minimum ISE. Then, in a similar manner, the index I_{n-2} is optimized. However, if changing I_{n-1} within the range (I_{n-2}, I_n) does not yield to a lower value of ISE, the optimization in this step is stopped. Using a similar approach, in the second step, the indices of the dominant points after I_n, i.e., $\{I_{n+1}, \ldots I_N\}$, are optimized. After the optimization of both the steps, AEV can be computed in the following manner. For convenience, we define the set of dominant points obtained after both optimization steps as $\mathbf{I}_{i,n}^{\text{opt}}$. If the ISE corresponding to $\mathbf{I}_i$ is denoted by ISE_i and the ISE corresponding to $\mathbf{I}_{i,n}^{\text{opt}}$ is denoted by $ISE_{i,n}$, then AEV of the nth point is:

$$AEV_{i,n} = ISE_i - ISE_{i,n} \tag{9}$$

After computing the AEV for all the dominant points in the ith iteration, the point with the minimum value of AEV is removed and the list of optimal points corresponding to its removal is retained (or can be recomputed). The algorithm can be terminated by specifying a termination condition which may be based upon the maximum number of dominant points, or the maximum integral square error, or the

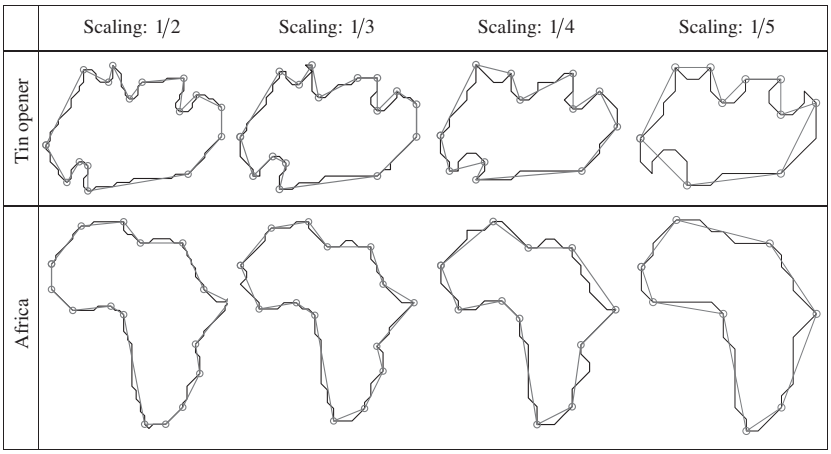

Fig. 9. Effect of scaling on RDP (mod).

maximum tolerable deviation (similar to d_{tol} in Eq. (7)). In [11], Masood proposed to use the following condition as the termination condition and the value of $d_{tol}=0.9$:

$$\max\left((d_m)^2\right) > d_{\text{tol}}. \tag{10}$$

where d_m is the deviation of the pixels on the digital curve from the polygon obtained by the dominant points.

3.2.2. Non-parametric adaptation of Masood

The method of Masood can be modified in the following manner. First, given a sequence of dominant points specified by the indices $\mathbf{I}_i=\{I_n;\ n=1 \text{ to } N\}_i$, we define a maximum deviation corresponding to the portion of digital curve corresponding to two consecutive dominant points as follows:

$$d_n = \max\left(\frac{\left|x_m(y_n-y_{n+1})+y_m(x_{n+1}-x_n)+y_{n+1}x_n-y_nx_{n+1}\right|}{\sqrt{(x_{n+1}-x_n)^2+(y_n-y_{n+1})^2}}; m\in\{I_n, I_n+1, \ldots, I_{n+1}\}\right);$$
$$\text{for } n=1 \text{ to } (N-1). \tag{11}$$

For convenience, the set of these maximum deviations for the given set of dominant points $\mathbf{I}_i=\{I_n;\ n=1 \text{ to } N\}_i$ is denoted as $\mathbf{D}(\mathbf{I}_i)=\{d_n;\ n=1 \text{ to } (N-1)\}$. The definition of AEV in the proposed modification is given as:

$$AEV_{i,n} = \max\left(\mathbf{D}\left(\mathbf{I}_{i,n}^{\text{opt}}\right)\right) \tag{12}$$

Further, in the optimization step (for the hypothesis that dominant point I_n shall be deleted) done for the sequence $\{I_1, \ldots, I_{n-1}\}$, instead of minimizing the integral square error (ISE), the goal is to minimize the maximum deviations d_{n-1}, d_{n-2}, and so on.

Finally, the termination condition is modified as follows:

$$\min\left\{AEV_{i,n}; n=2 \text{ to } (N-1)\right\} > d_{n-1,n+1}^{\max} \tag{13}$$

$$d_{n-1,n+1}^{\max} = s_{n-1,n+1}\phi_{n-1,n+1} \tag{14}$$

where $s_{n-1,n+1}$ is the length of the line segment formed by joining the dominant points I_{n-1} and I_{n+1} and $\phi_{n-1,n+1}$ is the angle made by the line segment with the $x-$axis.

The pseudocodes of the original and proposed modifications of Masood are presented in Fig. 5 and the modifications have been highlighted for the ease of comparison.

3.2.3. Comparison of the Masood original and Masood modified methods

18 digital curves used in recent publications [10,12] and in Section 3.1.3 are considered in this detailed benchmarking. For comparison, two values of the control parameter d_{tol} of the original method of Masood, $d_{tol}=0.9$ (recommended by Masood in [11]) and $d_{tol}=1.2$(taken as another control parameter for comparison) are used, and compared against the proposed modification which does not require user specified control parameter. The results are plotted in Fig. 6 and quantitative comparisons are provided in Table 2. Fig. 6 shows that the proposed modification provides

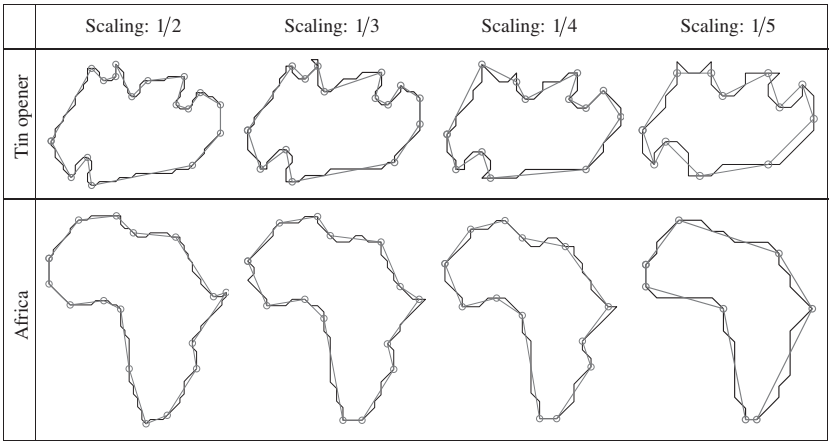

Fig. 10. Effect of scaling on Masood (mod).

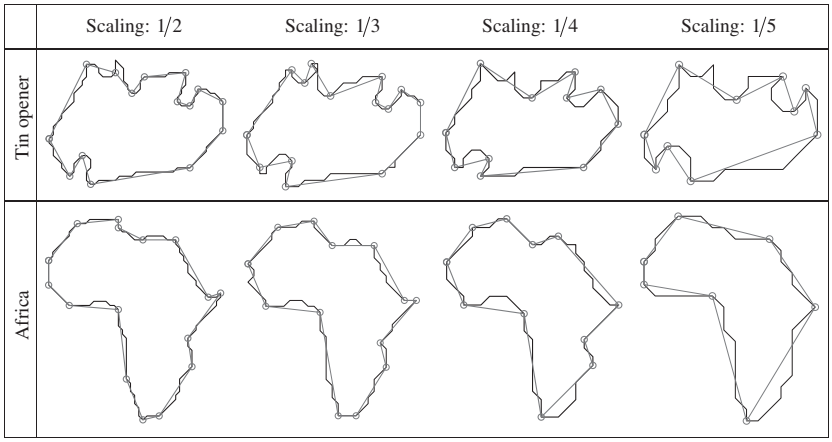

Fig. 11. Effect of scaling on Carmona (mod).

good approximation to all the digital curves. In Table 2 the number of pixels M in the digital curves, number of dominant points N found by a method, the maximum deviation $\max(d_m)$ of the polygon from the digital curve, the integral square error (ISE), the figure of merit (FOM), and the compression ratio $CR = M/N$ are listed. In general it is desired that $\max(d_m)$ and ISE are as less as possible and FOM and CR are as large as possible [32].

The value of the maximum deviation $\max(d_m)$ for the modified method is between 0.79 and 1.27 while it varies from 0.90 to 1.54 for the original method of Masood with $d_{tol} = 0.9$ and $d_{tol} = 1.2$. In fact for each digital curve, the value of $\max(d_m)$ for the modified method is always lesser than the original method of Masood with $d_{tol} = 1.2$. On the other hand, ISE of the original method with $d_{tol} = 0.9$ and $d_{tol} = 1.2$ is lower than the modified method for 15 images. This is because the original method Masood focuses on the minimization of ISE in each iteration, while the modified method focuses upon d_n. The modified method has a better CR than the original method with $d_{tol} = 0.9$ for all the digital curves and the original method with $d_{tol} = 1.2$ for 9 digital curves.

Further, in Fig. 6, we bring the curves of turtle, Africa, maple leaf, and dinosaur 1 to the notice. For turtle, we see that the modified method chooses much fewer dominant points ($N = 33$) than the original method with $d_{tol} = 0.9$ ($N = 38$) and $d_{tol} = 1.2$ ($N = 48$), while representing the digital curve effectively. Similar observations are noted for Africa and dinosaur 1. In maple leaf, though the number of dominant points obtained using the original method with $d_{tol} = 1.2$ and the modified method are same, the locations of the dominant points are different. Most differences occur in the concave regions of the digital curve and the locations where the curvature of the digital curve changes fast.

3.3. Carmona-Poyato's method of suppression of break points

3.3.1. Original method

Carmona-Poyato [12] (which we call Carmona for conciseness) is another method that begins with the list of break points as the initial set of dominant points (like Masood) and iteratively deletes points from the list of dominant points. However, beyond this initial similarity, the approach taken by Carmona is quite different. We highlight that there are two control parameters in Carmona's method [12], d_{tol} and r_{tol}. However, only r_{tol} is user specified and is used for termination condition only. On the other hand, d_{tol} is an internal control parameter used for controlling the iterative process and as a condition for deleting the dominant points in an iteration. It begins with a small value and slowly increases with the iteration number. The method is now summarized below.

Let the sequence of dominant points in a particular iteration be denoted by $\{P_n(x_n, y_n);\ n = 1 \text{ to } N\}_i$ where i denotes the iteration number. For explaining the method of Carmona, it shall be handy to define a distance d_n^C and a length l_n as follows:

$$d_n^C = \frac{\left|x_n(y_{n-1} - y_{n+1}) + y_n(x_{n+1} - x_{n-1}) + y_{n+1}x_{n-1} - y_{n-1}x_{n+1}\right|}{\sqrt{(x_{n+1} - x_{n-1})^2 + (y_{n-1} - y_{n+1})^2}}; \quad (15)$$

$n = 1 \text{ to } (N-1).$

Table 4
Performance parameters of RDP(mod) for digital curves with different scalings.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Tin opener						
Scaling 1	278	42	0.95	33.99	0.19	6.62
Scaling 1/2	135	17	1.41	43.16	0.18	7.94
Scaling 1/3	88	17	1.06	16.76	0.31	5.18
Scaling 1/4	62	13	1.46	17.15	0.28	4.77
Scaling 1/5	49	11	1.41	24.22	0.18	4.46
Africa						
Scaling 1	291	38	1.00	37.02	0.21	7.66
Scaling 1/2	137	17	1.34	30.85	0.26	8.06
Scaling 1/3	90	15	1.18	17.00	0.35	6.00
Scaling 1/4	64	12	1.41	17.13	0.31	5.33
Scaling 1/5	49	9	1.17	8.08	0.67	5.44

Table 5
Performance parameters of Masood(mod) for digital curves with different scalings.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Tin opener						
Scaling 1	278	45	1.00	20.69	0.30	6.18
Scaling 1/2	135	18	1.09	28.10	0.27	7.50
Scaling 1/3	88	16	0.98	16.92	0.33	5.50
Scaling 1/4	62	14	0.95	11.56	0.38	4.43
Scaling 1/5	49	13	1.00	12.22	0.31	3.77
Africa						
Scaling 1	291	39	0.97	25.89	0.29	7.46
Scaling 1/2	137	17	0.99	27.31	0.30	8.06
Scaling 1/3	90	15	1.18	15.40	0.39	6.00
Scaling 1/4	64	14	0.81	8.86	0.52	4.57
Scaling 1/5	49	9	0.89	8.00	0.68	5.44

Table 6
Performance parameters of Carmona(mod) for digital curves with different scalings.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Tin opener						
Scaling 1	278	29	2.28	145.12	0.07	9.59
Scaling 1/2	135	16	2.19	62.50	0.14	8.44
Scaling 1/3	88	16	1.37	26.79	0.21	5.50
Scaling 1/4	62	12	1.46	19.67	0.26	5.17
Scaling 1/5	49	11	1.79	27.16	0.16	4.45
Africa						
Scaling 1	291	35	1.24	52.62	0.16	8.31
Scaling 1/2	137	17	1.74	44.19	0.18	8.06
Scaling 1/3	90	14	1.66	25.27	0.25	6.43
Scaling 1/4	64	12	1.70	21.55	0.25	5.33
Scaling 1/5	49	8	1.54	14.70	0.42	6.13

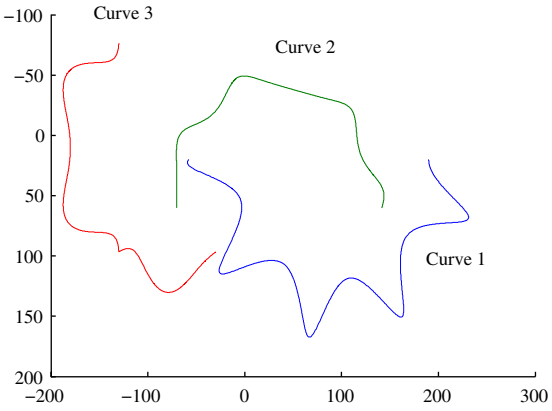

Fig. 12. Three non-digital curves given by Eqs. (20) and (21) and the parameters in Table 7.

$$l_n = \sqrt{(x_n - x_{n-1})^2 + (y_n - y_{n-1})^2} + \sqrt{(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2} - \sqrt{(x_{n+1} - x_{n-1})^2 + (y_{n+1} - y_{n-1})^2}. \quad (16)$$

The distance d_n^C is the distance of the nth dominant point from the line segment joining its adjacent dominant points and l_n is the difference between the perimeter of the polygon formed by $\{P_n(x_n, y_n);\ n = 1 \text{ to } N\}_i$ and the perimeter of the polygon formed by deleting the nth dominant point from $\{P_n(x_n, y_n);\ n = 1 \text{ to } N\}_i$. The superscript C in d_n^C denotes the method Carmona and is used to distinguish d_n^C from d_n in Eq. (11).

Carmona recommends that if the digital curve is a closed curve, a most suitable initial dominant point $P_{n=1}$ should be chosen before beginning the iterative procedure. Assuming that the initial sequence of break points is $\{P_n^{break}\}$. For this sequence, the distances d_n^C are computed and the point with the maximum value of d_n^C is identified, i.e. $n' = \arg(\max(d_n^C))$. Then, $P_{n=1} = P_{n'}^{break}$ and the remaining points in $\{P_n(x_n, y_n)\}_{i=0}$ follow the sequence.

The initial value of $d_{tol}(i = 0)$ is set as zero. In each iteration, the value of d_{tol} is increased by 0.5. Within an iteration, the first point for which $d_n^C < d_{tol}$ is deleted. This process is repeated for the newly obtained reduced sequence of dominant points, i.e., the first point for which $d_n^C < d_{tol}$ is deleted. This process of deletion is repeated till no point satisfies $d_n^C < d_{tol}$. At this point the current iteration is completed, the termination condition is checked and if the algorithm cannot be terminated then the next iteration is initiated. For the termination condition, a relative parameter r_i is defined as follows:

$$r_i = \frac{\max\left(\left\{l_n^{deleted}\right\}\right)}{\max(\{d_n\})}, \quad (17)$$

where d_n here is computed using Eq. (11) and $l_n^{deleted}$ correspond to the dominant points deleted in the current iteration. If at the end of the ith iteration, $r_i < r_{tol}$, where r_{tol} is a user specified control parameter, the algorithm is terminated.

3.3.2. Non-parametric adaptation of Carmona

Carmona can be made independent of user specified control parameter using the error bound in Eq. (4) by modifying the termination condition of Carmona. We compute d_n here is computed using Eq. (11). Further, we define:

$$d_n^{max} = s_n \phi_n \quad (18)$$

where s_n is the length of the line segment formed by joining the dominant points P_n and P_{n+1} and ϕ_n is the angle made by the line segment with the $x-$axis. Then at the end of an iteration, if there is a dominant point such that:

$$d_n > d_n^{max} \quad (19)$$

then the algorithm is terminated.

The pseudocodes of the original and modified methods are presented in Fig. 7 and the modifications have been highlighted for the ease of comparison.

3.3.3. Comparison of the Carmona original and Carmona modified methods

18 digital curves used in recent publications [10,12] and in Section 3.1.3 are considered. For comparison, two values of the control parameter r_{tol} of the original method of Carmona, $r_{tol} = 0.3$ and $r_{tol} = 0.7$(recommended in [12] for these digital curves) are used, and compared against the proposed modification which does not require user specified control parameter. The results are plotted in Fig. 8 and quantitative comparisons are provided in Table 3. Fig. 8 shows that the proposed modification provides good approximation to all the digital curves. In Table 3, the number of pixels M in the digital curves, number of dominant points N found by a method, the maximum deviation max(d_m) of the polygon from the digital curve, the integral square error (ISE), the figure of merit (FOM), and the compression ratio CR $= M/N$ are listed. In general it is desired that

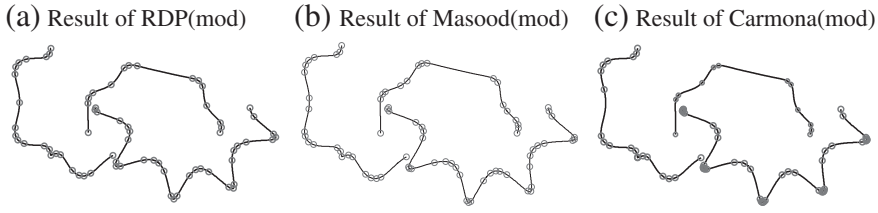

Fig. 13. Result of the three modified methods for non-digital curves given in Fig. 12.

Table 7
The parameters of three non-digital curves given by Eqs. (20) and (21).

Curve	a	m	n_1	n_2	n_3	θ_0	x_0	y_0	θ
Curve 1	100	9	9	14	11	0	90	20	$\theta = p\pi/1000$;
Curve 2		7	9	3	11	π	30	60	$p = 0$ to 1000
Curve 3		6	1	1	6	$\pi/3$	-80	10	

Table 8
Performance parameters of the three modified methods for non-digital curves given in Fig. 12.

	No. of data points (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
RDP(mod)	3003(1001 per curve)	70	1.40	355.59	0.12	42.90
Masood(mod)	3003(1001 per curve)	77	0.88	212.82	0.18	39.00
Carmona(mod)	3003(1001 per curve)	76	4.04	1754.49	0.02	39.51

max(d_m) and ISE are as less as possible and FOM and CR are as large as possible [32].

There are several interesting observations. First, see the results of Hammer in Table 3. We see that for $r_{tol} = 0.3$, the number of dominant points detected by the original method of Carmona is very large and the compression ratio is very poor. On the other hand, for $r_{tol} = 0.7$, the original method of Carmona misses some important features of the shape (like the top portion of the hammer). The modified method provides a better balance. Similar observations apply for Africa, plane 2, and plane 4. In such cases, the values of max(d_m), ISE, FOM, and CR for the modified method are in between the values of these parameters for the original method of Carmona with $r_{tol} = 0.3$ and $r_{tol} = 0.7$. Second, we consider the curves of hand, maple leaf, dinosaur 3, sword fish, plane 3, and plane 5. For these curves, we note that the dominant points detected by the original method with $r_{tol} = 0.3$ are the same as the dominant points detected by the modified method. This is because for such curves, the value of one of the elements in $\{d_n\}$ (used in Eqs. (17) and (19)) is larger than $d_n^{\max}$ and is sufficiently high to reduce the value of r_i below 0.3. Third, we consider the curves of screw driver, tin-opener, turtle, rabbit, and plane 1 in which the number of dominant points obtained by the modified method are larger than the original method with and $r_{tol} = 0.7$. For all these curves, the ISE for the modified method is significantly lower than the original method with $r_{tol} = 0.3$ and $r_{tol} = 0.7$ though the increase in the number of dominant points is not significant. Also, in most cases, the value of max(d_m) for the modified method is close to the value of max(d_m) for the original method with $r_{tol} = 0.3$. Fourth, we consider the digital curves of dog and dinosaur 1, and dinosaur 2, for which the modified and the original methods with $r_{tol} = 0.3$ and $r_{tol} = 0.7$ result into exactly the same dominant points. For these curves, the dominant point deleted in the last iteration reduced the value of r_i below 0.3 (from a value of r_i more than 0.7 in the last iteration) as well as the one dominant point satisfying $d_n > d_n^{\max}$. In addition to the above, it is worth noting that the value of max(d_m) ranges from 0.49 to 6.75 for the original method and from 1.29 to 2.73 for the modified method. This indicates that the modified method provides a better balance of the maximum deviation max(d_m).

Table 9
Performance parameters of the three modified methods for semi-digitized curves given in Fig. 14.

	No. of data points (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
RDP(mod)	100	15	1.52	34.21	0.20	6.67
Masood(mod)	100	14	1.08	19.55	0.37	7.14
Carmona(mod)	100	8	3.38	160.59	0.08	12.50

4. Discussions

4.1. About the algorithms in Section 3

In the previous section, three different dominant point detection methods were considered and adapted into the proposed non-parametric framework. Through this framework, the methods were made independent of user-specified control parameters. In addition, the comparison results showed that the modified method provide a more balanced performance in comparison to the corresponding original methods (the performance of which vary with the user specified control parameters).

It is important to emphasize that we do not intend to propose a new dominant point detection method or to provide a decision about the superiority of any one method among the three methods discussed in Section 3. Instead, the intention is to demonstrate how the proposed framework can be incorporated in methods of different types and nature. Even so, it is interesting to note how the modified RDP, modified Masood, and modified Carmona (all of which use the proposed bound) perform against each other. In this regard, the main concept of these algorithms is not disturbed and thus the modified versions of these methods retain the characteristics and nature of the original methods.

In order to compare the methods, the maximum deviations for these three methods are examined and compared against Eq. (4). From Eq. (4), assuming the smallest value of s to be $\sqrt{5}$, the maximum possible value of maximum deviation is less than 1.5315. The maximum deviations max(d_m) obtained for modified RDP, modified Masood, and

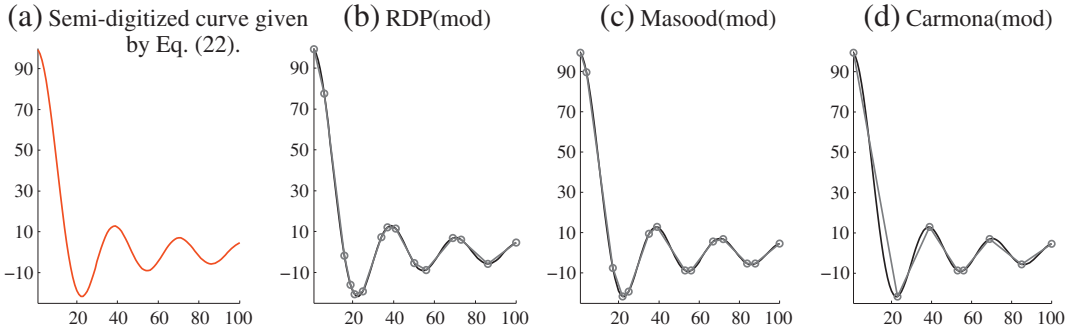

Fig. 14. Semi-digitized curve and the dominant points detected by the three modified methods in Sections 3.1.2, 3.2.2, and 3.3.2.

modified Carmona are 1.53, 1.27, and 2.73, respectively. RDP uses a splitting approach and it stops splitting further when the maximum deviation is below the chosen threshold. In the modified version, the threshold is computed using Eq. (4). Thus, as soon as the threshold reaches just below the bound, the method stops splitting further. So, it is expected that the actual maximum deviation of the modified RDP method is slightly lesser than the bound in Eq. (4).

Masood, after deleting a dominant point, optimizes the indices of the current dominant point such that an error parameter (ISE in the original and d_n given by Eq. (11) in the modified version) is as less as possible. Thus, it is a general characteristic of Masood that it gives a set of dominant points that fit quite closely with digital curve. This property is retained in the modified version as well (as seen in Fig. 6) and the maximum deviation of the modified Masood being less than RDP is a result of the re-optimization of the indices of the dominant points in Masood's method.

Carmona deletes one or few dominant points in each iteration such that the overall maximum deviation of the dominant points at the end of each iteration is equal to or larger than the internal control parameter d_{tol}. Thus, Carmona works by increasing the maximum deviation of the dominant points rather than reducing the maximum deviation. So, it is expected that the maximum deviation of Carmona is higher than the threshold in the final iteration, in both the original and the modified version. Thus, the maximum deviation of modified version of Carmona is higher than the bound.

Through the above discussion, two points are highlighted. First, the framework can be incorporated in dominant point detection methods without significantly altering their working principle and natural characteristics of the algorithms. Second, even while retaining the natural characteristics, incorporation of the framework can make the methods control parameter free and give a balanced performance.

The performance metrics used for comparison of the various algorithms also deserve a note. Especially, it is highlighted that the figure of merit (FOM) is well known to be biased towards small values of ISE [31–33]. For instance, zero value of ISE inherently results into infinity value of FOM, irrespective of the compression ratio. However, FOM is still used by many researchers as one of the basic merits. Thus, we have also used FOM in the current work. For other metrics, relevant discussions can be found in [31–33].

4.2. About scaling of digital curves and impact on dominant point detection

In this section, the effect of scaling on the performance of the three modified algorithms (Sections 3.1.2, 3.2.2, and 3.3.2) is illustrated. Two curves—tin opener and Africa—are used for this purpose. The resolution of each curve is reduced by a factor of 1/2, 1/3, 1/4, and 1/5. The modified versions of RDP, Masood, and Carmona are applied on these curves and the results are plotted in Figs. 9, 10, and 11 respectively. The performance parameters are tabulated in Tables 4, 5, and 6 respectively. The results for scaling 1 in Tables 4, 5, and 6 are taken from Tables 1, 2, and 3 respectively. The performance parameters indicate that when the curves are significantly de-scaled (for example by a factor 1/5) and loose the details, the compression ratio decreases for all the three methods. However, the behavior of the maximum deviation and the ISE does not change significantly. This implies that the methods retain their natural fitting characteristics. It is also interesting to note that all the three methods represent the curves well.

4.3. About using the proposed framework for non-digitized and semi-digitized curves

In this section, the applicability of the three modified algorithms (Sections 3.1.2, 3.2.2, and 3.3.2) for non-digitized and semi-digitized curves is demonstrated. First, we consider a set of non-digitized curves given by the following parametric equations:

$$r = a\left(\left(\cos\left(\frac{m\theta}{4}\right)\right)^{n_2} + \left(\sin\left(\frac{m\theta}{4}\right)\right)^{n_3}\right)^{-\frac{1}{n_1}} \tag{20}$$

$$x = r\cos(\theta + \theta_0) + x_0; y = r\sin(\theta + \theta_0) + y_0 \tag{21}$$

where a, m, n_1, n_2, n_3, θ_0, x_0, and y_0 are the parameters of the curve. Using Eqs. (20) and (21), three curves with the set of parameters in Table 7 are generated. The values of the x and y coordinates given by Eq. (21) are kept in the form of double-precision floating numbers in Matlab. The curves are shown in Fig. 12.

For these curves, the result of RDP(mod), Masood (mod), and Carmona(mod) are given in Fig. 13 and Table 8. The results clearly show that the modified methods based on the proposed non-parametric framework perform well even for non-digitized (real valued) curves.

Now we consider a semi-digitized curve, similar to data plots in which one axis (x axis) is digitized. A data plot given by the following equation and plotted in Fig. 14(a) is considered as an example:

$$y = 100\frac{\sin(x/5)}{(x/5)};\ x \in \{1, 2, \ldots, 100\} \tag{22}$$

For this plot, the dominant points detected by the three modified algorithms (Sections 3.1.2, 3.2.2, and 3.3.2) are shown in Fig. 14(b–d) respectively. The performance parameters are tabulated in Table 9.

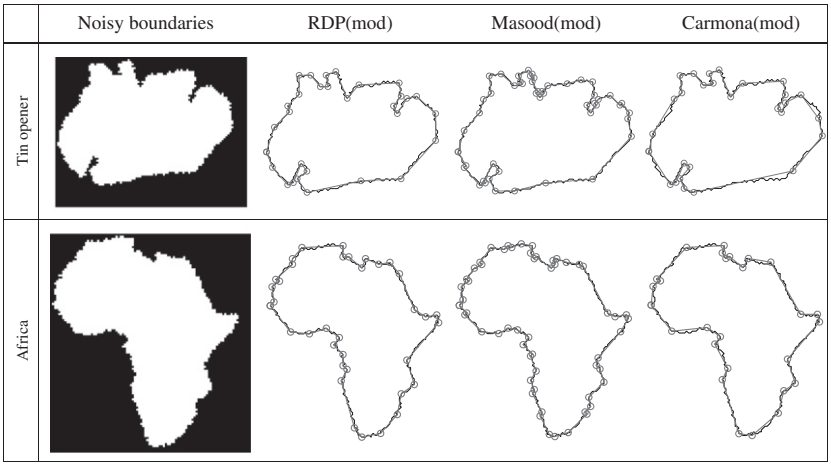

Fig. 15. Performance of the three modified methods for noisy digital curves.

Table 10
Performance parameters of the three modified methods for noisy digital curves.

	No. of pixels (M)	No. of dominant points (N)	max(d_m)	ISE	FOM	CR
Tin opener						
RDP(mod)	278	42	0.95	33.99	0.19	6.62
RDP(mod) noisy	296	31	1.40	98.20	0.10	9.55
Masood(mod)	278	45	1.00	20.69	0.30	6.18
Masood(mod) noisy	296	50	1.20	60.29	0.10	5.92
Carmona(mod)	278	29	2.28	145.12	0.07	9.59
Carmona(mod) noisy	296	29	3.41	237.69	0.04	10.21
Africa						
RDP(mod)	291	38	1.00	37.02	0.21	7.66
RDP(mod) noisy	321	37	1.36	93.35	0.09	8.68
Masood(mod)	291	39	0.97	25.89	0.29	7.46
Masood(mod) noisy	321	57	0.99	65.33	0.09	5.63
Carmona(mod)	291	35	1.24	52.62	0.16	8.31
Carmona(mod) noisy	321	28	2.04	166.37	0.07	11.46

The results clearly demonstrate the applicability of the proposed framework for semi-digitized curves like data plots as well.

4.4. About using the proposed framework for noisy digital curves

This section presents the performance of the three modified algorithms (Sections 3.1.2, 3.2.2, and 3.3.2) for noisy digital curves. For this purpose, we added noise to the digital curves using the Kanungo model [34]. The parameters used for Kanungo noise model were $\alpha_0 = \beta_0 = 4$, $\alpha = \beta = 2$, and $\eta_f = \eta_b = 0$. The performance of the three methods for two curves—tin opener and Africa—is presented in Fig. 15 and tabulated in Table 10. Results for RDP(mod), Masood(mod) and Carmona(mod) in Table 10 correspond to the results in Tables 1, 2, and 3 respectively. It is noted that the compression ratios (CR) of RDP(mod) and Carmona(mod) increase for the noisy curves, while the compression ratio (CR) of Masood(mod) decreases for noisy curves. This is consistent with the nature of Masood's method which supports very close fitting and consequently results in more dominant points in order to fit closely to the noise. On the other hand, RDP and Carmona both have relatively more smoothing effects as compared to Masood.

5. Conclusion

In this paper, a non-parametric framework for dominant point detection methods is proposed. The approach is based upon theoretical bound of the deviation of the pixels obtained by the digitization of a line segment. The approach is to use this bound in a dominant point detection method as either the optimization goal or the termination condition or both. It is shown that this approach can be incorporated in various types of dominant point detection methods easily to make them independent of control parameter and related heuristics. This is illustrated by modifying three different dominant point detection methods (RDP [21,22], Masood [10], and Carmona [12]). The results show that as compared to the use of control parameters in the original versions of these methods, the modifications of the methods using the non-parametric approach provide a more balanced performance and good approximation of the digital curves. It is also shown that the modified versions of the dominant point detection methods can still retain their original natural characteristics. This framework will be useful for applications, the performance of which suffers from heuristic choices of control parameters. It is also useful for applications where the a priori information about the input data is limited and heuristics for choosing the control parameters may not be available. The utility of the proposed framework is also shown for non-digital, semi-digital, and noisy digital curves. Though the approach is applied for three methods only in this paper, the approach can be suitably applied in most dominant point detection methods to make them free of heuristics.

Appendix A

Consider the effect of digitization on the slope of a line segment connecting two points (which may or may not be pixels). Here, an upper bound for the deviation of the pixels obtained by the digitization of the line segment is derived. Due to digitization in the case of digital images, a general point $P(x, y)$ is approximated by a pixel $P'(x', y')$ as follows:

$$x' = \text{round}(x); \quad y' = \text{round}(y) \tag{23}$$

where round(x) denotes the rounding of the value of real number x to its nearest integer. $P'(x', y')$ satisfies the following:

$$x', y' \in \mathbb{Z} \tag{24}$$

$$x' = x + \Delta x; \quad y' = y + \Delta y \tag{25}$$

$$-0.5 \le \Delta x \le 0.5, -0.5 \le \Delta y \le 0.5 \tag{26}$$

Let the slope of the line P_1P_2 (actual line) be denoted as m and the slope of the line $P_1'P_2'$ (digital line) be denoted as m'. Then,

$$m = \tan\phi = \frac{y_2 - y_1}{x_2 - x_1} \tag{27}$$

$$m' = \frac{y'_2 - y'_1}{x'_2 - x'_1} = \left(m + \frac{\Delta y_2 - \Delta y_1}{x_2 - x_1}\right) \Big/ \left(1 + \frac{\Delta x_2 - \Delta x_1}{x_2 - x_1}\right) \tag{28}$$

The angular difference between the numeric tangent and the digital tangent is used as the estimate of the error. This angular difference is given as:

$$\partial\phi = \left|\tan^{-1}(m) - \tan^{-1}\left(m'\right)\right| = \left|\tan^{-1}\left(\frac{m - m'}{1 + mm'}\right)\right| \tag{29}$$

Substituting Eq. (28) in Eq. (29), we get:

$$\begin{aligned}
\partial\phi &= \left|\tan^{-1}\left(\frac{\left(1 + \frac{\Delta x_2 - \Delta x_1}{x_2 - x_1}\right)m - \left(m + \frac{\Delta y_2 - \Delta y_1}{x_2 - x_1}\right)}{\left(1 + \frac{\Delta x_2 - \Delta x_1}{x_2 - x_1}\right) + m\left(m + \frac{\Delta y_2 - \Delta y_1}{x_2 - x_1}\right)}\right)\right| \\
&= \left|\tan^{-1}\left(\frac{\left(\frac{\Delta x_2 - \Delta x_1}{x_2 - x_1}\right)m - \left(\frac{\Delta y_2 - \Delta y_1}{x_2 - x_1}\right)}{(1 + m^2) + \left(\frac{\Delta x_2 - \Delta x_1}{x_2 - x_1}\right) + m\left(\frac{\Delta y_2 - \Delta y_1}{x_2 - x_1}\right)}\right)\right| \\
&= \left|\tan^{-1}\left(\frac{m(\Delta x_2 - \Delta x_1) - (\Delta y_2 - \Delta y_1)}{(1 + m^2)(x_2 - x_1) + (\Delta x_2 - \Delta x_1) + m(\Delta y_2 - \Delta y_1)}\right)\right|
\end{aligned} \tag{30}$$

Using Eq. (27) in Eq. (30), and substituting

$$s = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{31}$$

$$t = \frac{(\Delta x_2 - \Delta x_1)(x_2 - x_1)}{s^2} + \frac{(\Delta y_2 - \Delta y_1)(y_2 - y_1)}{s^2} \tag{32}$$

we get the following:

$$\partial\phi = \left|\tan^{-1}\left(\left(\frac{x_2 - x_1}{s^2}\right)(1 + t)^{-1}(m(\Delta x_2 - \Delta x_1) - (\Delta y_2 - \Delta y_1))\right)\right| \tag{33}$$

Now we highlight the following points that are together used in Eq. (33) in order to derive the analytical error bound.

- Due to Eq. (26), the maximum value of $|\Delta x_2 - \Delta x_1|$ and $|\Delta y_2 - \Delta y_1|$ is 1.
- $|(x_2 - x_1)/s|$ and $|(y_2 - y_1)/s|$ are both less than or equal to 1 due to the definition of s in Eq. (31).
- For any digital line made of more than 3 pixels for 4-connected digital curve and more than 2 pixels for 8-connected digital curve, s is always more than $\sqrt{2}$.
- As a consequence of the above points, $|t|<1$.
- In general, the exact values of $|\Delta x_2 - \Delta x_1|$ and $|\Delta y_2 - \Delta y_1|$ are not known, which implies that the value of t is not known except for the above mentioned fact that $|t|<1$.

Thus, in order to derive the bound, infinite geometric series expansion is used in Eq. (33) and $\partial\phi$ can be written as:

$$\partial\phi = \left| \tan^{-1}\left(\left(\frac{x_2 - x_1}{s^2}\right)(m(\Delta x_2 - \Delta x_1) - (\Delta y_2 - \Delta y_1)) \left(\sum_{n=0}^{\infty} (-t)^n \right) \right) \right| \tag{34}$$

Further we note that, $\partial\phi$ has a maximum value when $|\Delta x_2 - \Delta x_1| = |\Delta y_2 - \Delta y_1| = 1$. Thus, using the definition of ϕ in Eq. (27), the maximum value of t is given by:

$$t_{\max} = \left(\frac{1}{s}\right)(|\cos\phi| + |\sin\phi|) \tag{35}$$

Thus, the maximum value of $\partial\phi$ is gives as:

$$\partial\phi_{\max} = \max\left(\tan^{-1}\left\{ \frac{1}{s}(|\sin\phi \pm \cos\phi|)\left(\left| \sum_{n=0}^{\infty} (-t_{\max})^n \right| \right) \right\} \right) \tag{36}$$

Since $t_{\max} \le 1$, Eq. (36) is bounded. Truncating the series by retaining up to second order terms only, Eq. (36) can be written as:

$$\partial\phi_{\max} = \max\left(\tan^{-1}\left\{ \frac{1}{s}(|\sin\phi \pm \cos\phi|)\left(1 - t_{\max} + t_{\max}^2\right) \right\} \right) + O\left(t_{\max}^3\right) \tag{37}$$

References

[1] S. Lavallee, R. Szeliski, Recovering the position and orientation of free-form objects from image contours using 3D distance maps, IEEE Trans. Pattern Anal. Mach. Intell. 17 (4) (1995) 378–390.

[2] J.H. Elder, R.M. Goldberg, Image editing in the contour domain, IEEE Trans. Pattern Anal. Mach. Intell. 23 (3) (2001) 291–296.

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

[4] R. Yang, Z. Zhang, Eye gaze correction with stereovision for video-teleconferencing, IEEE Trans. Pattern Anal. Mach. Intell. 26 (7) (2004) 956–960.

[5] A. Kolesnikov, P. Fränti, Data reduction of large vector graphics, Pattern Recogn. 38 (3) (2005) 381–394.

[6] D. Brunner, P. Soille, Iterative area filtering of multichannel images, Image Vis. Comput. 25 (8) (2007) 1352–1364.

[7] F. Mokhtarian, A. Mackworth, Scale-based description and recognition of planar curves and two-dimensional shapes, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-8 (1) (1986) 34–43.

[8] D.K. Prasad, M.K.H. Leung, S.Y. Cho, Edge curvature and convexity based ellipse detection method, Pattern Recogn. 45 (9) (2012) 3204–3221.

[9] H. Imai, M. Iri, Polygonal approximation of a curve (formulations and algorithms), In: in: G.T. Toussaint (Ed.), Computational Morphology, 1988, pp. 71–86, Amsterdam.

[10] A. Masood, Dominant point detection by reverse polygonization of digital curves, Image Vis. Comput. 26 (5) (2008) 702–715.

[11] A. Masood, S.A. Haq, A novel approach to polygonal approximation of digital curves, J. Vis. Commun. Image Represent. 18 (3) (2007) 264–274.

[12] A. Carmona-Poyato, F.J. Madrid-Cuevas, R. Medina-Carnicer, R. Muñoz-Salinas, Polygonal approximation of digital planar curves through break point suppression, Pattern Recogn. 43 (1) (2010) 14–25.

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

[14] W.Y. Wu, An adaptive method for detecting dominant points, Pattern Recogn. 36 (10) (2003) 2231–2237.

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

[16] P. Bhowmick, B.B. Bhattacharya, Fast polygonal approximation of digital curves using relaxed straightness properties, IEEE Trans. Pattern Anal. Mach. Intell. 29 (9) (2007) 1590–1602.

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

[18] C.-H. Teh, R.T. Chin, On the detection of dominant points on digital curves, IEEE Trans. Pattern Anal. Mach. Intell. 11 (8) (1989) 859–872.

[19] J.C. Perez, E. Vidal, Optimum polygonal approximation of digitized curves, Pattern Recognit. Lett. 15 (8) (1994) 743–750.

[20] D.G. Lowe, Three-dimensional object recognition from single two-dimensional images, Artif. Intell. 31 (3) (1987) 355–395.

[21] U. Ramer, An iterative procedure for the polygonal approximation of plane curves, Comput. Graphics Image Process. 1 (3) (1972) 244–256.

[22] 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.

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

[24] B.K. Ray, K.S. Ray, An algorithm for detection of dominant points and polygonal approximation of digitized curves, Pattern Recognit. Lett. 13 (12) (1992) 849–856.

[25] P.V. Sankar, C.U. Sharma, A parallel procedure for the detection of dominant points on a digital curve, Comput. Graphics Image Process. 7 (4) (1978) 403–412.

[26] M. Salotti, Optimal polygonal approximation of digitized curves using the sum of square deviations criterion, Pattern Recogn. 35 (2) (2002) 435–443.

[27] N. Ansari, K.W. Huang, Non-parametric dominant point detection, Pattern Recogn. 24 (9) (1991) 849–862.

[28] T.M. Cronin, A boundary concavity code to support dominant point detection, Pattern Recognit. Lett. 20 (6) (1999) 617–634.

[29] B. Sarkar, S. Roy, D. Sarkar, Hierarchical representation of digitized curves through dominant point detection, Pattern Recognit. Lett. 24 (15) (2003) 2869–2882.

[30] D.K. Prasad, C. Quek, M.K.H. Leung, S.Y. Cho, A parameter independent line fitting method, In: *Asian Conference on Pattern Recognition (ACPR)*, Beijing, China, 2011, pp. 441–445.

[31] D.K. Prasad, M.K.H. Leung, Polygonal representation of digital curves, In: in: S.G. Stanciu (Ed.), Digital Image Processing, 2012, pp. 71–90, (InTech).

[32] P.L. Rosin, Techniques for assessing polygonal approximations of curves, IEEE Trans. Pattern Anal. Mach. Intell. 19 (6) (1997) 659–666.

[33] A. Carmona-Poyato, R. Medina-Carnicer, F.J. Madrid-Cuevas, R. Muoz-Salinas, N.L. Fernndez-Garca, A new measurement for assessing polygonal approximation of curves, Pattern Recogn. 44 (1) (2011) 45–54.

[34] T. Kanungo, R.M. Haralick, H.S. Baird, W. Stuetzle, M.D., Document degradation models: parameter estimation and model validation, In: IAPR Wolkshop on Machine Vision Application, Kawasaki, Japan, 1994.