Image and Vision Computing 30 (2012) 843–859

Contents lists available at SciVerse ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

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

ARTICLE INFO

Article history:
Received 27 December 2011
Received in revised form 22 April 2012
Accepted 23 June 2012

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

ABSTRACT

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.

© 2012 Elsevier B.V. All rights reserved.

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-#

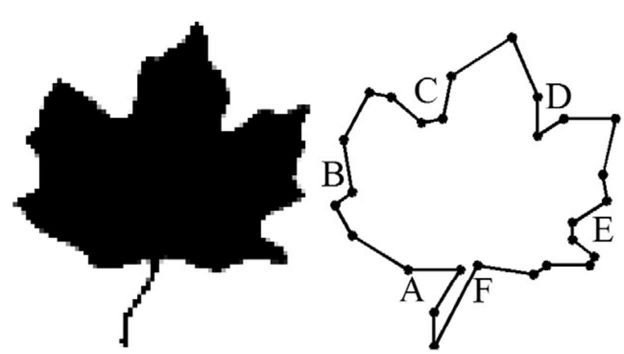

Fig. 1. An example of a digital shape and its polygonal approximation. The dominant points are denoted using the dots. The boundary of the maple leaf shape consists of 244 pixels. The polygonal approximation uses 27 dominant points.

☆ This paper has been recommended for acceptance by Sven Dickinson.
* Corresponding author.
E-mail address: dilipprasad@gmail.com (D.K. Prasad).

0262-8856/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2012.06.010

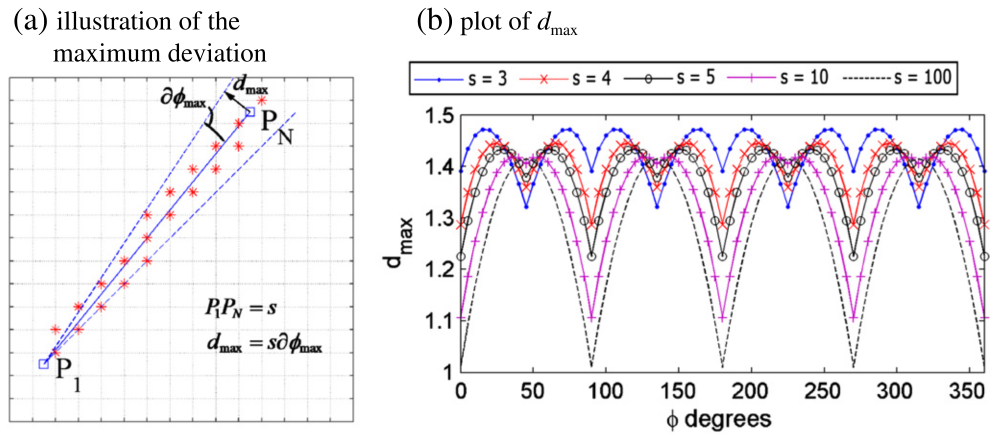

Fig. 2. The maximum deviation $d_{\max}$ of a continuous line segment and the digital line segment obtained from the continuous line segment.

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= $\{DP, \quad \mathrm{RDP\_max}\ (P_1, P_{\max})\}$.
DP= $\{DP, \quad \mathrm{RDP\_max}\ (P_{\max}, P_N)\}$.
}
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= $\{DP, \quad \mathrm{RDP\_max}\ (P_1, P_{\max})\}$.
DP= $\{DP, \quad \mathrm{RDP\_max}\ (P_{\max}, P_N)\}$.
}
End
Remove redundant points in DP.
Return(DP).
}

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

Download English Version:

https://daneshyari.com/en/article/526777

Download Persian Version:

https://daneshyari.com/article/526777

Daneshyari.com