THE DETECTION OF DOMINANT POINTS ON DIGITAL CURVES BY SCALE-SPACE FILTERING

SOO-CHANG PEI and CHAO-NAN LIN

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, Republic of China

Abstract—The detection of dominant points is an important preprocessing step for shape recognition. An effective method of scale-space filtering with a Gaussian kernel is introduced to detect dominant points on digital curves. The conventional polygonal approximation algorithms are time-consuming and need input parameter tuning for Gaussian smoothing the noise and quantization error, also they are sensitive to scaling and rotation of the object curve. The above difficulty can be overcome by finding out the dominant points at each scale by scale-space filtering. By tracing back the dominant point contours in the scale-space image, the stable cardinal curvature points can be detected very accurately. This new method requires no input parameters, and the resultant dominant points do not change under translation, rotation and scaling. Meanwhile a fast convolution algorithm is proposed to detect the dominant points at each scale.

Dominant points   Scale-space   Gaussian smoothing   Curvature

1. INTRODUCTION

Various representations of two-dimensional shape have been developed in the computational vision literature.[1] It has been suggested from the viewpoint of the human visual system that the dominant points having high curvature are sufficient to characterize the shape of the object. In general, the current existing methods for detecting dominant points are divided into two major groups: one is to detect the dominant points directly through angle or corner detection schemes,[2–7] and the other is to obtain a piecewise linear polygonal approximation of the digital curve subject to a certain constraint on the goodness of fit.[8–10]

The recent Teh–Chin algorithm[7] is a good example of detecting dominant points based on angle detection. The algorithm does not require input parameters and works well on an object curve which is not corrupted with noise. However, it is very sensitive to noise, some false dominant points will be detected due to quantization error and noise effect. On the other hand, the curvature guided polygonal approximation method[10] makes use of Gaussian smoothing to reduce the effect of noise, but it requires two input parameters, namely, the width of a Gaussian filter and the collinearity tolerance for doing a split-and-merge algorithm. There is a trade-off in selecting the width of the Gaussian filter, a larger width will remove small details of the boundary curvature, a smaller width will permit false concavities and convexities. Recently, Ansari and Huang used the advantages of both approaches, they introduced a new method[11] which was non-parametric with no input parameters and less sensitive to noise. However, this method requires the support region for each contour point to be determined, and then the contour is smoothed by an adaptive Gaussian filter with a width proportional to the support region. This complicates the smoothing procedure a lot and increases the computational efforts and complexities.

Mokhtarian and Mackworth[12] have suggested a number of necessary criteria which any reliable method for curve representation and recognition must satisfy:

(1) The feature selection or extraction method must be computed efficiently.

(2) The extracted feature should be invariant under translation, rotation and scaling.

(3) The feature representation should contain information about the curve at varying levels of detail, and should uniquely specify a single curve.

In order to satisfy the above conditions, we introduce the scale-space filtering to find out the relevant dominant points at each scale. Scale-space filtering[13] is a useful signal description method that deals gracefully with the problem about scales by treating the size parameter of the smoothing kernel as a continuous parameter. As the scale parameter is varied, the dominant points of each scale in the smoothed signal in general move continuously. By tracing back the dominant point contour in the scale-space image, we are able to detect the stable cardinal curvature points very accurately. They are extreme curvature points of the digital curve that are stable with respect to Gaussian smoothing for a reasonable wide range of Gaussian filter width, and can represent a unique shape attribute of the curve. Also the resultant dominant points are shown to be invariant under translation, rotation and scaling. Meanwhile a fast convolution algorithm is proposed to extract the dominant points at each scale very efficiently.

The arrangement of this paper is organized as follows. In Section 2, we introduce how to detect the dominant points at each scale by scale-space filtering.

In Section 3, we explain how to correct the moved dominant points on each scale into the right locations by tracing back the scale-space image. Section 4 describes how to determine an appropriate scale to find out the stable cardinal curvature points accurately. Also a fast convolution algorithm is introduced in Section 5 to compute the derivative of curvature efficiently. Finally conclusions are made in Section 6.

2. DETECTION OF DOMINANT POINTS BY SCALE-SPACE FILTERING

Points with high curvature along a shape contour are rich in information content and are sufficient to characterize the object contour. We call these points the dominant points. In this section, the curvature computation and the dominant point detection will be described in detail.

2.1. Curvature computation on a curve

Consider a path length variable t along the curve C and express the curve in terms of two functions $x(t)$ and $y(t)$

$$C = \{x(t), y(t)\} \tag{1}$$

where t is the path length of the curve, $x(t)$ the x-coordinate of the curve at path length t, and $y(t)$ the y-coordinate of the curve at path length t. Let K be the curvature of a planar curve at a point p which is defined as the instantaneous rate of change of the slope angle of the tangent at point p with respect to length t

$$K = \frac{d\theta}{dt}. \tag{2}$$

K can be computed if the curve is expressed in terms of the derivatives of functions $x(t)$ and $y(t)$. Define

$$y' = \frac{dy}{dx}, \quad y'' = \frac{d^2y}{dx^2}. \tag{3}$$

It is known that

$$K = \frac{y''}{(1 + (y')^2)^{3/2}}. \tag{4}$$

Since the curve is a function of path length t, the above y' and y'' can be expressed in terms of the first and second derivatives of $x(t)$ and $y(t)$. Denote

$$\dot{x} = \frac{dx}{dt}, \quad \ddot{x} = \frac{d^2x}{dt^2}, \quad \dot{y} = \frac{dy}{dt}, \quad \ddot{y} = \frac{d^2y}{dt^2}. \tag{5}$$

Then

$$y' = \frac{\dot{y}}{\dot{x}}, \quad y'' = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{\dot{x}^3} \tag{6}$$

and

$$K = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}}. \tag{7}$$

2.2. Finding out the relevant dominant points by scale-space filtering

Scale-space filtering is useful to describe a signal qualitatively.[13] A unified and organized description can be produced by treating each contour in the scale-space image as a single physical event, observed through a continuum of Gaussian smoothing filters, rather than as a set of unrelated events. Hence we can find out the relevant dominant points from the contour in the scale-space image. In this section, we introduce how to derive a scale-space image from a shape and how to find out the dominant points from the scale-space image.

Let $g(t, \sigma)$ be the Gaussian kernel

$$g(t, \sigma) = \frac{1}{\sqrt{(2\pi)}\sigma} e^{-t^2/2\sigma^2} \tag{8}$$

where t is the path length of the shape contour and σ the filter width (spatial support) of the Gaussian kernel. By changing the value of σ, we can get the Gaussian kernel at various scales.

In order to reduce the quantization error and noise in computing the curvature of the digital curve, the digital curve is first smoothed by the Gaussian filter with small scale σ, i.e.

$$x(t) = x(t)*g(t, \sigma) = \frac{1}{\sqrt{(2\pi)}\sigma} \int_{-\infty}^{\infty} x(\tau)e^{-(t-\tau)^2/2\sigma^2} d\tau \tag{9a}$$

$$y(t) = y(t)*g(t, \sigma) = \frac{1}{\sqrt{(2\pi)}\sigma} \int_{-\infty}^{\infty} y(\tau)e^{-(t-\tau)^2/2\sigma^2} d\tau \tag{9b}$$

where $*$ is the convolution operation and $\sigma = 2$ pixels is used in our paper. By equation (7), the curvature of the contour can be evaluated by using the above smoothed $x(t)$ and $y(t)$

$$K(t) = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}}. \tag{10}$$

The curvature of the contour at various scales is computed by convolving $K(t)$ with a one-dimensional kernel $g(t, \sigma)$

$$K(t, \sigma) = K(t)*g(t, \sigma) = \frac{1}{\sqrt{(2\pi)}\sigma} \int_{-\infty}^{\infty} K(\tau)e^{-(t-\tau)^2/2\sigma^2} d\tau \tag{11}$$

where $K(t, \sigma)$ is the curvature at scale σ. Since the dominant points are located at the extreme of curvature which is equivalent to the zero-crossing of the derivative of the curvature

$$\dot{K}(t, \sigma) = \frac{dK(t, \sigma)}{dt}. \tag{12}$$

Equations (11) and (12) are combined to reduce the quantization error

$$\dot{K}(t, \sigma) = \frac{d}{dt}\{K(t)*g(t, \sigma)\} = K(t)*\frac{dg(t, \sigma)}{dt}. \tag{13}$$

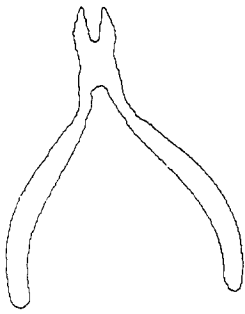Fig. 1. Original curve of a wire cutter.

(a) $\sigma = 3$ (b) $\sigma = 19$
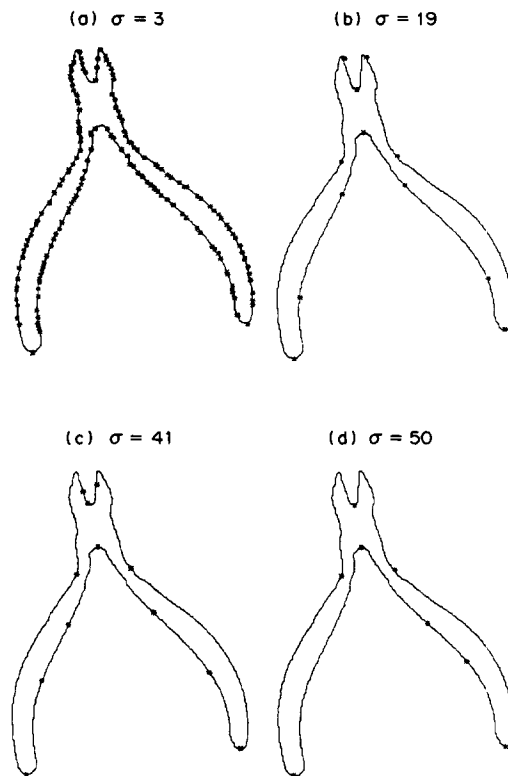
(c) $\sigma = 41$ (d) $\sigma = 50$

Fig. 3. The detected dominant points on the contour at various scales. Notice that the top two sharp tips in (c) drift from the original positions very severely.

Then we can find the dominant points at each scale. Figure 1 shows the original contour of a wire cutter. As the scale parameter σ is varied and increased by a small amount, the extrema or zero-crossing in the smoothed curvature and its derivatives in general move continuously. The behavior of extrema over scale is characterized in terms of the scale-space image as shown in Fig. 2. The scale-space images contain several contours which are closed at the top and open at the bottom, now extrema points are created at higher levels of detail (more smoothing). One can track back the extreme points at lower levels in the neighborhood of the previous ones at higher levels. The positions of dominant points on the contour at various scales are illustrated in Figs 3(a)–(d). In the next section, we will describe how to combine various levels of information in the scale-space image in order to obtain a representation for the curve.

3. POSITION CORRECTION OF DOMINANT POINTS BY TRACING BACK ALONG SCALE-SPACE CONTOURS

From Figs 3(a) to (d) notice that the locations of dominant points will drift as the scale increases (see the top two sharp tips of the wire cutter in Fig. 3(c)).

In order to correct the shifting, we must trace it back along the contour in the scale-space image, as shown in Fig. 4(a).

The location position of the dominant points at smaller scale is much more accurate than the ones at larger scale. In this paper, we start the scale-space filtering from the scale $\sigma = 3$, and compute the curva-
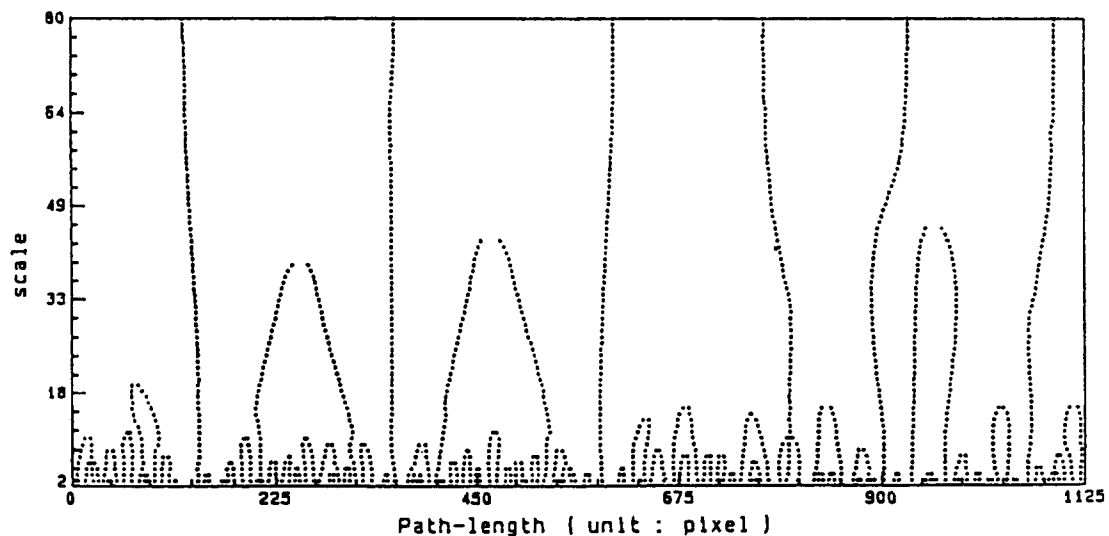

Path-length (unit : pixel)

Fig. 2. Scale-space image of the wire cutter in Fig. 1, with the dominant points drifted as the scale increases.

ture of various levels for the scales $\sigma = 3, 5, 7, 9, \ldots, 79$. Let array $p3(i)$ record the root location of dominant points at the smallest scale, array $p2(i)$ the location of dominant points at current σ, and array $p1(i)$ the location of dominant points at the preceding scale $\sigma - 2$. The tracing back step is done immediately following the detection of the dominant points at each scale, this will correct the shifted dominant points into the right position, the major algorithm is summarized as follows:

(1) Start at scale $\sigma = 3$, find the dominant points at the extreme of curvature $K(t, 3)$, which are located at path length $t = n_1, n_2, n_3, \ldots, n_{M1}$. Then save them in array $p3(i)$ and $p1(i)$ for $i = 1, 2, \ldots, M1$.

(2) Increasing the filter width by 2, $\sigma = \sigma + 2$. Find the dominant points at the extreme of curvature $K(t, \sigma)$, which are located at path length $t = m_1, m_2, \ldots, m_{M2}$. Then save them in array $p2(i)$ for $i = 1, 2, \ldots, M2$. By the scale-space theory,[14] the number of dominant points is monotonously decreasing as the scale is increased, i.e. $M2 \leq M1$.
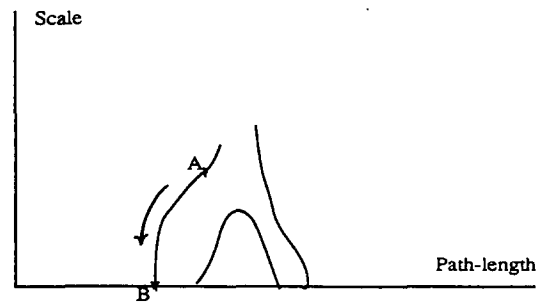

(a) $\sigma = 5$ (b) $\sigma = 9$

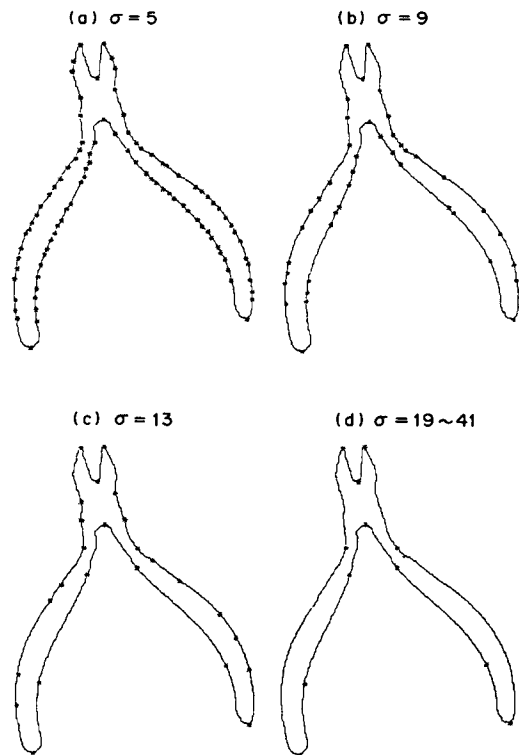(c) $\sigma = 13$ (d) $\sigma = 19 \sim 41$

Fig. 5. The detected dominant points on the curve at various scales are traced back into the right locations, as compared with Fig. 3. There are 12 stable extreme curvature dominant points detected in the range of Gaussian smoothing $\sigma = 19-41$.

Fig. 4(a). Position correction of dominant points by tracing back along scale-space contours.

(3) Trace the shifted dominant point location back to $\sigma = 3$. The algorithm is listed as follows:

For $i = 1$ to $M2$

search $p1(k)$ so that $\underset{k}{\text{Min}} |p1(j) - p2(i)|$,
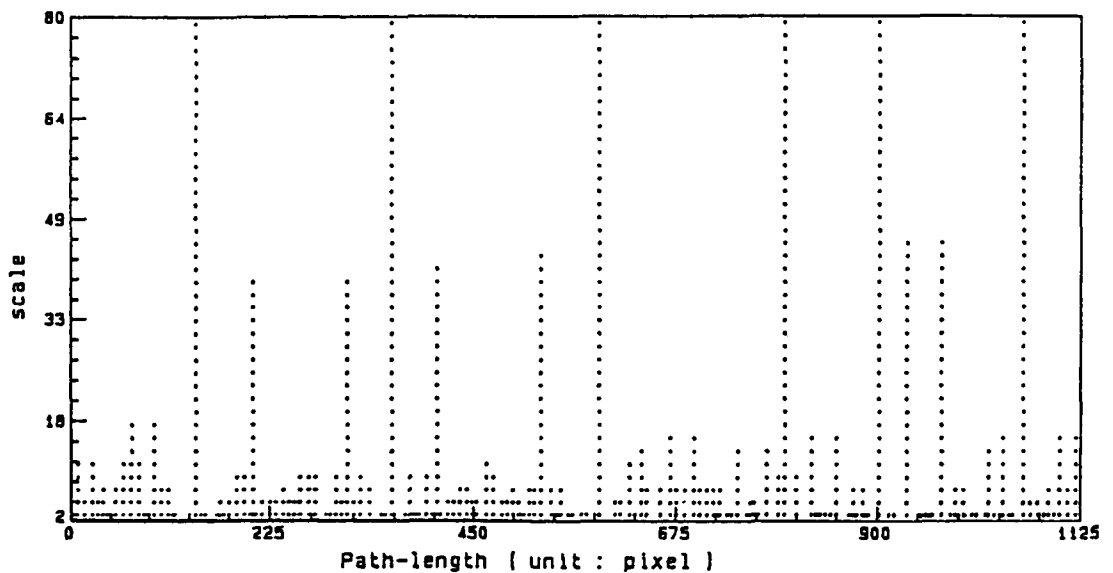
for $1 \leq j \leq M1$ $p3(i) = p3(k)$

Fig. 4(b). Tracing back scale-space image of Fig. 2.

end for

(4) For $i = 1$ to $M2$

$\quad p1(i) = p2(i)$

end for

go to step (2).

This tracing back algorithm is very simple and can be computed very efficiently. Figure 4(b) illustrates the tracing back scale-space image of Fig. 2. The dominant points detected on the curve at various scales are traced back into the right locations as shown in Figs 5(a)-(d). By comparison of Figs 3(c) and 5(d) with the same scale $\sigma = 41$, the dominant points found by tracing back are more accurate. Also we notice that there are 12 extreme curvature dominant points in Fig. 5(d) that are stable with respect to Gaussian smoothing for a reasonable range of values of $\sigma = 19$–41. We shall refer to those stable local extreme curvature points as the cardinal curvature points.[10]

4. TERMINATION AT AN APPROPRIATE SCALE VALUE TO DETECT THE CARDINAL CURVATURE POINTS

How to decide the scale value to terminate Gaussian smoothing? The resultant cardinal curvature points

are stable with respect to a reasonable degree of scaling. The simplest method is that the number of dominant points is specified in advance. The scale is gradually increased until the number of dominant points at one scale is equal to the specified one. Another preferred method is that we terminate the algorithm when the number of dominant points does not change in a duration $\Delta\sigma$ above some threshold. We modified step (4) of the above algorithm in Section 3 as

(4) If $M1 = M2$ then

$\quad$ counter $=$ counter $+ 1$

$\quad$ If counter $>$ threshold, find the cardinal dominant points.

$\quad$ else if counter $<$ threshold, go to step (2).

else if $M1 \neq M2$, then

$\quad$ counter $= 0$

$\quad M1 = M2$

$\quad$ go to step (2)

where the counter measures the duration $\Delta\sigma$ where the number of dominant points does not change, i.e. $M1 = M2$, the typical threshold value is chosen from 6 to 8 in our computer simulations. Figures 6(a)-(d) show another more complex example of the Africa
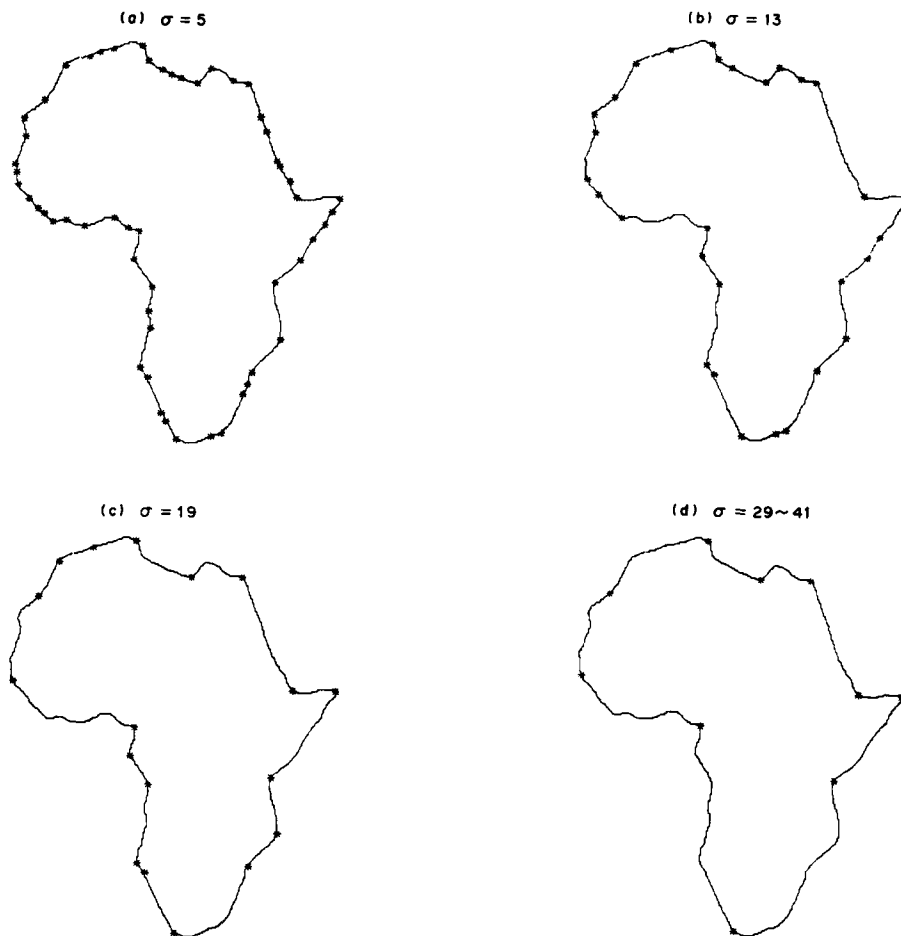

Fig. 6. The traced-back detected dominant points of the Africa curve at various scales, in which 10 cardinal curvature points in (d) are accurately detected and stable in the scale range from $\sigma = 29$ to 41.

curve in which 10 cardinal curvature points are accurately detected and stable in the range from $\sigma = 29$ to 41.

5. FAST ALGORITHM

Chen et al.[15] proposed a fast algorithm for computing the convolution of an image with LOG (Laplacian of Gaussian) mask. They used a well-known property that a LOG of width σ can be decomposed as a Gaussian mask and a LOG of width $\sigma_1 < \sigma$. Since the Gaussian kernel is neither time-limited nor band-limited, they must carefully establish effective, independent limits on the LOG and Gaussian mask required for proper performance.[16,17]

Here, we can derive a similar fast algorithm for computing the convolution of a curvature with the derivative of Gaussian kernels (see equation (13)). This fast technique is described in Section 5.1.

5.1. Fast convolution with the derivative of the Gaussian kernel

There is a special property about the Gaussian function as follows.

Theorem 1. The convolution of two Gaussian functions with width σ_1 and σ_2 will produce another Gaussian function with width $\sigma = (\sigma_1^2 + \sigma_2^2)^{1/2}$. This means that

$$g(t,\sigma) = g(t,\sigma_1) * g(t,\sigma_2), \quad \text{where } \sigma = \sqrt{(\sigma_1^2 + \sigma_2^2)}.$$
(14)

By Theorem 1, we derive the convolution of the curvature with the derivative of the Gaussian kernel as follows:

$$\dot{K}(t,\sigma) = \frac{d}{dt}\{K(t) * g(t,\sigma)\} = \frac{d}{dt}\{K(t) * g(t,\sigma_1) * g(t,\Delta\sigma)\}$$

$$= \left\{ K(t) * \frac{d}{dt}g(t,\sigma_1) \right\} * g(t,\Delta\sigma)$$

with $\sigma = \sqrt{(\sigma^2 - \sigma_1^2)}.$
(15)

Thus, the derivative of curvature at current scale σ can be computed efficiently by convolving the derivative of curvature at the preceding scale σ_1 with a Gaussian kernel with width $\Delta\sigma$. This can save a lot of derivative operations at each scale except at the smallest starting scale $\sigma = 3$.

Since the Gaussian smoothing function is not time-limited, if we take the finite Gaussian filter length as $[-3\sigma, 3\sigma]$, there must occur some truncation error. Also we sample the continuous Gaussian filter into a discrete Gaussian filter, since the Gaussian filter is not band-limited, this will cause some aliasing error during the sampling process. These aliasing and truncation errors will accumulate as this iterative convolution process goes on. In order to overcome this accumulation error effect, we adopt the sampling interval for the Gaussian kernel as 1 pixel for all scales, also the scale

interval $\Delta\sigma$ is switched from 2 to 4 pixels at large scales. This can reduce the aliasing error effectively. By equation (15), we summarize the fast algorithm as follows:

(1) By equation (13), start computing the derivative of curvature at the smallest scale $\sigma_1 = 3$, and denote it by $\dot{K}(t,\sigma_1)$.

(2) Set the scale interval to 2 pixels, thus the next scale is $\sigma_2 = \sigma_1 + 2$, then $\dot{K}(t,\sigma_2)$ can be computed simply by convolving $\dot{K}(t,\sigma_1)$ with the Gaussian kernel $g(t,\Delta\sigma)$, this means that

$$\dot{K}(t,\sigma_2) = \dot{K}(t,\sigma_1) * g(t,\Delta\sigma).$$
(16)

(3) Repeat step (2) until the scale $\sigma = 12$.

(4) Reset the scale interval $\Delta\sigma$ to 4 pixels to reduce aliasing error. Repeat step (2) until the dominant points have been found.

Combining this fast method and the tracing-back algorithm, we can find the relevant dominant points rapidly.

In order to see that the detected dominant points by this approach are very stable and remain unchanged with respect to translation, rotation and scaling, we perform several experiments the results of which are shown in Figs 7 and 8. The detected dominant points of the original curves at normal scale are shown in Figs 7(a) and 8(a), respectively. Figures 7(b) and 8(b) show the results when the two curves are reduced by half, i.e. scaling = 0.5. The detected curve shown in
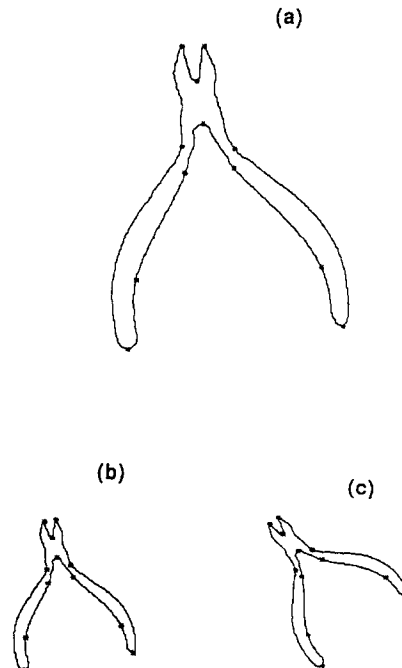
(a)

(b)

(c)

Fig. 7. The detected 12 dominant points by scale-space filtering are very stable and remained unchanged with respect to translation, rotation and scaling: (a) original curve; (b) scaling = 0.5; (c) translation = (10, 10) pixels, scaling = 0.5 and rotation = 30 deg.
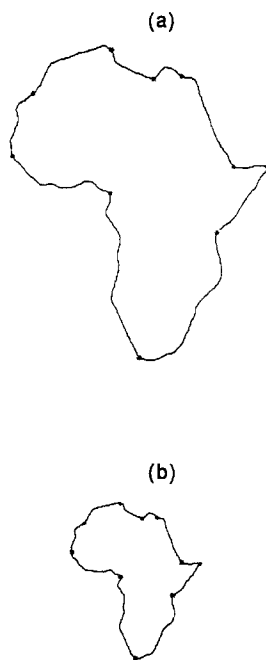
(a)

(b)

Fig. 8. Ten stable cardinal dominant points of the Africa curve are detected under (a) original scale and (b) half scale at scaling = 0.5.

Fig. 7(c) is obtained if the wire cutter curve is translated, scaled and rotated under translation = (10, 10) pixels, scaling = 0.5 and rotation = 30 deg. Notice that the detected dominant points by this approach remain very stable and do not change under translation, rotation and scaling. Hence the dominant points found by the scale-space algorithm satisfy Mokhtarian and Mackworth's criteria[12] discussed above for reliable recognition.

6. CONCLUSIONS

The information of a curve is concentrated and characterized at the dominant points with high curvature. In this paper, we have introduced the method of scale-space filtering with a Gaussian kernel to effectively detect these cardinal curvature points on the digital curve. These extracted features can uniquely specify a curve qualitatively.

The uniqueness of this method is that the dominant points found remain very stable, and will not change as the curve translates, rotates and scales. Thus we can recognize shapes using the dominant points under distortion of translation, rotation and scaling.

Another advantage of finding dominant points is

that it is very useful in the recognition of partial shapes.[18,19]

REFERENCES

1. S. Marshall, Review of shape coding techniques, Image Vision Comput. 7, 281–294 (1989).
2. A. Rosenfeld and E. Johnston, Angle detection on digital curves, IEEE Trans. Comput. 22, 875–878 (1973).
3. A. Rosenfeld and J. S. Weseka, An improved method of angle detection on digital curves, IEEE Trans. Comput. 24, 940–941 (1975).
4. H. Freeman and L. S. Davis, A corner-finding algorithm for chain coded curves, IEEE Trans. Comput. 26, 297–303 (1977).
5. P. V. Sankar and C. V. Sharma, A parallel procedure for the detection of dominant points on a digital curve, Comput. Graphics Image Process. 7, 403–412 (1978).
6. I. M. Anderson and J. C. Bezdek, Curvature and tangential deflection of discrete arcs: a theory based on the commutator of scatter matrix pairs and its application to vertex detection in planar shape data, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-6, 27–40 (1984).
7. C. H. Teh and R. T. Chin, On the detection of dominant points on digital curves, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-11, 859–872 (1989).
8. T. Pavlidis, Algorithms for shape analysis and waveforms, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-2, 301–312 (1980).
9. J. G. Dunham, Optimum uniform piecewise linear approximation of planar curve, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-8, 67–75 (1986).
10. N. Ansari and J. Delp, On detecting dominant points, Pattern Recognition 24, 441–451 (1991).
11. N. Ansari and K. W. Huang, Non-parametric dominant point detection, Pattern Recognition 24, 849–862 (1991).
12. F. Mokhtarian and A. Mackworth, Scale-based description and recognition of planar curves and two-dimensional shapes, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-8, 34–43 (1986).
13. J. Babaud, A. P. Witkin and M. Baudin, Uniqueness of the Gaussian kernel for scale-space filtering, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-8, 26–33 (1986).
14. A. L. Yuille and T. A. Poggio, Scaling theorems for zero crossing, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-8, 15–25 (1986).
15. J. S. Chen, A. Huertas and G. Medioni, Fast convolution with Laplacian-of-Gaussian masks, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-9, 584–590 (1987).
16. G. E. Sotax, Jr and K. L. Boyer, Comments on fast convolution with Laplacian-of-Gaussian masks, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-11, 1329–1332 (1989).
17. G. E. Sotax, Jr and K. L. Boyer, The Laplacian-of-Gaussian kernel: a formal analysis and design procedure for fast, accurate convolution and full-frame output, Comput. Vision Graphics Image Process. 48, 147–189 (1989).
18. L. Gupta and K. Malakapalli, Robust partial shape classification using invariant breakpoints and dynamic alignment, Pattern Recognition 23, 1103–1111 (1990).
19. H. C. Lin and M. D. Srinath, Partial shape classification using contour matching in distance transformation, IEEE Trans. Pattern Analysis Mach. Intell. PAMI-12, 1072–1079 (1990).

About the Author—Soo-Chang Pei was born in Soo-Auo, Taiwan, Republic of China, on 20 February 1949. He received the B.S. degree from National Taiwan University in 1970 and the M.S. and Ph.D. degrees from the University of California, Santa Barbara, in 1972 and 1975, respectively, all in electrical engineering. He

was an engineering officer in the Chinese Navy Shipyard at Peng Fu Island from 1970 to 1971 and a Research Assistant at the University of California, Santa Barbara, from 1971 to 1975. He was Professor and Chairman in the Department of Electrical Engineering at Tatung Institute of Technology from 1981 to 1983. Presently, he is the professor of Department of Electrical Engineering at National Taiwan University. His research interests include digital signal processing, digital picture processing, optical information processing, laser and holography. Dr Pei is a member of the IEEE, Eta Kappa Nu and the Optical Society of America.

About the Author—CHAO-NAN LIN was born in Taiwan, Republic of China, on 7 September 1966. He received the B.S. degree from National Tsing Hua University, Hsinchu, in 1989 and the M.S. degree from National Taiwan University, Taipei, in 1991, both in electrical engineering. Presently, he is a communication officer in the Chinese Army. His research interests include digital signal processing, image processing and computer graphics.