Pergamon

0031–3203(94)00086–7

FITTING DIGITAL CURVE USING CIRCULAR ARCS

SOO-CHANG PEI and JI-HWEI HORNG

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

Abstract—A smoothing procedure is proposed, where the Gaussian filter is used with an adaptive mechanism to suppress the noise effect and quantization error of a digital curve. Those points of the smoothed curve where curvature changes abruptly are detected as breakpoints. Circular arcs are suitably designed between breakpoints to fit the input curve. Experimental results indicate that our curve-fitting method provides good approximations of the input curves.

Curve-fitting   Circular arc   Gaussian filter   Breakpoint

1. INTRODUCTION

Many methods for constructing polygonal approximation of planar curve have been proposed,[1–7] which are based on the local maximum points of curvature detected by various techniques. They provide good approximations of the input curves subject to the constraint that the fundamental elements of approximation are straight line segments.

In some situations, we are not satisfied by these approximations even though a nearly optimal one under the constraint is given. For example, the boundary of a ball image can hardly be approximated by a polygon unless a large number of breakpoints are used. To improve the performance of approximation, the constraint should be suitably relaxed. That is, it should be permitted to choose some high-order curves as the fundamental elements.

In this paper, a subset of second-order curves, the circular arcs, are chosen as the fundamental elements. The reason for this choice is that it can provide satisfactory results and keep low computational complexity. Of course, the technique of breakpoint detection should be reconsidered under the newly defined constraint. Furthermore, unlike the polygonal approximation, where the polygon is constructed by connecting consecutive breakpoints directly, the design of circular arcs between breakpoints is a little more complex task. A technique is introduced in this paper to do that.

Owing to the noise effect and discrete nature of digital curve, spurious breakpoints along the curve are introduced. An adaptive smoothing procedure is proposed to preprocess the input curve.

The overall system block diagram is shown in Flow-chart 1.

In Section 2, the proposed adaptive smoothing procedure for noise reduction is described. The rationale and algorithm of breakpoint detection are given in Section 3. In Section 4, a technique of designing circular arcs between breakpoints is introduced. Experi-

mental results are presented in Section 5. Our method is compared with the Teh and Chin algorithm[3] in Section 6. Finally, we provide a conclusion in Section 7.

2. ADAPTIVE GAUSSIAN FILTERING

To reduce the noise effect and quantization error of a digital curve, Gaussian filter is usually used with a suitable standard deviation. The conventional Gaussian filter is defined by

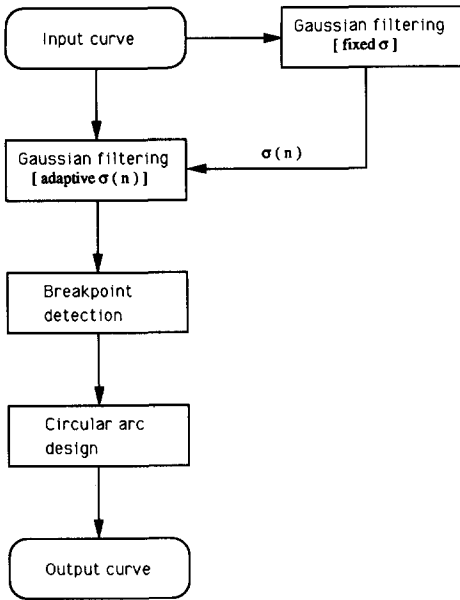$$G_\sigma(n) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{n^2}{2\sigma^2}\right),$$

where σ is the standard deviation.

There is a tradeoff in selecting the value of σ. A larger value of σ will remove small details of the curve, while a smaller value will permit false breakpoints. A solution of the σ value selecting problem is to smooth the curve adaptively. A larger value of σ is chosen to smooth the coarser portions of the curve and a smaller value is chosen to smooth the finer portions.

In general, if the Gaussian filter with a fixed value of σ is applied to smooth a curve, those points at coarser portions of the curve will result in less shift than those at finer portions. That is, the amount of shift resulting from the Gaussian filtering with a fixed value of σ gives an index to coarseness. The Gaussian filter can thus be applied to smooth the curve with the value of σ adapted according to this index.

The adaptive smoothing procedure we proposed is summarized as follows:

(1) Smoothing the curve using the Gaussian filter with a fixed value of σ.

(2) Calculating the shift function by subtracting the original curve from its smoothed version point by point.

(3) Defining a monotonously decreasing function. Selecting σ value for each point by mapping its corresponding shift value via the chosen function.

```
┌─────────────────┐        ┌─────────────────┐
│   Input curve   │───────▶│ Gaussian filtering │
└─────────────────┘        │   [ fixed σ ]    │
         │                 └─────────────────┘
         ▼                          │
┌─────────────────┐    σ(n)         │
│ Gaussian filtering │◀──────────────┘
│ [ adaptive σ(n) ] │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│   Breakpoint    │
│   detection     │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│  Circular arc   │
│     design      │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│  Output curve   │
└─────────────────┘
```

Flowchart 1. The overall system block diagram.

(4) Smoothing the original curve adaptively using the Gaussian filter with the σ value selected in the previous step for each point.

Remarks:

● The result of this procedure is insensitive to the fixed value of σ chosen in Step 1 within reasonable range.

● If the input curve is highly fluctuant, the shift function will behave in a similar manner. Which will result in an undesirable fluctuant output curve. To reduce this effect, we use a smoothed version of the shift function instead of itself to select σ value in Step 3.

3. BREAKPOINT DETECTION

The best advantage of choosing the circular arcs to construct approximation of the input curve is that the curvature along a circular arc is constant. Therefore, fitting a curve using circular arcs is equivalent to making a piecewise constant approximation of its curvature function. The method proposed in reference (8) is capable of doing this but very time consuming. In this paper, we propose a simple method which is similar in spirit to it but requires less computing time.

The breakpoint detection method we proposed is described as follows:

(1) Calculating the curvature function of the smoothed curve.
(2) Calculating the absolute value of derivative of the curvature function.
(3) Detecting local maxima of the resulting function.

The method proposed in reference (8) repeatedly convolves the signal with a very small averaging mask weighted by a measure of the signal continuity at each

point. A possible choice of the weight is

$$w(n) = \exp\left(-\frac{|S'(n)|^2}{2k^2} \right),$$

where $S'(n)$ is the derivative of the signal $S(n)$, and k is a parameter.

Notice that, the weight is a decreasing function of the derivative of the signal. Local maxima of the derivative of the signal will be sharpened by convolving with the mask. After convergence, these local maxima will be the breakpoints of the resulting piecewise constant signal. The result is exactly the same as our simple method.

4. CIRCULAR ARC DESIGN

Based on the breakpoints detected in the previous section, the approximation can now be constructed by designing a circular arc between each pair of adjacent breakpoints. The circular arc used to approximate the digital curve segment between a pair of adjacent breakpoints should, of course, have them as the two end points.[9] In order to uniquely specify the circular arc, we constrain that the angular difference between tangents of the two end points should be preserved. Our circular arc design technique is summarized as follows: (see Fig. 1)

(1) Input a pair of adjacent breakpoints B_1, B_2 and the angular difference between tangents of them θ.

(2) If the angular difference θ is smaller than a given threshold, use a straight line segment to approximate the input curve between the two breakpoints B_1, B_2 and go to (1); otherwise, go to the next step.

(3) Calculate the radius of the circular arc by the equation

$$r = \sqrt{\frac{l^2}{2(1 - \cos\theta)}},$$

where l is the length of the vector $\mathbf{B_1 B_2}$.

(4) If $-\pi < \theta < 0$ or $\pi < \theta < 2\pi$, the center of the circular arc O is at $B_1 + r[\cos(\phi - \psi), \sin(\phi - \psi)]$; otherwise, it is at $B_1 + r[\cos(\phi + \psi), \sin(\phi + \psi)]$. Where
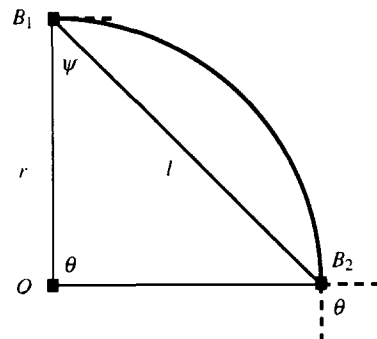
Fig. 1. Illustration of the circular arc design.

ϕ is the angle of the vector $\mathbf{B_1 B_2}$ and ψ is the angle $(\pi - \theta)/2$.

(5) The following point sequence is used to approximate the input curve between the two breakpoints B_1, B_2.

$$\hat{C}_{B_1,B_2}(n) = O + r[\cos(\varphi + n\theta/N), \sin(\varphi + n\theta/N)],$$
$$n = 1, \dots, N.$$

Where the value of N can be chosen arbitrarily and φ is the angle of the vector $\mathbf{OB_1}$.

(6) go to (1).

The radius and center of the circular arc used in the above technique are defined as the radius and center of the contour circle where the arc lies upon, respectively.

Notice that, if a polygon-like curve is given, the design of most arcs will be terminated at Step 2. Thus, the computational complexity remains the same as those polygonal approximation methods.

5. EXPERIMENTAL RESULTS

The boundary of a digital wrench image is plotted in Fig. 2. To reduce the quantization error, the Gaussian filter is applied to smooth the digital boundary. The result of convolving the boundary with the Gaussian filter is plotted in Fig. 3, where the σ value is set to 3. Small details of the boundary are removed by this process. Figure 4 shows the result of convolving with the adaptive Gaussian filter, where the σ value is adapted between 0.7 and 6. Small details are preserved and quantization error is largely reduced. The approximation designed by our method is plotted in Fig. 5, where breakpoints are marked by crosses. The monotonously decreasing function used in this experiment is plotted in Fig. 6, which is defined by the equation

$$\sigma = \sigma_{\min} + \frac{\sigma_{\max} - \sigma_{\min}}{1 + \exp\left(-8 + 16\dfrac{d - d_{\min}}{d_{\max} - d_{\min}}\right)},$$
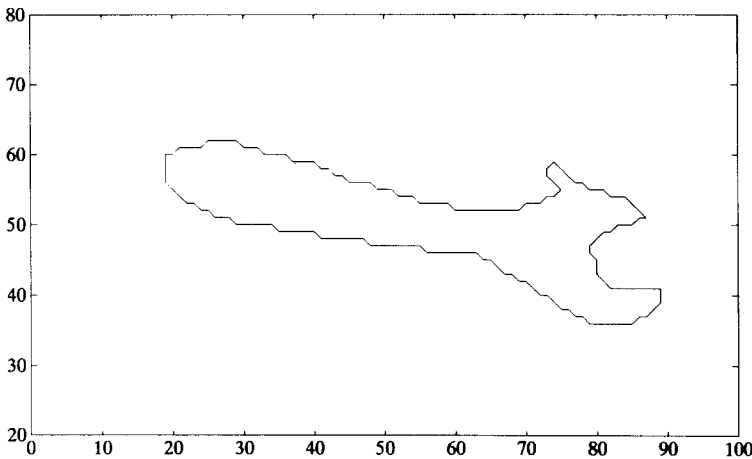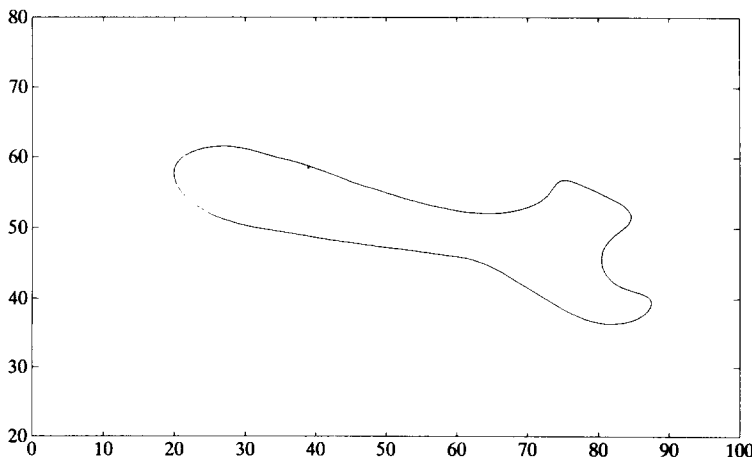

Fig. 2. A wrench boundary.

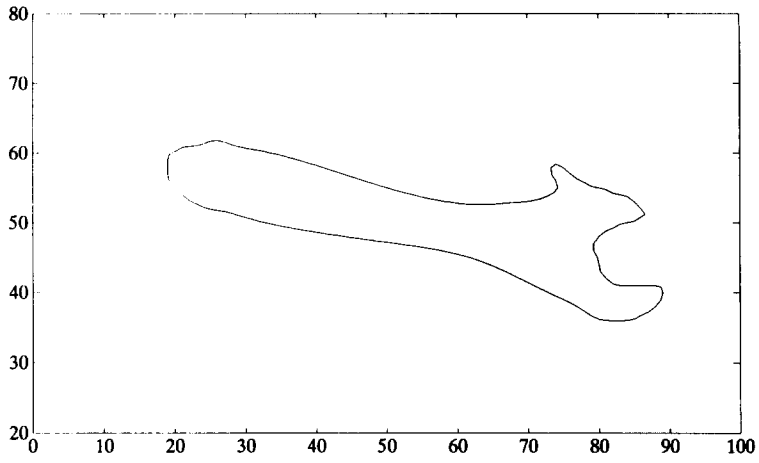Fig. 3. The Gaussian filtered wrench boundary.

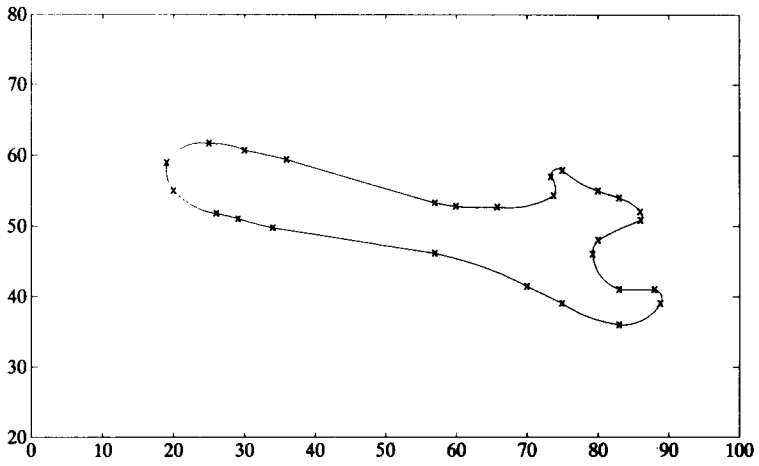Fig. 4. Adaptive Gaussian filtered wrench boundary.

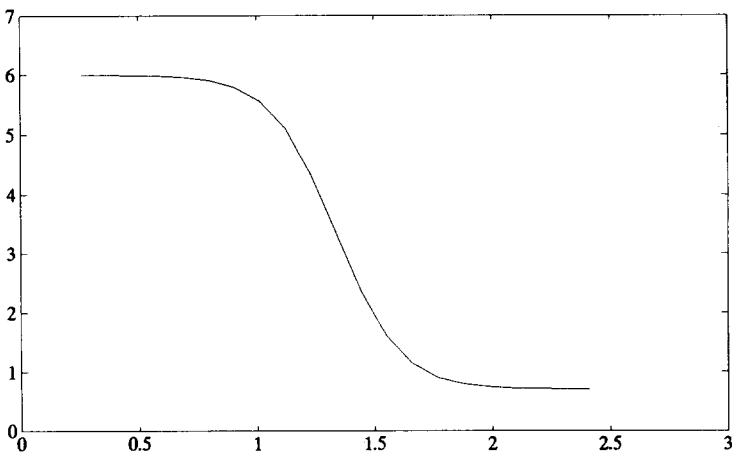Fig. 5. The breakpoints and designed curve.

Fig. 6. A monotonously decreasing function.

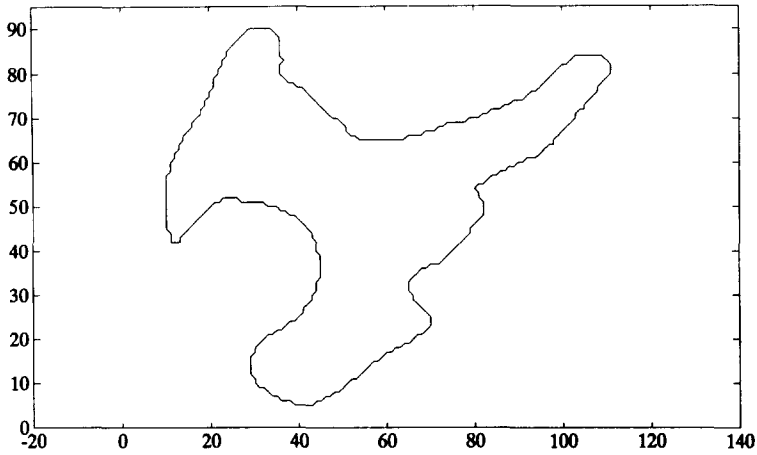Fig. 7. The boundary of an industrial part.

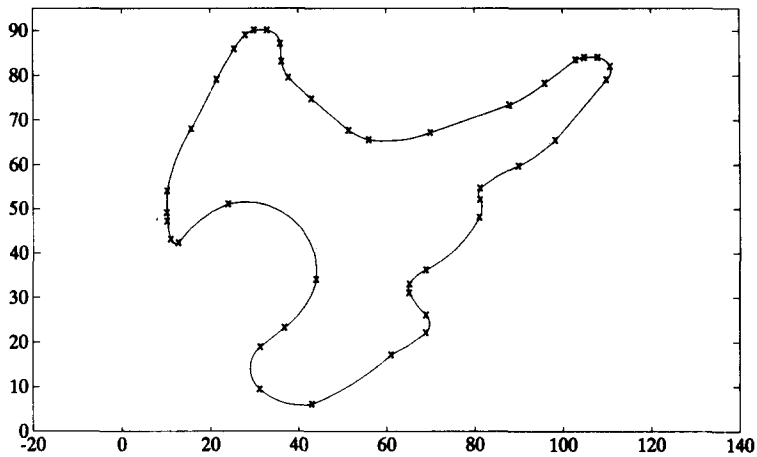Fig. 8. The breakpoints and designed curve.

where the values of σ_{min} and σ_{max} are set to 0.7 and 6, respectively, in this experiment, d_{min} and d_{max} are the global minimum and maximum values of the shift function.

Figure 7 shows the boundary of an industrial part, the experimental result of this input is plotted in Fig. 8.

6. COMPARISON AND DISCUSSION

In order to get into a deeper sense about the performance of the present method, we have applied our method to the four digital curve used in reference (3), namely, a chromosome-shaped curve (Fig. 9a), a leaf-shaped curve (Fig. 10a), a figure-8 curve (Fig. 11a), and a curve with four semicircles (Fig. 12a). The results of the Teh and Chin algorithm are shown in part (b) of Figs 9–12 by solid lines and the results of our method are shown in part (c). The k cosine measure is selected as the measure of significance, which is required in the Teh and Chin algorithm.

Some features of our method and the Teh and Chin algorithm are tabulated in Table 1, including the number of dominant points, the integral square error, and the maximum error. The error between a point of a digital curve C and the approximating curve $\hat{C}$ designed by our method is defined as $e_i = d_i - r_i$ where d_i is the distance between the point and the center of the contour circle that the approximating arc lies on, and r_i is the radius of the contour circle. The two error norms between C and its approximating curve $\hat{C}$ are defined by
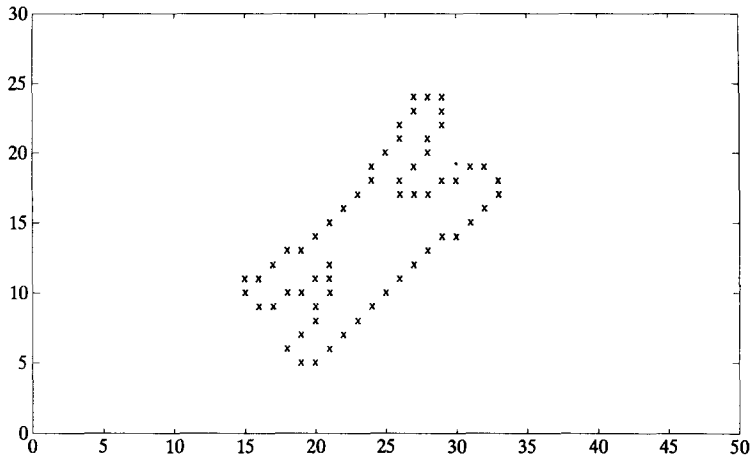
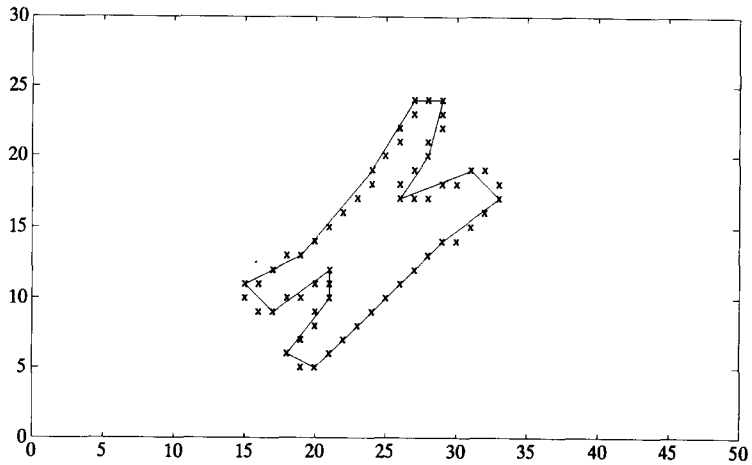(a) Integral square error

$$E_2 = \sum_{i=1}^{n} e_i^2,$$

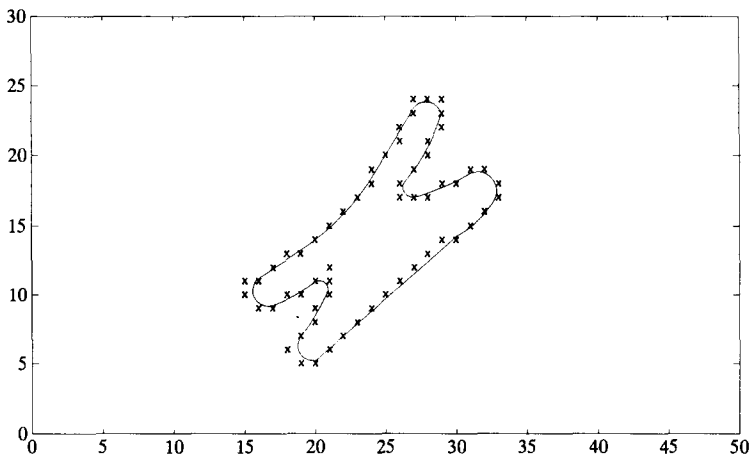(b) Maximum error

$$E_\infty = \max_{1 \le i \le n} e_i.$$

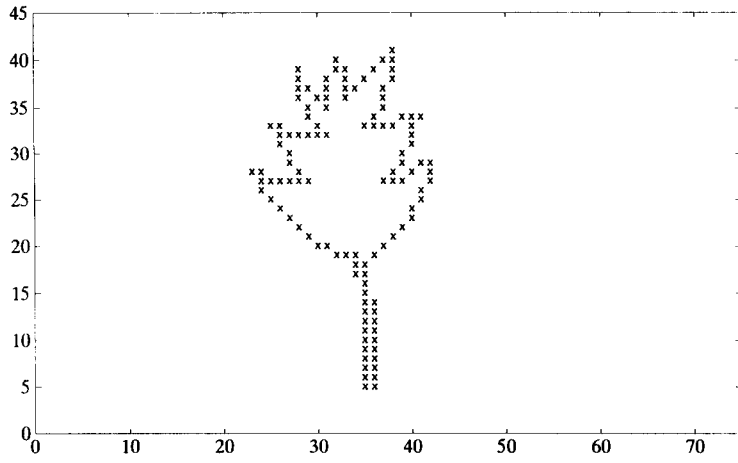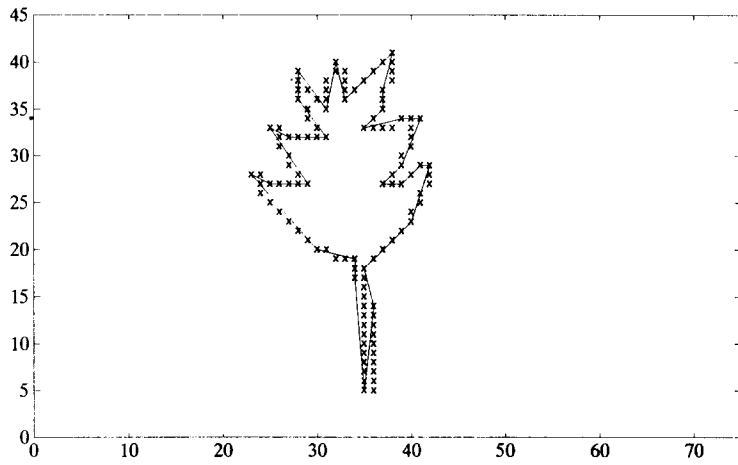As shown in the table, our method outperforms the existing methods in applying to those smooth curves,
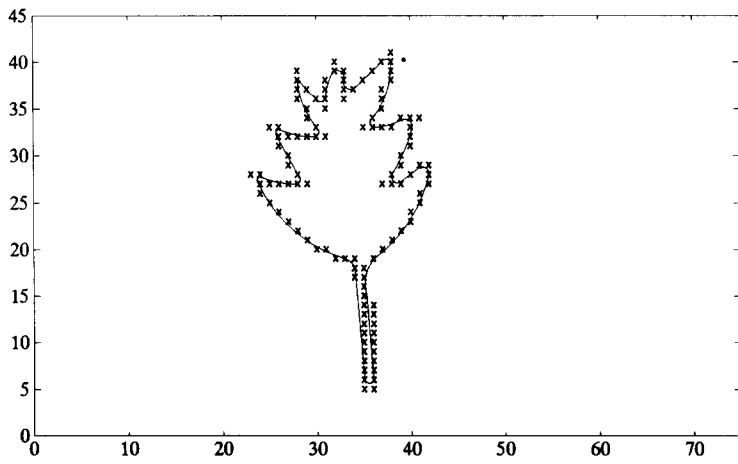
Fig. 9. (a) A chromosome-shaped curve. (b) The result of the Teh and Chin algorithm (k cosine). (c) The result of the present method.
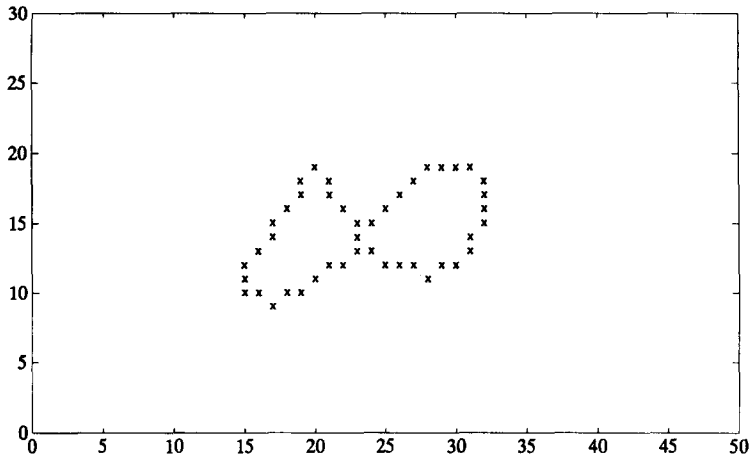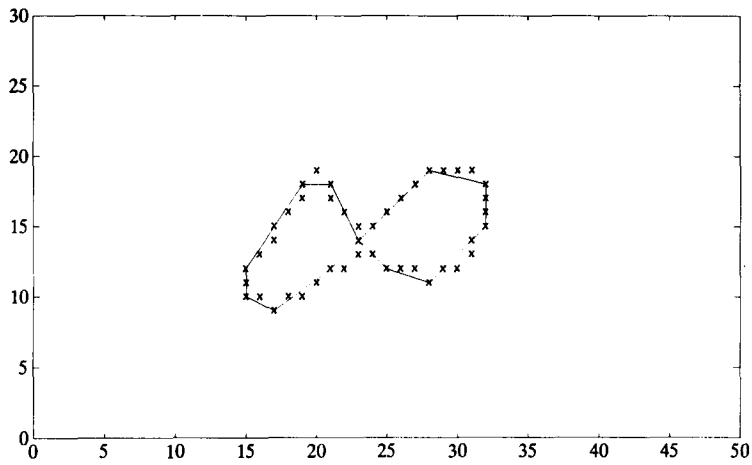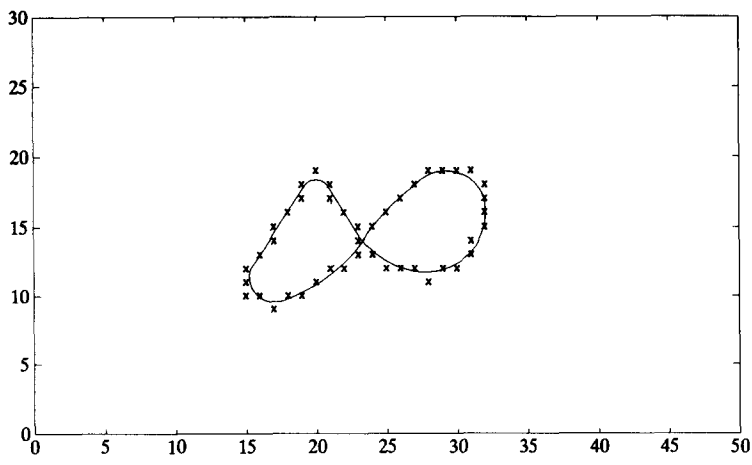
(a)

(b)

(c)

Fig. 10. (a) A leaf-shaped curve. (b) The result of the Teh and Chin algorithm (k cosine). (c) The result of the present method.
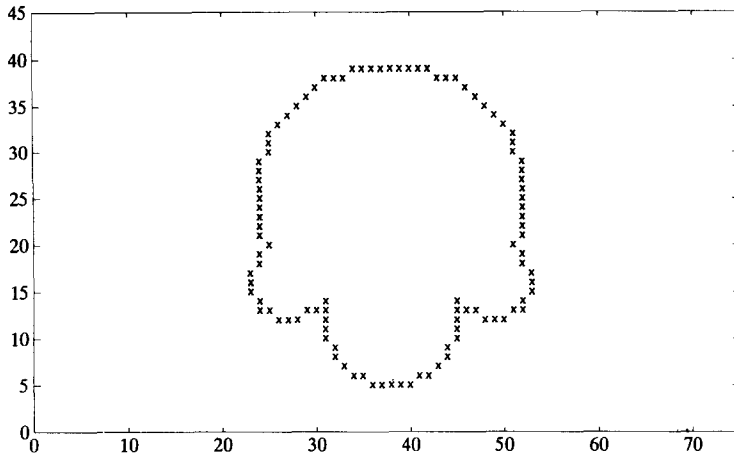
(a)

(b)

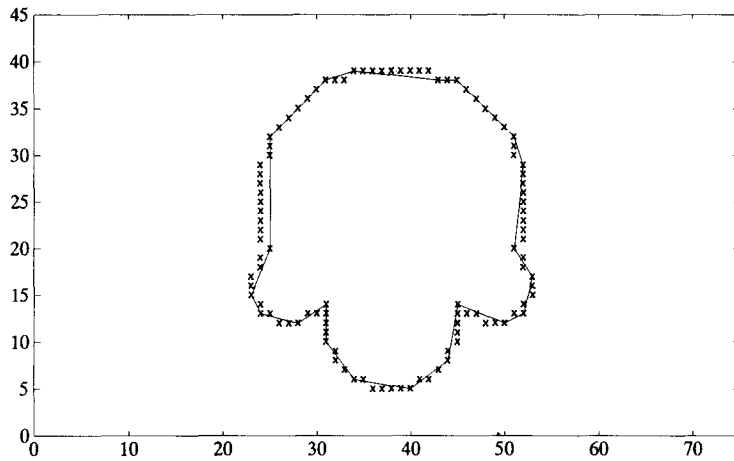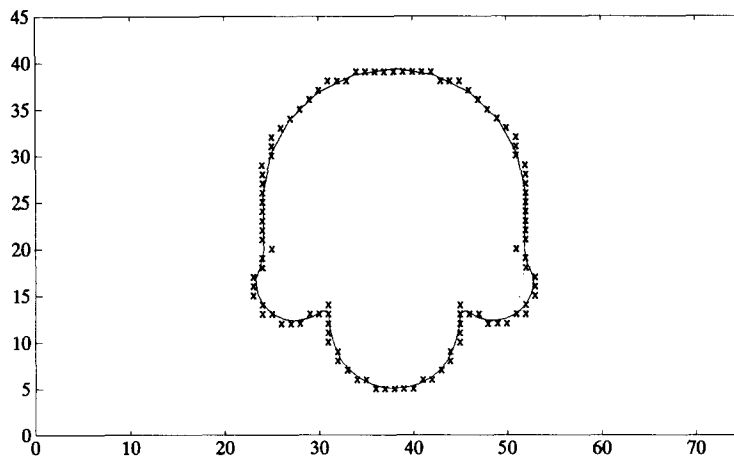

(c)

Fig. 11. (a) A figure-8 curve. (b) The result of the Teh and Chin algorithm (k cosine). (c) The result of the present method.

Fig. 12. (a) A curve with four semi-circles. (b) The result of the Teh and Chin algorithm (k cosine). (c) The result of the present method.

Table 1. Results of the present method and the Teh and Chin algorithm

Digital curve	Fig. 9(a)	Fig. 10(a)	Fig. 11(a)	Fig. 12(a)
Number of points	60	120	45	102
Results of the present method				
Number of dominant points	15	31	9	12
Integral square error	6.18	19.66	4.24	10.91
Maximum error	1.13	1.14	0.72	0.85
Results of the Teh and Chin algorithm				
Number of dominant points	15	29	13	22
Integral square error	7.20	14.96	5.93	20.61
Maximum error	0.74	0.99	1.00	1.00

the results shown in Figs 11 and 12 are two examples. For polygon-like curves, our approximations are slightly worse than those polygonal approximations. In fact, most of the integral square error are contributed by sharp corner points and, of course, the maximum error results from one of them. Except for these sharp corners, our approximation fits the curve well.

7. CONCLUSION

We have proposed a method for the design of circular arcs to approximate digital curves. Unlike the dominant points for polygonal approximation, a completely different set of dominant points are detected using a new proposed method to satisfy the new demand. In addition, we have proposed an adaptive smoothing procedure to remove noise while preserve details of the input curve. Our method has been tested on a number of digital curves and satisfactory results have been obtained.

REFERENCES

1. N. Ansari and E. J. Delp, On detecting dominant points, *Pattern Recognition* 24(5), 441–451 (1991).

2. N. Ansari and K.-W. Huang, Non-parametric dominant point detection, *Pattern Recognition* 24(9), 849–862, (1991).
3. C.-H. Teh and R. T. Chin, On the detection of dominant points on digital curves, *IEEE Trans. Pattern Anal. Machine Intell.* 11(8), 859–872 (August 1989).
4. A. Rattarangsi and R. T. Chin, Scale-based detection of corners of planar curves, *IEEE Trans. Pattern Anal. Machine Intell.* 14(4), 430–449 (April 1992).
5. B. K. Ray and K. S. Ray, Detection of significant points and polygonal approximation of digitized curves, *Pattern Recognition Letters* 13, 443–452 (1992).
6. B. K. Ray and K. S. Ray, An algorithm for detection of dominant points and polygonal approximation of digitized curves, *Pattern Recognition Letters* 13, 849–856 (1992).
7. W.-Y. Wu and M. J. Wang, Detecting the dominant points by the curvature-based polygonal approximation, *CVGIP: Graphical Models and Image Processing* 55(2), 79–88 (March 1993).
8. P. Saint-Marc, J.-S. Chen and G. Medioni, Adaptive smoothing: a general tool for early vision, *IEEE Trans. Pattern Anal. Machine Intell.* 13(6), 514–529 (June 1991).
9. C. A. Cabrelli and U. M. Molter, Automatic representation of binary images, *IEEE Trans. Pattern Anal. Machine Intell.* 12(12), 1190–1196 (December 1990).

About the Author—SOO-CHANG PEI was born in Soo-Auo, Taiwan, China, on 20 February 1949. He received the B.S. degree from the National Taiwan University in 1970 and M.S. and Ph.D. degrees from the University of California, Santa Barbara, in 1972 and 1975, respectively, all in electrical engineering. He was a 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 the Department of Electrical Engineering at the 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—JI-HWEI HORNG was born in Kinmen, Republic of China, on 13 April 1968. He received the B.S. degree from the Tamkang University in 1990 and M.S. degree from the National Taiwan University in 1992. Now, he is at the National Taiwan University studying for a Ph.D. degree. His research interests include image analysis and pattern recognition.