Multiscale Curvature-Based Shape Representation Using B-Spline Wavelets

Yu-Ping Wang, S. L. Lee, and Kazuo Toraichi, *Member, IEEE*

***Abstract*— This paper presents a new multiscale curvature-based shape representation technique with application to curve data compression using B-spline wavelets. The evolution of the curve is implemented in the B-spline scale-space, which enjoys a number of advantages over the classical Gaussian scale-space, for instance, the availability of fast algorithms. The B-spline wavelet transforms are used to efficiently estimate the multiscale curvature functions. Based on the curvature scale-space image, we introduce a coarse-to-fine matching algorithm which automatically detects the dominant points and uses them as knots for curve interpolation.**

***Index Terms*— B-spline, curvature function, curve evolution, curve fitting, diffusion equation.**

I. Introduction

SHAPE representation or analysis is an important issue in many applications of machine vision, such as character recognition, automatic visual inspection, fingerprint identification, and contour matching for medical imaging. More specifically, the contour of a shape contains a substantial amount of information about the original shape [7], [12]. One of the most important parameters characterizing contours is the curvature at each of their constituent points, particularly at places characterized by inflection or curvature modulus maxima, which are usually called *visual primitives* [7]. The curvature function of the curve is invariant under affine geometric transformation, i.e., scaling, rotation and translation. There are psychophysical results showing that curvature plays a fundamental role in human shape perception and representation [12].

Since the visual primitives, usually characterized by high curvature of the curve, occur at different density, it is necessary that these features be described in a multiscale sense. Thus, the curvature scale-space image was introduced by Mokhtarian and Mackworth [8] as a tool for representing planar curves. This representation is computed by convolving a path-based parametric representation of the curve with a Gaussian function with varying standard deviation. Most classical multiscale representations of planar curves (see, for example, [7] and [8]) are based on the Gaussian function because the evolution of curve in such a scale-space satisfies the causality criteria, i.e., no new feature points are created when the scale increases [6].

This paper presents a curvature-based multiscale shape representation using B-spline wavelets. In a broad sense, we attempt to develop a B-spline based multiresolution shape representation algorithm as an alternative to the Gaussian-based multiscale description. Recently an affine invariant multiscale representation of planar shapes using B-splines was proposed in [9] where the scale was selected as the order of the B-splines. In this paper we adopt a different approach, we use the dilated B-splines of fixed order for scale-space filtering. This allows us to use the efficient subdivision algorithm.

The remainder of the paper is organized as follows. Sections II and III review and present fast filterbank implementation of scale-space filtering using B-spline technique. Section IV studies the multiscale curvature image of a curve, in particular the property of fractal curve in the B-spline scale-space. Section V introduces an efficient algorithm for curve data compression application. Some experiments are performed to illustrate the performance of the B-spline wavelets. Finally some conclusions and discussions are presented.

II. Curve Parameterization and Diffusion in the B-Spline Scale-Space

A. B-Spline Kernels

We briefly describe the B-spline theory used in the paper. More details can be found in [1], [2], [4], and [5].

The continuous B-spline of order n is denoted by $\beta^n(x)$, which can be generated by repeated $n+1$ convolutions of the B-spline of order 0, as follows:

$$\beta^n(x) = \beta^0 * \beta^{n-1}(x) = \overbrace{\beta^0 * \beta^0 * \cdots * \beta^0}^{n+1}(x) \qquad (1)$$

where $\beta^0(x)$ is the zeroth-order B-spline, i.e., an impulse with support $[\frac{-1}{2}, \frac{1}{2}]$.

The discrete B-spline of order n at scale level m is defined as

$$B_m^n = \overbrace{B_m^0 * B_m^0 * \cdots * B_m^0}^{n+1} \qquad (2)$$

where $B_m^0 = 1/m[1, 1, \cdots, 1]$ is a normalized sampled pulse of width m.

Manuscript received November 5, 1997; revised February 16, 1999. This work was supported by the Wavelets Strategic Research Program funded by the National Science and Technology Board and Ministry of Education under Grant RP960 601/A. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kannan Ramchandran.

Y.-P. Wang is with the School of Medicine, Washington University, St. Louis, MO 63110 USA (e-mail: wyp@cauchy.wustl.edu).

S. L. Lee is with with the Center for Wavelets, Approximation, and Information Processing, National University of Singapore, Republic of Singapore 119260 (e-mail: wyp@wavelets.math.nus.edu.sg).

K. Toraichi is with the Institute of Information Science and Electronics, University of Tsukuba, Ibaraki 305, Japan.

Publisher Item Identifier S 1057-7149(99)08749-7.

1057–7149/99$10.00 © 1999 IEEE

The discrete sampled B-spline of order n at resolution m is denoted as $b_m^n(k)$, which is obtained by directly sampling the nth-order continuous B-spline at the scale m, i.e.,

$$b_m^n(k) = \frac{1}{m}\beta^n\left(\frac{k}{m}\right) \quad \forall\, k \in \mathbb{Z}. \tag{3}$$

When $m = 1$, the discrete sampled B-spline of order n is just denoted as $b^n(k) = \beta^n(k)$. The inverse operator $(b^n)^{-1}$ is defined as

$$(b^n)^{-1} * b^n(k) = \delta(k). \tag{4}$$

One of the notable properties of B-splines is that they are m-refinable

$$\frac{1}{m}\beta^n\left(\frac{x}{m}\right) = \sum_{k=-\infty}^{+\infty} B_m^n(k)\beta^n(x-k) \tag{5}$$

the property that gives efficient multilevel algorithms or subdivision algorithms (see Section III).

A. Curve Parameterization and Diffusion Using B-Spline Bases

Suppose the input is a discrete set of points $\mathbf{V}_i = [x_i, y_i]^T$, $(1 \le i \le N)$. First we parameterize the curve using a B-spline basis

$$\mathbf{C}_0(u) = \sum_k \mathbf{c}_0(k)\beta^l(u-k) \tag{6}$$

where $\mathbf{c}_0(u) = [x(u, 0), y(u, 0)]^T$. The approximation coefficient vector $\mathbf{c}_0(k)$ is given by

$$\mathbf{c}_0(k) = (b^l)^{-1} * \mathbf{V}(k) \tag{7}$$

where $(b^l)^{-1}$ can be computed efficiently [3]. If we take B-spline of order $l = 0$, the approximation coefficients $\mathbf{c}_0(k)$ are just the original sampling points of the curve, as often used in the Gaussian smoothing.

Next, the evolution of the curve at different instances or resolution levels s is achieved by convolving the curve with a dilated B-spline kernel,

$$\mathbf{C}(u, s) = \mathbf{C}_0 * \beta_s^n(u) \tag{8}$$

where $\beta_s^n(u) = (1/s)\beta^n(u/s)$ denotes the nth-order B-spline at resolution s.

We call the above representation *the B-spline scale-space,* which is quite similar to the Gaussian scale-space where the evolution of curve is driven by heat diffusion equation. Due to the central limit theorem, B-splines approximate the Gaussian when the order goes to infinity. But using B-splines we are able to obtain efficient subdivision algorithms for curve generation and compute the geometric descriptors of the curve easily.

III. Curve Subdivision at Continuous Scales and Dyadic Scales

Because of the m-refinable property (5) of B-splines, we are able to obtain the evolving curve at a certain instant s in an efficient way. In [5], a fast and parallel algorithm for the implementation of the smoothing operation (8) is derived. Here we just present the formula and one can refer to [5] for more details. The evolving curve at the rational scale m_1/m_2 can be computed using the filterbank algorithm

$$\mathbf{C}\left(u, \frac{m_1}{m_2}\right) = m_2\left(b^{l+n+1} * B_{m_2}^l * B_{m_1}^n * \mathbf{c}_{0_{\uparrow m_2}}\right)_{\downarrow m_2}(u) \tag{9}$$

where $\mathbf{c}_{0_{\uparrow m_2}}$ represents the upsampling operation applied to $\mathbf{c}_0(k)$ given in (7). The operation $\downarrow m_2$ denotes the downsampling operation by a factor of m_2.

Equation (9) can be realized by a moving average technique due to the definition of $B_{m_2}^l$ in (2), as detailed in [5]. The complexity of the algorithm is $\mathcal{O}(N)$, independent of the scale level, where N is the number of original curve sampling points.

If we restrict the scale level to the dyadic, $m_2/m_1 = 2^m$, we can obtain the recursive smoothing between the levels 2^m and 2^{m-1}. At level 2^m, the formula (9) simplifies to

$$\mathbf{C}(u, 2^m) = (b^{l+n+1} * B_{2^m}^n * \mathbf{c}_0)(u) = B_{2^m}^n * \breve{\mathbf{C}}_0(u) \tag{10}$$

where $\breve{\mathbf{C}}_0 = b^{l+n+1} * \mathbf{c}_0$ is regarded as the original approximation of the curve. It is easy to establish the relationship of the evolving curve between two successive dyadic levels as follows [4], [5]:

$$\mathbf{C}(u, 2^m) = \frac{1}{2^{n+1}}\sum_{j=0}^{n+1}\binom{n+1}{m}\mathbf{C}(u - j2^{m-1}, 2^{m-1}). \tag{11}$$

Hence, the evolving curve at dyadic levels can be implemented with only additions due to the binomial kernels in the convolution [4], [5].

If the curve is represented using B-spline bases, it is easy to extract the geometric information which are usually characterized by the differential operators. If we define the B-spline wavelet transforms $W_1\mathbf{C}(u, s)$ and $W_2\mathbf{C}(u, s)$ of the curve $\mathbf{C}(u, s)$ to be the convolution of $\mathbf{C}_0(u)$ with the first and second derivatives of the nth-order B-spline, respectively, as in (8), then [4]

$$W_1\mathbf{C}(u, s) = \Delta_u\mathbf{C}(u, s) \quad \text{and} \quad W_2\mathbf{C}(u, s) = \Delta_u^2\mathbf{C}(u, s) \tag{12}$$

where $\Delta_u f(u, s) = f(u+\frac{1}{2}, s) - f(u-\frac{1}{2}, s)$. In other words, they are just the first and second order differences of the smoothing curve (9).

IV. Geometric Information Extraction in the B-Spline Scale-Space

A. Computation of the Curvature Function Using the B-Spline Wavelets

The most important and significant description of a planar curve is its curvature function. The curvature function of the

curve $\mathbf{C}_0(u) = [x(u), y(u)]^T$ is defined as

$$\kappa(u) = \frac{\dot{x}(u)\ddot{y}(u) - \dot{y}(u)\ddot{x}(u)}{((\dot{x}(u))^2 + (\dot{y}(u))^2)^{3/2}} \tag{13}$$

where $\dot{x}(u)$, $\ddot{x}(u)$ represent the first and second derivative of $x(u)$, and similarly for $\dot{y}(u)$, $\ddot{y}(u)$. If u is the normalized arc length parameter, then (13) can be rewritten as

$$\kappa(u) = \dot{x}(u)\ddot{y}(u) - \dot{y}(u)\ddot{x}(u). \tag{14}$$

As the curvature is sensitive to noise, the localization of the curvature points is obtained by integrating multiple scales. For the evolving curve according to (8), it is easy to show that the curvature function at the scale s can be estimated as follows:

$$\kappa(u, s) = \frac{W_1x(u,s)W_2y(u,s) - W_1y(u,s)W_2x(u,s)}{(W_1x(u,s)^2 + W_1y(u,s)^2)^{3/2}} \tag{15}$$

where the two wavelet transform vectors $[W_1x(u, s), W_1y(u, s)]^T$ and $[W_2x(u, s), W_2y(u, s)]^T$ can be computed efficiently using the fast subdivision scheme discussed in Section III.

B. Curvature Scale-Space Image of the Curve and Its Properties

For different scale level, the function defined implicitly by

$$\kappa(u, s) = 0 \tag{16}$$

is called the curvature scale-space image of a curve by Mokhtarian and Mackworth [8], or the curvature primal sketch by Asada and Brady [7]. Many tasks in computer vision such as the curve compression, automatic object recognition can be performed by identifying the significant points in the shape contour, which are generally points with high absolute curvature values.

Using a little more general mathematical analysis, it can be shown that the curvature edge changes in B-spline scale-space have the same behavior as that of the Gaussian scale-space. Furthermore, in a discrete sense the extrema or zero-crossing number of the curve coordinates does not increase as the scale varies from low to high level. In other words, the scaling or causality property for Gaussian kernel also holds in the case of B-splines. For detailed discussion, refer to [5].

C. Analysis of Fractal Curve in the B-Spline Scale-Space

Fractal curve or surface can be found widely in nature, for example, in natural landscape, in the turbulent flow, and even in the fluctuations of the stock market. In this section we analyze the behavior of such typical shape or texture in the B-spline scale-space, which also shows why multiscale technique is necessary for fractal geometry description. All fractal curves encountered in physical models have certain properties: Each segment is statistically similar to all others; they are statistically invariant over wide transformations of scale. The popular model is the fractional Brownian motion [10], which is self-similar in a stochastic sense with the power spectrum proportional to $1/|\omega|^{2H+1}$, for some exponent $1 > H > 0$, i.e.,

$$P(\omega) = \lambda\frac{1}{|\omega|^{2H+1}}, \qquad 1 > H > 0 \tag{17}$$

where λ is an absolute constant. The Hurst exponent H is used to describe the smoothness or roughness of the curve and the fractal (Hausdorff) dimension is given by $D = 2 - H$.

The behavior of a stochastic process is usually characterized qualitatively by its extrema value distribution [11]. The density of local extrema D for a stationary Gaussian process can be obtained according to the second- and fourth-order derivatives of the autocorrelation function R or equivalently according to the second and fourth-order moments of its spectrum P:

$$D = \frac{1}{2\pi}\sqrt{-\frac{R^{(4)}(0)}{R^{(2)}(0)}} \tag{18}$$

and the derivatives of the autocorrelation function at zero can be expressed in terms of the power spectrum by

$$\begin{aligned} -R^{(2)}(0) &= \int_{-\infty}^{\infty}\omega^2 P(\omega)d\omega, \\ R^{(4)}(0) &= \int_{-\infty}^{\infty}\omega^4 P(\omega)d\omega. \end{aligned} \tag{19}$$

If the fractal curve evolves as in (8), it is easy to verify that the power spectrum P_s of the curve at scale s is related to the power spectrum P_0 of the curve at scale zero by

$$P_s(\omega) = |\hat{\beta}^n(s\omega)|^2 P_0(\omega). \tag{20}$$

Since the B-spline of order n is asymptotically equal to Gaussian function by the following relation [5]:

$$\beta^n(x) \approx \sqrt{\frac{6}{\pi(n+1)}}\exp\left(-\frac{6x^2}{n+1}\right) \tag{21}$$

their Fourier transforms are related by

$$\hat{\beta}^n(\omega) \approx \frac{1}{2}\exp\left(-\frac{n+1}{24}\omega^2\right). \tag{22}$$

From this relation and

$$\int_{-\infty}^{\infty} t^m e^{-\alpha t^2}dt = \frac{\Gamma(\frac{m+1}{2})}{\alpha^{(m+1)/2}} \tag{23}$$

we can easily estimate the moments of the power spectrum (19). For fractional Brownian motion, the second and fourth moments of its power spectrum at scale s are

$$\begin{aligned} -R^{(4)}(0) &\approx \lambda\int_{-\infty}^{\infty}\omega^4\frac{1}{4}\exp\left(-\frac{n+1}{12}s^2\omega^2\right)\omega^{-(2H+1)}d\omega \\ &= \frac{\lambda}{4}\frac{\Gamma(1-H)}{\left(\frac{n+1}{12}s^2\right)^{1-H}}, \\ R^{(4)}(0) &\approx \lambda\int_{-\infty}^{\infty}\omega^4\frac{1}{4}\exp\left(-\frac{n+1}{12}s^2\omega^2\right)\omega^{-(2H+1)}d\omega \\ &= \frac{\lambda}{4}\frac{\Gamma(2-H)}{\left(\frac{n+1}{12}s^2\right)^{2-H}}. \end{aligned} \tag{24}$$

Therefore, the density of local extrema of the fractional Brownian noise at scale s is

$$D_s = \frac{\sqrt{3}}{2\pi}\frac{1}{s}\frac{1}{\sqrt{n+1}}\sqrt{1-H}, \quad 0 < H < 1. \tag{25}$$

From (25), we can draw the following conclusion about the influence of scale and order of B-spline on the evolution of fractal curve. If a fractal curve diffused in the B-spline scale-space, the density of local extrema decreases with scale as s^{-1} or decreases with the order of B-spline as $n^{-(1/2)}$. If the scale is fixed, for larger value of H which corresponds to smoother curve there is lower density of extrema, and vice versa. This theoretical analysis is consistent with the practical observations.

Fig. 2 shows the evolution of a Von Koch snowflake curve in the B-spline scale-space. With the increase of scale, the curve becomes more smooth and has less density of extrema.

V. An Algorithm for the Curve Data Compression Application

The curvature scale-space image of a curve is equivalently characterized by its curvature extrema at different scales, which provide significant information about the contour of the object. They can be used as a compact and effective representation for shape analysis. In this section, we introduce a coarse-to-fine matching algorithm to detect the dominant points of the curve and apply them to represent a curve.

A. An Algorithm for Curve Data Compression

We present the detailed description of our algorithm in four steps.

1) Obtaining the Curvature Image of B-Spline Scale-Space: At first, we use the fast algorithm introduced in Section III to obtain the evolving curve $\mathbf{C}(u_i, s_k)$, $1 \le i \le N$, $s_{\min} \le s_k \le s_{\max}$ at different scales, where N is the number of original curve sampling points and $s_{\min}$, $s_{\max}$ are the minimal and maximal scales. Then the curvature estimation at different scales are obtained using (15). At this step, one has the freedom to choose the discrete scale parameters as either the continuous scales or dyadic scales as in Section III. Usually the choice of the dense scale resolutions gives better results than the usual dyadic scales.

2) Coarse to Fine Matching of the Dominant Points: The locations of peaks in the curvature scale-space image correspond to the corner points of the curve. It turns out that by looking at the movement of the peaks over several scales we can localize different types of corner points. At finer scales we can obtain a good localization of corner points. But due to the influence of noise or high frequency details, the spurious points are detected. At coarser scale, we will obtain the overall picture of the curve, but the locations of corner points are not accurate due to the smoothing procedure. Therefore, we have to combine the multiscale information to trace the trajectories of these dominant points and correct their locations. See Fig. 1 for better illustration. After extracting the local peaks across several scales, we then chain these fingerprints using the following coarse-to-fine strategy.

1) The local maxima points are classified into local positive maxima and local negative minima points. The polarization does not change with the scale. From the coarse to fine tracking, these two classes of points are matched across scales, respectively.

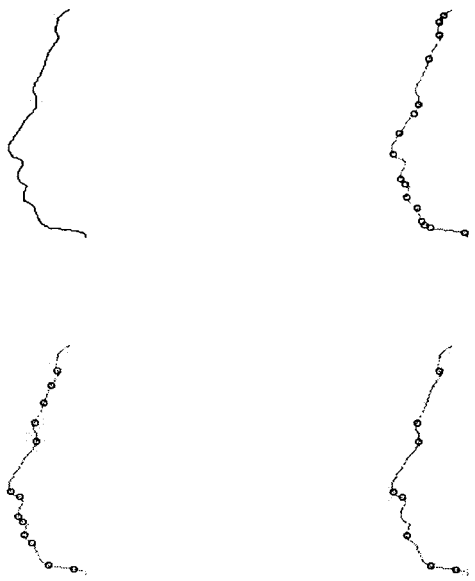

Fig. 1. Multiscale corner detection of a face contour (upper left). More corner points are detected at finer scales (upper right). In order to suppress noise, coarser scales should be selected. But the locations of these points are "migrated" (lower left). The correct detector should integrate multiple scales together (lower right).

2) A local maximum point P^{i+1} at level $i+1$ is said to correspond to another local maximum point P^i at the finer scale i, if

$$d(p_k^{i+1}, p_l^i) \le \varepsilon, \qquad p_k^{i+1} \in D_l^i \tag{26}$$

where ε is the predetermined threshold and D_l^i denotes the neighborhood of the points p_l^i. Because of the little "migration" of the extrema locations, the D_l^i is chosen to be monotonically decreasing from the coarse scales to fine scales so that in finer scales the search is performed in a small neighborhood and vice versa. The distance $d(p_k^{i+1}, p_l^i)$ can be measured in the sense of either $l^2(\mathbb{Z})$ or $l^1(\mathbb{Z})$. If the two points in the neighboring scales match each other, then the location of P_k^{i+1} is corrected to be the same as that of P_l^i. This matching procedure is implemented from coarse scale $s_{\max}$ to fine scale $s_{\min}$ and repeated several times until the locations of the corner points are found.

3) Postprocessing of the Corner Candidates: In the second phase, the candidate corners are detected and those with significant changes in direction are identified as corners. At finer scales, there may be more corner candidates whose locations are very close and are not necessary for restoration. In order to alleviate such a problem, we introduce the following geometric criteria to reduce the corner candidates. Our idea is, if the three consecutive corner candidates P_{k-1}^i, P_k^i, P_{i+1}^i are on the same straight line we consider that P_k^i as a phantom corner and remove it. In the implementation we use the following formula to judge P_k^i:

$$\overline{P_{k-1}^i P_k^i} + \overline{P_k^i P_{k+1}^i} \ge \lambda \overline{P_{k-1}^i P_{k+1}^i} \tag{27}$$

where $\overline{P_{k-1}^i P_k^i}$ denotes the length of the line segment between the two points P_{k-1}^i and P_k^i. The parameter λ is usually taken as 0.9–1.

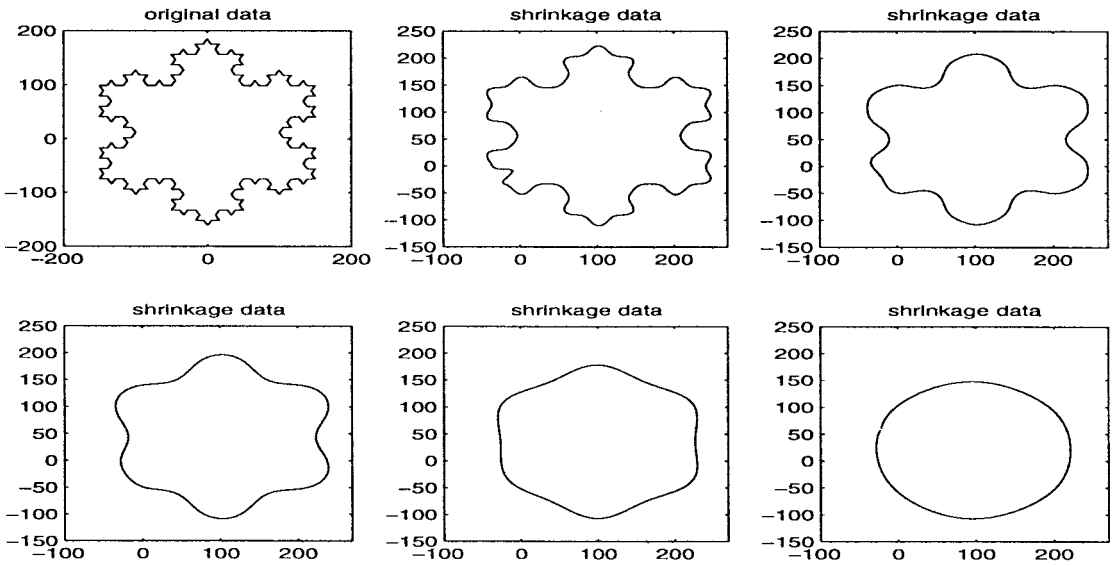

Fig. 2. Shrinkage of the Koch snowflake in the B-spline scale-space. As the scale level gets larger, the Koch snowflake becomes more and more smooth and shrinks to a circle. The evolution of a curve has a similar behavior as that of Gaussian scale-space except that the shrinking speed is different.

One may choose other criteria to refine the detection of corner points. For example, if the distance of two corner candidates are within a predetermined threshold, then one of them will be deleted. In this way, the data points to be stored can be greatly reduced.

4) Curve Reconstruction Using Interpolation or Fitting Method: Once the above procedure is completed, we select the dominant points at a certain scale and store their positions. Thus a compressed form of curve is obtained. In order to recover the original curve, we use these corner points as control knots and selecting some interpolation methods such the spline and polynomial interpolation to get the coordinates of the curve at other locations. One may obtain a better compression ratio at large scale since there are few corner points, but the recovered curve may also be poor. On the contrary, good reconstruction quality can be obtained with the cost of lower compression ratio at small scale. So a proper scale should be chosen depending on the needs at hand.

B. Experimental Examples

In this section we present some examples to show the procedure in implementing the above algorithms. Fig. 1 shows an example for corner detection of a face contour. As can be seen, at finer scales more points are detected and vice versa. So the multiscale information should be integrated to find the correct points.

Fig. 3 is a typical terrain map containing different features at different locations. We first obtain the curvature image at different scales. According to the algorithm presented in Section III, we can choose the scale level freely. In practice we usually choose the scale level as 1, 3, 5, 7, 9, $\cdots$. The estimated curvature values at several scales are given in Fig. 4. We then do the coarse to fine matching across these scales. The detected corner points at level 1 are drawn in Fig. 5 as crosses. For better illustration the detected corner points in a small portion of the curve are shown in Fig. 7. It can be

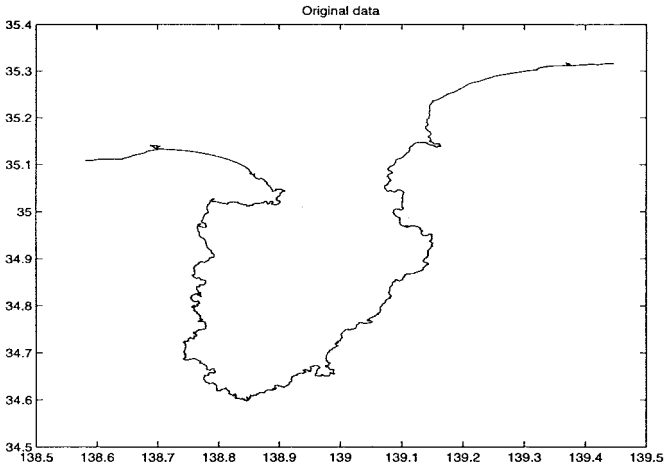

Fig. 3. Original terrain map curve with 2048 sampling points.

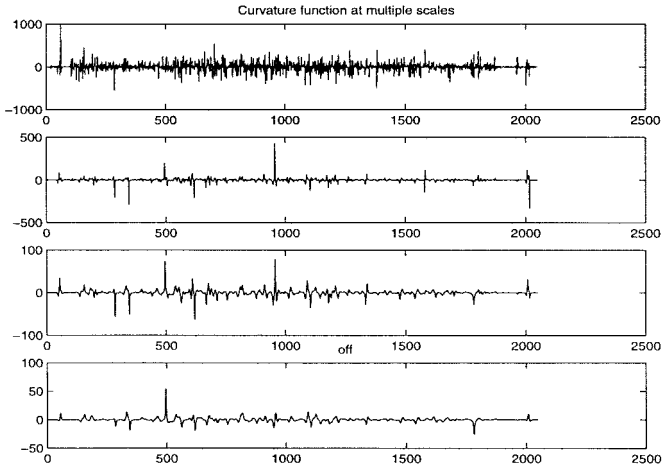

Fig. 4. Curvature functions at scale 1, 3, 5, 7. The curvature scale-space image is a hierarchical organization of the inflection points of the curve as indicated by the zero-crossings of the curvature function at multiple scales. In practice, only a discrete set of scales are sufficient for application. Using B-spline technology, we can compute the scale-space image efficiently at any scales.

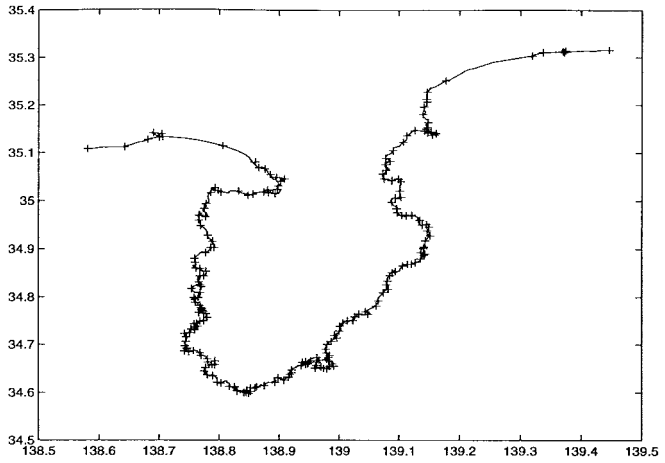

Fig. 5. Detected corner points in scale 1, which are denoted as cross "+" in the curve. The number of detected corner points is 221.

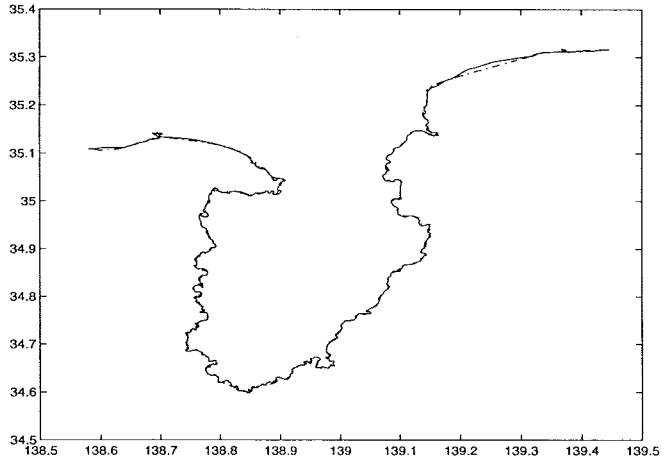

Fig. 6. Recovered curve using spline fitting. The recovered curve is drawn in dashed dot line. Compression ratio = 9.2670, error = 0.0035.

seen around the x-coordinate 138.7, more corner points have to be assigned in order to record the fine change of the curve orientation. Between the x-coordinate 138.7 and 138.8, only one point is necessary for the representation of this smooth segment. In Fig. 6, the reconstructed curve is drawn in dotted line, which gives almost the same curve as the original one. In the paper we select the spline fitting for reconstruction. The compression ratio for this curve is defined as the ratio between the number of the original curve points over that of the compressed data points. The compression ratio in this case is 9.2670, i.e., only 221 points are needed to represent the original curve with 2048 points. The reconstruction error is defined as

$$\text{Error} = \sqrt{\frac{\sum_{i=1}^{n} (d_i)^2}{n}}$$

where d_i is the distance from reconstruction data points to the original data points and n is the number of data points. Fig. 8 shows the result of curve compression at scale 3. We can see that there is a tradeoff in selecting the width or scale of the B-spline filter. A large width will remove small

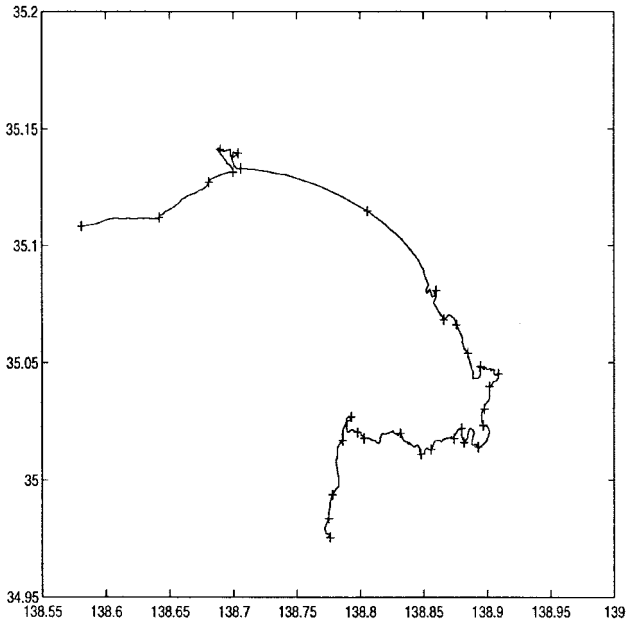

Fig. 7. Zoom in a small portion of the curve to show the multiscale property of the shape descriptor.

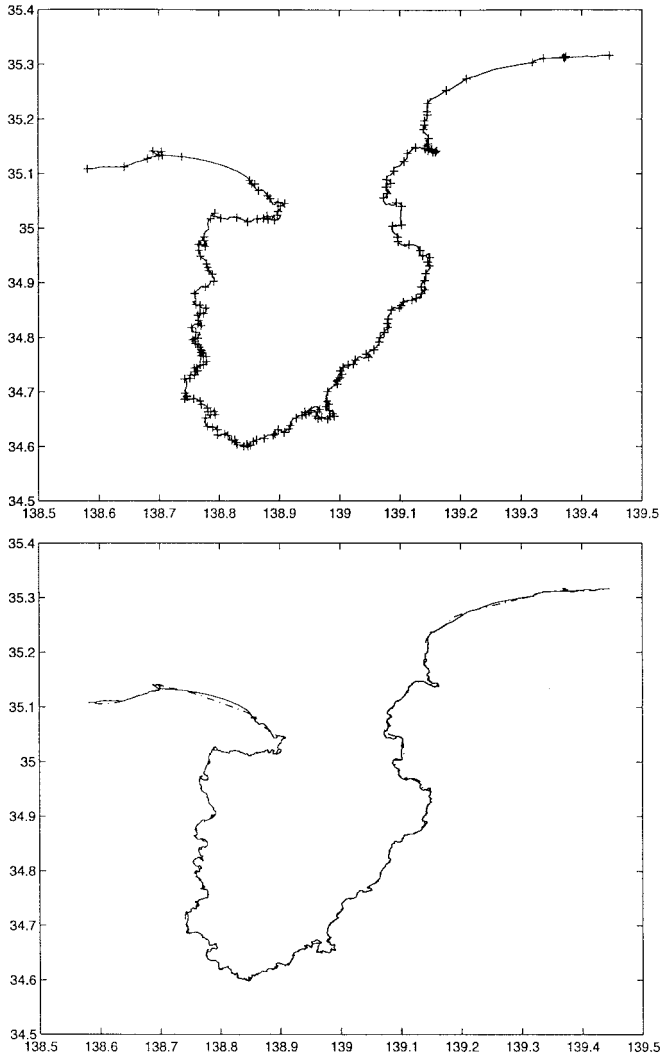

Fig. 8. Curve data compression at scale 3. Top: detected corner points; number = 210. Bottom: recovered curve; compression ratio = 9.7524, error = 0.0031.

details of the boundary curvature while a small width will permit false concavities and convexities. Therefore, at the finer scale the reconstruction error is lower with comparatively low compression ratio. At the coarser scale the reconstruction error is large with high compression ratio. The compression ratio is also related to the type of curve data at hand. An intermediate scale level is recommended in practice.

VI. Discussions and Conclusions

We have presented an algorithm for multiscale shape representation based on B-spline wavelets.

The classical multiscale curve representation is mainly based on the Gaussian scale-space representation. According to the criteria of Mokhtarian and Mackworth [8] for multiscale curve representation, one basic requirement is the *efficiency*. Using B-spline wavelets, we can obtain efficient subdivision algorithms and compute the geometric information of the curve easily. This shows that the B-spline scale-space is very suitable for multiscale curve representation.

The curve representations in the B-spline scale-space also enjoy other advantages over the classical Gaussian scale-spaces. For example, B-spline representation is defined directly on an initial discrete set of samples, and the outputs of B-spline wavelet transforms are directly used for curvature computation as shown in (15), thus avoiding possible problems caused by discretization of continuous Gaussian scale-space. In other words, the proposed approach is *easy to implement*. Furthermore, since B-spline representation is a piecewise polynomial representation, it is also easy to determine the properties of the shape of the curve. All of these indicate that the multiscale B-spline representation satisfies the criteria as described in [8].

References

[1] R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling.* San Francisco, CA: Morgan Kaufmann, 1987.

[2] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part 1 and Part 2," *IEEE Trans. Signal Processing* vol. 41, pp. 821–847, 1993.

[3] ——, "Fast B-spline algorithms for continuous image representation and interpolation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 277–285, 1991.

[4] Y.-P. Wang, "Image representations using multiscale differential operators," *IEEE Trans. Image Processing*, vol. 8, Dec. 1999, to be published; available at http://wavelets.math.nus.edu.sg/~wyp/download_papers/Edgespline.ps.gz.

[5] Y.-P. Wang and S. L. Lee, "Scale-space derived from B-splines," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 1040–1055, Oct. 1998.

[6] J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda, "Uniqueness of Gaussian kernel for scale-space filtering," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 26–33, 1986.

[7] A. Asada and M. Brady, "The curvature primal sketch," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 2–3, 1986.

[8] F. Mokhtarian and A. K. Mackworth, "A theory of multiscale, curvature-based shape representation for planar curves," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 789–805, Aug. 1992.

[9] G. Sapiro, A. Cohen, and A. M. Bruckstein, "A subdivision scheme for continuous-scale B-splines and affine-invariant progressive smoothing," *J. Math. Imag. Vis.*, vol. 7, pp. 23–40, 1997.

[10] G. W. Wornell, "Wavelet-based representations for the $1/f$ family of fractal processes," *Proc. IEEE*, vol. 81, Oct. 1993.

[11] E. Suhir, *Applied Probability for Engineers and Scientists.* New York: McGraw-Hill, 1997.

[12] F. Attneave, "Some informational aspects of visual perception," *Psychol. Rev.*, vol. 61, pp. 183–193, 1954.

Yu-Ping Wang received the B.Sc. degree in applied mathematics in 1990 from Tianjin University, China. He received the Ph.D. degree in communication and electronic systems at the Image Processing Center and the M.Sc. degree in computational mathematics at the Center for Applied Mathematics, Xi'an Jiaotong University, China, in 1996 and 1993, respectively.

He was a Research Fellow with the Wavelets Strategic Research Program, National University of Singapore, since 1996. He joined Washington University, St. Louis, MO, in March 1999, where he works on medical image analysis. His current research interests include computer vision, signal analysis, and various applications of wavelets, partial differential equations, fractals and splines.

S. L. Lee received the B.Sc. (hons) and M.Sc. degrees from the University of Malaya, Kuala Lumpur, in 1969 and 1971, respectively, and the Ph.D. degree in mathematics from the University of Alberta, Edmonton, Alta., Canada, in 1974.

He is a Professor of mathematics at the National University of Singapore (NUS), and the Director, Center for Wavelets, Approximation and Information Processing, NUS. His research interests include spline approximation, computer aided geometric design, and wavelets.

Kazuo Toraichi (M'99) was born in Tokyo, Japan, in 1943. He received the B.E. degree in electronics engineering from Tokyo Denki University, Tokyo, Japan, in 1966, and the M.E. and Ph.D. degrees in electronics engineering from Hokkaido University, Hokkaido, Japan, in 1968 and 1971, respectively.

From 1971 to 1980, he worked in the Electrotechnical Laboratory of MITI. In 1980, he became an Associate Professor at the Institute of Information Sciences and Electronics, University of Tsukuba, Ibaraki, Japan, where he is currently a Professor. His main fields of research are wisdom systems and medical electronics.

Dr. Toraichi was the recipient of Research Futherance from the Takayanagi Foundation for Electronics Science and Technology, Japan, in 1989. Dr. Toraichi is a member of the Institute of Electrical Engineers of Japan.