An Analytical Curvature B-Spline Algorithm for Effective Curve Modeling

Joi San Tan[(✉)], Ibrahim Venkat, and Bahari Belaton

School of Computer Sciences, Universiti Sains Malaysia, 11800 Pulau Pinang, Malaysia
hiki_3joi@yahoo.com, {ibra,bahari}@usm.my

Abstract. This paper presents a new algorithm based on an analytical approach for generating non-uniform cubic B-spline which has potential applications in curve modeling. For a given set of data points, knot vectors are computed using the centripetal approach. Next, number of data points is assimilated around the high curvature areas and the parametrization aspect is computed by the inverse chord length with the new set of data points. Second order derivatives are used to determine the high curvature areas of the curves. The method proposed here enables to construct the curves smoothly around high curvature areas by assigning adequate number of data points for the B-splines. Experimental validations justify the fact that the average curve fitting error yielded by the proposed approach is the lowest when compared to other standard curve models.

Keywords: Non-uniform cubic B-spline · Centripetal method · Second order derivative · Inverse chord length

1 Introduction

Curve modeling plays an important role in describing shapes. There are a few existing techniques for describing boundary curves or analysis of shapes such as B-splines [1–5], chain code [6–8], Fourier descriptors [9, 10], polygonal approximation [11–13], and moments [14]. B-spline has attractive properties such as smoothness, continuity, invariance transformation, and local influence [1]. Compared to the other techniques, its main attraction is the piecewise polynomial characteristics which enable local controllability of the curves by using only a few of the neighboring points. Also, in view of these properties, the construction of a B-spline curve is always complex as it involves excessive computational cost especially on the tuning of parameters. Parameters such as control points, knot vectors and traveling nodes (parametrization) are required to be precise enough in order to construct a smooth curve. Besides tuning of parameters, interpolating a curve from a given diverse set of data points (an example is shown in Fig. 1) imposes a major challenge in curve modeling. A new approach based on tangent estimation for cubic B-spline has been proposed by Chen et al. [15]. Basically, the tangent points which are computed using a heuristic method serve as seed points. These seed points will define the seed segments and Bezier technique is used to construct the approximation curves for these segments. Further, these approximation curves are

sequentially extended to other tangent points by means of a curve unclamping technique [16, 17]. Genetic Algorithm (GA) has been proposed by Bein et al. [18] to solve uniform cubic B-splines using densely sampled curves. The knot points and sharpness information of the input curve points are basically represented as genes (chromosomes) in the GA approach.
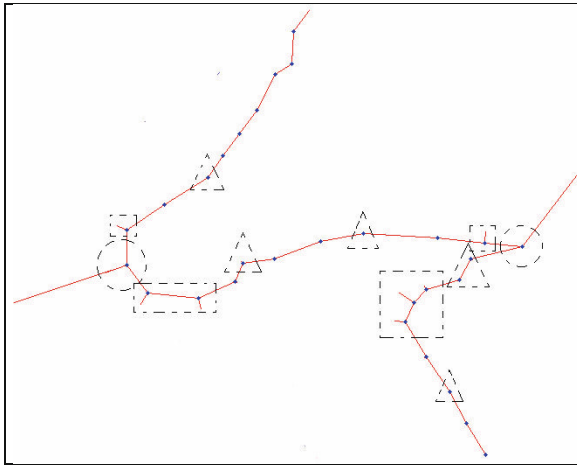

Fig. 1. An example curve showing high and low curvature areas. High curvature points (dots that connect every line) are obviously detected on the high variation areas as indicated (circle). Notice that, the points located around the convex and concave areas are also determined as high curvatures points (circles and boxes).
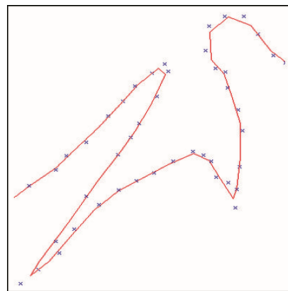

Fig. 2. An example of a poorly generated boundary curve. Shape of the curve is not well presented especially at the high variation areas. The degree of bending around these areas is not satisfactory.

The fitness function is defined between the constructed B-spline curve and the given input curve. Besides this, the fitness function has also been combined with a number of control points. The fitness error tends to reduce when the number of control points is increased. A serious limitation inherent in this GA-based interpolation method is the

increased computational cost as a consequence of excessive tuning of parameters. Without a good estimation of parameters, the curves will not be able to present the best shape especially at the convex and concave areas. Hence, a new Analytical Curvature (AC) B-spline algorithm here is proposed to solve the above mentioned potential problems. Initially, the sequence of the data points to generate a curve is arranged in sequential order. Next, curvature of the initially generated curve is estimated using second order derivatives. The high curvature (convex and concave) areas are selected and recorded into a list. Knot vectors are computed using the centripetal method. A new set of data points is generated optimally by gradually increasing the number of points at the high curvature areas based on the recorded list. Ultimately, the traveling nodes (parametrization) are computed using inverse chord length with the new set of data points. This paper is structured as follows. Section 2 briefly describes relevant related works pertaining to curvature models and B-splines. Section 3 explains the methodology aspect of the proposed AC B-spline algorithm. Section 4 presents experimental results and discussions. Section 5 finally concludes the paper.

2   Related Work

Curvature is basically computed using tangent angles [19, 20]. For example, suppose that there are two points A and B with a fixed chord length on the curve. Tangents are calculated for A and B and the angle between positive directions of both tangents are estimated. Then, the limit of the angle between the tangents at both the points yields the curvature. Arc-chord distance [21] was proposed to determine the peak points of the curve. First, a chord is chosen based on a point and its perpendicular arc. Then, along the chord, local maximum is computed to determine the local point. These two methods can determine curvatures to a reasonable precision. However, they involve the tedious process of estimating the curvature for the whole curve. Second order polynomial [22] is simpler compared to the above methods. It uses second order derivatives and neighboring points to calculate the curvature. There are techniques [23, 24] using second order derivatives and B-spline curves to detect corners and represent the curve. But, these approaches are different from the method proposed here. The curvature model proposed by Abbas et al. [23] focusses on uniform cubic B-spline curves instead of non-uniform curves. B-spline curve heavily depends on knot vectors and parametrization, which can also be defined as a set of traveling nodes. The traveling nodes tend to move between the knot intervals in order to compose the B-spline curve using the data points. Hence, it is important to clearly understand the relationship between the knot vectors and the parametrization. The estimation of knot vectors plays an important role in B-spline based curve modeling approaches. Genetic algorithms (GA) has been proposed to estimate the knot vectors by Rony and Michel [25]. Knot vectors are assigned as the control variables (chromosomes) in the GA approach. This method requires excessive tuning of parameters in order to obtain good results. Gaussian mixture model (GMM) [26] is used to construct the best knot vectors.

Initial location of the knot vectors is determined using the Monte Carlo approach. Next, GMM is used to determine the probability distribution of all knots based on the data points. The drawback of this method is the size of the data points. It is more suitable to estimate the knot vectors for large scale data. Uniform [27], chord length [28], centripetal [29], inverse chord length [3] and Foley and Nielson [30] are the well-known parametrization techniques which have been widely implemented. Haron et al. [31] proposed a new methodology to overcome the drawback of the hybrid parametrization technique which will produce a singular matrix when the distance between the traveling nodes is equal to zero. The traveling nodes are computed using an exponential approach and a new parameter which is associated with the maximum value of the B-spline function is calculated. The difference values between the traveling nodes and the new parameter are calculated. Finally, a new set of traveling nodes is estimated using the difference values. A refined centripetal [32] is proposed to improve the traveling nodes. The parametrization is carried out using osculating circle at each point. Besides, a fine wiggle validation method is also proposed to determine the performance of all the methods. Precise parametrization is obtained from the refined centripetal, but it involves a considerable computational cost when compared to the conventional methods.

3   Analytical Curvature (AC) B-Spline Interpolation Method

3.1   Analytical Curvature

Analytical curvature is used to determine the high variation (bending) areas on the curves. The curvature values that are estimated in these areas are mainly higher compared to the low variation areas. A corner regardless of being convex or concave is defined as a high curvature point [33]. In the AC B-spline, one of the main goals is to identify the set of high curvature points on the curves so as to increase the number of data points within these areas. Basically, the curvature is calculated with the parameterized curve, $C(s) = (x(s), y(s))$, using the second order derivatives method [33] with the equations below:

$$x(s) = a_3 s^2 + a_2 s + a \tag{1}$$

$$y(s) = b_3 s^2 + b_2 s + b_1 \tag{2}$$

with $s \in [-q_l, q_r]$. Parameter, s, is the coverage range of C within $-q_l$ and q_r in determining the curvatures. When s is equal to $-q_l$, 0, and q_r,

$$a_3 q_l^2 - a_2 q_l + a_1 = x_{i-m_L} \tag{3}$$

$$a_1 = x_i \tag{4}$$

$$a_3 q_r^2 - a_2 q_r + a_1 = x_{i+m_R} \tag{5}$$

x_{i-m_L} and x_{i+m_R} are the left and right side of x_i data point. The number of selected neighboring points is determined based on m where $m = 0.02n$ and n is the total number of data points. The selection is more flexible when the neighboring points are increased according to the number of data points. This adjustment is able to reduce certain amount of noise while generating appropriate high curvature points. The parameter q is usually assigned as 1. However, in this method, q is determined based on the distance between the neighboring points. q_l is the Euclidean distance between x_{i-m_L} and x_i where q_r is the Euclidean distance between x_i and x_{i+m_R}. Coefficients b_1, b_2 and b_3 can be computed using,

$$b_3 q_l^2 - b_2 q_l + b_1 = y_{i-m_L} \tag{6}$$

$$b_1 = y_i \tag{7}$$

$$b_3 q_r^2 - b_2 q_r + b_1 = y_{i+m_R} \tag{8}$$

Hence, a_2, a_3, b_2 and b_3 are computed using Eqs. (3)–(8). Then, the curvature is calculated as follows:

$$k = \frac{2(a_2 b_3 - a_3 b_2)}{(a_2^2 + b_2^2)^{\frac{3}{2}}} \tag{9}$$

It is constructed based on the formula:

$$k = \frac{\begin{vmatrix} x' & y' \\ x'' & y'' \end{vmatrix}}{\left(x'^2 + y'^2\right)^{\frac{3}{2}}} \tag{10}$$

Positive and negative curvature values are generated from the set P of data points based on Eq. (9). The negative values are ignored because the values only indicate the concave points whereas positive values indicate the convex points. Hence, absolute curvature values, $|k_i|$, are computed, in order to select high curvature points

$$K = \begin{cases} 1 & \text{if } |k_i| > \xi \\ 0 & \text{otherwise} \end{cases} \quad i = 1, \dots n \tag{11}$$

where

$$\xi = \frac{|max(k) - min(k)|}{2n} \tag{12}$$

ξ is the threshold value in Eq. (11). Then, a list, $\tilde{L}$, which defines the high curvature areas is sorted out from K. For example, if the high curvature points in P are from K_i^{th} point till K_{i+1}^{th} point, then $\tilde{L}_i \in [K_i, K_{i+1}]$, $i = 1, \dots v$ and v represents the number of high curvature areas. In the AC B-spline, in order to join the high and low curvature areas

smoothly, two neighboring points for both sides are included into $\tilde{L}$ where $\tilde{L}_i \in [K_i - 2, K_{i+1} + 2]$ as shown in Fig. 1. The neighboring points are indicated with the triangles.

3.2   B-Splines

The piecewise polynomial function of B-spline is given below:

$$N_{k,t}(u) = \sum_{i=1}^{n} \rho_i B_{i,k}(u), \quad i = 1, \ldots, n \tag{13}$$

The piecewise polynomial function can be defined recursively:

$$B_{i,1}(u) = \begin{cases} 1 & \text{if } t_i \leq u < t_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

$$B_{i,k}(u) = \frac{(u - t_i)B_{i,k-1}(u)}{t_{i+k-1} - t_i} + \frac{(t_{i+k} - u)B_{i+1,k-1}(u)}{t_{i+k} - t_{i+1}} \tag{15}$$

where ρ is the control points of the B-splines, k is the $(d + 1)^{\text{th}}$ order, d is the degree of the basis function polynomial, t represents the knot vectors, and u is the traveling node (parametrization). A smooth B-spline curve can be constructed with the following characteristics:

1. Control points ρ that are especially located at high variation areas are allocated within the same knot interval, $[t_i, t_{i+1}]$.
2. When the control points are close to each other, the speed of the traveling node u is decreased.
3. When the control points are far from each other, the speed of the traveling node u is increased.

A new constructive scheme to define all the parameters to construct a B-spline curve are proposed as follows:

1. Non-uniform cubic B-splines $d = 3$ are used to represent the curve;
2. Centripetal method using the set P of data points is used to estimate the knot vectors;
3. Increase the number of data points on the high curvature areas; A new set P' of data points is generated.
4. u is computed using the inverse chord length [3] with the aid of P'.

Next the proposed AC B-spline interpolation algorithm is described, which consists of the following steps:

1. Calculate the knot vectors t using the centripetal method with P

$$\triangle t_i = \|P_{i+1} - P_i\|^\alpha, \quad i = 1,\ldots,n \quad \text{and} \quad \alpha = \frac{1}{2} \tag{16}$$

2. for $(i = 1; i < 100; i++)$ {
   for $(j = 1; j < m + 1; m++)$ {
3. A new interval list, $\psi \in [P_{K_j-2}, P_{K_{j+1}+2}]$, containing i points is generated.
4. ψ is inserted into P starting P_{K_j-2} till $P_{K_{j+1}+2}$ }
5. A new set of data points, P_b' is generated where $b = 1,\ldots,w$ and w is the size of the new data points.
6. Inverse Chord Length is used to compute u with P'

$$\triangle u_i = u_{i+1} + \frac{u'_{max}}{\sqrt[\gamma]{\|P_i' - P_{i-1}'\|}} \cdot \frac{1}{\sum_{i=2}^{w} \frac{1}{\sqrt[\gamma]{\|P_i' - P_{i-1}'\|}}} \tag{17}$$

where $i = 1,\ldots,w$, $u'_{max} = w - 2$(for cubic B-spline) and $\gamma > 100$. The estimation of parameter γ (refer Fig. 3) has an impact on the curve fitting error.

7. The B-spline curve, N, with control points, ρ, knot vectors, t, and traveling node, u are identified. The data points, P, are assigned as the control points.
8. Curve fitting error, ε, between N and P is calculated using the formula:

$$\varepsilon_i = \sum_{i=1}^{n} \|P_i - N_i\|, \quad i = 1,\ldots,n \tag{18}$$

where N is the nearest neighboring point set with P.

9. If $(\varepsilon_i - \varepsilon_{i-1}) < 0.005$, stop the iteration. }

Most of the knot vectors are estimated uniformly. However, it may not be suitable when the distribution of the data points is in irregular form as shown in Fig. 3b.
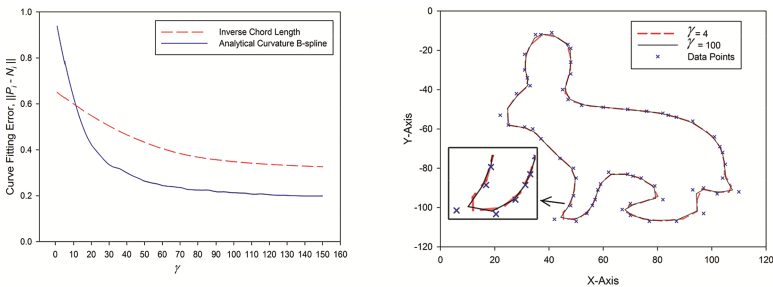

Fig. 3. Identification and empirical experiment of γ value. (a) Identification of γ value. The γ value is sequentially increased and the curve fitting error is calculated simultaneously. Two methods are implemented in order to observe the variation of the γ. The error values gradually become constant as the value of γ is increasing for both methods. (b) $\gamma = 4$ and $\gamma = 100$. The shape model yielded by the proposed AC B-spline algorithm for a given input. The curve with $\gamma = 100$ is smoother compared to $\gamma = 4$.
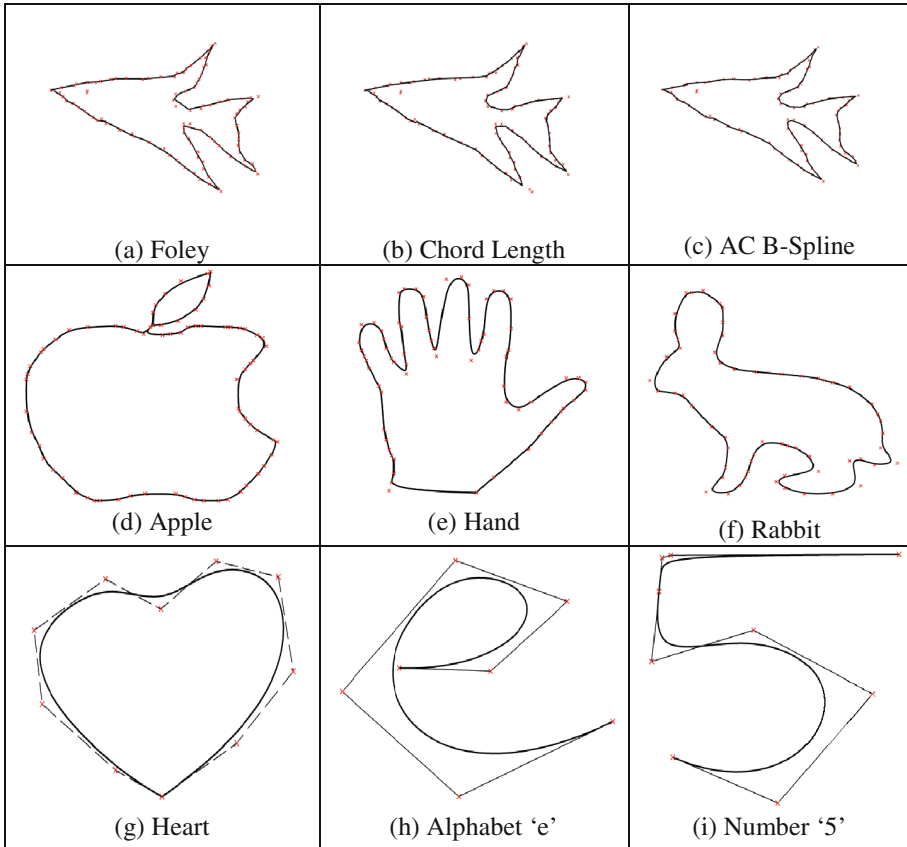
Fig. 4. Some sample results from various datasets are shown, where the 'cross' signs represent the data points and also the control points. Figure 4a–c represent some example results generated by the Foley, chord length and the proposed AC B-spline approaches. It can be seen that the proposed AC B-spline approach fits smoother curves when compared to the Foley and chord length approach especially at the convex and concave areas. Figure 4d–f represent three sample shapes that have been extracted using the Harris corner detector. We can observe that the shapes of these images are fitted reasonably well using the AC B-spline. Figure 4g–i represent three sample results (randomly chosen) fitted by selecting the least number of data points from the images. The purpose is to test the effectiveness of the proposed AC B-spline approach when less number of control points is being considered. It can be observed that, the shape of the heart, the alphabet 'e' and numerical '5' are all constructed smoothly as well.

Hence, non-uniform estimation for knot vectors is considered. By considering the traveling nodes along the curve where the distance of each two adjacent points is proportional to u, the control points, especially those located at high variation areas or complex turning points must be allocated into the same knot interval. Hence, centripetal method (Eq. (16)) is chosen to estimate t. The knot vectors must be a non-decreasing sequence of real numbers. In our case, it is a periodic B-spline which closes on itself. The high variation areas are usually difficult to handle in boundary curve or shape

analysis as shown in Fig. 2. The number of data points is increased on the high curvature areas, so that smooth shapes are obtained. The number of points for the insertion is decided automatically (step 3 and step 4 of the method) by estimating the curve fitting error between P and N.

Inverse chord length is proposed in this method because it is able to fulfill the characteristics that have been described earlier. The traveling nodes that are computed using inverse chord length are able to control the shape of a given input object based on the distance between the control points. The parameter γ is estimated empirically as shown in Fig. 3a. Figure 3b illustrates the construction of the shape of a rabbit (input object) when different values of γ are applied. γ with 4 is the value that was suggested by [3].

4   Results and Discussions

For experimental evaluations, Chui-Rangarajan synthesized dataset [34], SIID silhouette dataset [35] and some randomly created shapes are used for testing purposes. Feature extraction is carried out on SIID dataset using Harris corner detector [36] because only images are provided instead of data points. B-splines curve generated using uniform, centripetal, chord length, inverse chord length, Foley and Deboor [37] parametrization techniques with the uniform knot vectors are compared with the AC B-spline in this experiment. One hundred of fish shapes from the synthesized dataset are tested with all of the methods and the average of the curve fitting error is estimated for every method. The average error values of the fish shapes are presented in Table 1.

Table 1. Comparison of the proposed AC B-spline approach with other standard curve models in terms of average curve fitting error. One hundred fish shapes from the Chui-Rangarajan synthesized dataset [34] have been used for this experimental evaluation.

Standard methods	Average curve fitting error
Uniform	0.3223
Centripetal	0.3330
Chord length	0.5227
Inverse chord length	0.2898
Foley	0.5645
Deboor algorithm	0.2902
AC B-spline	0.2749

Overall, the lowest curve fitting error is achieved by the AC B-spline and highest error is yielded by the Foley parametrization approach. The effectiveness of these methods is also evaluated when the number of data points is gradually increased in order to produce a smoother curve. In general, we observe that as the number of the data points

increases, the shape of the curve gets smoother and the curve fitting error tend to decrease. The graph shown in Fig. 5 compares the performance of the proposed AC B-spline with some standard curve modeling approaches. The average curve fitting error for all the methods tends to reach a constant when the number of data points is gradually increased. AC B-spline shows good results with the lowest error values.
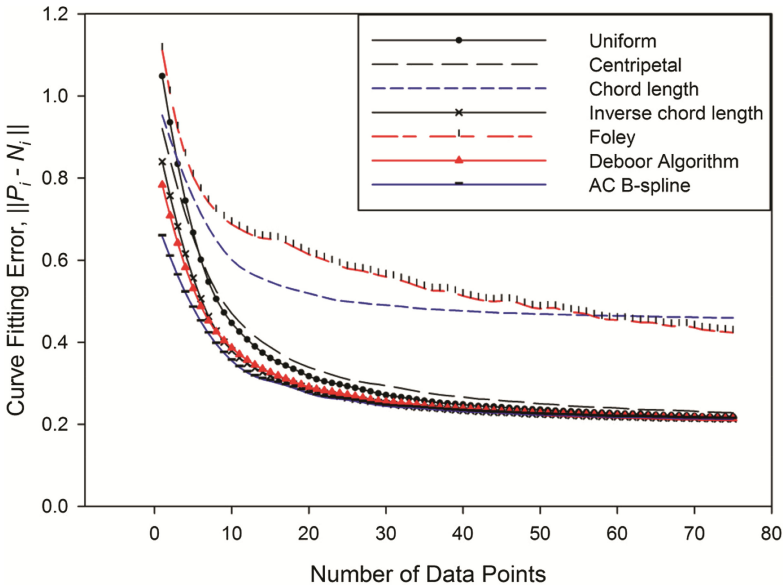

Fig. 5. Graph showing the variations of the average curve fitting error number when the data points are being gradually incremented. The graph shows that the proposed AC B-spline approach has the lowest error when compared to other standard curve models.

Inverse chord length and De Boor algorithm also exhibit satisfactory results compared to other methods. Foley and chord length become constant, however, the error values are always higher than the other methods. Figure 4a–c represent some fish shapes taken from the Chui-Rangarajan synthesized dataset. We can observe that the shape of the fish fitted by the AC B-spline approach is better when compared to Foley and chord length parametrization which use uniform knot vectors. In particular, Fig. 4c witnesses the fact that the shape of the curve formed is smoother at the high convex and concave curvature areas. Some more example shapes fitted using the AC B-spline approach have been shown in Fig. 4d–i.

5   Conclusion

In this work, it is shown that the analytical curvature (AC) B-spline method proposed here is able to provide effective results in curve modeling. The method is proposed based on the understanding of: (1) control points especially on high curvature areas must be allocated within a knot interval, (2) the traveling nodes (parametrization) must be

controlled carefully in order to model a curve. Hence, a second order derivative based method is used to determine the high curvature points and the number of data points is increased within these areas. Next, the new set of data points can be used to estimate the parametrization using inverse chord length in order to produce a smooth curve. AC B-spline shows better results when compared with the other methods. For future work, improvement on the proposed method can be explored by manipulating the control points, besides extending the approach to 3D curve modeling.

References

1. Cohen, F.S., Wang, J.-Y.: Part I: modeling image curves using invariant 3-D object curve models-a path to 3-D recognition and shape estimation from image contours. IEEE Trans. Pattern Anal. Mach. Intell. **16**, 1–12 (1994)
2. Farin, G.E.: Curves and Surfaces for CAGD: A Practical Guide. Morgan Kaufmann, Burlington (2002)
3. Huang, Z., Cohen, F.S.: Affine-invariant B-spline moments for curve matching. IEEE Trans. Image Process. **5**, 1473–1480 (1996)
4. Sederberg, T.W.: Computer Aided Geometric Design. CAGD Course Notes. Brigham Young University, Provo, UT, 84602 (2012)
5. Wang, J.-Y., Cohen, F.S.: Part II: 3-D object recognition and shape estimation from image contours using B-splines, shape invariant matching, and neural network. IEEE Trans. Pattern Anal. Mach. Intell. **16**, 13–23 (1994)
6. Dai, X., Khorram, S.: A feature-based image registration algorithm using improved chain-code representation combined with invariant moments. IEEE Trans. Geosci. Remote Sens. **37**, 2351–2362 (1999)
7. Lin, Y.-L., Wang, M.-J.J.: Automated body feature extraction from 2D images. Expert Syst. Appl. **38**, 2585–2591 (2011)
8. Vaddi, R.S., Boggavarapu, L.N.P., Vankayalapati, H.D., Anne, K.R.: Contour detection using freeman chain code and approximation methods for the real time object detection. Asian J. Comput. Sci. Inf. Technol. **1** (2013)
9. Lee, C.P., Tan, A.W.C., Tan, S.C.: Gait recognition via optimally interpolated deformable contours. Pattern Recogn. Lett. **34**, 663–669 (2013)
10. Mebatsion, H.K., Paliwal, J., Jayas, D.S.: Evaluation of variations in the shape of grain types using principal components analysis of the elliptic Fourier descriptors. Comput. Electron. Agric. **80**, 63–70 (2012)
11. Kolesnikov, A.: ISE-bounded polygonal approximation of digital curves. Pattern Recogn. Lett. **33**, 1329–1337 (2012)
12. Marji, M., Siy, P.: A new algorithm for dominant points detection and polygonization of digital curves. Pattern Recogn. **36**, 2239–2251 (2003)
13. Yin, P.-Y.: A discrete particle swarm algorithm for optimal polygonal approximation of digital curves. J. Vis. Commun. Image Represent. **15**, 241–260 (2004)

14. Anuar, F.M., Setchi, R., Lai, Y.K.: Trademark image retrieval using an integrated shape descriptor. Expert Syst. Appl. **40**, 105–121 (2013)
15. Chen, X.-D., Ma, W., Paul, J.-C.: Cubic B-spline curve approximation by curve unclamping. Comput. Aided Des. **42**, 523–534 (2010)
16. Hu, S.-M., Tai, C.-L., Zhang, S.-H.: An extension algorithm for B-splines by curve unclamping. Comput. Aided Des. **34**, 415–419 (2002)
17. Piegl, L., Tiller, W.: Curve and Surface Basics. Springer, Heidelberg (1995)
18. Bein, M., Fellner, D.W., Stork, A.E.: Genetic B-spline approximation on combined B-reps. Vis. Comput. **27**, 485–494 (2011)
19. Liu, H., Latecki, L.J., Liu, W.: A unified curvature definition for regular, polygonal, and digital planar curves. Int. J. Comput. Vis. **80**, 104–124 (2008)
20. Hermann, S., Klette, R.: Multigrid analysis of curvature estimators. In: Proceedings of the Image Vision Computing, pp. 108–112. New Zealand (2003)
21. Marji, M., Klette, R., Siy, P.: Corner detection and curve partitioning using arc-chord distance. In: Klette, R., Žunić, J. (eds.) IWCIA 2004. LNCS, vol. 3322, pp. 512–521. Springer, Heidelberg (2004)
22. Kovalevsky, V.: Curvature in digital 2D images. Int. J. Pattern Recogn. Artif. Intell. **15**, 1183–1200 (2001)
23. Abbas, A., Nasri, A., Maekawa, T.: Generating B-spline curves with points, normals and curvature constraints: a constructive approach. Vis. Comput. **26**, 823–829 (2010)
24. Medioni, G., Yasumoto, Y.: Corner detection and curve representation using cubic B-splines. In: 1986 IEEE International Conference on Robotics and Automation. Proceedings, vol. 3, pp. 764–769. IEEE (1986)
25. Goldenthal, R., Bercovier, M.: Spline curve approximation and design by optimal control over the knots. In: Hahmann, S., Brunnett, G., Farin, G., Goldman, R. (eds.) Geometric Modelling, pp. 53–64. Springer, Vienna (2004)
26. Zhao, X., Zhang, C., Yang, B., Li, P.: Adaptive knot placement using a GMM-based continuous optimization algorithm in B-spline curve approximation. Comput. Aided Des. **43**, 598–604 (2011)
27. Lim, C.-G.: A universal parametrization in B-spline curve and surface interpolation. Comput. Aided Geom. Des. **16**, 407–422 (1999)
28. Lü, W.: Curves with chord length parameterization. Comput. Aided Geom. Des. **26**, 342–350 (2009)
29. Hoschek, J., Lasser, D., Schumaker, L.L.: Fundamentals of Computer Aided Geometric Design. AK Peters, Ltd., Natick (1993)
30. Foley, T.A., Nielson, G.M.: Knot Selection for Parametric Spline Interpolation. Academic Press Professional, Inc., Waltham (1989)
31. Haron, H., Rehman, A., Adi, D.I.S., Lim, S.P., Saba, T.: Parameterization method on B-spline curve. Math. Probl. Eng. **2012**, (2012)
32. Fang, J.-J., Hung, C.-L.: An improved parameterization method for B-spline curve and surface interpolation. Comput. Aided Des. **45**, 1005–1028 (2013)
33. Hermann, S., Klette, R.: A comparative study on 2d curvature estimators. pp. (2006)
34. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. Comput. Vis. Image Underst. **89**, 114–141 (2003)
35. Sebastian, T.B., Klein, P.N., Kimia, B.B.: Recognition of shapes by editing their shock graphs. IEEE Trans. Pattern Anal. Mach. Intell. **26**, 550–571 (2004)

36. Harris, C., Stephens, M.: A combined corner and edge detector. In: Alvey Vision Conference, vol. 15, Manchester, UK (1988)
37. De Boor, C.: A practical guide to splines. In: Mathematics of Computation. Springer, New York (1978)