The Open
University

Open Research Online

The Open University's repository of research publications
and other research outputs

A Novel Half-Way Shifting Bezier Curve Model

Conference or Workshop Item

How to cite:

Sohel, Ferdous; Dooley, Laurence S. and Karmakar, Gour C. (2005). A Novel Half-Way Shifting Bezier Curve Model. In: 10th IEEE International Conference on Convergent Technologies (TENCON '05), 21-25 Nov 2005, Melbourne, Australia.

For guidance on citations see FAQs.

oro.open.ac.uk

A Novel Half-Way Shifting Bezier Curve Model

Ferdous Ahmed Sohel, Laurence S. Dooley, and Gour C. Karmakar

Gippsland School of Information Technology
Monash University, Churchill, Victoria – 3842, Australia
E-mail: {Ferdous.Sohel, Laurence.Dooley, Gour.Karmakar}@infotech.monash.edu.au

Abstract

Bezier curves can cause a considerable gap to occur between the approximation curve and its control polygon, due to considering only the global information of the control points. In order to reduce this error in curve representations, localised information needs to be incorporated, with the main philosophy to narrow down the gap by shifting the Bezier curve points closer to the control polygon. To integrate this idea into the theoretical framework of the classical Bezier curve model, this paper presents a novel Half-way shifting Bezier Curve (HBC) model, which automatically incorporates localised information along with the global Bezier information. Both subjective and objective performance evaluations of the HBC model using upon a number of objects having arbitrary shape confirm its considerable improvement over the classical Bezier curve model without increasing the order of computational complexity.

Index Terms – Bezier curve, localised information, global information, half-way shifting.

I. INTRODUCTION

Bezier curves were independently developed by P. de Casteljau and P.E. Bézier and have been applied to many computer-aided design (CAD) applications. While their origin can be traced back to the design of car body shapes in the sixties, their usage is no longer confined to this field. Indeed, their robustness in curve and surface representation has meant that Bezier curve usage now pervades many areas of multimedia technology including shape description of characters [1-2] and objects [3], active shape lip modelling [4], shape error concealment for MPEG-4 objects [5] and surface mapping [6].

Bezier curves are defined by a set of *control points* which, depending upon their number and orientation, govern the shape of the curve. Bezier curves however, only consider the global information about the control points [7] and calculate the curve points in a linear iterative subdivision approach of the edges of the control polygon (CP). As a result, there is often a large error between the curve and its CP especially in the middle region of the curve, thereby restricting the maximum curve length for a given number of control points. A composite Bezier [8] strategy comprising multiple segments, and hence more control points can be used to address this shortcoming, however when describing a shape using a Bezier curve, most of the control points are required to be defined outside the original shape which will not necessarily be inside the coordinate system, so increasing the computational overhead in many applications. *Degree elevation* [9] has been exploited to form a curve with an increased number of control points though all these points, except the two end points have to be recalculated so incurring a significant computational overhead. *Subdivision* and *refinement* techniques have also been used to reduce the gap between the Bezier curve and its CP by increasing the number of curve segments. When the control points of a curve are known, two sets of new control points that are closer to the curve can be calculated using subdivision algorithms such as *midpoint subdivision* [10] or *arbitrary subdivision* [11]. All these algorithms however increase the number of curve segments and thus the number of control points. Moreover, to ensure that curve segments are conjoint, the number of subdivisions has also always to be constrained.

All the aforementioned algorithms minimise the gap between the Bezier curve and its CP by increasing the number of control points. In communication applications this means a higher coding and transmission cost to represent a particular shape. To overcome this problem, this paper presents *a novel Half-way shifting Bezier Curve (HBC) model*, which incorporates localised information within the classical Bezier curve framework by shifting the curve point at the midpoint between the curve point and the CP, with no increase in computational complexity. It is particularly noteworthy that this new model can be

seamlessly integrated into Bezier refinement algorithms such as *degree elevation* and s*ubdivision* and it will be shown that HBC retains many of the core properties of the classical Bezier curve. The performance of the model as a generic shape descriptor for a number of arbitrary shapes has been extensively analysed using both quantitative and qualitative metrics, with results clearly confirming its superiority compared with the classical BC.

The remainder of the paper is structured as follows: Section II provides a short overview of the classical Bezier curve identifying problems due to the global control, while Section III discusses the theoretical basis of the new HBC model along with proofs that the key properties of the classical Bezier curve are retained. Section IV presents some experimental results confirming the superior performance of HBC model relative to the original Bezier curve, with some conclusions given in Section V.

II.   OVERVIEW OF THE CLASSICAL BEZIER CURVE

The Bezier curve (BC) is defined in iteratively linear weighted subdivision, having variation diminishing property, of the edges of the control polygon. The variation diminishing property is in the sense that it starts with the edges of the control polygon and decrements the number of edges by one in each iteration and stops when the final point is generated for a particular weight t. The set of $N+1$ starting points is referred to as the *control points* which govern the characteristics of the Bezier curve of degree N. The polygon connecting the control points is called the *control polygon* (CP). The Casteljau form of the Bezier curve for an ordered set of control points $V = \{v_1, v_2, ..., v_{N+1}\}$ is defined as:-

$$v_i^r(t) = \begin{cases} i^{th} \ member of \ V, v_i; & if \ r=1; \\ (1-t)v_i^{r-1}(t)+tv_{i+1}^{r-1}(t); & r=2,\cdots,N+1; \ i=1,\cdots,N-r; \ 0 \le t \le 1 \end{cases} \quad (1)$$

where t is the weight of subdivision which determines the number of points on the Bezier curve. The *final* generation $v_1^{N+1}(t)$ is the Bezier curve.

From (1) it is evident that a particular BC point is generated by blending all control points. This implies that BC only considers the global information of a shape and yields a gap between the curve and its CP. Figure 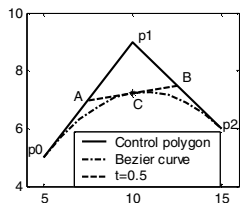1 shows an example of a quadratic BC using control point p_0, p_1, p_2 illustrating the large gap that occurs between the curve and

Figure 1. A quadratic Bezier curve illustrating the gap.

its CP. For $t = 0.5$, the inner part of ΔAp_1B is never reached and curve point C is generated on line AB. This gap which manifests itself as a significant shape distortion is due to the classical BC considering only the global information. In order to reduce this error, a novel strategy incorporating local information about the CP is presented in the next section.

III.   HALF-WAY SHIFTING BEZIER CURVE (HBC) MODEL

To minimise the gap between a Bezier curve and its control polygon the generated Bezier curve point for each t must be shifted inside the area between the control polygon and the Bezier curve using *half-way shifting* as follows:-
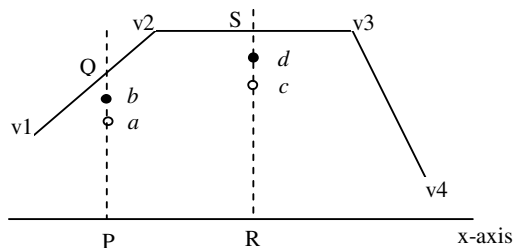
Figure 2. Half-way Bezier curve point generation illustration.

The HBC point is then calculated as the mid-point between the BC point and this pivot point. Mid point (half-way shifting) is subsequently chosen as the new point, since being average it will retain both the properties of the Bezier curve as well as the CP. This process is illustrated in Figure 2, where a is the Bezier curve point for a particular t. The minimum distance edge from t is $v_1 v_2$ and PQ is the line parallel to the y- axis through a. The intersection point between PQ and $v_1 v_2$ is Q. After *half-way shifting* the HBC point becomes b. A similarly translation takes place for point c, with the corresponding HBC point now being d. However, for a particular BC point; if the edge of the CP with the minimum distance is parallel or almost parallel to the y- axis (or indeed the y- axis itself) then there will be no intersection point between the edge and line passing through the BC point or the intersection point be far outside the CP and so the curve is discontinuous. Two examples illustrating these problems and solutions are presented in Figure 3.

Figure 3(a) shows a discontinuous curve caused by the CP edges being parallel to the y- axis. This can be solved as follows so that when the minimum distance edge is

parallel to the y-axis, a line parallel to the x-axis rather than the y-axis is drawn through the Bezier curve point.

Firstly, the edge of the control polygon, which ensures the minimum distance between the CP and the Bezier curve point, is determined. A line parallel to the y-axis and passing through the Bezier curve point is then generated, with the intersection point of this line and the minimum distance edge being the *pivot* point.
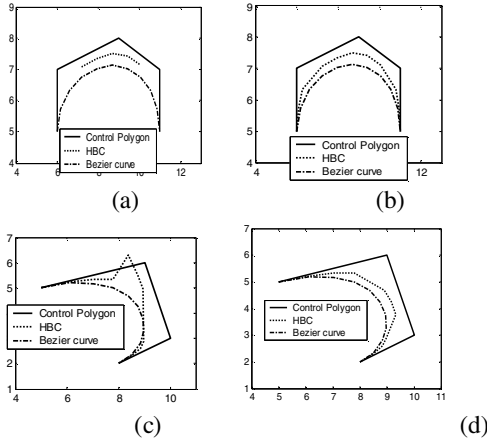

(a) (b)

(c) (d)

Figure 3. Problems and their solutions when any edge of the control polygon is parallel or almost parallel to the y-axis.

The ordinate value of the HBC point is then calculated from the half-way interpolation of the ordinates of Bezier curve point and the intersection point, and the abscissa is retained as the Bezier curve point. The resultant curve is shown in Figure 3(b).
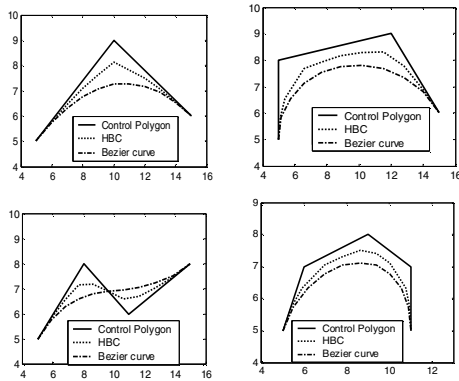

Figure 4. HBC and Bezier curve of different degrees and orientation of the control points.

A further complication is that curves generated in this manner may have unnecessary oscillations and even in certain cases go outside the CP (Figure 3(c)) when any CP edge is almost parallel to the y-axis. To overcome this problem, the actual slope of a particular edge is considered. If the edge is flat with respect to the x-axis then a line parallel to the y-axis will be drawn and the HBC points generated, otherwise the edge will be steeper and a line parallel to the x-axis will be drawn and the HBC points generated accordingly. Deciding upon whether a curve is flat or steep is easily detected by checking the absolute value of the slope. If it is greater than 1 then the curve is steep, otherwise it is flat. The resultant model will then generate the curve shown in Figure 3 (d) for the same control points as in Figure 3(c). Some example curves generated using this approach is presented in Figure 4. It is clear from the figures that HBC is containing more information of the control points and producing less distance from the CP.

The HBC can be formally defined as in Algorithm 1.

Algorithm 1. Half-way Shifting Bezier curve Algorithm.

Input: A set of control points, $V = \{v_1, v_2, \cdots, v_n\}$, which will govern the shape of the curve.
Variables: t is the weight, $V = \{v_1, v_2, \cdots, v_n\}$ is the set of control points.
Output: The half-way shifting Bezier curve.

1. For each value of t
2.   Find the BC point $a = V(t)$ by (1).
3.   Find the edge $v_i v_{i+1}$ of the CP with minimum distance from a. (In case of ties select the first edge among them)
4.   IF the slope of the edge $v_i v_{i+1}$ is less than 1 THEN
            Draw a line l parallel to the y-axis through a.
6.   Find the intersection point Q between line l and edge $v_i v_{i+1}$.
7.   Calculate HBC point by:-
8.   HBC$_x$ (t)= Q_x and HBC$_y$ (t)= $(a_y + Q_y)/2.0$.
9.   ELSE
10.   Draw a line l parallel to the x-axis through a.
11.   Find the intersection point Q between line l and edge $v_i v_{i+1}$.
12.   Calculate HBC point by:-
13.   HBC$_x$ (t)= $(a_x + Q_x)/2.0$ and HBC$_y$ (t)= Q_y.
14. END.

As the foundation HBC model is underpinned by the classical Bezier curve theory, many of the core properties [7] are preserved. The following examines some of these properties as well as also considering the computational complexity overhead of the HBC model.

Lemma 1: End point interpolation: The HBC always passes through its first and last control points.

Proof: Any Bezier curve interpolates its end control points [7] for the starting control point $u=0$ and for the end $u=1$. So the corresponding pivot point and hence the HBC points will be that particular end-point.

Lemma 2: Convex Hull Property: The HBC lies within the convex hull of its control points.

Proof: The Bezier curve lies within its control polygon [7]. And the HBC points lie in area between the CP and the BC inclusive. So the HBC will lie within the convex hull of its control points.

Lemma 3: Linear Precision: When all the control points are on a straight line the HBC will also be a straight line.

Proof: By Lemma 2, HBC lies within the convex hull of its control points. Therefore, when all the control points are on a straight line, HBC will be a straight line.

Computational complexity analysis: The HBC model has the same order of complexity as the original BC. The BC in (1) requires $O(N)$ iterations for an Nth order curve. Similarly for HBC as summarised in Algorithm 1: Step 2 takes $O(N)$ time as it is actually BC point generation, while Step 3 also takes $O(N)$ since it is a linear search among the distances from the edges. For the next part of the algorithm, either Steps 4-8 or Steps 10-13 are executed, which both take constant time, irrespective of the degree of the curve. Thus, the overall complexity of HBC is $O(N)$, which is the same of the classical BC.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

While Figure 4 is showing some examples of HBC compared with the classical Bezier curve for some hypothetical control point sets, HBC can be of better use in real applications. Next two experimental examples of shape description are presented to support this statement.

Before analysing the experimental results, some theoretical context will be provided. Cubic Bezier curves have been used for shape description [3] using an a *priori* number of curve segments (segment rate-SR) each having the same number of shape points. It means the entire shape is divided into SR number of segments, and each segment described by a single cubic Bezier curve. The control points are selected from the shape points, with the points for a curve segment, approximating the shape

$P = \{p_1, p_2, \ldots, p_M\}$, between p_i to p_{i+m} (where $m = \frac{M}{SR}$) are:-

$$v_1 = p_i; \quad v_2 = p_{i+\left\lceil \frac{m}{4} \right\rceil}; \quad v_3 = p_{i+\left\lceil \frac{3 \times m}{4} \right\rceil}; \quad v_4 = p_{i+m} \qquad (2)$$

In the experiments, the same control points generated in (2) were used by both the Bezier curve and HBC model for two different natural images [3]. The minimum gap between a point on the contour and the approximated shape represents the shape distortion at that contour point.
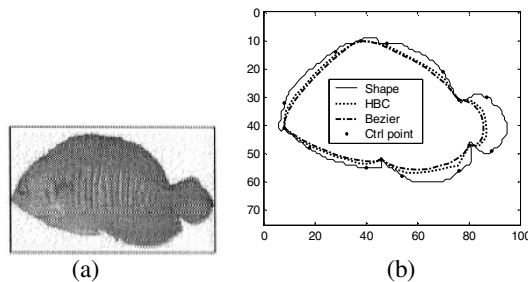

Figure 5. a) Fish image; b) Shape described by 5 segments.

For numerical analysis purposes, the popular and widely used *shape distortion measurement* metrics [12] $D_{\max}$ and D_{ovl} were used for *class one* (peak) and *class two* (mean-squared) distortion values respectively with the edge distortion measured by using [13]. In Figure 5(b), shape descriptions of the fish object are shown for a fixed segment rate (SR=5). The Bezier curve produced a maximum distortion of 9.5 *pel* at the tail portion of the object, while the HBC model generated a corresponding maximum distortion of 8.2 *pel*. When the entire shape was considered, HBC model provided a better shape description in comparison to the Bezier curve, as confirmed by the numerical results in Table 1, for the maximum and overall (Ovl) distortion values, for various segment numbers. For instance, with 5 segments the Bezier curve and HBC representations had overall distortions of 14.1 and 9.1 *pel²* respectively. This improvement highlights the fact that HBC considered localised information in addition to the inherent global information of the Bezier curve.

Table 1. Distortion (units: max distortion = *pel*; Ovl distortion = *pel²*) in shape representation.

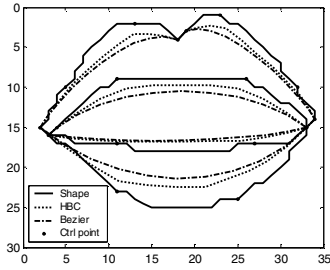|		SR = 5		SR = 6		SR = 7		SR = 8	
		Max	Ovl	Max	Ovl	Max	Ovl	Max	Ovl
Fish	BC	9.5	14.1	6.3	7.1	6.2	4.08	4.7	2.9
	HBC	8.2	9.1	6.0	5.0	5.7	2.6	4.2	1.9

Figure 6: Lip shape description.

The next series of experiments was performed upon the *lip* shape used in [4], which was divided into five segments using the curvature information of the shape. For each segment, the control points were generated using (2) with m being the total number of shape points for a particular segment. Figure 6 shows the comparative results between BC and HBC for this shape, with HBC again generating better results than the classical BC. The *class one* distortion values for HBC and BC were respectively 2.8 and 3.9 *pel*, while the corresponding *class two* distortions were 1.8 and 3.6 pel^2.
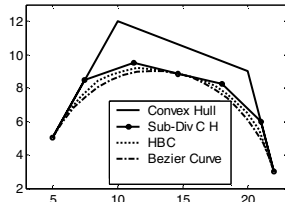

Figure 7. HBC curve and Bezier curve using Bezier subdivisions.

A further series of experiments was conducted to illustrate the potential of HBC model using the *midpoint subdivision* algorithm [8] of Bezier curve, with the results shown in Figure 7. The HBC was drawn using the resultant control points generated in [8]. Again the HBC model generated better curves compared with the original Bezier curve using the same subdivided control points set.

Table 2. Area coverage (in pel^2) by curve of different degrees.

Degree of the curve	Control polygon	HBC	Bezier Curve
2	17.5	14.53	11.55
3	21.0	18.03	15.14
4	14.5	12.56	10.52
5	387.5	277.06	224.20
10	280.5	236.81	213.84

The final experiments tested the performance of the HBC model for higher degree curves. Different control point sets were used for different degree curves; however, for a particular degree the same set was used for both HBC and BC. The curves were closed by joining the first and last curve points and the total area covered by the curves used as a comparison metric, with the results given in Table 2. This shows that HBC consistently covered a greater area for all degree curves compared with the BC, with the CP being the upper bound, so confirming that HBC more closely follows the CP than the Bezier curve.

V. CONCLUSIONS

While the Bezier curve is a well established tool for a wide range of applications, its main drawback is that it does not incorporate localised information. This paper has focused upon bridging this by integrating localised information into the BC framework. *A novel half-way shifting Bezier curve* (HBC*) model* has been presented and mathematically proven that it retains the core properties of the BC. The qualitative and quantitative results using different shapes and also the examples in the theoretical analysis confirmed that HBC model exhibited considerable improvement over the Bezier curve in terms of shape distortion performance and kept the same computational complexity order.

REFERENCES

[1] M. Sarfraz, and M.A. Khan, "Automatic outline capture of Arabic fonts", *Information Sciences*, Elsevier Science Inc., 2002, pp. 269-281.

[2] H.-M. Yang, J.-J. Lu, and H.-J. Lee, "A Bezier curve-based approach to shape description for Chinese calligraphy characters", *Proceedings of 6th International Conf. on Doc. Analysis and Recog.*, 2001, pp. 276-280.

[3] L. Cinque, S. Levialdi, and A. Malizia, "Shape description using cubic polynomial Bezier curves", *Pattern Recognition Letters*, Elsevier Science Inc., 1998, pp. 821-828.

[4] I. Shdaifat, R. Grigat, and D. Langmann, "Active shape lip modeling" *Proceedings of International Conference on Image Processing,* 2003, pp. 875-878.

[5] L.D. Soares, and F. Pereira, "Spatial shape error concealment for object-based image and video coding", *IEEE Trans. on Image Proc.*, no.13(4), 2004, pp.586-599.

[6] R. Zhang, and G. Wang, "Some estimates of the height of rational Bernstein-Bezier triangular surfaces", *Proceedings of the Geometric Modeling and Processing*, 2004, pp.79-84.

[7] F.S. Hill Jr., *Computer Graphics*, Prentice Hall, Englewood Cliffs, 1990.

[8] R.H. Bartels, J.C. Beatty, and B.A. Barsky, *An Introduction to Splines for use in Computer Graphics & Geometric Modeling*, Morgan Kaufmann Publishers, 1987.

[9] L.M. Kocić, "Modification of Bézier curves and surfaces by degree-elevation technique", *Computer-Aided Design*, vol. 23, no. 10, 1991, pp. 692-699.

[10] J.M. Lane, and R.F. Riesenfeld, "A theoretical development for the computer generation of piecewise polynomial surfaces", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no.1, 1980, pp. 35-46.

[11] M. Hosaka, and F. Kimura, "A theory and method for free form shape construction", *Journal of Information Processing*, vol.3, no.3, 1980, pp. 140-151.

[12] G.M. Schuster, and A.K. Katsaggelos, *Rate-Distortion Based Video Compression-Optimal Video Frame Compression and Object Boundary Encoding*, Kluwer Academic Publishers, 1997.

[13] F.A. Sohel, L.S. Dooley, and G.C. Karmakar, "Accurate distortion measurement for generic shape coding," *Pattern Recognition Letters*, Elsevier Science Inc., in press.