

Quasi-Bezier Curves Integrating Localised Information

Ferdous A. Sohel¹, Gour C. Karmakar, Laurence S. Dooley, and John Arkininstall

ABSTRACT

Bezier curves (BC) have become fundamental tools in many challenging and varied applications, ranging from computer aided geometric design to generic object shape descriptors. A major limitation of the classical Bezier curve however, is that only global information about its control points (CP) is considered, so there can often be a large gap between the curve and its control polygon, leading to large distortion in shape representation. While strategies such as degree elevation, composite BC, refinement and subdivision reduce this gap, they also increase the number of CP and hence bit-rate, and computational complexity. This paper presents novel contributions to BC theory, with the introduction of *quasi-Bezier curves* (QBC), which seamlessly integrate localised CP information into the inherent global Bezier framework, with no increase in either the number of CP or order of computational complexity. QBC crucially retains the core properties of the classical BC, such as geometric continuity and affine invariance, and can be embedded into the vertex-based shape coding and shape descriptor framework to enhance rate-distortion performance. The performance of QBC has been empirically tested upon a number of natural and synthetically shaped objects, with both qualitative and quantitative results confirming its consistently superior approximation performance in comparison with both the classical BC and other established BC-based shape descriptor methods.

¹ Corresponding author: E-mail: Ferdous.Sohel@infotech.monash.edu.au; Ferdous.Sohel@ieee.org; Tel.: +61-3-990-26133; Fax: +61-3-990-26842. Mailing address:- GSIT, Monash University, Churchill, Victoria – 3842, Australia.

Index Terms – Vertex-based shape coding, image processing, video processing, and Bezier curve.

I. INTRODUCTION

Bezier curves (BC) were independently developed by P. de Casteljaou and P. E. Bézier, and have subsequently been applied to a wide range of computer-aided design applications. While their origin can be traced back to the design of car body shapes in the automobile industries, their usage is no longer confined to this field. Indeed, their robustness in curve representation means BC now pervades many areas of multimedia technology, including shape description of characters [1] and objects [2], shape coding and error concealment for MPEG-4 coded objects [3].

The classical BC is defined by a set of *control points* (CP) which, when joined together, form the *control polygon*, with the number and orientation of the vectors connecting the CP governing the shape of the curve. One limitation of BC theory is that only global information about the CP is considered [4] because each BC point is produced by blending all CP. As a consequence, a large gap can occur between the curve and its control polygon, leading to high distortions in shape approximation.

A number of approaches have been proposed to reduce this gap, including *degree elevation* [5], *Composite Bezier curve* [6] and *refinement* and *subdivision* [7]-[8]. *Degree elevation* forms a curve with an increased number of CP by one in each iteration, though all of these, except the two end-points, need to be recalculated, so the computational overhead is commensurately increased. Moreover, a higher degree curve is always computationally expensive than a lower degree curve. *Composite Bezier curves* (CBC) [6] model a shape by dividing it into multiple

segments, each of which is defined by a simple BC. The main drawback of CBC is however, that the number of segments required increases with shape complexity, as segment division is not very strategic. This was the primary driver behind the evolution of the *refinement* and *subdivision* techniques [6]. In the latter, the BC is arbitrarily subdivided into two [8], with a new set of CP being calculated from the initial CP set for each part, that is guaranteed closer to the curve. In the special case, where the two lengths are equal, the technique is referred to as *midpoint subdivision* [7]. These algorithms however, increase the number of curve segments and thereby the number of CP. Indeed, *arbitrary subdivision* and CBC double the number of curve segments, which commensurately increases the number of CP, meaning a high bit-rate encoding overhead is required.

While these techniques successfully reduce the distance between a Bezier approximation and its control polygon, they also increase the number of CP, leading to a higher coding or descriptor length. This was the motivation behind this research, namely to reduce the gap between the curve and its control polygon without increasing the number of CP. Such an objective mandates an augmentation to the fundamental theoretical basis of the BC, which this paper addresses by introducing two novel BC enhancements², namely *quasi-Bezier curves (QBC)*, theory which considers local information within the classical BC framework, without any increase in either the number of CP or computational complexity incurred. It is especially noteworthy that QBC can be seamlessly integrated into all Bezier *variants* including the aforementioned *degree elevation*, *composite* and *subdivision* techniques, while concomitantly retaining all the central properties of the BC.

² The preliminary idea behind this work was presented at IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2005) [9].

Moreover, the B-spline is a generalisation of the Bezier curve [4], for example, quadratic B-splines are piecewise Bezier curves. Quadratic B-splines have been efficiently used in the classical vertex-based *operational-rate-distortion* (ORD) shape coding framework [10]-[15]. The performance of QBC as a generic shape descriptor is rigorously analysed for a number of natural and synthetic shapes, as well as embedding it within the classical vertex-based ORD optimal shape coding framework. Experimental results corroborate the theoretical basis of QBC by consistently providing superior shape approximations in comparison with the classical BC and its major variants.

The remainder of the paper is organised as follows: Section II presents a short overview of classical BC theory and the classical vertex-based ORD optimal shape coding framework, while Section III introduces the mathematical foundations of the new QBC together with appropriate proofs that all the core properties of the BC are retained and also the model of QBC based ORD optimal vertex-based shape coding framework. Section IV provides a comprehensive analysis of the improved performance of QBC, with some conclusions drawn in Section V.

II. OVERVIEW OF THE CLASSICAL BEZIER CURVE THEORY AND THE VERTEX-BASED ORD OPTIMAL SHAPE CODING FRAMEWORK

This section presents a short overview of the Bezier curve theory followed by the B-spline based shape coding framework.

II A. The Bezier Curve Theory

The classical BC is a linearly-weighted interpolation which exhibits the *variation diminishing* property of the edges of a generated polygon. Commencing with a set of points which form the initial (control) polygon, this property relates to the fact that during each iteration a particular u ,

the number of interpolated points decreases by one and ends when the final point is generated. Hence, as u varies, it produces a segment of curve in form of a blending polynomial. As the iteration goes, the levels of intermediate points increase, the degree of each blending polynomial increases, and the number of curve segments reduces. After the last iteration, a single blending polynomial of degree N i.e., the N degree BC, is produced, from the set of $N+1$ CP. The Casteljaou form of the BC for an ordered set of CP $P = \{p_0, p_1, \dots, p_N\}$ is iteratively defined as:

$$p_i^r(u) = \begin{cases} i^{\text{th}} \text{ member of } P, p_i; & \text{if } r=0; \\ (1-u)p_i^{r-1}(u) + up_{i+1}^{r-1}(u); & r=1, \dots, N; \quad i=0, \dots, N-r; \quad 0 \leq u \leq 1 \end{cases} \quad (1)$$

where u is the interpolation weight, which is determined by the number of points on the BC. The final iteration $p_0^N(u)$ is the Bezier curve of P .

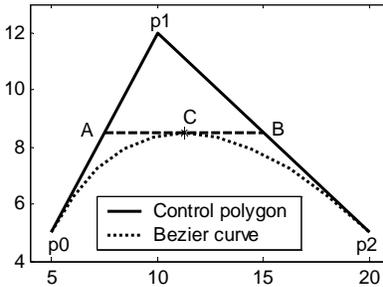


Figure 1: A quadratic BC example illustrating the gap.

Figure 1 shows a quadratic BC produced using CP p_0, p_1 and p_2 . The large gap between the BC approximation and its control polygon represents a substantial shape distortion (error) caused by the fundamental BC limitation of considering only global CP information. If for a particular value $u = 0.5$, points A and B are generated by (1), then the inner area of $\Delta p_1 B$ is never reached and the final BC point C will be generated on line AB . This inadequacy has led to many variants of

the classical BC being proposed [5]-[8], all attempting to some degree, to reduce this gap however at the cost of increasing the number of CP hence the bit requirement. To minimise the gap between the curve and its control polygon without increasing the number of CP, it is required to move the Bezier point inside ΔAp_1B .

II B. The Vertex-Based ORD Optimal Shape Coding Framework

In [10], a rigorous review of shape coding algorithms was presented with the conclusion that the classical vertex-based shape coding framework was optimal in an ORD sense. With both polygonal and quadratic B-spline based shape encoding strategies being deployed in [10], these have become the kernel for several other shape coding algorithms [11]-[16] within the ORD framework. However, being higher order curve, the B-spline based algorithms require lower bit-rate than those of the polygon based algorithms for the same experimental set up and the same test shapes. The general aim of all these algorithms is that for some prescribed distortion, a shape contour is optimally encoded in terms of the number of bits, by selecting a set of CP that incurs the lowest bit rate and vice versa. To define mathematically, let boundary $B = \{b_0, b_1, \dots, b_{N_B-1}\}$ is an ordered set of shape points, where N_B is the total number of points. $S = \{s_0, s_1, \dots, s_{N_S-1}\}$ is an ordered set of CP used to approximate B , where N_S is the total number of CP and $S \subseteq C$, where C is the ordered set of vertices in the admissible control point band (ACB), the source of potential CP. For a representative example, the ORD B-spline based shape coding algorithm for determining the optimal S for boundary B within RD constraints is formalised in Algorithm 1 with the details can be found in [10], [11], [14] and [15].

Algorithm 1: The B-spline based ORD optimal shape coding algorithm.

Inputs: B – the boundary; T_{\max} and T_{\min} – the peak admissible distortion bounds.

Variables: State $(c_{i,m}, c_{j,n})$ refers to encoding up to $c_{j,n}$ from b_0 with $c_{i,m}$ immediately preceding $c_{j,n}$; $MinRate(c_{i,m}, c_{j,n})$ – current minimum bit-rate required to encode $(c_{j,n}, c_{k,l})$; $pred(c_{j,n}, c_{k,l})$ – predecessor of $(c_{j,n}, c_{k,l})$ that maintains the $MinRate(c_{i,m}, c_{j,n})$; $N[i]$ – the number of vertices in C associated to b_i .

Output: S – the ordered set of CP approximating B .

1. Determine the admissible distortion $T[i]$ for $0 < i < N_B - 1$ using T_{\max} and T_{\min} ;
2. Form the ACB C using width $w[i]$ for $0 < i < N_B - 1$ according to [15];
3. Initialise $MinRate(c_{0,0}, c_{1,0})$ with the total bits required to encode the first boundary point b_0 ;
4. Set $MinRate(c_{i,m}, c_{j,n})$, $0 < i < N_B - 1$, $0 \leq m < N[i]$, $i < j < N_B$, $0 \leq n < N[j]$ to infinity;
5. FOR each vertex $c_{i,m}$, $0 \leq i < N_B - 2$, $0 \leq m < N[i]$
6. FOR each vertex $c_{j,n}$, $i < j < N_B - 1$, $0 \leq n < N[j]$
7. FOR each vertex $c_{k,l}$, $j < k < N_B$, $0 \leq l < N[k]$
8. Determine the B-spline curve Q_{BS} using CP set $(c_{i,m}, c_{j,n}, c_{k,l})$;
9. Check the admissible distortion using (Q_{BS}, T) ;
10. IF the admissible distortion is maintained
11. Determine bit-rate $r(c_{i,m}, c_{j,n}, c_{k,l})$ and weight $w(c_{i,m}, c_{j,n}, c_{k,l})$;
12. IF $((MinRate(c_{i,m}, c_{j,n}) + w(c_{i,m}, c_{j,n}, c_{k,l})) < MinRate(c_{j,n}, c_{k,l}))$ THEN
13. $MinRate(c_{j,n}, c_{k,l}) = MinRate(c_{i,m}, c_{j,n}) + w(c_{i,m}, c_{j,n}, c_{k,l})$;
14. $pred(c_{j,n}, c_{k,l}) = c_{i,m}$;
15. Obtain S with properly indexed values from $pred$.

From (1), polynomial form of a quadratic Bezier curve Q_{BC} for the ordered CP set $\{p_0, p_1, p_2\}$ is obtained as:

$$Q_{BC}((p_0, p_1, p_2), u) = (1-u)^2 p_0 + 2 \cdot u \cdot (1-u) p_1 + u^2 p_2, \quad 0 \leq u \leq 1 \quad (2)$$

Again, the polynomial form of a quadratic B-spline curve segment Q_{BS} for the ordered CP set $\{p_0, p_1, p_2\}$ is defined as:

$$Q_{BS}((p_0, p_1, p_2), u) = \frac{1}{2} \cdot (1-u)^2 p_0 + (-u^2 + u + 0.5) p_1 + \frac{1}{2} \cdot u^2 p_2, \quad 0 \leq u \leq 1 \quad (3)$$

From (2) and (3):

$$Q_{BS}((p_0, p_1, p_2), u) \equiv Q_{BC}\left(\left(\frac{p_0+p_1}{2}, p_1, \frac{p_1+p_2}{2}\right), u\right), 0 \leq u \leq 1 \quad (4)$$

Equation (4) confirms that a Bezier curve can be represented in B-spline form and a quadratic B-spline curve is piecewise Bezier where the end CP for the BC are the midpoints of the control polygon of the B-spline curve, which can also be illustrated by the example in Figure 2.

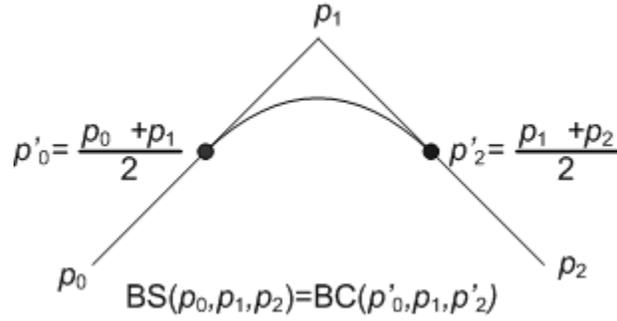


Figure 2: The relationship between Bezier and B-spline curves.

Therefore, in Step 8 of Algorithm 1, Bezier curves can be equivalently used instead of the B-splines, with proper CP of course. This leads the way to embed the proposed QBC within the existing B-spline based ORD optimal shape coding framework to improve the rate-distortion performance.

Section III introduces novel strategies namely the *quasi-Bezier curves (QBC)* that reduce the gap between the classical Bezier curve and its control polygon, and also the mechanism to embed QBC into B-spline based shape coding framework.

III. QUASI-BEZIER CURVE THEORY INTEGRATING LOCAL INFORMATION

In this section the quasi BC theory is firstly developed, including a series of formal proofs to confirm all core properties of the classical BC are upheld in the new representation. A short delineation is then presented as to how QBC may be seamlessly embedded in the *operational-rate-distortion* (ORD) optimal vertex-based shape coding framework [10] to enhance its *rate distortion* (RD) performance.

III A. The Quasi-Bezier Curve

Enhancement of the *quadratic* BC is initially presented, before the theory is generalised for any arbitrary degree N . To reduce the gap illustrated in the example in Figure 1, curve points need to be generated inside the triangular area. For this purpose, the *centre of gravity* (CoG) G of $\Delta p_1 B$ in Figure 3(a) can be exploited, by shifting a particular point generated by the classical BC, towards it. If this point, for a particular u is moved directly to the CoG, three major problems arise: 1) End-point interpolation, which is one of the most important BC properties is no longer upheld, since for the extreme values $u=0$ or $u=1$, the CoG can never be an end CP. For $u=0$, the corresponding triangle will be the line $p_0 p_1$ in Figure 3(a), so the CoG of the triangle is not at point p_0 , rather specifically, it will be at the midpoint of line $p_0 p_1$. As a consequence, a point shifted directly to the CoG violates the end-point interpolation for the first CP. Similarly for $u=1$, the CoG will be on line $p_1 p_2$ but, not at point p_2 and so again direct shifting to the CoG invalidates this important BC property. 2) The length of the generated curve will be shorter than the BC since the CoG for various u values are confined to within a small region. As it is just discussed, that shifting directly to the CoG does not uphold the end-point interpolation property rather it begins the curve at the midpoint of line $p_0 p_1$ and ends at the middle of $p_1 p_2$. Therefore,

the resulting curve is shorter than the BC. 3) The resulting curve also may not be smooth, since the curve connecting the corresponding CoG may form an unwanted zigzag.

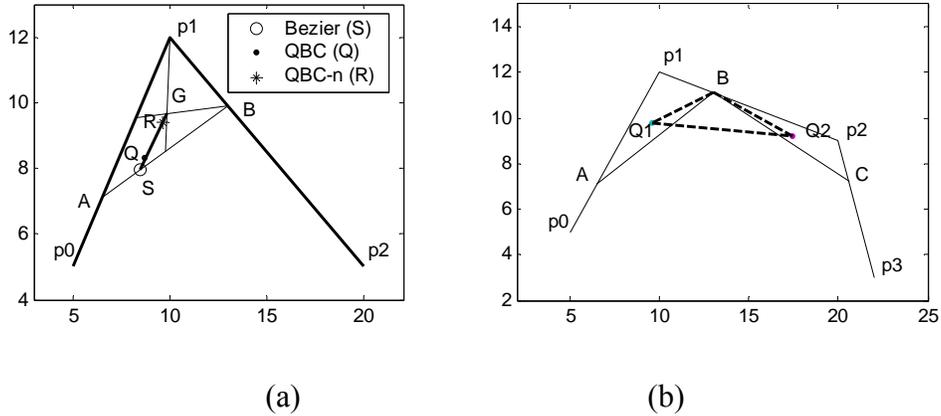


Figure 3: QBC examples for a) Quadratic; b) Cubic.

To obtain a smooth curve, all generated points need to be regularly distributed over the entire curve which is again controlled by the values of u in accordance with the direction guided by the CP. For example, there is regularity between a constant increment du in the parameter domain u and the corresponding increment in arc length, say dl on BC. It is noteworthy to mention that it is not necessarily constant. To ensure a large and smooth curve, it is essential to maintain the end-point interpolation property as well as the regular distribution of the generated points. This can be achieved by generating the points using a suitably weighted linear interpolation between the BC point and its CoG. If the original BC ratio $u:(1-u)$ is used as the interpolation weighting factor to shift a BC point, the end-point interpolation property for the last CP will not be satisfied, since for $u=1$ the shifting ratio is $1:0$ and as a result the Bezier point at p_2 will be shifted to the corresponding CoG which will be at the midpoint of the line p_1p_2 . However, as will be proven in *Lemma 1*, the ratio $(u(1-u)):(1-u(1-u))$ for a BC point and its CoG guarantees the end-point

interpolation criterion and concomitantly ensures a smooth curve, because, the generated points are regularly dispersed over the entire curve due to the values of the shifting parameters in the direction guided by the CP set. Moreover, $u(u-1)$ is the lowest order polynomial that maintains the required shifting ratio for the end-point interpolation property. This strategy of shifting a BC point using the above ratio is the basis of the new *quasi Bezier curves* (QBC), which is pictorially depicted in Figure 3(a), where S is the BC point for $u = 0.3$ and G is the CoG of ΔAp_1B . In QBC, S moves to any point on line SG and the shifted point Q always segments line SG such that $SQ:QG = (u(1-u)):(1-u(1-u))$. For a particular $u = 0.3$ this ratio is $SQ:QG = 0.21:0.79$, and the quadratic QBC can be formulated as:

$$p(u) = \frac{1}{3}(1-u)^2(3u^2 - 2u + 3)p_0 + \frac{2}{3}u(1-u)(3u^2 - 3u + 4)p_1 + \frac{1}{3}u^2(3u^2 - 4u + 4)p_2 \quad ; \quad 0 \leq u \leq 1 \quad (5)$$

where $\{p_0, p_1, p_2\}$ is the set of CP.

The cubic QBC is shown in Figure 3(b), where points Q_1 and Q_2 are generated using the quadratic QBC described above for CP sets $\{p_0, p_1, p_2\}$ and $\{p_1, p_2, p_3\}$ respectively. A new quadratic CP set is then formed with $\{Q_1, B, Q_2\}$, where B is produced by the weighted $(u:(1-u))$ interpolation of successive initial CP p_1 and p_2 during the BC generation process. B is so chosen because of its influence on both Q_1 and Q_2 . The final curve point is generated by quadratic QBC with CP $\{Q_1, B, Q_2\}$.

The quadratic QBC can be iteratively extended to an arbitrary degree N by using two consecutively generated quadratic QBC points in the previous iteration, together with a polygon point between them, to form another quadratic QBC, until it converges to a single point for each value of u , thereby generating the entire QBC polynomial for the values of u in the range.

Depending on the iteration, the polygon point is selected either from the CP or from Bezier generated intermediate points during interpolation. This polygon point will actually be the common point which has been involved in generation of both these QBC points. This is formulated as:

$$p_i^r(u) = \begin{cases} \frac{1}{3}(1-u)^2(3u^2-2u+3)p_i + \frac{2}{3}u(1-u)(3u^2-3u+4)p_{i+1} + \frac{1}{3}u^2(3u^2-4u+4)p_{i+2}; & 0 \leq i \leq N-2; r=0; 0 \leq u \leq 1 \\ \frac{1}{3}(1-u)^2(3u^2-2u+3)p_i^{r-1}(u) + \frac{2}{3}u(1-u)(3u^2-3u+4)s_i^r(u) + \frac{1}{3}u^2(3u^2-4u+4)p_{i+1}^{r-1}(u); & 1 \leq r \leq N-2; 0 \leq i \leq N-r-2; 0 \leq u \leq 1 \end{cases} \quad (6)$$

$$s_i^r(u) = \begin{cases} u \cdot p_{i+2} + (1-u) \cdot p_{i+1}; & \text{if } r = 1 \\ p_{i+2}; & \text{if } r = 2 \\ s_{i+1}^{r-2}(u); & \text{else} \end{cases} \quad (7)$$

The first and last of the three CP required for a quadratic QBC are chosen from the QBC points generated in the previous iteration, while the polygon point $s_i^r(u)$ is selected from either the initial CP or the interpolation points according to (7), so the final generation $p_0^{N-2}(u)$ is the resulting QBC.

As $0 \leq u \leq 1$, the value of $u(1-u)$ in QBC is generally small and consequently, the corresponding displacement distance of a BC point towards the CoG is also small. To create a larger displacement, so further reducing the gap, a normalised shifting parameter can be introduced, which is normalised with respect to the value of the following expression:

$$\max_{0 \leq u_j \leq 1} \{u_j(1-u_j)\} \quad (8)$$

which is 0.25 with $u_j = 0.5$. The normalised shifting parameter thus becomes $u(1-u)/0.25 : (1-u(1-u)/0.25)$. This ensures a smooth curve since the generated points are well distributed over the entire curve and also that the gap between the curve and control polygon is reduced further. Note, when $u = 0.5$, S shifts to the CoG of the triangle, which is the maximum possible shift within this framework, while concomitantly maintaining the end-point interpolation and the smoothness properties of the classical BC. When the normalised parameter is used, QBC is referred to as QBC- n , so in Figure 3(a), for $u = 0.3$. R is the QBC- n point, where the shifting parameter is $SR : RG = 0.84 : 0.16$. Applying the same rationale as for QBC, the generic QBC- n can be formally expressed as:

$$p_i^r(u) = \begin{cases} \frac{1}{0.75}(1-u)^2(3u^2 - 2u + 0.75)p_i + \frac{2}{0.75}u(1-u)(3u^2 - 3u + 1.75)p_{i+1} + \frac{1}{0.75}u^2(3u^2 - 4u + 1.75)p_{i+2}; & 0 \leq i \leq N-2; r=0; 0 \leq u \leq 1 \\ \frac{1}{0.75}(1-u)^2(3u^2 - 2u + 0.75)p_i^{r-1}(u) + \frac{2}{0.75}u(1-u)(3u^2 - 3u + 1.75)s_i^r(u) + \frac{1}{0.75}u^2(3u^2 - 4u + 1.75)p_{i+1}^{r-1}(u); & 1 \leq r \leq N-2; 0 \leq i \leq N-r-2; 0 \leq u \leq 1 \end{cases} \quad (9)$$

where the polygon point $s_i^r(u)$ is selected from either the initial CP or the interpolation point as found by (7), so $p_0^{N-2}(u)$ is the resulting QBC- n .

III B. Properties of Quasi-Bezier Curves

As the foundations of both QBC frameworks are underpinned by classical BC theory, all the core properties [4] are preserved. The following examines some of these, where without loss of generality; all proofs are provided for QBC, though they are equally applicable to QBC- n .

Lemma 1: End-point interpolation: The QBC always interpolates its first and last CP.

Proof: Any Bezier curve interpolates its end points [4] for the starting ($u = 0$) and end ($u = 1$) CP. QBC makes a parametric shift of the classical BC point towards the CoG by the

ratio $(u(1-u)):(1-u(1-u))$. For both $u=0$ and $u=1$, $(u(1-u)):(1-u(1-u))=0:1$, which means the end-points are shift invariant in QBC, i.e. the end-points of QBC and BC are the same. This is also evidenced in (5) and (6), i.e. $p(0) = p_0$ and $p(1) = p_N$.

Lemma 2: Convex Hull Property: QBC always lies within the convex hull of its CP.

Proof: Suppose a curve is defined as $p(u) = \sum_{0 \leq k \leq N} \alpha_k(u) p_k$, $\alpha_k(u) \geq 0$ where p_k is the k^{th} CP. If

$\sum_{0 \leq k \leq N} \alpha_k(u) = 1, \forall u$, the curve $p(u)$ lies within its convex hull [4]. QBC in (6) can be expressed in

the form $p(u) = \sum_{0 \leq k \leq N} \alpha_k(u) p_k, \forall u$. It follows from (5) that $\sum_{0 \leq k \leq N} \alpha_k(u) = 1, \forall u$, so the quadratic QBC

lies within the convex hull of its control polygon, i.e. within the corresponding enclosed triangular area. It follows from (7) that $s_i^r(u)$ always lies on the control polygon, so any QBC point will always lie within the corresponding triangle and QBC therefore must lie within the convex hull of the CP.

Lemma 3: Affine Invariance: QBC is invariant under affine transformations.

Proof: A BC is affine invariant if the curve drawn with affine transformed CP is the same as the entire affine transformed curve with the same parameters, i.e.

$$\sum_{k=0}^N (R \cdot p_k + t) \alpha_k(u) = \sum_{k=0}^N R \cdot p_k \alpha_k(u) + t \text{ where } R \text{ is a transformation matrix and } t \text{ is an offset vector}$$

[4]. QBC with affine transformed CP can be expressed as:-

$$\sum_{k=0}^N (R \cdot p_k + t) \alpha_k(u) = \sum_{k=0}^N R \cdot p_k \alpha_k(u) + \sum_{k=0}^N t \alpha_k(u) = \sum_{k=0}^N R \cdot p_k \alpha_k(u) + t, \text{ since from Lemma 2 } \sum_{0 \leq k \leq N} \alpha_k(u) = 1;$$

therefore QBC is affine invariant.

Computational complexity analysis: QBC has the same order of computational complexity as the classical BC, since for a N degree curve, QBC in (6) requires $(N - 2)$ iterations to locate the final curve point for each value of u , as it started with a quadratic curve. In contrast, the classical BC in (1) takes N iterations, so the computational order in both cases is $O(N)$ iterations.

III C. QBC in the ORD Optimal vertex-based shape coding framework:

Katsaggelos et al. [10] proposed the framework for ORD optimal vertex-based shape coding using B-splines and polygons, which has subsequently been deployed in [11], [12] and extended in [14], [15]. It is already shown in Section IIB that quadratic Bezier curves can be equivalently used instead of the B-splines. Therefore, to improve the rate-distortion performance of these algorithms a series of conjoint QBC curves can be used to approximate the shape. Though since QBC possesses an end-point interpolation property (*Lemma 1*) similar to the Bezier, to ensure a series of conjoint curves so that adjacent curves have some common CP, the points are to be coordinated in a similar fashion of (4), where two QBC curves abut. Therefore, as shown in (10) the QBC will replace the B-spline based framework:

$$Q_{BS}((p_0, p_1, p_2), u) \longleftrightarrow Q_{QBC}\left(\left(\frac{p_0+p_1}{2}, p_1, \frac{p_1+p_2}{2}\right), u\right), 0 \leq u \leq 1 \quad (10)$$

where \longleftrightarrow denotes that the right-hand-side curve will replace the left-hand-side curve. For a series curves using the CP set $S = \{s_0, s_1, \dots, s_{N_s-1}\}$, the i^{th} and $(i+1)^{th}$ curve segments are respectively defined, within the range $0 \leq u \leq 1$, as:

$$Q_i(u) = Q_{BS_i}((s_{i-1}, s_i, s_{i+1}), u) \longleftrightarrow Q_{QBC_i}\left(\left(\frac{s_{i-1}+s_i}{2}, s_i, \frac{s_i+s_{i+1}}{2}\right), u\right) \quad (11)$$

$$Q_{i+1}(u) = Q_{BS_{i+1}}((s_i, s_{i+1}, s_{i+2}), u) \longleftrightarrow Q_{QBC_{i+1}}\left(\left(\frac{s_i+s_{i+1}}{2}, s_{i+1}, \frac{s_{i+1}+s_{i+2}}{2}\right), u\right) \quad (12)$$

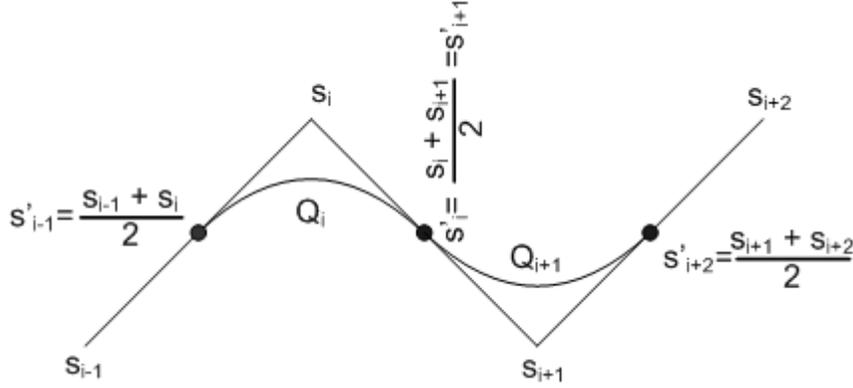


Figure 4: Illustration for series of conjoint QBC curves with in a quadratic B-spline framework.

These are also pictorially shown in Figure 4. It will now be proven in Lemma 4 that the resulting series of QBC curves maintain the geometric continuity at the knot (where two consecutive curves abut) points which is crucially an important property for the parametric curves when they are dealt with shapes [17].

Lemma 4: Geometric Continuity: The QBC curves series produced in accordance with (11) and (12) maintains the geometric continuity at the knot points.

Proof: From (11) and (12) for QBC curves using parameter u in (5), $Q_i(1) = -Q_{i+1}(0)$ which means the consecutive curve segments join at the end points and form a series of curves. Now, if $Q'_i(u)$ denotes the derivative of $Q_i(u)$ with respect to u , $Q'_i(1) = -Q'_{i+1}(0)$ which means the conjoint curves maintain the geometric continuity at the knots.

Lemma 5: Bounds for the ACB Width: Step 2 of Algorithm 1 determines the width $W[j]$ of the ACB for each boundary point b_i . It was proven in [15] that for B-spline based encoding:

$$W[j] \leq \min \left\{ \frac{3\delta + 4T_{\max} + 2T[j]}{6}, \frac{\rho\sqrt{2}}{4} \right\} + T[j] \quad (13)$$

where δ and ρ are respectively the longest chord length of the boundary and the largest run-length possible for the code employed. These bounds for QBC- n and QBC are respectively:

$$W[j] \leq \min \left\{ \frac{(5\delta + 6T_{\max} + 4T[j])}{20}, \frac{\rho\sqrt{2}}{6} \right\} + T[j] \quad (14)$$

$$W[j] \leq \min \left\{ \frac{11}{37} \cdot \frac{(37\delta + 48T_{\max} + 26T[j])}{26}, \frac{11\rho\sqrt{2}}{48} \right\} + T[j] \quad (15)$$

Proof: Figure 5(a) shows a uniform quadratic parametric curve (BC, QBC or QBC- n) with the ordered CP set $\{p'_0, p_1, p'_2\}$, with h being the minimum distance of the middle CP p_1 from the curve. It thus follows from [18] that for BC $2h \leq \max\{|p'_0 p_1|, |p_1 p'_2|\}$, where $|p_1 p'_2|$ is the length of edge $p_1 p'_2$. However, for example, in case of QBC- n the curve point is generated by shifting the BC point to the CoG of the triangle for $u = 0.5$ and hence this distance is reduced. This minimum distance becomes the maximum when the end CP p_0 and p_2 coincide and it is $\frac{1}{3}|p'_0 p_1|$, i.e., $3h \leq \max\{|p'_0 p_1|, |p_1 p'_2|\}$. Therefore, $6h \leq \max\{|p_0 p_1|, |p_1 p_2|\}$.

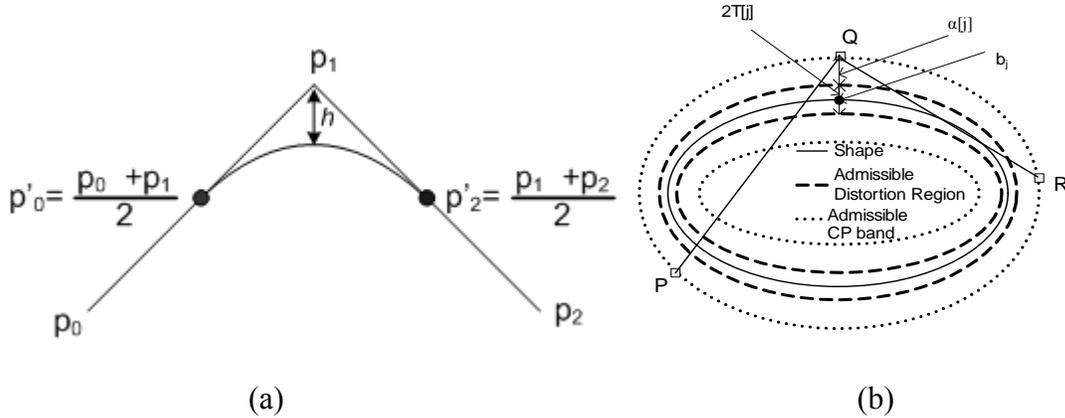


Figure 5: a) Distance between a quadratic BC or QBC curve and its CP, b) Maximal width of the admissible CP band calculation.

In the example shown in Figure 5(b), three CP P, Q & R are employed to encode a shape segment that includes the boundary point b_j which has an admissible distortion $T[j]$. Assuming $PQ \geq QR$, the distance of the QBC- n curve from Q is always $\leq \frac{1}{6}|PQ|$. Let $\alpha[j]$ denotes the difference between the corresponding admissible distortion and width of the admissible CP band, i.e., $W[j] = \alpha[j] + T[j]$. The maximum length of PQ is:

$\delta + T_{\max} + T_{\max} + \alpha_{\max} + \alpha_{\max} = \delta + 2T_{\max} + 2\alpha_{\max}$ where α_{\max} is the maximum value of α . So

$$\delta + 2T_{\max} + 2\alpha_{\max} \geq 6\alpha_{\max}.$$

$$\text{Hence, } \alpha_{\max} \leq \frac{\delta + 2T_{\max}}{4} \quad (16)$$

The corresponding $\alpha[j]$ for boundary point b_j is given by; $6\alpha[j] \leq \delta + T_{\max} + \alpha_{\max} + T[j] + \alpha[j]$.

$$\text{Hence, } \alpha[j] \leq \frac{1}{20}(5\delta + 6T_{\max} + 4T[j]) \quad (17)$$

The encoding strategy adopted can limit the length of an edge since for example, the logarithmic code [11] can support a maximum length of $\rho = 15$, while using a 3-connected chain as the direction encoder, it is able to encode a maximum length of $\rho\sqrt{2}$ (through the diagonal) so that:

$$\alpha[j] \leq \frac{\rho\sqrt{2}}{6} \quad (18)$$

$$\text{From (17) and (18) } \alpha[j] \leq \min\left\{\frac{(5\delta + 6T_{\max} + 4T[j])}{20}, \frac{\rho\sqrt{2}}{6}\right\}.$$

$$\text{Therefore, } W[j] \leq \min\left\{\frac{(5\delta + 6T_{\max} + 4T[j])}{20}, \frac{\rho\sqrt{2}}{6}\right\} + T[j].$$

Again, for QBC $\frac{24}{11} \cdot h \leq \max\{|p'_0 - p_1|, |p_1 - p'_2|\}$ and it can be similarly shown for QBC that

$$W[j] \leq \min\left\{\frac{11}{37} \cdot \frac{(37\delta + 48T_{\max} + 26T[j])}{26}, \frac{11\rho\sqrt{2}}{48}\right\} + T[j].$$

From the widths of the ACB for B-spline and the QBC curves shown in (13), (14) and (15) it is clear that the bound for QBC- n is the minimum while for B-splines it is maximum. The computational complexity of the framework of Algorithm 1 (the loops due to the $N[i]$'s in Steps 5-7) is primarily depends on the number of vertices in the ACB C , if all other parameters remain the same. The number of ACB vertices is directly proportional to the widths of the band. Therefore, a larger distortion bound will enforce a computationally expensive encoder if the admissible distortion and the shape properties are intended to be fully utilised in bit-rate reduction.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the performance of both QBC- n and QBC is initially compared with the classical BC from the perspective of curve representation, by using some hypothetical CP sets, before analysing the results upon a series of popular test shapes from the perspective of both shape descriptor and the enhanced QBC based ORD optimal shape encoding. To quantitatively evaluate the performance of QBC, the widely-used shape distortion measurement metrics [11] were employed. *Class one* distortion measures the maximum distortion D_{max} over the entire curve, while *Class two* distortion provides a measure of the mean-square (MS) distortion D_{ms} of the shape approximation. For the distortion measurement purpose the *accurate distortion measurement* technique [16] was employed.

IV A. Comparative results for QBC, BC and popular Bezier variants

Figure 6 shows a comparison between the classical BC, QBC- n and QBC for varying degrees and orientations. QBC- n is consistently the closest to the control polygon, followed by QBC, with BC providing the poorest approximation, reflecting the fact that both QBC- n and QBC integrate local information about each CP in addition to the inherent global BC information.

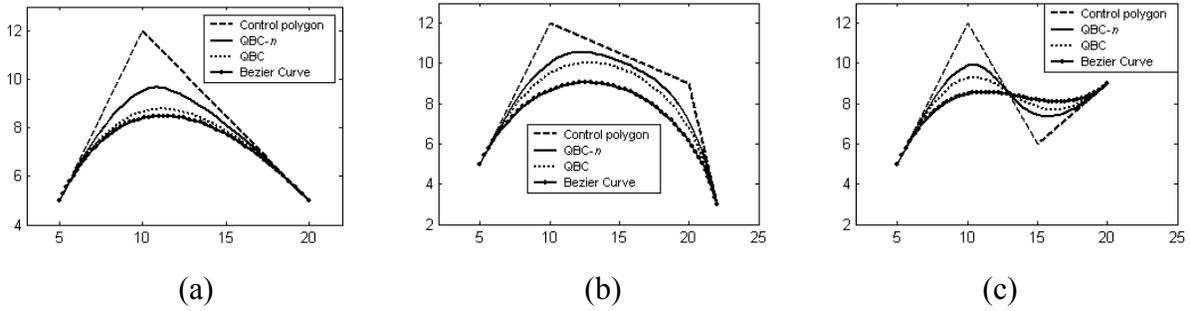


Figure 6: Curves of different degrees and orientations; a) Quadratic; b) Cubic; c) Cubic curves in a different orientation.

A series of experiments were conducted to compare both QBC- n and QBC with the aforementioned *degree elevation* [5] technique. A hypothetical CP set for a quadratic BC was employed for which BC, QBC and QBC- n respectively yielded maximum distortions of 3.6, 3.3 and 2.4 *pel* and *MS* distortions of 4.5, 3.6 and 1.9 *pel*². A new CP set for one degree elevation shown in Figure 5(a) was generated by degree elevation using the same CP set. It is visually apparent that the new control polygon is closer to both QBC- n and QBC than the classical BC, with the maximum and *MS* distortion values in Table 1 confirming the numerical superiority of the QBC approximations over BC.

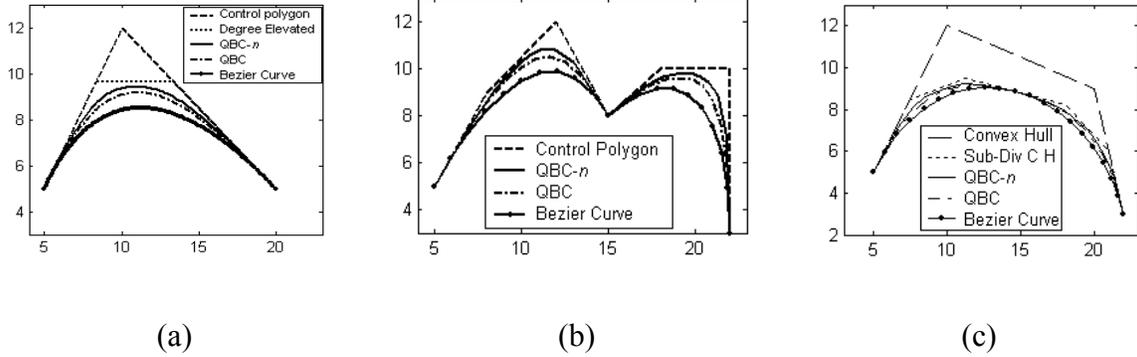


Figure 7: QBC- n , QBC and BC comparison; a) degree elevation; b) composite curve control polygons; c) subdivision (legend Sub-Div C H means sub-division convex hull).

Table 1: QBC- n , QBC and BC distortions for degree elevation.

Degree	<i>BC</i>		<i>QBC</i>		<i>QBC-n</i>	
	$D_{max} (pel)$	$D_{ms} (pel^2)$	$D_{max} (pel)$	$D_{ms} (pel^2)$	$D_{max} (pel)$	$D_{ms} (pel^2)$
2	3.6	4.5	3.3	3.6	2.4	1.9
3 (Elevated)	1.65	1.36	1.1	0.6	0.82	0.34

To test the effectiveness of QBC- n and QBC compared to the classical BC using a CBC approach, another experiment was conducted using the curve in Figure 7(b), which is intuitively divided into two segments. The corresponding control polygons, each defined by four CP are shown in Figure 7(b). The results reveal the control polygon for BC is further away than either QBC- n or QBC, with QBC- n generating the better approximation. The plots in Figure 7(c) illustrate the potential of QBC- n and QBC using the *midpoint subdivision* algorithm [7]. Both curves were drawn using the resultant CP generated by Bezier subdivision and reveal that both enhancements qualitatively generated better curve approximations than BC, using the same subdivided CP set.

IV B. Comparative results as a shape descriptor

Cubic BC was used for shape description in [2], with an *a priori* number of curve segments (segment rate-SR) each with the same number of contour points. The CP for the segments were determined as in [2] and for comparative purposes, the experiments used the same set of CP for the BC, QBC and QBC-*n*.

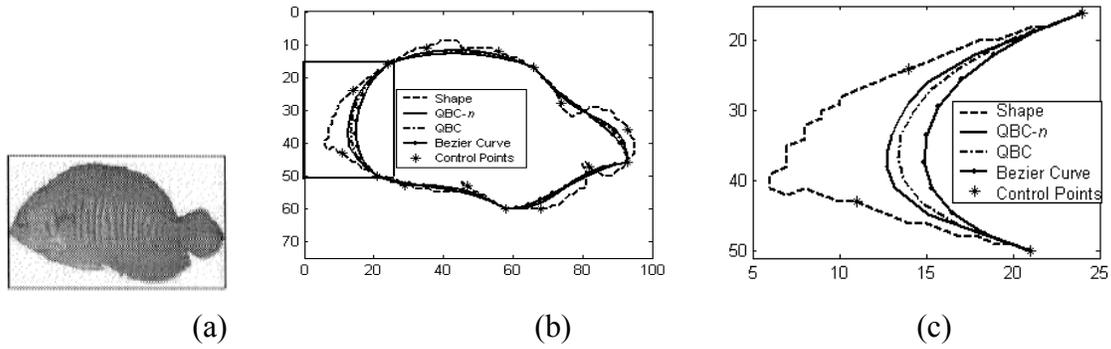


Figure 8: a) *Fish* shape of [2]; b) Shape described with SR= 5; c) Zoom-in on the highlighted portion.

The shape descriptions of the object-shape in Figure 8(a) are shown for SR=5 in Figure 8(b). The BC generated a *class one* distortion of $9.25pel$ for the highlighted head region, compared with the corresponding values for QBC and QBC-*n* of $7.8pel$ and $7pel$ respectively. For clarity, a magnified version of this region is shown in Figure 8(c). When the whole object was considered, QBC-*n* provided the best shape description, while BC performed worst as confirmed by the numerical results in Table 2 for various segment numbers. Table 2 also reveals QBC-*n* consistently provided better performance (lower distortion) even for a small number of curve segments. For instance, the *class one* and *class two* distortions for the BC with 6 segments were $7.8 pel$ and $6.7 pel^2$ respectively, while for 5 segments, it was $7.8 pel$ and $6.6 pel^2$ for QBC and $7 pel$ and $5.4 pel^2$ for QBC-*n* respectively. This improvement was a direct result of incorporating localised information into the classical BC global framework.

Table 2: *Class one* and *class two* distortion measures for the *fish*-shape with different segment rates (units: $D_{max} = pel$; $D_{ms} = pel^2$).

Fish Object	SR=5		SR=6		SR=7		SR=8		SR=9		SR=10	
	D_{max}	D_{ms}										
BC	9.25	9.6	7.8	6.7	6.3	3.8	5.3	3.4	3.7	2.1	3.6	1.2
QBC	7.8	6.6	6.5	5.8	5.5	2.8	4.7	2.4	3.2	1.5	3.2	0.9
QBC-<i>n</i>	7.0	5.4	6.0	4.6	5.0	2.3	4.3	2.0	3.0	1.2	2.9	0.7

From the results analysis above, it is evident that for the same set of CP on the shape both QBC and QBC-*n* produce better shape approximations than BC. The robustness of these enhancements were further tested by comparing them against shape approximating technique [1], which permit CP other than shape points, using their own set of CP derived for the classical BC. Finally a series of tests were conducted upon one of the Arabic character [1] which has strong localised information comprising very sharp peaks followed by sharp troughs over the entire shape. The respective results for BC, QBC and QBC-*n* are shown in Figure 9(a), (b) and (c). Although [1] produced an optimal set of CP in terms of minimum distortion for the BC representation, QBC and QBC-*n* generated a better approximation. The quantitative results in Table 3 again confirm this observation.

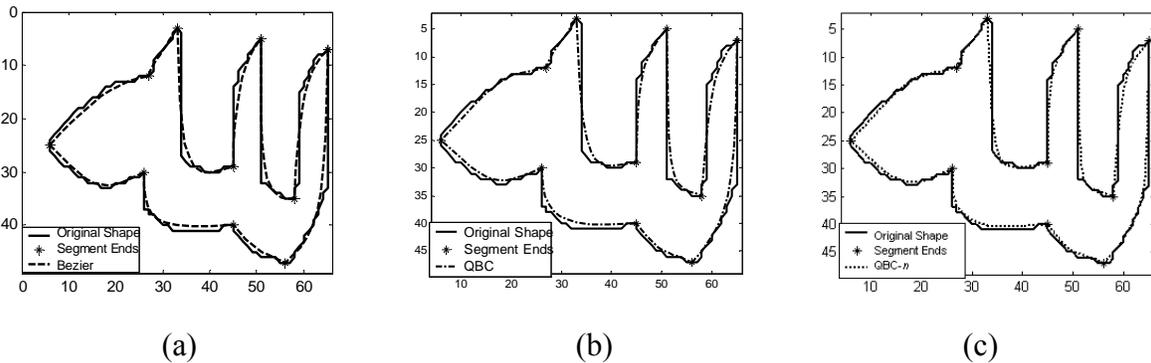


Figure 9: Shape modelling for Arabic-character [1] by a) BC; b) QBC; c) QBC-*n* approximations.

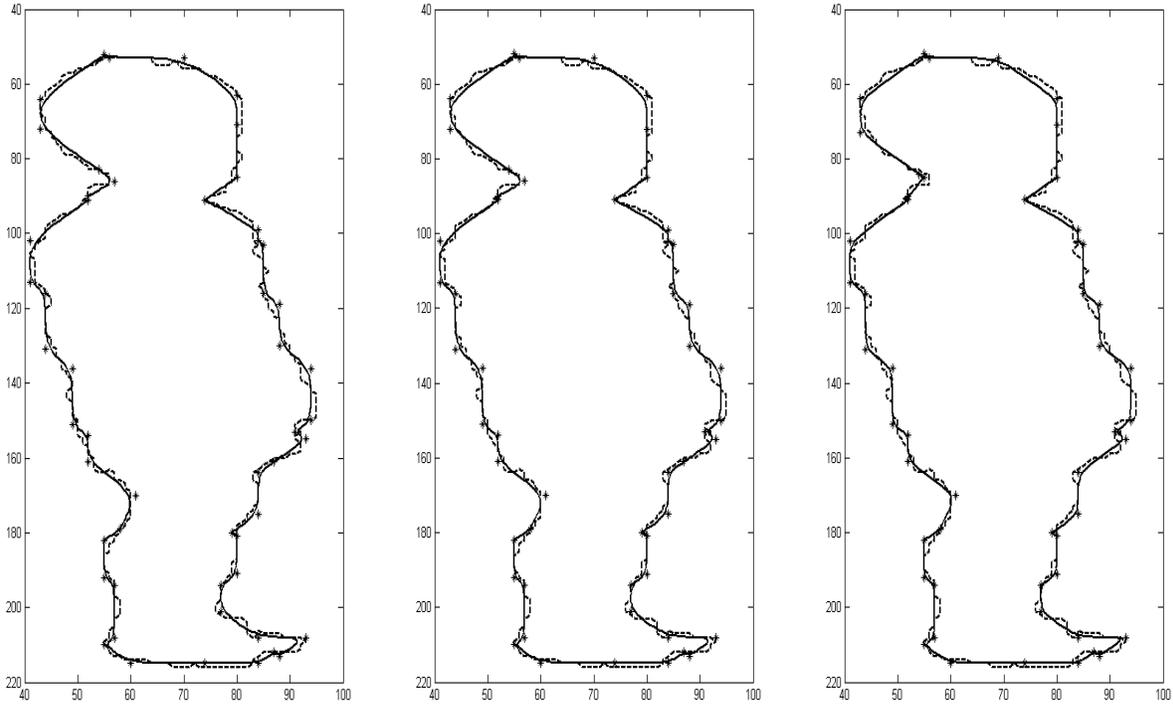
Table 3: Results summary obtained for shape description for the Arabic character.

<i>Class one distortion (pel)</i>			<i>Class two distortion (pel²)</i>		
BC	QBC	QBC-<i>n</i>	BC	QBC	QBC-<i>n</i>
1.45	1.44	1.2	0.34	0.34	0.23

IV C. Comparison with B-splines based ORD optimal shape coding framework

Though QBC primarily enhances the performance of the BC theory, since BC is the basis of B-spline curves, QBC can be used, with proper adjustments of the CP, in the B-spline based frameworks. Section IIIC discussed how quadratic QBC can be used within the B-spline based ORD optimal shape coding algorithms. Some related experimental results will now be presented.

For sake of equity in all the subsequent experiments the variable width admissible CP band [15], and the curvature based admissible distortion measurement strategy proposed in [13], since for binary shape coding purposes image intensity data may not always be available. Without loss of generality however, QBC is equally applicable to the image gradient based techniques [13], [14] provided the necessary image intensity data are available.



(a) B-spline

(b) QBC

(c) QBC- n

Figure 10: Results for the first Kid of the 1st frame of the Kids sequence with $T_{\max} = 2$ and $T_{\min} = 1pel$ (legends – *solid line*: Approximated boundary; *dashed line*: Original boundary; *asterisk*: CP).

A series of experiments were performed concentrating upon the required bit-rate for a prescribed set of admissible distortion values. The respective results produced by the different ORD algorithms upon the first Kid shape of the 1st frame of the Kids sequence are shown in Figure 10 (a)-(c) for a peak distortion bound of $T_{\max} = 2pel, T_{\min} = 1pel$ while Table 4 summarises the bit-rate requirement for both Kid shapes of the 1st frame using various admissible distortion combinations. The subjective results in Figure 10 show that the approximated shapes maintained the admissible distortions in all cases and also for both QBC cases the approximated curves possessed similar smoothness that of B-spline based algorithms. The results in Table 4 reveals

that both QBC and QBC- n based algorithms required lower bit-rate than those that of the B-spline based algorithms and also the QBC- n provides superior results over the QBC.

Table 4: Bit requirements for admissible T_{\max} and T_{\min} (in *pel*) for various ORD optimal shape coding algorithms upon the 1st frame of the *Kids* test sequence.

	$T_{\max} = 1, T_{\min} = 1$	$T_{\max} = 2, T_{\min} = 1$	$T_{\max} = 2, T_{\min} = 2$	$T_{\max} = 3, T_{\min} = 1$	$T_{\max} = 3, T_{\min} = 2$
B-Spline	1140	730	641	627	612
QBC	1136	728	628	616	609
QBC-n	1084	708	620	609	601

To substantiate the performance of the proposed QBC and QBC- n based ORD optimal shape coding algorithms compared with the existing B-spline based algorithms, a further series of experiments was conducted, this time using the MPEG-4 shape distortion metric D_n , often referred to as the *relative area error* (RAE) which is defined as the percentile ratio of the number of erroneously represented pels of an approximating shape to the total number of pels in the original shape [19]. Figure 11 plots the corresponding RD curves for B-splines, and the new QBC and QBC- n based algorithms using the 1st frame of the *Kids* sequence and this clearly reveals that both QBC and QBC- n based algorithms produced superior results over the existing B-spline based algorithms. This is because, the QBC curves closely follow the control polygon of the CP. Consequently, to obtain a curve similar to that of the B-spline in the sense of the maintaining admissible distortions, the distance between the consecutive CP in QBC becomes smaller than that of the B-splines and thereby requires a lower bit-rate and improves the overall rate-distortion performance.

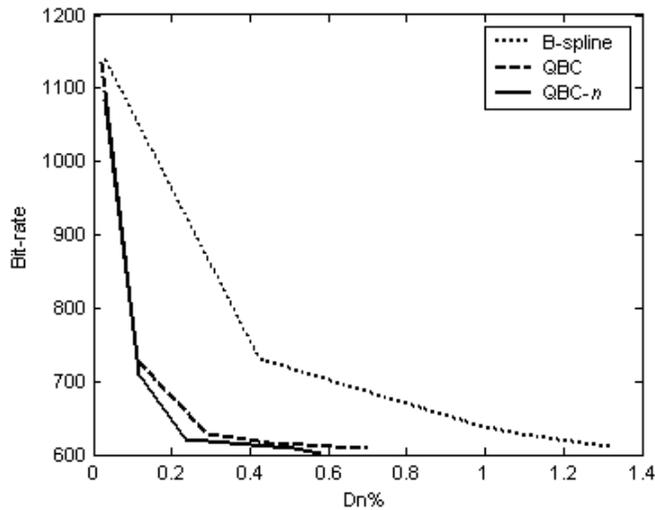


Figure 11: Comparative rate distortion performances for different ORD algorithms using the MPEG-4 D_n metric on the 1st frame of the *Kids* test sequence.

V. CONCLUSION

While the Bezier curve is a well established tool for a wide range of applications, its principal drawback is that it does not consider local shape information. This paper has focused specifically upon bridging this hiatus by integrating local information into the classical Bezier curve framework without increasing the number of control points. Two enhancements of Bezier (QBC and QBC- n) theory have been presented and mathematically proven they retain all the core properties of the classical Bezier curve. The qualitative and quantitative results using different control point sets, Bezier variants and test shapes have shown that QBC exhibited considerable improvement over the Bezier curve as well as other well-established shape descriptor methods, in terms of a consistently lower shape distortion performance, while retaining the same order of computational complexity. QBC can also be seamlessly integrated into these descriptor methods and operational rate distortion optimal vertex-based shape coding framework to improve their overall shape approximating performance. This paper has also determined the bounds of the admissible control point band for both QBC and QBC- n when these are embedded within the

classical vertex-based shape coding framework. Moreover, since these bounds are lower than the bound for the existing B-spline based encoding, the QBC based encoding will also reduce the overall computational cost.

VI. References

- [1] M. Sarfraz and M.A. Khan, "Automatic outline capture of Arabic fonts," *Information Sciences*, pp.269-281, 2002.
- [2] L. Cinque, S. Levialdi and A. Malizia, "Shape description using cubic polynomial Bezier curves," *Pattern Recognition Letters*, pp.821-828, 1998.
- [3] L.D. Soares and F. Pereira, "Spatial shape error concealment for object-based image and video coding," *IEEE Transactions on Image Processing*, vol.13, no.4, pp.586-599, 2004.
- [4] F. S. Hill Jr., *Computer Graphics*, Prentice Hall, Englewood Cliffs, 1990.
- [5] A.R. Forrest, "Interactive interpolation and approximation by Bézier polynomials," *Computer Journal*, vol.15, no. 1, 1972.
- [6] R.H. Bartels, J.C. Beatty and B.A. Barsky, *An Introduction to Splines for use in Computer Graphics & Geometric Modeling*, Morgan Kaufmann Publishers Inc, 1987.
- [7] 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, pp.35-46, 1980.
- [8] M. Hosaka and F. Kimura, "A theory and methods for free form shape construction," *Journal of Information Processing*, vol. 3, no. 3, pp. 140-151, 1980.
- [9] F.A. Sohel, G.C. Karmakar, L.S. Dooley, and J. Arkininstall, "Enhanced Bezier curve models incorporating local information," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. IV, pp. 253-256, 2005.
- [10] A.K. Katsaggelos, L.P. Kondi, F.W. Meier, J. Ostermann, and G. Schuster, "MPEG-4 and rate-distortion-based shape-coding techniques," *Proceedings of the IEEE*, vol.86, no.6, pp.1126-1154, 1998.
- [11] G.M. Schuster and A.K. Katsaggelos, *Rate-Distortion Based Video Compression-Optimal Video Frame Compression and Object Boundary Encoding*, Kluwer Academic Publishers, 1997.
- [12] G.M. Schuster, G. Melnikov, and A.K. Katsaggelos, "Operationally optimal vertex-based shape coding," *IEEE Signal Processing Magazine*, vol.15, no.6, pp.91-108, 1998.
- [13] L.P. Kondi, G. Melnikov, and A.K. Katsaggelos, "Jointly optimal coding of texture and shape," *Proceedings of International Conference on Image Processing (ICIP)*, vol.3, pp.94-97, 2001.

- [14] L.P. Kondi, G. Melnikov, and A.K. Katsaggelos, "Joint optimal object shape estimation and encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.14, no.4, pp.528-533, 2004.
- [15] F.A. Sohel, L.S. Dooley, and G.C. Karmakar, "New dynamic enhancements to the vertex-based rate-distortion optimal shape coding framework," *IEEE Transactions on Circuits and Systems for Video Technology*, in press.
- [16] F.A. Sohel, L.S. Dooley, and G.C. Karmakar, "Accurate distortion measurement for generic shape coding," *Pattern Recognition Letters*, vol.27, no.2, pp.133-142, 2006.
- [17] G.E. Farin, *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guide*, Academic Press, 1997.
- [18] D. Nairn, J. Peters, and D. Lutterkort, "Sharp, quantitative bounds on the distance between a polynomial piece and its Bezier control polygon," *Computer Aided Geometric Design*, vol.16, no.7, pp.613-631, 1999.
- [19] N. Brady, "MPEG-4 standardized methods for the compression of arbitrarily shaped video objects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.9, no.8, pp.1170-1189, 1999.