Fast Distortion Measurement Using Chord-Length Parameterization Within the Vertex-Based Rate-Distortion Optimal Shape Coding Framework

Ferdous Ahmed Sohel, *Student Member, IEEE*, Gour C. Karmakar, *Member, IEEE*, and Laurence S. Dooley, *Senior Member, IEEE*

***Abstract*—Existing vertex-based *operational rate-distortion* (ORD) optimal shape coding algorithms can use a number of different distortion measurement techniques, including the *shortest absolute distance* (SAD), the *distortion band* (DB), the *tolerance band* (TB), and the *accurate distortion measurement technique for shape coding* (ADMSC). From a computational time perspective, an N-point contour requires $O(N^2)$ time for DB and TB for both polygon and B-spline-based encoding, while SAD and ADMSC incur $O(N)$ time for polygonal encoding but $O(N^2)$ for B-spline based encoding, thereby rendering the ORD optimal algorithms computationally inefficient. This letter presents a novel *distortion measurement* strategy based on *chord-length parameterization* (DMCLP) of a boundary that incurs order $O(N)$ complexity for both polygon and B-spline-based encoding while preserving a comparable rate-distortion performance to the original ORD optimal shape coding algorithms.**

***Index Terms*—Image/video coding, transmission.**

I. Introduction

THE RIGOROUS review of shape coding algorithms presented in [1] draws the conclusion that the classical vertex-based shape coding framework is optimal in an *operational rate-distortion* (ORD) sense. With both polygonal and quadratic B-spline-based strategies being deployed, [1] has become the kernel for several other shape coding algorithms [2]–[5]. 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 *control points* (CPs) that incurs the lowest bit rate and vice versa. Distortion measures thus play a major role in these algorithms, as evidenced in the fixed *admissible distortion* (AD) framework proposed in [1], which applied either the *shortest absolute distance* (SAD) or *distortion band* (DB) measure. The generalization of DB is the *tolerance band* (TB), which supports variable AD [3]–[5], though it inherits the same performance characteristics of the DB, including the propensity for trivial solutions, a problem that is partially solved by using a *sliding window* (SW). The SW constrains the search space for the next CP to only points within the window, so it also improves the computational time requirements but crucially compromises bit-rate optimality. Furthermore, while both SAD and DB guarantee all points on the approximating curve are bounded by the peak distortion, they do not ensure that each boundary point maintains the AD, particularly at shape corners, though this anomaly has been solved by the new *accurate distortion measurement technique* for *shape coding* (ADMSC) [6]. Since distortion measurement is seamlessly integrated into the ORD optimal shape coding framework, the overall computational complexity of these algorithms is highly dependent on the distortion measurement overhead, and so to ensure computationally efficient encoding, it is essential to employ a fast technique. For both polygonal and B-spline-based encoding, the DB and TB incur $O(N^2)$ computational complexity, where N is the number of boundary points. Additionally, the computational impost for DB and TB proportionally increases with AD as the width of the respective admissible bands widens. In contrast, the SAD and ADMSC measures require $O(N)$ time for polygonal encoding but still mandate $O(N^2)$ time for B-spline-based encoding. B-spline encoding is mostly used since it affords a twofold superiority over its polygonal counterpart, in first requiring a lower bit-rate since higher degree curves are used and second curve smoothness, due to the inherent parametric continuity of B-splines.

To secure computationally fast distortion measurements within the shape coding framework while concomitantly preserving comparable quality, this letter presents a novel *distortion measurement technique* using *chord-length parameterization* (DMCLP) of the associated boundary to determine the approximating curve points and hence the distortion. It will be proven that for both polygon and B-spline-based encoding, DMCLP entails only $O(N)$ complexity, and it will also be shown that the empirical results ensure faster encoding for comparable *rate-distortion* (RD) performance.

The rest of this letter is organized as follows: Section II provides a short overview of the classical vertex-based shape coding framework together with a computational complexity analysis of both the TB and ADMSC techniques. Section III details the new DMCLP technique for both B-spline and polygon-based encoding, while Section IV provides a thorough discussion on the empirical results and performance comparison with DMCLP embedded into the original ORD framework. Some concluding remarks are given in Section V.

Manuscript received March 28, 2006; revised June 30, 2006. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mihaela van der Schaar.

The authors are with the Gippsland School of Information Technology, Monash University, VIC-3842, Victoria, Australia (e-mail: Ferdous.Sohel@infotech.monash.edu.au; Gour.Karmakar@infotech.monash.edu.au; Laurence.Dooley@infotech.monash.edu.au).

Digital Object Identifier 10.1109/LSP.2006.882099

II. EXISTING VERTEX-BASED ORD OPTIMAL SHAPE CODING FRAMEWORK

Existing ORD optimal shape coding algorithms seek to determine and encode a set of CP representing a particular shape. Let the boundary $B = \{b_0, b_1, \cdots, b_{N_B-1}\}$ be an ordered set of points, where N_B is the total number of boundary points. P is an ordered set of CP used to approximate B with $P \subseteq C$, where C is the ordered set of vertices in the admissible CP band. The original ORD framework detailed in [1]–[3] is valid for both polygonal and B-spline-based models and determines an optimal P for B within RD constraints. As the choice of the distortion metric clearly has a significant impact upon both computational speed and quality of this framework, existing distortion measurement techniques will first be briefly examined.

Tolerance Band (TB): TB works as follows [3], [4]: draw a circle around each b_r of radius $T[r]$ that is the AD at b_r and is determined from the prescribed admissible bounds ($T_{\max}$ and $T_{\min}$); TB consists of the set of all points that lie inside the circles; check the distortion, and if all points on a candidate edge (curve) lie inside the TB, it is considered that the candidate edge (curve) maintains the distortion criterion. Thus, every point on the candidate edge (curve) is required to be checked to see whether it belongs to the TB points set. The number of points compatible with the TB-grid on a candidate edge (curve) is of $O(N_B)$, while TB itself comprises a set of $O(N_B)$ points, so the entire checking process for a candidate curve necessitates $O(N_B^2)$ computational time in the worst case. However, as the distortion of each individual associated boundary point is not tested to see if it lies within TB, this inherits the same accuracy measurement problems as DB [6].

Accurate Distortion Measurement Technique for Shape Coding (ADMSC): In polygonal encoding, the edge-distortion for all associated boundary points are calculated from the candidate edge and checked against the corresponding AD that takes $O(N_B)$ computational time. B-spline encoding, however, requires $O(N_B^2)$ time to check the distortion because a B-spline curve is in fact a concatenation of piecewise polygon-edges, and so for each boundary point associated with a candidate curve, the individual distortion has to be measured from all edges that form the curve. The minimum edge-distortion is then taken as the distortion for that particular boundary point and checked with the corresponding AD.

III. DISTORTION MEASUREMENT TECHNIQUE USING CHORD-LENGTH PARAMETERIZATION

The proposed chord-length parameterization-based distortion measurement technique has been developed to be embedded within both B-spline and polygon-based framework, which will now be, respectively, considered.

B-Spline-Based Framework: B-spline is a member of the family of parametric curves and defined by a set of CP; a quadratic B-spline curve for CP set p_{u-1}, p_u, p_{u+1} is defined as

$$Q_u(p_{u-1}, p_u, p_{u+1}) = [t^2 \; t \; 1] \cdot [\text{B-spline Basis Matrix}] \cdot [p_{u-1} \quad p_u \quad p_{u+1}]^T, \; 0 \leq t \leq 1. \quad (1)$$

So for each value of parameter t, one B-spline point is generated, and the number of t values thereby determines the number of points on the B-spline curve. As mentioned in [1], each approximating curve segment is associated with a number of boundary points. The philosophy in this letter is that if each boundary point associated with a B-spline curve has its corresponding point on the curve, then the distortion can be measured as the Euclidean distance between these two points. This can be achieved provided that every associated boundary point for a candidate curve has a corresponding value of t and so one point on the candidate curve. If the distortion between respective boundary points and their corresponding curve points is less than or equal to their AD, the curve upholds the permitted distortion and is further considered in the RD optimization process. To determine this value of t, chord-length parameterization is applied since it produces a smooth curve as well as being extensively used in development of parametric curves [7]. For an arbitrary curve segment having start and end indexes k_1 and k_2, respectively, of the associated boundary points in B, the values of t can be determined as

$$t_r = \begin{cases} 0, & \text{if } r = k_1 \\ \dfrac{|b_{k_1} b_{k_1+1}| + |b_{k_1+1} b_{k_1+2}| + \cdots + |b_{r-1} b_r|}{|b_{k_1} b_{k_1+1}| + |b_{k_1+1} b_{k_1+2}| + \cdots + |b_{k_2-1} b_{k_2}|}, & \text{otherwise} \end{cases} \quad (2)$$

where t_r is associated to b_r, and $|b_{r-1} b_r|$ is the Euclidean distance between b_{r-1} and b_r.

Once a B-spline point is generated using t_r, distortion at boundary point b_r is determined as the Euclidean distance between these two points. It is then only required to ensure that b_r upholds the AD, so DMCLP inherently supports variable admissible distortion and in that sense is analogous to TB. The full DMCLP technique is formalized in Algorithm 1.

Algorithm 1: DMCLP for B-spline-based shape encoding.

Inputs: B—boundary, p_{u-1}, p_u, p_{u+1}—control points, T—AD, k_1, k_2—indexes of the associated boundary points.

Output: $Flag$—1 means distortion criteria upheld; 0 means not upheld.

1. $Flag = 1$; $chord\_len_{k_1} = 0$;
2. FOR $\forall r, k_1 < r \leq k_2$
3. $\quad chord\_len_r = chord\_len_{r-1} + |b_{r-1} b_r|$;
4. FOR $\forall r, k_1 \leq r \leq k_2$
5. $\quad t_r = (chord\_len_r / chord\_len_{k_2})$;
6. $\quad$ Calculate B-spline point q using t_r in (1);
7. $\quad dist = |q b_r|$;
8. $\quad$ IF $dist > T[r] Flag = 0$;

Fig. 1 illustrates the main difference between DMCLP and ADMSC. In Fig. 1(a), for the CP set p_{u-1}, p_u, p_{u+1}, the B-spline curve generates a series of polygons (1,2,3,4,5,6), with the shortest distance of these polygons from b_r being determined, and the minimum amongst these ($b_r 3$ in the example) is designated as the final distortion. Conversely for DMCLP,

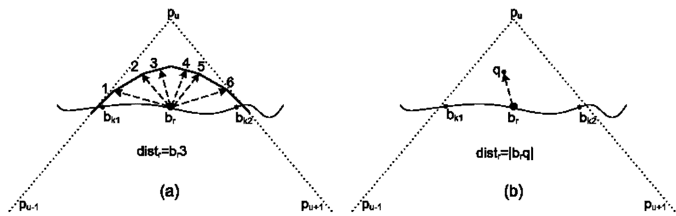

Fig. 1. Example illustrating the key difference between ADMSC and DMCLP. (a) ADMSC. (b) DMCLP.

each associated boundary point has its own value of t, so there is no need to generate the complete B-spline curve with only the corresponding B-spline point q being generated, so the distortion for this metric, as shown in Fig. 1(b), is $|b_r q|$.

Polygon-Based Framework: In polygonal encoding, two CP p_{u-1}, p_u are required for each candidate edge rather than the three in quadratic B-spline-based encoding. The value of t_r is again determined by chord-length parameterization using (2), although in this case, the corresponding edge-point q can be found by linear interpolation in (3) instead of (1) in Step 6

$$q = t_r \cdot p_u + (1 - t_r) \cdot p_{u-1}. \tag{3}$$

This means Algorithm 1 can therefore also be applied for polygon-based encoding with appropriate inputs and using (3) instead of (1) in Step 6. It will now be proven in Lemma 1 that DMCLP maintains AD for all boundary points.

Lemma 1: If $d[r]$ and $T[r]$ are, respectively, the generated and AD for an arbitrary boundary point b_r, the DMCLP technique always upholds the AD so that $d[r] \le T[r]$.

Proof: By embedding DMCLP into the vertex-based ORD optimal shape coding framework, every boundary point will have a respective point on the encoded shape. If the distortion defined in Step 7 of Algorithm 1 at b_r is $dist_r$ and distortion (shortest distance) for the same point with respect to the final approximated shape is $d[r]$, then $d[r] \le dist_r$. In the distortion test (Step 8), for any curve segment to be included in the encoded shape, $dist_r \le T[r]$ must hold true, so that $d[r] \le T[r]$.

Computational complexity analysis: In terms of the order of complexity for Algorithm 1, Steps 2 and 3 calculate the cumulative chord-length of the boundary at a cost, in the worst case of $O(N_B)$, while Steps 4–8 first determine the value of t and then generate the B-spline point using (1) before checking to see whether the AD is maintained, which also incurs $O(N_B)$. The overall complexity for Algorithm 1 in the worst case is therefore $O(N_B)$, which is one degree less than all existing distortion measurement techniques for B-spline-based encoding, while for polygonal encoding, DMCLP exhibits order of computational complexity $O(N_B)$, which is exactly the same as for SAD and ADMSC but one degree lower than both DB and TB.

IV. Results and Analysis

To substantiate that ORD optimal algorithms employing the new DMCLP distortion measure produce faster encoding with comparable RD results to existing distortion measurement techniques, experimental results upon the popular *Neck* region of the 31st frame of the *Miss America* video sequence will now be presented. To determine AD, the curvature-based scheme of [4] is adopted, since it works effectively for all image types, including binary. DMCLP, however, is equally applicable to the image gradient-based techniques [3]–[5], if the required information is available. For presentational clarity, the following nomenclature is used: *Encoding type—Distortion measurement technique*, with the former defining whether either B-spline or polygon-based encoding was used, and the latter the distortion measuring technique employed.

The first series of experiments was conducted to compare the computational time requirements incurred by algorithm combinations for various AD pairs. Table I displays the total CPU time for various algorithm implementations running on a 2.8-GHz Pentium-4 microprocessor, with 512 MB of RAM under Windows XP. It should be noted that to ensure trivial solutions did not occur in the TB measure, an SW was used that, for the purposes of equity in the experiments for all algorithms, had a length 21 pel as this was the maximum feasible length with a 15 codeword-based logarithmic code [2]. From the results in Table I, it is apparent those algorithms using DMCLP were computationally faster compared with the respective TB and ADMSC counterparts, to underscore the theoretical time complexity analysis presented in Sections II and III. For instance, with $T_{\max} = 3$, $T_{\min} = 1$ pel *B-spline-TB*, *B-spline-ADMSC*, and *B-spline-DMCLP*, respectively, required 620.3, 587.8, and 312.5 s, so reflecting that while TB and ADMSC cost $O(N^2)$ for B-spline-based encoding, the overhead for DMCLP is only $O(N)$. For polygonal encoding, DMCLP algorithms required slightly less time than their ADMSC-based counterparts, while the TB algorithms always incurred a significantly greater time overhead, because both DMCLP and ADMSC are of $O(N)$ compared with $O(N^2)$ for the TB. For B-spline-based algorithms using small AD, TB was actually faster than ADMSC, though for higher distortion values, the comparative time requirement for TB increased significantly. So, for example, with $T_{\max} = T_{\min} = 2$ pel, *B-spline-TB* and *B-spline-ADMSC* required 545.5 and 582.1 s, respectively, while the corresponding values for $T_{\max} = 3$, $T_{\min} = 2$ pel were, respectively, 680.4 and 591.6 s because at higher distortions, the TB width is extended so commensurately increasing the computational overhead.

To confirm that algorithms that integrate DMCLP produce analogous RD results to the original framework, a set of experiments concentrating upon the required bit-rate for a prescribed set of AD values was undertaken. The respective results produced by different algorithm combinations are displayed in Fig. 2(a)–(f) for the AD pair $T_{\max} = T_{\min} = 2$ pel, while the numerical results together with some other distortion pairings are summarized in Table I. The results in Fig. 2 reveal that all algorithms produced similar shapes, with the notable exceptions of the *Polygon-TB* and *B-spline-TB* algorithms, where, as indicated by the boxes, the distortion was greater than the prescribed peak admissible value. $T_{\max}$ and $T_{\min}$ were made equal so all boundary points had the same AD, thereby manifesting the aforementioned measuring problems inherent in TB. Furthermore, while TB-based algorithms failed to be constricted to the peak AD, in contrast, both DMCLP and ADMSC consistently maintained a bounded distortion.

TABLE I
COMPUTATIONAL TIME (seconds) AND BIT-RATE (bits) REQUIREMENTS (WITH THE OBTAINED DISTORTION IN PARENTHESIS WHENEVER THE PEAK DISTORTION FAILED TO BE BOUNDED) FOR DIFFERENT ADMISSIBLE DISTORTION PAIRS ($T_{\max}$, $T_{\min}$ IN pel) IN VARIOUS ORD OPTIMAL SHAPE CODING ALGORITHMS

	$T_{\max}=1$, $T_{\min}=1$		$T_{\max}=2$, $T_{\min}=1$		$T_{\max}=2$, $T_{\min}=2$		$T_{\max}=3$, $T_{\min}=1$		$T_{\max}=3$, $T_{\min}=2$	
Algorithm	**Time**	**Bit-rate**	**Time**	**Bit-rate**	**Time**	**Bit-rate**	**Time**	**Bit-rate**	**Time**	**Bit-rate**
Polygon-TB	4.26	115(2.24)	6.03	95(2.24)	7.73	79 (4.0)	11.35	71 (5.0)	12.66	70 (5.0)
Polygon-ADMSC	1.63	138	1.89	109	2.01	86	2.15	86	2.25	86
Polygon-DMCLP	1.61	146	1.83	112	1.92	93	1.97	92	2.02	87
B-spline-TB	390.60	133 (2.0)	510.50	87 (3.6)	545.50	78 (2.8)	620.30	76 (6.0)	680.40	75 (6.0)
B-spline-ADMSC	554.20	127	575.00	100	582.10	78	587.80	78	591.60	78
B-spline-DMCLP	270.20	132	290.30	102	297.00	78	312.50	80	314.30	78

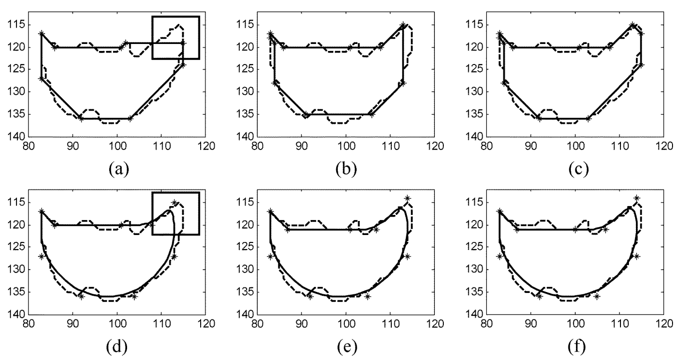

Fig. 2. Results for the *Neck* shape with $T_{\max} = T_{\min} = 2$ pel (legends—*solid line*: approximated boundary; *dashed line*: original boundary; *asterisk*: CP). (a) *Polygon-TB*. (b) *Polygon-ADMSC*. (c) *Polygon-DMCLP*. (d) *B-spline-TB*. (e) *B-spline-ADMSC*. (f) *B-spline-DMCLP*.

The numerical results in Table I show the bit-rate requirement for different combinations of the same AD pairings are comparable between respective algorithms, as, for example, in $T_{\max} = 2$, $T_{\min} = 1$ pel, *Polygon-TB*, *Polygon-ADMSC*, and *Polygon-DMCLP* required 95, 109 and 112 bits, respectively, while the corresponding *B-spline-TB*, *B-spline-ADMSC*, and *B-spline-DMCLP* combinations required 87, 100, and 102 bits. The comparatively small number of additional bits incurred for DMCLP-based algorithms compared to ADMSC was due to it being a slightly more relaxed distortion measure, in the sense that ADMSC always guarantees to calculate the absolute minimum distortion. A cursory review of the TB results fallaciously reveals they consistently required fewer bits for encoding, until cognizance is taken that these algorithms did not always uphold the AD constraint as in the *B-spline-TB* case, which generated a maximum distortion of 3.6 pel despite being supposedly bound to a peak of 2 pel. This meant TB ignored particular parts of the shape leading to a lower bit-rate than ADMSC and DMCLP-based algorithms, which both guaranteed peak AD. It is also worth highlighting that for the same AD, B-splines always required a lower bit-rate than its polygon counterpart, provided the AD was maintained. For instance, with $T_{\max} = T_{\min} = 2$ pel, *Polygon-ADMSC* required 86 bits compared to only 78 bits for *B-spline-ADMSC*, endorsing the earlier comment about higher degree B-splines requiring lower bit-rates than lower degree polygon approximations.

V. CONCLUSION

The computational speed of any algorithm is a vital benchmark of quality, as many applications mandate fast and efficient throughput. This letter has presented a novel DMCLP that can be seamlessly embedded into the classical vertex-based RD optimal shape coding framework to reduce the computational time complexity order. It has been proven that for a N-point boundary, DMCLP requires $O(N)$ computational time for both B-spline and polygon-based encoding in contrast to the conventional *tolerance band* approach, which incurs $O(N^2)$ complexity in both cases. Experimental results have shown that algorithms embedding DMCLP consistently provided both comparable RD performance to existing algorithms and faster encoding due to the lower complexity order.

REFERENCES

[1] A. K. Katsaggelos, L. P. Kondi, F. W. Meier, J. Ostermann, and G. M. Schuster, “MPEG-4 and rate-distortion-based shape-coding techniques,” *Proc. IEEE*, vol. 86, no. 6, pp. 1126–1154, Jun. 1998.
[2] G. M. Schuster and A. K. Katsaggelos, *Rate-Distortion Based Video Compression: Optimal Video Frame Compression and Object Boundary Encoding*. Norwell, MA: Kluwer, 1997.
[3] L. P. Kondi, G. Melnikov, and A. K. Katsaggelos, “Joint optimal object shape estimation and encoding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 4, pp. 528–533, Apr. 2004.
[4] ——, “Jointly optimal coding of texture and shape,” in *Proc. Int. Conf. Image Processing*, 2001, vol. 3, pp. 94–97.
[5] S. K. Bandyopadhyay and L. P. Kondi, “Optimal bit allocation for joint contour-based shape coding and shape adaptive texture coding,” in *Proc. Int. Conf. Image Processing*, 2005, vol. 1, pp. 589–592.
[6] F. A. Sohel, L. S. Dooley, and G. C. Karmakar, “Accurate distortion measurement for generic shape coding,” *Pattern Recognit. Lett.*, vol. 27, no. 2, pp. 133–142, 2006.
[7] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design—A Practical Guide*, 4th ed. New York: Academic, 1997.