PREDICTIVE SHAPE CODING USING GENERIC POLYGON APPROXIMATION

Jong-il Kim and Brian L. Evans

Laboratory for Image and Video Engineering
Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin, TX 78712-1084

E-mail: {`jikim,bevans`}`@ece.utexas.edu`
Web: http://anchovy.ece.utexas.edu/

ABSTRACT

We introduce an efficient predictive binary shape coding method that consists of (1) global motion estimation, (2) local motion estimation, (3) matched segment coding, and (4) residual segment coding. Global and local motion estimation use contour pel matching and knowledge of previously reconstructed contours. After motion compensation, we code the one-dimensional reference contour indices of the matched contour positions. The final step codes the mismatched contour segments using residual coding. We use a maximum shape distortion tolerance parameter ($d_{\max}$), which is zero for lossless coding, for both motion estimation and residual coding. We apply the new shape coding method to MPEG-4 binary mask test sequences in QCIF and SIF formats for a wide range of $d_{\max}$ values. The key contribution of our method is in lossy shape coding in which the average coding gain is more than 100% over generalized differential chain coding. The proposed scheme can be applied to MPEG-4 compliant shape coding and to efficient shape representation for future MPEG-7 applications.

1. INTRODUCTION

Since the publication in 1961 of Freeman chain codes [1] for compact representations of shapes, improvements in chain codes have yielded simple and popular lossless contour compression methods [2, 3]. In 1991, Lu and Durham developed highly sophisticated chain coding schemes using arithmetic coding that offered 50% improvement over Freeman's chain coding [4] and 25% improvement over differential chain coding. The upper bound on the coding gain over Freeman chain codes is 150% [4]. When constraints are placed on the quality of service, an additional limitation exists to apply chain codes for lossy channels. In 1997, O'Connell generalized a vertex-based chain coding scheme [5] specifically for non-predictive binary shape coding. This idea extends chain codes for lossy shape coding, so that the relative position of the vertices after polygon approximation is coded in a uniform manner as in differential chain coding. Predictive polygons and spline coding are implemented in an object-based analysis-synthesis coder at very low bitrates based on the source model of flexible two-dimensional objects which move translationally in the image plane [6]. We propose a new scheme that applies vertex-based generalized chain coding scheme in intra-binary shape coding, and extends polygon approximation to inter-binary shape coding using lossless and lossy contour motion estimation. We generalize polygon approximation to estimate lossy contour motions and encode the residual contours. The residual mismatched contour segments is coded by additional vertices, so that we code the motion matched and mismatched segment of a shape in uniform way using polygon approximation.

2. BACKGROUND

2.1. Non-predictive Lossless Shape Coding Using Chain Coding

Chain coding has been widely used for non-predictive lossless content-based shape coding because of its simplicity. Since the advent of Freeman chain codes [1], several improved chain coding schemes have been proposed. Of these schemes, differential chain coding with arithmetic coding exhibits the best coding gain [4]. The eight-directional chain code and its differential chain code (if the previous code is 0) are described in Figure 1. Generally, Huffman coding has been applied to encode the chain and differential chain codes. Because of the high correlation of consecutive chain codes, an entropy coding scheme with a higher-order Markov model for differential chain code outperforms all other schemes. The entropy E_c of the differential chain code with a zero-order Markov model, which is the average bit for a contour pel, is given by

$$E_c = -\sum_{i=-3}^{4} p(i) \cdot \log_2 p(i) \tag{1}$$

where $p(i)$ is the probability that the differential chain code equal to i. For head-and-shoulder shapes, E_c is approximately 1.5.

This work was supported by a US National Science Foundation CAREER Award under Grant MIP-9702707.

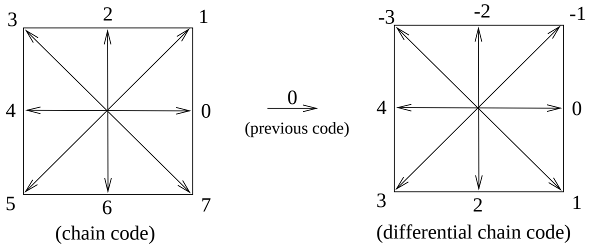

Figure 1: Chain and differential chain codes

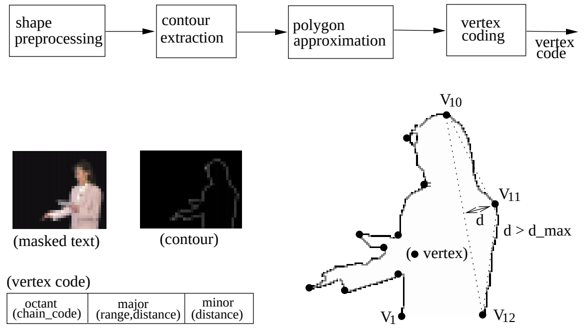

Figure 2: Generalized chain coding with polygon approximation

2.2. Lossy Shape Coding Using Generalized Chain Coding

O'Connell's generalized chain coding scheme known as the object-adaptive vertex-based shape coding method [5] encodes the vertices of a polygon approximation of the shape by adapting the dynamic range of the relative locations of the vertices and by exploiting an octant based representation of each vertex. Figure 2 illustrates the generalized chain coding approach for lossy shape coding.

Shape preprocessing in Figure 2 implies spatial conversion and contour smoothing of the binary mask. A iterated-refinement vertex-selection method [6] allows the selection of an ordered list of vertices for the contours, as shown in Figure 2. Twelve vertices exist for the given desired maximum distortion $d_{\max}$. If the maximum distance between the straight line from adjacent vertices (V_{10} and V_{12}) and the original contour pels is larger than the given tolerance ($d_{\max}$), then the position of the original contour point is selected as the new vertex, such as V_{11} in Figure 2. Low-curvature contour segments between consecutive vertices may be better approximated by higher-order splines. For the contours having high curvature (e.g., line segments), a polygon representation performs better. The selected ordered vertices are encoded using generalized chain codes which consist of an octant code, a major code, and a minor code, as shown in Figure 2. Freeman chain codes specify a direction between two pels having unit distance, as shown in Figure 1. This approach is extended to lossy shape coding with polygon approximation having an arbitrary length between vertices. So, an octant code can be regarded as a chain code for adjacent vertices. As examples of clockwise contour tracing, the chain code region between 0 and 7 (the 7th octant) in Figure 1 corresponds to octant code 0, and octant code 7 is for the 0th octant of region 1 and 0. So, the octant code for V_{10} and V_{11} in Figure 2 is 1, and the octant code for V_{11} and V_{12} is 2. Similarly, a differential octant code can be derived from the differential chain code.

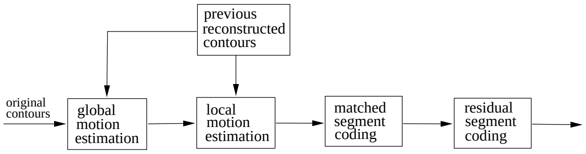

Figure 3: Proposed generalized predictive binary shape coding

3. GENERALIZED PREDICTIVE SHAPE CODING

Figure 3 depicts the block diagram of our proposed shape coding scheme. In general, multiple contours exist in a Video Object Plane (VOP) or in an object of given picture. Instead of applying polygon approximation to each contour as in non-predictive (intra) shape coding (e.g., generalized chain coding), we search global and local contour motions to exploit the correlations in the current and previously reconstructed contours. Since the contour pel matching algorithm (which counts the number of matched pels) is used for both local and global motion estimation, the increase in complexity of these motion estimation steps is minimal when compared to non-predictive coding schemes. After contour matching, matched contour positions are represented by the reference contour index of the previous contour in the matched segment coding step, and mismatched segments (residual segments) are further coded in the residual segment coding step. More details follow.

3.1. Contour Motion Estimation using the Contour Pel Matching Algorithm

Contour motion estimation is based on the assumption that objects move translationally in an image plane as in an object-oriented coding scheme. A contour is a collection of contour pels, and each contour pel contains 2-D spatial position information. If the contour length of current contour is N_c and the reference contour length is N_r, and if I_c and I_r are current and reference contour sets, then set I_c has N_c elements and set I_r has N_r elements. A contour pel matching algorithm finds the offset vector (mv_x, mv_y) to maximize the number of the element in $I_c \cap I_r^{(mv_x,mv_y)}$, with $I_r^{(mv_x,mv_y)}$ in (2), where mv_x and mv_y are motion vectors for horizontal and vertical direction:

$$I_r^{(mv_x,mv_y)} \equiv I_r(x_i + mv_x, y_i + mv_y) \qquad (2)$$

Here, $I_r(x_i, y_i)$ is the i-th element in set I_r. If we apply pel matching algorithm without any constraints, then the number of comparison operation needed is $2 \cdot N_c \cdot N_r$ times the motion search area. But some heuristics can be imposed to reduce complexity for motion estimation without loss of generality such as

- $\bigcap_{i=1}^{N_c} I_r(x_i, y_i) = \bigcap_{i=1}^{N_r} I_r(x_i, y_i) = \emptyset$
- $|N_c - N_r| < \alpha$, where $\alpha \approx 0.1 \cdot N_c$.
- start the motion search from the center of gravity of each contour.
- isolated (unconnected) matched point is considered as mismatched point.
- trace direction is same for I_r and I_c, so that matched points are connected in the same direction (either clockwise, or counter clockwise).
- reference point of indexing for the contour is the first pel met during scanning by raster order, so that reference index point need not to be sent for the decoder.

Unlike texture motion estimation, contour motion estimation is simple. Instead of using the entire binary mask, only the x and y spatial position variables of the contour pels are compared with the reference contour pels. In global contour motion estimation, the contour pel matching algorithm is applied to all of the contours in a VOP. After global contour motion, local motion is searched in single contour unit using the contour pel matching algorithm. If there are N contours in a VOP, and all contours are coded in predictive coding mode, then the global motion vector is followed by N contour motion vectors. If a contour is not matched well, then it is coded by a non-predictive coding scheme such as generalized chain coding. The motion estimation is either lossless or lossy. A line widening scheme is applied to estimate lossy contour motion estimation. Line widening is the extension of polygon approximation to contour motion estimation. When a contour pel in current contour is within $d_{\max}$ range from a reference contour pel, the pel is considered as matched pel in lossy motion estimation. When $d_{\max}$ is zero, motion estimation becomes lossless. As $d_{\max}$ increases, the portion of matched segment region increases, so that for slow moving object with high correlation of the topologies, most of the contour points are matched. To achieve better coding efficiency, motion vectors can be coded using variable length code which has smaller code length for the small motion vector.

3.2. Coding of matched and mismatched contour positions -Reference Index Based Coding

Examples of matched and unmatched segments after contour motion compensation are shown in Figure 4, where matched and mismatched status of (b) and (d) are indicated by Figure 4 (e). Matched regions are from solid circle to solid rectangle in a counter clockwise tracing in Figure 4 (b) and (d), and vice versa for mismatched segments. V_j can be considered as vertex points as in Figure 2, but the coding scheme is different for matched and mismatched segments. Since a segment is either matched or mismatched, and each segment is composed of start and end points, the number of V_j is always even, and the matched contour segments are between an odd index of j and even index j. If the motion matched region increases as $d_{\max}$ increases in lossy motion estimation, then the number of V_j is decreased and the length of matched segment is increased as shown in Figure 4 (b) for $d_{\max}$ is one and (d) for $d_{\max}$ is two.

These contour points V_j will be coded to report matched or unmatched status to the decoder. All V_j points are both in the current and reference contours. The positions of all V_j, which are 2-D variables, can be exactly mapped into the reference one-dimensional contour index. Using this reduction in dimensionality, we can achieve higher coding gain when compared to 2-D vertex encoding as in conventional polygon coding. For example, if the number of reference contour pels (reference contour length) is L_r, it is possible to code any V_j using a fixed-length code of $\lceil \log_2(L_r) \rceil$ bits, where $\lceil \cdot \rceil$ indicates the ceiling of $\cdot$. The initial point, which corresponds to the contour pel at index 1, is marked by X in Figure 4 and is available at the decoder, because the initial index of the reference contour is the first pel encountered during raster scan. The initial index is 1, and following pels in counter clockwise direction have increasing indices. The last index (L_r) is the index of the adjacent contour pel to the initial point in the clockwise direction. Once we have the information about the initial index, all of the reference contour points are mapped into a one-dimensional index value. For example, let $L_r = 100$ in Figure 4 (d). Since the initial index is between V_1 and V_2, the possible index values for (V_2,V_3,V_4,V_1) are (30,38,45,58). All of the V_j indices can be encoded using 7 bits.

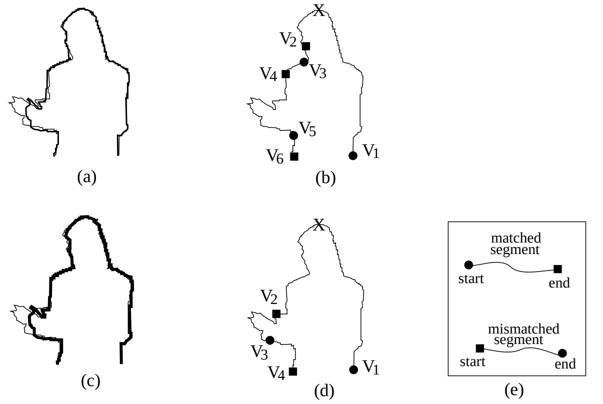

Figure 4: Lossy motion estimation (a) overlapped figure of original and reconstructed contours with line widening at $d_{\max} = 1$; (b) matched and mismatched segment of (a) in counter clockwise tracing; (c) overlapped figure of original and reconstructed contours with line widening at $d_{\max} = 2$; (d) matched and mismatched segment of (c) in counter clockwise tracing; and (e) matched and mismatched segment status indication.

Further coding efficiency can be obtained by using the feature of differential V_j value and sending the largest segment first. Additional bit savings can be reaped if we apply travel length information, which is the sum of the decoded contour length. The difference between the index of adjacent V_j points is always smaller than either the maximum segment length or the remaining length (contour length - travel length). By successively applying this theory to the remaining V_j values, we can achieve a high coding gain over fix-length codes, especially for a sequence which has a large number of V_j with large differential distances. In lossless shape coding, the unmatched segments are coded by chain coding. In lossy shape coding, after lossy motion estimation, the unmatched segments are approximated using poly-

sequence	*format*	*entropy (E_c)*	*contour pels/VOP*
`weather`	QCIF	1.58	285
`cyclamen`	SIF	1.72	2122
`kids`	SIF	1.78	963
`logo`	SIF	1.71	1761

Table 1: Test sequences for computer simulation

sequence	*bound*	*intra*	*proposed*	*gain %*
`weather`	450	466	394	16
`cyclamen`	3649	3052	2763	9
`kids`	1714	1740	1716	1
`logo`	3011	3223	1209	62

Table 2: Lossless Coding Results in non-predictive and predictive schemes (bits/VOP)

gons. If the tolerance is greater $d_{\max}$, then additional vertices are appended between V_j.

4. SIMULATIONS

We processed the test sequences in Table 1 for 10 sec using 10 frames/sec. The first VOP (frame) is coded using non-predictive coding scheme as generalized chain coding, and the remaining VOPs are coded by our proposed method.

The error criterion given by (3) is the ratio of mismatched area to the total object area. The mismatched area includes both the erosive and dilative errors.

$$E_n = \frac{\sum \text{mismatched pels}}{\sum \text{original contour pels}} \qquad (3)$$

For multiple contours, the search range of global motion is between -16 and 15. For local motion, the search range is confined to one-half of the global range. This is a reasonable heuristic because once the global motion is detected, the local motion range is relatively small vs. the global motion range. From Table 1, we can compute the lower bound of chain coding based on the zero-order Markov model. The lower bound for chain code bit in a VOP is E_c (bits/pel)$\cdot$ (contour pels/VOP), which is in the second column of Table 2 computed from Table 1. For non-predictive coding, we modify the generalized chain coding to include an escape code. If a contour has long line segment, then escape and length codes encode the line as a group. Cyclamen sequence has long line segments, so bit reduction is achieved for non-predictive coding given in column 2 of Table 2 that is below the differential chain code bound. Using the proposed predictive coding scheme, we obtained a coding gain between 1% to 62% for the lossless case.

Figures 5 and 6 show rate-distortion curves for `weather` and `logo` sequences. The largest coding gain is for `logo` because of its high temporal correlation. The smallest coding gain is for `kids` which contains fast motion and abrupt topological changes. If we use higher frame rate source, then the proposed coding scheme will show better results because of the increased temporal similarity.

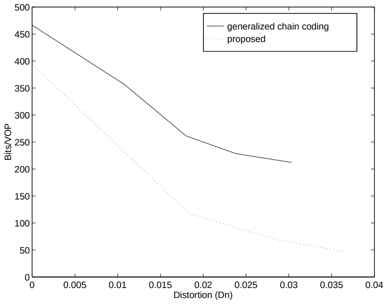

Figure 5: Weather rate-distortion curve

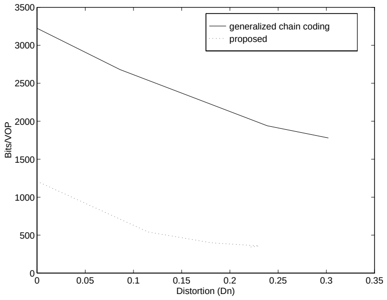

Figure 6: Logo rate-distortion curve

5. CONCLUSION

We propose a new predictive binary shape coding method. As a true realization of the polygon approximation in each encoding stage, the new coding scheme guarantees subjective quality with greatly improved coding efficiency over conventional approaches in both the lossless and lossy cases. Our method can be incorporated into an MPEG-4 encoder.

6. REFERENCES

[1] H. Freeman, “On the encoding of arbitrary geometric configurations,” *IRE Trans. on Electronic Computers*, vol. EC-10, pp. 260–268, June 1961.

[2] T. Kaneko and M. Okudaira, “Encoding of arbitrary curves based on the chain code representation,” *IEEE Trans. on Communications*, vol. 33, pp. 697–707, July 1985.

[3] R. Gonzalez and P. Woods, *Digital Image Processing.* Addison-Wesley, 1992.

[4] C.-C. Lu and J. G. Dunham, “Highly efficient coding schemes for contour lines based on chain code representations,” *IEEE Trans. on Communications*, vol. 39, pp. 1511–1514, Oct. 1991.

[5] K. J. O’connell, “Object-adaptive vertex-based shape coding method,” *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 7, pp. 251–255, Feb. 1997.

[6] P. Gerkin, “Object-based analysis-synthesis coding of image sequences at very low bit rates,” *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 4, pp. 228–235, June 1994.