JOINTLY OPTIMAL CODING OF TEXTURE AND SHAPE

Lisimachos P. Kondi

Dept. of Electrical Engineering
State University of New York at Buffalo
Buffalo, NY 14260
USA

Gerry Melnikov, Aggelos K. Katsaggelos

Dept. of Electrical and Computer Engineering
Northwestern University
Evanston, IL 60208
USA

ABSTRACT

A major problem in object oriented video coding and MPEG-4 is the encoding of object boundaries. Traditionally, and within MPEG-4, the encoding of shape and texture information are separate steps (the extraction of shape is not considered by the standards). In this paper, we present a vertex-based shape coding method which is optimal in the operational rate-distortion sense and takes into account the texture information of the video frames. This is accomplished by utilizing a variable-width tolerance band which is proportional to the degree of trust in the accuracy of the shape information at that location. Thus, in areas where the confidence in the estimation of the boundary is not high and/or coding errors in the boundary will not affect the application (object oriented coding, MPEG-4, etc.) significantly, a larger boundary approximation error is allowed. We present experimental results which demonstrate the effectiveness of the proposed algorithm.

1. INTRODUCTION

In recent years, significant research has been performed on shape coding. Interest in shape coding increased with the advent of *object oriented video coding* [1]. In contrast with block-based video coding algorithms, object oriented video coding performs motion compensation using objects instead of rectangular blocks. Motion compensation performed this way can be more effective. However, the shapes of these objects need to be encoded and transmitted. Thus, efficient shape coding algorithms would need to be developed in order for object oriented video coding to be efficient. The newly introduced standard MPEG-4 would provide *interactivity* by transmitting object information along with the texture. Thus, the receiver would be able to manipulate the objects in the scene. Again, efficient shape coding algorithms are essential in order for MPEG-4 to be able to provide interactivity.

In our previous work [2, 3, 4], we introduced efficient shape coding schemes (polygon and/or B-spline based), which are optimal in the operational rate-distortion sense. These schemes utilize Graph Theory and Dynamic Programming in order to reduce their computational complexity and typically outperform shape coding techniques proposed by MPEG-4 [5]. A review of the methods which appeared in the literature on shape coding can be found in [4, 5].

A variety of distortion measures can be applied to these operational rate-distortion optimal shape coding techniques. If errors at all parts of the boundary are given the same weight, small but distinct features of the original boundary can disappear (be "cut off") in its approximation. Thus, we might be spending more bits to encode parts of the boundary where high accuracy is not required while fewer bits are left to encode other parts of the boundary which are more important.

In this work, we propose the use of an adaptive distortion measure which is based on texture information or on shape curvature. As an example, the magnitude of the gradient is found; in areas where it is high, a better approximation is forced, whereas in areas where it is low, a higher approximation error is allowed [6, 7]. The justification for this is that in areas of low gradient magnitude, a higher approximation error would be less perceivable. Furthermore, if a gradient-based boundary estimation method was employed, our confidence in the accuracy of the boundary estimation would not be very high in areas of low gradient magnitude. This is also true in the case of object oriented video coding. The motion and object estimation cannot be very accurate in areas with low gradient magnitude and furthermore, larger boundary approximation errors in these areas would have low impact on motion compensation. Although in this paper the adaptive distortion measure is based on the gradient magnitude and the curvature, it could be based on a number of other measures.

The MPEG-4 standard allows for the encoding of the texture of video objects using *Shape-Adaptive Discrete Cosine Transform (SA-DCT)*. Usually, video frames are encoded by taking the Discrete Cosine Transform of 8×8 image blocks. For the texture of objects, however, some 8×8 blocks that are close to the boundary will be partially occupied by the object. Using an 8×8 DCT, we would need to transmit 64 coefficients for these blocks, although the actual number of pels would be smaller. SA-DCT provides for a way of encoding such blocks using a number of coefficients that is equal to the number of the object pels in the block. This is accomplished by shifting the object pels towards the origin of the block and then taking one dinensional DCTs row-wise and then column-wise. The length of these one-dimensional DCTs can be less than eight.

Compression is accomplished by quantization of the DCT coefficient followed by entropy coding. It is expected that if a block contains an edge or an area with high gradient magnitude, the entropy of the quantized DCT coefficients of this block will be high and a large number of bits will be required for its encoding. If SA-DCT is used and the segmentation is based on gradient, it is beneficial for the shape information to be accurate in areas with high gradient. This way, the number of areas of high gradient within the object will be minimized and the encoding efficiency of SA-DCT will be higher.

0-7803-6725-1/01/$10.00 ©2001 IEEE

2. ALGORITHM

In our previous work [2, 3, 4] we proposed schemes which approximate a given shape using a curve of any order (straight lines, B-splines, etc). Given a total bit budget, the algorithm provides the boundary approximation which results in the minimum distortion. Conversely, given a maximum allowable distortion, the algorithm yields the boundary approximation which requires the lowest number of bits for its encoding. The output of the algorithm is the set of the control points of the curve. The min-max and the min-average distortion criteria have been used. Our schemes are optimal in the Operational Rate-Distortion (ORD) sense and utilize Graph Theory and Dynamic Programming. The solution depends on the distortion measure and the encoding scheme for the control points.

In the following discussion, we focus on approximations using B-splines. The following notation will be used. Let $B = \{b_0, \ldots, b_{N_B-1}\}$ denote the connected boundary which is an ordered set, where b_j is the j-th point of B and N_B is the total number of points in B. Note that in the case of a closed boundary, $b_0 = b_{N_B-1}$. Let $P = \{p_0, \ldots, p_{N_P+1}\}$ denote the set of control points of the B-spline curve, which is also an ordered set, with N_P the total number of curve segments. Every second order B-spline curve segment Q_k is defined by three control points p_{k-1}, p_k, p_{k+1}. Note that every curve segment shares control points with its neighboring curve segments. Since P is an ordered set, the ordering rule and the set of control points uniquely define the curve. In general, the B-spline curve could be permitted to place its control points anywhere on the image plane. We restrict the possible locations for control points to a set A.

We assume that the control points of the curve are encoded differentially, which is an efficient method for natural boundaries since the locations of the control points are highly correlated. We denote the required bit rate for the differential encoding of control point p_{k+1}, given control point p_{k-1} and p_k, by $r(p_{k-1}, p_k, p_{k+1})$. Hence the bit rate $R(p_0, \ldots, p_{N_P+1})$ for the entire approximation curve is equal to

$$R(p_0, \ldots, p_{N_P+1}) = \sum_{k=0}^{N_P} r(p_{k-1}, p_k, p_{k+1}), \qquad (1)$$

where $r(p_{-1}, p_0, p_1)$ is set equal to the number of bits needed to encode the absolute position of the first two control points p_0 and p_1, which are identical. As mentioned before, the first two control points p_0 and p_1 are identical and so are the last two control points, p_{N_P} and p_{N_P+1}. This results in the fact that the B-spline approximation starts at p_0 and ends at p_{N_P+1}. Hence p_{N_P+1} does not need to be encoded and therefore $r(p_{N_P-1}, p_{N_P}, p_{N_P+1})$ is always zero. For a closed boundary, the first two control points are identical to the last two, hence the rate $r(p_{N_P-2}, p_{N_P-1}, p_{N_P})$ is also set to zero since the control point p_{N_P} does not need to be encoded. Note that the rate $r(p_{k-1}, p_k, p_{k+1})$ depends on the specific control point encoding scheme.

Besides the rate, we also need the curve segment distortion for our proposed curve approximation scheme, which we define as $d(p_{k-1}, p_k, p_{k+1})$. One popular distortion measure for curve approximation is the maximum absolute distance.

So far we have only discussed the segment distortion measures, i.e., the measures which judge the approximation of a certain partial boundary by a given curve segment. In general we are interested in a curve distortion measure $D(p_0, \ldots, p_{N_P+1})$ which can be used to determine the quality of approximation of an entire curve. As mentioned above, we are using the maximum absolute distance distortion measure. This can be expressed as follows, using the segment distortion measures defined above,

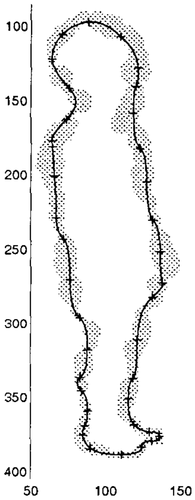

Fig. 1. An example of a tolerance band. The boundary approximation (solid line) and the corresponding control points (x) are also shown.

$$D(p_0, \ldots, p_{N_P+1}) = \max_{k \in [1, \ldots, N_P]} d(p_{k-1}, p_k, p_{k+1}). \qquad (2)$$

In our previous work, we defined a "distortion band" with width $2 \cdot D_{max}$ along the boundary B. The B-spline approximation must lie within the distortion band. In this paper, we allow the distortion band to have variable width along the boundary. We call this new band a *tolerance* band (an example is shown in Fig. 1). The definition of the tolerance band requires a D_{max} for every boundary point. We denote this as $D_{max}[i], i = 0, \ldots, N_B - 1$. In order to construct the tolerance band, we draw circles from each boundary point b_i with radius $D_{max}[i]$. The tolerance band consists of the set of all point that lie inside the circles.

We can now define the segment distortion measure which we will use by

$$d(p_{k-1}, p_k, p_{k+1}) = \begin{cases} 0: & \text{all points of } Q_k(p_{k-1}, p_k, p_{k+1}) \\ & \text{are inside the tolerance band} \\ \infty: & \text{any point of } Q_k(p_{k-1}, p_k, p_{k+1}) \\ & \text{is outside the tolerance band} \end{cases} \qquad (3)$$

This distortion measure takes a curve segment Q_k, given by the three control points p_{k-1}, p_k and p_{k+1}, as input and checks if the curve segment is inside the tolerance band.

As mentioned earlier, we define $D_{max}[i]$ in a way that is proportional to the significance of that segment or feature of the boundary. As an example, we define $D_{max}[i]$ to be inversely proportional to the image intensity gradient. The algorithm proceeds as follows: The gradient is first calculated for the whole image; that is, for an image $f(x, y)$, it is defined as:

$$\nabla f(x, y) = [\partial f / \partial x \; \partial f / \partial y]^T = [f_x \; f_y]^T. \qquad (4)$$

As an example, if the Sobel edge detector is used [8], estimates of the gradients are given by

$$\hat{f}_x = \mathbf{w}_1^T \mathbf{x}, \tag{5}$$

$$\hat{f}_y = \mathbf{w}_2^T \mathbf{x}, \tag{6}$$

where $\mathbf{x}$ is the vector containing image pels in a local image neighborhood and $\mathbf{w}_1$ and $\mathbf{w}_2$ are the Sobel edge detector masks, where,

$$\mathbf{w}_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \tag{7}$$

and

$$\mathbf{w}_2 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{8}$$

The magnitude of the gradient is then computed by

$$|\nabla f(x,y)| = \sqrt{f_x^2(x,y) + f_y^2(x,y)}. \tag{9}$$

The minimum and maximum of the magnitude of the image gradient for the whole image are computed next, denoted respectively as $gradmin$ and $gradmax$. Let us also denote the desired minimum and maximum values of $D_{max}[i]$ as T_{min} and T_{max}, respectively. Then, a linear mapping is performed between the gradient value of each boundary point and the width of the distortion band. If the magnitude of the gradient at the boundary point b_i is grad[i], then the width of the tolerance band at this point is given by:

$$D_{max}[i] = T_{min} + \lambda(grad[i] - gradmax) \tag{10}$$

where

$$\lambda = \frac{T_{max} - T_{min}}{gradmin - gradmax}. \tag{11}$$

In practice, we need to define a threshold for the gradient magnitude. The boundary points whose gradient magnitude exceeds the threshold should have the minimum possible $D_{max}[i]$. Clearly, $gradmax$ is equal to the threshold in that case. This mapping was used in generating the variable distortion band in Fig. 1.

Alternatively, the curvature of the boundary to be encoded could be used instead of the gradient of the image intensity. The curvature is defined as the change in tangent angle produced by an infinitesimal change of arc length. If the boundary is represented as a parametric curve $[r(t), c(t)]$, the curvature κ at $[r(t), c(t)]$ is computed by [9]

$$\kappa(t) = \frac{\frac{dr}{dt}(t)\frac{d^2c}{dt^2}(t) - \frac{d^2r}{dt^2}(t)\frac{dc}{dt}(t)}{[(\frac{dr(t)}{dt})^2 + (\frac{dc(t)}{dt})^2]^{3/2}}. \tag{12}$$

The width of the distortion band is set to be inversely proportional to the curvature, so that segments of the boundary with fine details (i.e., large curvature) are better preserved.

3. EXPERIMENTAL RESULTS

We coded both the intensity and shape of frame 0 of the "Kids" sequence shown in Fig. 2 and Fig. 3, respectively.

A number of experiments were conducted, some of which are reported below.

In one experiment, B-splines were used for the boundary approximation. The intensity of the objects was coded using Shape-Adaptive DCT with a Quantization Parameter (QP) equal to 32

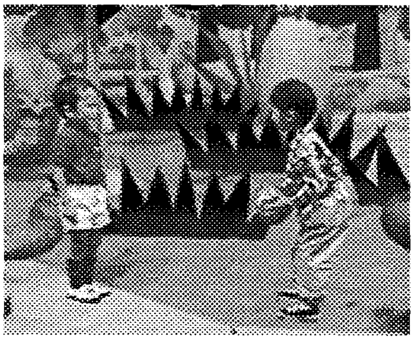

Fig. 2. Frame 0 of the "Kids" sequence

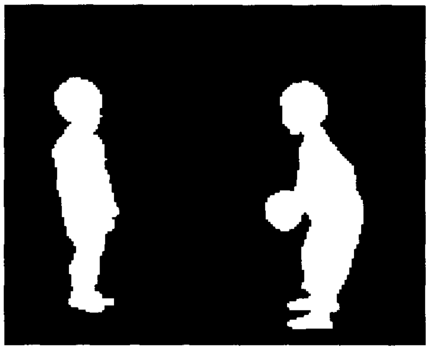

Fig. 3. Segmentation of frame 0 of the "Kids" sequence

Tol. Band	Shape Bits	Int. Bits	Total Bits	PSNR
1 pel	467	16699	17166	20.33
3 pels	323	16845	17168	20.45
Variable	345	16655	17000	20.34

Table 1. Results of the first experiment.

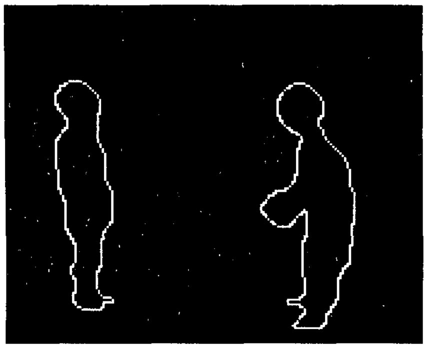

Fig. 4. Result of the variable width tolerance band

(coarse quantization). A fixed-width tolerance band of one and three pels, was used, as well as, a variable-width tolerance band which is inversely related to the gradient magnitude, as discussed in this paper ($T_{max} = 3$, $T_{min} = 0.8$), were used in encoding the shape information. In Table 1 the number of bits required for shape and intensity encoding are shown, along with the PSNR in dB. It can be seen that, using a variable distortion band, the same quality is obtained with fewer bits. Figure 4 shows the boundary approximation using a variable tolerance band as discussed in this paper. Figure 5 shows a result of a fixed width tolarance band width of 1 pel.

The above experiment was repeated using a QP of 32 and straight lines instead of B-splines to encode the boundaries. The results of this experiment are shown in Table 2. Similar observations as before are made.

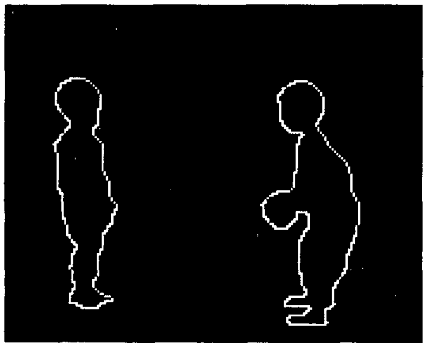

Fig. 5. Result of the fixed width tolerance band algorithm ($D_{max} = 1$)

Tol. Band	Shape Bits	Int. Bits	Total Bits	PSNR
1 pel	528	17112	17640	20.31
3 pels	290	16855	17145	20.43
Variable	343	16453	16796	20.43

Table 2. Results of the second experiment.

4. CONCLUSIONS

From the above experiments, we can see that use of a variable tolerance band results in savings in encoded bits when compared with the 1 pel wide tolerance band while preserving the important features of the shape. Furthermore, the encoding of the texture using SA-DCT is more efficient when the shape is encoded using a variable-width tolerance band than when a fixed band of width 1 pel is used. The 3 pel wide tolerance band results in savings in shape but not in texture encoding. For these reasons, be believe that the variable-width tolerance band has significant advantages over a fixed-width tolerance band.

5. REFERENCES

[1] H.G. Musmann, M. Hötter, and J. Ostermann, "Object-oriented analysis-synthesis coding of moving images," *Signal Processing: Image Communication*, vol. 1, no. 2, pp. 117–138, Oct. 1989.

[2] G. M. Schuster and A. K. Katsaggelos, *Rate-Distortion Based Video Compression, Optimal Video frame compression and Object boundary encoding*, Kluwer Academic Press, 1997.

[3] G. M. Schuster and A. K. Katsaggelos, "An optimal boundary encoding scheme in the rate distortion sense," *IEEE Transactions on Image Processing*, vol. 7, no. 1, pp. 13–25, Jan. 1998.

[4] 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, Nov. 1998.

[5] A. K. Katsaggelos, L. P. Kondi, F. W. Meier, J. Ostermann, and G. M. Schuster, "MPEG-4 and rate-distortion-based shape-coding techniques," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1126–1154, June 1998.

[6] L. P. Kondi, F. W. Meier, G. M. Schuster, and A. K. Katsaggelos, "Joint optimal object shape estimation and encoding," in *Proceedings SPIE Conf. on Visual Comm. and Image Proc.*, 1998, pp. 14–25.

[7] K. J. Kim, C. W. Lim, M. G. Kang, and K. T. Park, "Adaptive approximation bounds for vertex based contour encoding," *IEEE Transactions on Image Processing*, vol. 8, no. 8, pp. 1142–1147, Aug. 1999.

[8] A. K. Jain, *Fundamentals of digital image processing*, Prentice-Hall, 1989.

[9] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, vol. I, Addison-Wesley, 1992.