OBJECT-BASED VIDEO COMPRESSION SCHEME WITH OPTIMAL BIT ALLOCATION AMONG SHAPE, MOTION AND TEXTURE

Haohong Wang, Guido. M. Schuster, and Aggelos. K. Katsaggelos*

Image and Video Processing Lab (IVPL)
Department of Electrical and Computer Engineering
Northwestern University, Evanston, IL 60208, USA
Email: {haohong, aggk}@ece.northwestern.edu

* Abteilung Elektrotechnik
Hochschule für Technik, Rapperswil
CH-8040 Rapperswil, Switzerland
Email: guido.schuster@hsr.ch

ABSTRACT

In object-based video, the encoding of the video data is decoupled into the encoding of shape, motion and texture information, which enables certain functionalities like content-based interactivity and scalability. However, the problem of how to jointly encode these separate signals to reach the best coding efficiency has never been solved thoroughly. In this paper, we present an operational rate-distortion optimal bit allocation scheme that provides a solution to this problem. Our approach is based on the Lagrangian relaxation and dynamic programming. Experimental results indicate that the proposed optimal encoding approach has considerable gains over an ad-hoc method without optimization. Furthermore the proposed algorithm is much more efficient than exhaustive search.

1. INTRODUCTION

In recent years, object-based video coding has become one of the most important topics in the visual communication field, as the fast growing modern multimedia applications require a number of new functionalities in addition to the conventional efficient video compression. For example, in order to allocate limited bandwidth resources to different semantic parts of a scene and to satisfy the needs of each individual mobile multimedia applications, like videoconferencing and computer games, require content-based interactivity and content-based scalability.

Compared with the conventional frame-based video, which is represented by encoding a sequence of rectangular frames, object-based video coding is based on the new concept of encoding arbitrarily shaped video objects. In object-based video compression, the optimal bit allocation among shape, texture, and motion is not only a fundamental problem, but also a bottleneck for solving other important optimization problems related to source coding. For example, in joint source-channel coding, in order to get an optimal unequal (for shape and texture) error protection scheme, the optimal source-coding scheme needs to be found in the first place.

The difficulty of the optimal bit allocation problem lies in that the coding of an object shape and the coding of its texture have certain dependencies on each other. Furthermore, in the process of coding an object texture, the adoption of predictive coding and motion compensation increases the dependencies on macroblocks within the VOP, and makes the problem even more complicated. Therefore, a jointly optimal coding scheme of shape, texture and motion needs to be developed. In [1], an optimal vertex-based shape encoder is proposed, taking into consideration the texture information of the video frames. In [2], a joint shape and texture rate control algorithm is proposed for MPEG-4 encoders. However, none of these approaches provides an optimal solution to the bit allocation problem.

In this paper, we are proposing an operational rate-distortion optimal bit allocation scheme for object-based video coding. Our algorithm is based on Lagrangian relaxation and dynamic programming. We implemented our scheme on the MPEG-4 verification model. The rest of the paper is organized as follows. The next section provides a brief overview of object-based video coding approaches. In section 3, the problem is formally defined. Section 4 demonstrates the optimal solution. Section 5 describes some implementation issues and section 6 reports the experimental results. We draw conclusions in the last section.

2. OBJECT-BASED VIDEO CODING

The object-based video encoder essentially consists of separate encoding schemes for shape and texture. The encoding method for texture is quite mature. However, shape coding is still a relatively new research topic with new approaches being reported recently [3]. The shape coding methods can be classified into bitmap-based encoding and contour-based encoding. The former method encodes for each pixel whether it belongs to the object or not, while the latter one encodes the outline of the object. This paper is focusing on the bitmap-based encoding method adopted by MPEG-4. We assume the binary shape information is coded utilizing the macroblock-based structure, where binary alpha data are grouped within 16x16 binary alpha blocks (BAB).

The BAB can be coded in various ways. The CAE (Context-based Arithmetic Encoding) method of MPEG-4 uses a template of 10 pixels to define the context for predicting the alpha value of the current pixel. A probability table is predefined for the context of each

pixel. The sequence of pixels within the BAB drives an arithmetic encoder with its pair of alpha value and probability. Note that the concept of CAE makes the encoding of a BAB depend on its neighbors to the left, above, and above-left.

The texture coding approaches are similar for almost all of the existing video coding standards, where VOPs are divided into 8x8 blocks followed by 2D 8x8 DCT transforms. The resulting DCT coefficients are quantized and entropy coded. For P-VOP, the texture data are predictively coded, that is, for each block, a motion vector and the corresponding motion-compensated residue data are generated after the motion estimation. The motion vectors are coded differentially, while the residue data are coded as that of the intra-coded texture data mentioned above. It is important to note that the differential coding of the motion vector makes the encoding of current macroblock texture depend on its neighbor macroblocks.

3. PROBLEM FORMULATION

The problem at hand is to control both the shape and texture coding parameters to minimize the total (shape, motion plus texture) bit rate required to transmit a video sequences at some acceptable level of quality. We can write this optimization formally as

$$\text{Minimize } R, \text{ subject to } D \le D_{max}, \quad (1)$$

where R is the total bit rate per frame, D is the distortion, and D_{max} is the maximum tolerable distortion. The same techniques can be applied to solve the dual problem, that is,

$$\text{Minimize } D, \text{ subject to } R \le R_{budget}, \quad (2)$$

where R_{budget} is the bit budget.

Our study of the optimal bit allocation among shape, motion and texture is restricted to the frame level. In other words, we do not attempt to optimally allocate the bits among the different frames of a video sequence. The reader interested in that problem is referred to [4].

A. Rate

Let us denote by $\{m_1, m_2, \ldots, m_M\}$ the total M macroblocks in the frame, and s_i the associated shape data of m_i. Let us denote by $V=\{V_1, V_2, \ldots, V_M\}$ the set of admissible shape decision vectors for the macroblocks, and v_i a shape decision vector for m_i ($v_i \in V_i$). Then s_i can be represented by $s_i=g_i(v_{i-a}, \ldots, v_i)$, where g_i is a shape function and a is the number of previous macroblocks it depends on. Also, let us denote by $R_{Si}(v_{i-a}, \ldots, v_i)$ the shape bit rate for m_i.

In encoding the texture, not all the macroblocks in the VOP need to be coded. That is, for those macroblock outside the shape boundary, the encoding process can be skipped, because the composition process of the decoder will eventually remove them. Let us denote by $W=\{W_1, W2, \ldots, W_M\}$ the set of admissible texture decision vectors (which include the motion vectors), w_i a texture decision vector for m_i ($w_i \in W_i$), and $R_{Ti}(s_i, w_{i-b}, \ldots, w_i)$ the texture rate for macroblock m_i, where b is the number of previous macroblocks that m_i depends on. Clearly,

$$R = R_{syntax} + \sum_{i=1}^{M} [R_{Si}(v_{i-a}, \ldots, v_i) + R_{Ti}(s_i, w_{i-b}, \ldots, w_i)], \quad (3)$$

where R_{syntax} represents the bits allocated on the data structure syntax of the VOP.

B. Distortion

We use the mean-squared error (MSE) as the distortion metric. Let us denote by $D_i(s_i, w_i)$ the distortion for macroblock m_i. Clearly,

$$D = \sum_{i=1}^{M} D_i(s_i, w_{i-b}, \ldots, w_i) \quad (4)$$

and

$$D_i = \sum_{x=0}^{15}\sum_{y=0}^{15} d_{i,Y}(x,y)^2 + \sum_{x=0}^{7}\sum_{y=0}^{7} [d_{i,U}(x,y)^2 + d_{i,V}(x,y)^2], \quad (5)$$

where $d_{i,Y}(x,y)$, $d_{i,U}(x,y)$, and $d_{i,V}(x,y)$ are differential intensity values for the Y, U and V components at pixel (x,y). The peak signal-to-noise ratio (PSNR) in dB can be obtained by

$$D_{PSNR} = 10 \log_{10} \frac{1.5 \times M \times 16^2 \times 255^2}{D}, \quad (6)$$

where the factor 1.5 comes from the downsampling of the chrominance components by a factor of 2.

4. OPTIMAL SOLUTION

We define a set of admissible decision vectors $U=V \times W$, and for each macroblock m_i, its associated decision vector $u_i=(v_i, w_i)$. Then (1) can be rewritten and simplified as

$$\text{Minimize } \sum_{i=1}^{M} [R_i(u_{i-a}, \ldots, u_i)],$$

$$\text{such that } \sum_{i=1}^{M} D_i(u_{i-a}, \ldots, u_i) \le D_{\max}, \quad (7)$$

where $R_i(u_{i-a}, \ldots, v_i) = R_{Si}(v_{i-a}, \ldots, v_i) + R_{Ti}(s_i, w_{i-b}, \ldots, w_i)$, and $D_i(u_{i-a}, \ldots, u_i) = D_i(s_i, w_{i-b}, \ldots, w_i)$ (without loss of generality, we assume that $a \ge b$).

We derive a solution to problem (7) using the Lagrange multiplier method to relax the constraint, so that the relaxed problem can be solved using a shortest path algorithm. The same steps can be followed in solving the dual problem (2). We first define the Lagrangian cost function

$$J_\lambda(u) = R + \lambda D = R_{syntax} + \sum_{i=1}^{M} [R_i(u_{i-a}, \ldots, u_i) + \lambda D_i(u_{i-a}, \ldots, u_i)], \quad (8)$$

where λ is the Lagrange multiplier. It has been shown in [5] and [6] that if there is a λ^* such that $u^* = \arg[\min_u J_{\lambda^*}(u)]$, and which leads to $D=D_{max}$, then u^* is also an optimal solution to (7). It is well known that when λ sweeps from zero to infinity, the solution to (7)

traces the convex hull of the operational rate distortion function, which is a non-increasing function. Hence, bisection or the fast convex search presented in [7] can be used to find λ^*. Therefore, if we can find the optimal solution to the unconstrained problem

$$\min \sum_{i=1}^{M} [R_i(u_{i-a},...,u_i) + \lambda D_i(u_{i-a},...,u_i)], \quad (9)$$

we can find the optimal λ^*, and the convex hull approximation to the constrained problem (7).

To implement the algorithm for solving the optimization problem (9), we create a cost function $C_k(u_{k-a},..., u_k)$, which represents the minimum total rate and distortion up to and including macroblock m_k given that $u_{k-a},...,u_k$ are decision vectors for macroblocks $m_{k-b},..., m_k$. Clearly,

$$J_\lambda(u) = \min_{u_{M-a},...,u_M} C_M(u_{M-a},...,u_M). \quad (10)$$

The key observation for deriving an efficient algorithm is the fact that given $a+1$ decision vectors $u_{k-a-1}, ..., u_{k-1}$ for macroblocks $m_{k-a-1},...,m_{k-1}$, and the cost function $C_{k-1}(u_{k-a-1},...,u_{k-1})$, the selection of the next decision vector u_k is independent of the selection of the previous decision vectors $u_1, u_2, ..., u_{k-a-2}$. This is true since the cost function can be expressed recursively as

$$C_k(u_{k-a},...,u_k) = \min_{u_{k-a-1}}[C_{k-1}(u_{k-a-1},...,u_{k-1})] + R_k(u_{k-a},...,u_k) + \lambda D_k(u_{k-a},...u_k). \quad (11)$$

The recursive representation of the cost function above makes the future step of the optimization process independent from its past step, which is the foundation of dynamic programming.

The problem can be converted into a graph theory problem of finding the shortest path in a directed acyclic graph (DAG) [7]. The computational complexity of the algorithm is $O(M \times |U|^{max(a,b)+1})$ ($|U|$ is the cardinalities of U), which depends directly on the value of a and b, but still much more efficient than the exponential computational complexity of an exhaustive search algorithm.

5. IMPLEMENTATION ISSUES

We implemented the proposed optimal bit allocation scheme based on the MPEG-4 verification model [8].

In MPEG-4, each BAB can be assigned a mode from seven options, transparent, opaque, intra CAE coded, inter CAE coded w/o MVD (Motion Vector Difference), etc. The BAB can have a lossy representation by successively downsampling with a conversion ratio factor (CR) of two or four, and following an upsampling back to the full-resolution. In Inter-mode, the motion vector (MV) of a BAB is decoded from a motion vector predictor (MVP) and an offset MVD (MVD=MV-MVP), where MVP is taken from an ordered list of candidate motion vectors. The selected MV is the one that is close to MVP and minimizes the16x16 motion compensation error computed by comparing the BAB indicated by MV and current BAB.

In MPEG-4, the texture content of the macroblock's bitstream depends to a great extent on the reconstructed shape information.

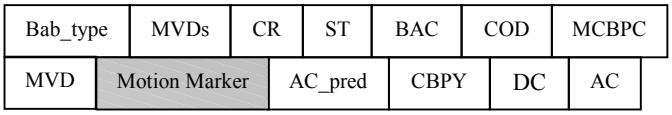


Figure 1. Data partitioned bitstream syntax for P-VOP

Our optimization work addresses the making of decisions on the coding parameters so that the video is coded to meet certain constraints in video quality, bit rate, etc. Since MPEG-4 only standardizes the video decoding, our work is therefore based on the standard decoder structure. From the video data structure shown in Fig. 1, the adjustable parameters could only include bab_type, CR, ST, MCBP, CBPY, MVDs, MVD, and quantization step size, which decides DC and AC data. We assume that the M macroblocks are arranged in K columns, and they are numbered following the horizontal scan order. Every macroblock has a bab_type $b_i \in B_i$, an MVDs $ms_i \in MS_i$, a CR $c_i \in C_i$, an ST $s_i \in S_i$, an MCBP $p_i \in P_i$, an MVD $mv_i \in MV_i$, and a CBPY $y_i \in Y_i$ associated with it, where B_i is the set of all admissible BAB type for m_i, MS_i is the set of all admissible shape motion vectors for m_i, C_i is the set of all admissible CR for m_i, S_i is the set of all admissible ST for m_i, P_i is the set of all admissible MCBP for m_i, MV_i is the set of all admissible texture motion vector for m_i, and Y_i is the set of all admissible CBPY for m_i. Let us define a decision vector $u_i=[b_i, ms_i, c_i, s_i, p_i, mv_i, y_i] \in U_i$ for every m_i. Then $U_i = B_i \times MS_i \times C_i \times S_i \times P_i \times MV_i \times Y_i$ is the admissible decision vector set for m_i. The encoding of current macroblock will only depend on macroblock to the left, above, above-left and above-right. So, by substituting a with $K+1$ in (9), we are solving the problem of $\min \sum_{i=1}^{M} [R_i(u_{i-K-1},...,u_i) + \lambda D_i(u_{i-K-1},...,u_i)]$ using the proposed algorithm.

6. EXPERIMENTAL RESULTS

A number of experiments have been conducted, some of which are reported here. We encoded the first four frames of the "Akiyo" sequence as I-P-P-P VOPs using the proposed optimal approach and MoMuSys (a software implementation of MPEG-4 Video Verification Model) by exhaustively trying all combinations of the parameters (Alpha_TH and QP). Figure 2 shows the R-D curves for both experiments. Clearly, our result in addition to providing solutions on the convex hull of all operating points, also demonstrates some gains in RD quality, due to the selection of adjustable parameters to get better extended shape and texture data.

Figure 3 shows a comparison of the reconstructed first two "Bream" frames obtained using the proposed optimal approach (see Fig. 3(a)) and using MoMuSys without

optimization (see Fig. 3(b)) by given a bit budget of *7,000* bits. The PSNR gained by optimization is over *6dB*.

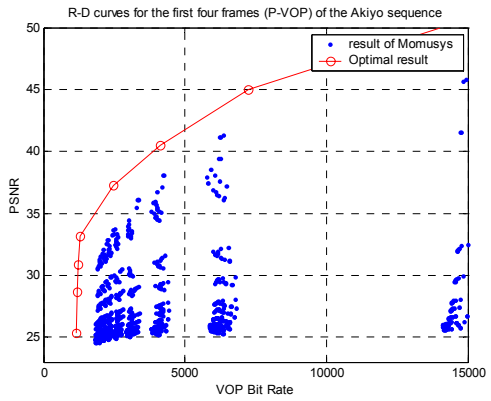
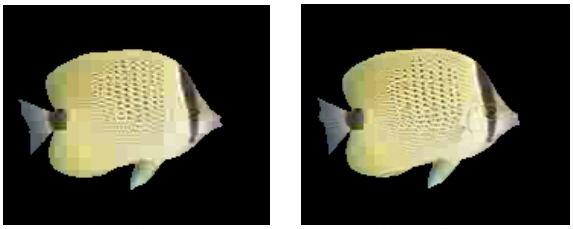
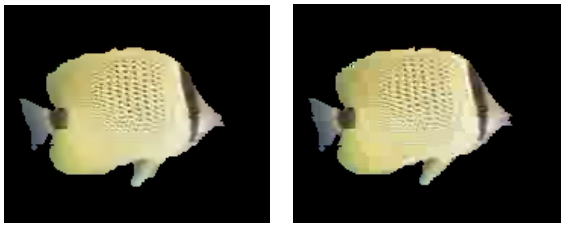


Figure 2. Comparison of R-D curves



Frame 0 (I-VOP), PSNR=28.22 *Frame 1 (P-VOP), PSNR=31.42*

(a) Results obtained using optimal approach



Frame 0 (I-VOP), PSNR=22.09 *Frame 1 (P-VOP), PSNR=22.41*

(b) Results obtained using MoMuSys without optimization

Figure 3. Comparison of reconstructed image quality

Figure 4 shows the R-D curves for intra and inter modes and corresponding bit allocation between shape and texture obtained using the proposed optimal approach for encoding the first *30* frames of the "Bream" sequence. It is easy to notice from the left figure that the Inter-mode encoding saves significant amount of bits compared to the Intra-mode encoding as expected. In Intra-mode encoding, the shape bits increase rapidly from *300* to around *800* when the PSNR is between *20* and *28dB*, while the texture bits stay at around *3,200*. However, when the PSNR keeps increasing to *42.7dB*, the texture bits go up steeply from *3,200* to over *15,000* and shape bits only change slightly. A similar behavior is observed for inter-mode encoding. As it can also be inferred from the figure, shape information plays a more important role in Inter-mode encoding (consuming up to 34% of the total bit budget) than in Intra-mode. In Intra-mode encoding, although shape information only occupies 0.3-20% of the total VOP bits, it still has a strong impact on the quality of the video, demonstrating the importance of optimal bit allocation between shape and texture.

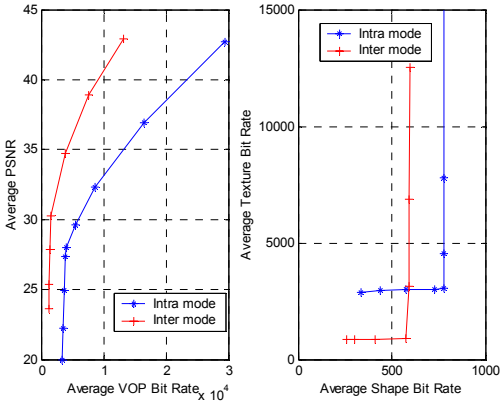


Figure 4. R-D curve and corresponding bit allocations

7. CONCLUSIONS

In this paper, we presented an operational rate-distortion optimal bit allocation scheme among shape, motion and texture for the encoding of object-based video. By applying the Lagrange multiplier method and dynamic programming, our optimal approach has considerable computational savings over an exhaustive search algorithm. Our experimental results indicate that the optimal joint shape and texture encoding has considerable quantitative gains over an ad-hoc encoding method without optimization. Our work not only improves the source coding efficiency, but also makes possible the deeper understanding of the relative importance of shape and texture.

REFERENCES

[1] L. P. Kondi, G. Melnikov, and A. K. Katsaggelos, "Joint optimal coding of texture and shape", in *Proc. IEEE International Conference on Image Processing*, Volume III, pp. 94-97, Thessaloniki, Greece, October 2001.

[2] A. Vetro, H. Sun, and Y. Wang, "Joint shape and texture rate control for MPEG-4 encoders", in *Proc. IEEE International Conference on Circuits and Systems*, pp. 285-288, Montery, USA, June 1998.

[3] A. K. Katsaggelos, L. Kondi, F. W. Meier, J. Ostermann, and G. M. Schuster, "MPEG-4 and rate distortion based shape coding techniques", *Proc. IEEE*, pp.1126-1154, June 1998.

[4] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders", *IEEE Trans. Image Processing*, Vol. 3, pp. 533-545, Sept. 1994.

[5] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources", Oper. Res., vol. 11, pp. 399-417, 1963.

[6] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1445-1453, Sept. 1988.

[7] G. M. Schuster and A. K. Katsaggelos, *Rate-Distortion based video compression: optimal video frame compression and object boundary encoding*, Kluwer Academic Publishers, 1997.

[8] *MPEG-4 video VM 18.0, ISO/IEC JTC1/SC29/WG11 N3908*, Pisa, Jan. 2001.