RATE-DISTORTION OPTIMIZED STREAMING FOR 3-D WAVELET VIDEO

Chuo-Ling Chang, Sangeun Han and Bernd Girod

Information Systems Laboratory, Department of Electrical Engineering, Stanford University
{chuoling,sehan,bgirod}@Stanford.EDU

ABSTRACT

We propose a rate-distortion optimized framework to stream scalable bitstreams of 3-D wavelet video stored at the sender to a remote receiver. Based on the source rate-distortion profiles, the desired playout deadline and transmission rate, and the network characteristics, the receiver issues customized requests throughout the video playout session in order to retrieve the data that minimize the distortion in the reconstructed frames. Rate-distortion optimized data request is formulated as a convex optimization problem. Experimental results show that the proposed scheme improves the video quality by up to 3.1 dB over the heuristic scheme at the same rate, or correspondingly reduces the required the transmission rate by up to 60% for the same quality.

1. INTRODUCTION

Over the years, many researchers have proposed 3-D wavelet coding of video sequences. Thanks to the multi-resolution nature of wavelet transforms as well as efficient embedded coding of the wavelet coefficients, 3-D subband video coding provides great support for scalability, a very desirable feature when transmitting video over the network. However, linear transforms applied in the temporal direction may be inefficient if the motion between frames is not fully exploited.

Many attempts have been made to incorporate motion compensation into the 3-D wavelet video coding framework [1]-[3]. Earlier works are somewhat unsatisfactory in terms of the rate-distortion coding performance because the motion vector field is severely restricted and the temporal transform is usually limited to the two-tap Haar wavelet. Recently, a technique called *motion-compensated lifting* has been proposed [4]-[6], which successfully incorporates unrestricted motion compensation into 3-D wavelet coding and provides compression efficiency approaching the state-of-the-art predictive video coding schemes. However, despite the increasing interest in 3-D wavelet video, efficient streaming of such data sets that exploits the rate-distortion performance as well as the inherent support for scalability is seldom addressed.

In [7], Chou and Miao proposed a framework for rate-distortion optimized packet scheduling of video and audio data. In general, it can be applied to streaming of the 3-D wavelet video data set. However, in their framework, the data set has to be assembled into packets before the optimization for packet scheduling takes place. Therefore, the packet content cannot dynamically adapt to the network characteristics as well as the state of the data previously transmitted to the receiver. Additionally, the optimization process is formulated as a combinatorial problem which requires high complexity to solve.

We have previously proposed a rate-distortion optimized framework for interactive streaming of light fields coded as scalable bitstreams [8][9]. For streaming of 3-D wavelet video, we follow a similar approach in this work. The source rate-distortion profiles, the desired playout deadline and transmission rate, and the network characteristics are all taken into account to optimize the streaming strategy. Due to the fine scalability property of embedded wavelet coefficient coding, the optimization process is approximated as a convex optimization problem, which can be efficiently solved by standard optimization techniques.

The remainder of the paper is organized as follows: In Section 2, we describe the structure of the 3-D wavelet video coding scheme adopted in this work. In Section 3, we present the proposed framework for streaming the 3-D wavelet video over networks. Finally, experimental results comparing to a heuristic scheme are presented in Section 4.

2. MOTION-COMPENSATED 3-D WAVELET VIDEO CODING

In this work, a 3-D wavelet coder using motion-compensated lifting is adopted to encode the video sequence [4]-[6]. Lifting is a procedure that can be used to implement discrete wavelet transforms (DWT). Wavelet analysis can be factorized into one or more lifting steps, each consisting of a prediction and an update filter. For reconstruction, wavelet synthesis can be similarly implemented with the inverse lifting procedure. As long as the prediction and update filters in wavelet synthesis are identical to those in wavelet analysis, the reversibility of the transform is ensured.

For 3-D wavelet video coding, motion compensation is incorporated into the prediction and update filters. A multi-level temporal DWT implemented using motion-compensated lifting is first applied across the video frames to decompose them into temporal subbands. To further exploit the coherence among neighboring pixels within each temporal subband, a multi-level 2-D spatial DWT is then applied to decompose the subband into wavelet coefficients. The SPIHT (Set Partitioning in Hierarchical Trees) [10] algorithm is finally used to encode the wavelet coefficients of each subband into a scalable bitstream. The SPIHT algorithm provides a scalable representation so that different reconstruction qualities of the wavelet coefficients, hence different reconstruction qualities of the video frames, can be obtained by truncating the coded bitstreams at different lengths.

To decode a particular video frame, only a few subbands relevant to synthesizing the frame need to be reconstructed. The truncated bitstreams of these subbands available at the decoder are decoded into reconstructed wavelet coefficients by the inverse SPIHT algorithm. Then the inverse 2-D spatial DWT is applied to reconstruct the temporal subbands. Finally, the inverse motion-compensated lifting procedure is performed to carry out the inverse temporal DWT in order to reconstruct the video frame.

0-7803-8554-3/04/$20.00

3. RATE-DISTORTION OPTIMIZED STREAMING FOR SCALABLE VIDEO

We consider the video streaming scenario where the video sequence consisting of N frames, F_n, $n = 1, \cdots, N$, is encoded as N scalable bitstreams stored at the sender. Each bitstream corresponds to a temporal subband S_n. The viewer at a remote receiver constantly issues requests to retrieve the appropriate data from the sender via the network throughout the playout session. In this paper, we do not consider streaming of the motion vectors and control information and assume that they are transmitted to the receiver before the playout session. Assume that the viewer starts the playout session at time instant t_{st}, and one frame is decoded every time duration T_d. Given the required playout delay, D_{max}, from t_{st} to the time data required for reconstructing the first frame F_1 are being decoded, a sequence of *decoding instants*, $\{d_k\}$, is defined as $d_k = t_{st} + D_{max} + (k-1) \cdot T_d$ where $k \in \mathcal{Z}^+$. As a result, data required for reconstructing F_k are decoded at $\{d_k\}$.

We also define a sequence of *requesting instants*, $\{r_i\}$, where $r_{i+1} = r_i + T_r$ and T_r denotes a fixed time duration. A requesting instant r_i is associated with a window of frames and hence a set of subbands related to reconstructing these frames, denoted as $\{S_n : n \in \mathcal{Z}_i\}$. At r_i, the receiver initiates a request for the data of the associated subbands from the sender. The request is optimized such that the resulting distortion in the reconstructed video sequence after receiving the requested data is minimized. It takes T_{proc} processing time to generate the request, and the request is fragmented into N_p requesting packets issued at $r_i + T_{proc}$ through the backward channel of the network. Upon receiving a requesting packet, the sender transmits a responding packet containing the requested data to the receiver through the forward channel. The q-th requesting packet from the i-th request, where $q = 1, \cdots, N_p$, and the corresponding responding packet are indexed by $(i-1) \cdot N_p + q$.

Both the backward and the forward channel are modelled as an independent time-invariant packet erasure channel with random delays. Each packet is delayed or lost independently from other packets. The cumulative distribution function of the round trip time (RTT) is denoted as $F_{RTT}(\tau) = Pr\{RTT \leq \tau\}$. Additionally, we denote the target forward payload transmission rate by C_f, measured in bit-per-second. Given C_f and T_r, the payload length in each response is determined by $L = \lfloor T_r \frac{C_f}{8} \rfloor$ in bytes.

3.1. Packet Buffer and Decoding Buffer

The request contained in a requesting packet as well as the content of the responding packet can be described by two N-vectors, $\mathbf{s}$ and $\mathbf{e}$. The n-th element of $\mathbf{s}$ and $\mathbf{e}$, s_n and e_n, denote the preceding and ending position (in bytes) of the bitstream for S_n contained in the packet respectively. In other words, the bitstream segment for S_n starts at byte $s_n + 1$ and ends at byte e_n of the bitstream.

Between two adjacent decoding instants, the receiver continuously receives packets and keeps them in the *packet buffer* which holds up to N_{pb} most recently received packets. In addition, the *decoding buffer* stores the bitstreams for the N_{db} most recently referred subbands. The decoding buffer state can also be described with an N-vector, $\mathbf{b}$, where the n-th element, b_n, denotes the length (in bytes) of the bitstream for S_n in the decoding buffer.

At d_k, we repeatedly update the decoding buffer by the packets in the packet buffer in the ascending order of their index. A bitstream segment in the packet can update the decoding buffer only if it satisfies all of the following three conditions. First, the segment belongs to one of the N_{db} most recently referred subbands (including those for reconstructing F_k). Second, all the bits in the bitstream preceding the segment are already in the decoding buffer, i.e., $s_n \leq b_n$. Third, the segment extends the bitstream already in the decoding buffer, i.e., $e_n > b_n$. After d_k, the packets are still kept in the packet buffer since they may be needed in later decoding instants, for instance, for those bitstream segments contained in the packets that later satisfy the above conditions.

3.2. Rate-Distortion Optimized Data Request

We define the rate for coding S_n as the length of the bitstream decoded to reconstruct S_n. The distortion is defined as the mean squared reconstruction error of the reconstructed S_n. During SPIHT-encoding for S_n, rate and distortion are recorded whenever there is a change in the distortion due to the rate increment, resulting in a sequence of $(R_n^{j_n}, D_n^{j_n})$ pairs with $j_n \in \{0, \cdots, j_n^{max}\}$ indexing the recording order. $R_n^{j_n}$ is expressed in bytes but is not necessarily an integer. Here we first assume that these R-D pairs are transmitted to the receiver before the playout session.

The first R_n^0 bytes of a bitstream contain essential header information for the bitstream to be decodable. Therefore, we assume that these initial bytes are transmitted separately before the playout session. The viewer can always reconstruct a subband at the lowest quality when no subsequent bitstream segments are received.

To optimize the i-th request, we assume it is given that the first $\hat{b}_n$ bytes of the bitstream for S_n, where $\hat{b}_n \in \{R_n^{j_n}\}$, are already in the decoding buffer by the time the responding packets update the decoding buffer. In general, the decoding buffer state in the future is a variable. Prediction of the state will be discussed in Section 3.4.

At r_i, the task of rate-distortion optimized data request is to determine the length of the additional bitstream segment for each S_n where $n \in \mathcal{Z}_i$ to be requested in order to minimize the expected total distortion in the reconstructed video sequence, subject to the payload length constraint.

To associate the distortion in the reconstructed frames to that in the reconstructed subbands, we make several simplifications and assumptions. By neglecting the effects from motion compensation, the inverse lifting procedure applied to synthesize a frame from the temporal subbands is identical to ordinary temporal wavelet synthesis. Therefore, a reconstructed frame can be modelled as a linear combination of the reconstructed subbands. We denote the wavelet synthesis filter coefficient associated to a input subband S_n and an output frame F_k by w_k^n. Note that only a few of such coefficients are nonzero for synthesizing F_k, depending on the decomposition levels and the wavelet kernel applied for the temporal DWT. We further assume that the distortion in a reconstructed temporal subband is proportional to that in its spatially decomposed wavelet coefficients. In addition, assume the reconstruction error in one subband is uncorrelated to that in any other subband, and hence the distortions from different subbands are additive in the reconstructed frame. Based on these assumptions, minimizing the expected total distortion is equivalent to:

$$\text{minimize} \sum_{k=1}^{N} F_{RTT}(d_k - r_i - T_{proc}) \sum_{n \in \mathcal{Z}_i} (w_k^n)^2 D_n^{j_n} \quad (1a)$$

$$\text{subject to} \sum_{n \in \mathcal{Z}_i} (R_n^{j_n} - \hat{b}_n) \leq L \quad (1b)$$

$$R_n^{j_n} \geq \hat{b}_n, \quad n \in \mathcal{Z}_i \quad (1c)$$

Note that the distortion in F_k is weighted by $F_{RTT}(d_k - r_i - T_{proc})$ since a packet responding to the i-th request contributes to the distortion reduction in F_k only if it arrives by d_k. The unknown variables in (1) are j_n for every $n \in \mathcal{Z}_i$ where $j_n \in \{0, \cdots, j_n^{max}\}$ determines the final bitstream length $\hat{R}_n = R_n^{j_n}$ and its corresponding distortion $\hat{D}_n = D_n^{j_n}$ for S_n.

Finally the optimized request is fragmented into N_p requesting packets, and each is responded by one responding packet from the sender. Ideally, the fragmentation is applied such that the bitstream segment for a subband is entirely contained in a single packet to avoid dependencies among the N_p packets. However, we set a maximum packet length in order to avoid further packet fragmentation incurred in the network. Thus a bitstream segment too long to fit into a single packet is then divided into several segments, and each is requested in a different requesting packet.

3.3. Convex Optimization Approximation

The formulation in (1) is a combinatorial optimization problem. To reduce the complexity, we approximate the formulation by a convex optimization problem. Specifically, since the recorded $(R_n^{j_n}, D_n^{j_n})$ pairs are usually densely located and they approximately form a decreasing convex function they can be well fitted with a weighted sum of V terms of exponential functions, resulting in the continuous distortion-rate function $D_n(R_n)$:

$$D_n(R_n) = \sum_{v=1}^{V} c_{n,v} \cdot exp(-\lambda_{n,v} \cdot R_n), \quad c_{n,v} \geq 0 \qquad (2)$$

$D_n(R_n)$ is a convex function. Additionally, it has analytically derivable gradient and Hessian, which greatly facilitate the optimization process. Furthermore, only $2V$ parameters instead of a long list of R-D pairs for each subband now need to be transmitted.

If the first $\tilde{b}_n$ bytes of the bitstream for S_n are already in the decoding buffer, (1) can be replaced by:

$$\text{minimize} \sum_{k=1}^{N} F_{RTT}(d_k - r_i - T_{proc}) \sum_{n \in \mathcal{Z}_i} (w_k^n)^2 D_n(R_n) \qquad (3a)$$

$$\text{subject to} \sum_{n \in \mathcal{Z}_i} (R_n - \tilde{b}_n) \leq L \qquad (3b)$$

$$\tilde{b}_n \leq R_n \leq R_n^{j_n^{max}}, \quad n \in \mathcal{Z}_i \qquad (3c)$$

The formulation in (3) consists of a convex cost function with linear constraints, thus formulates a convex optimization problem that approximates the original problem in (1). This formulation can be solved efficiently using standard optimization techniques. To conform with the original problem, the optimal solution, R_n, should be rounded to the nearest $R_n^{j_n}$. However, for efficient signalling of the request, we round it to the nearest integer and denote the result by $\tilde{R}_n$. This also makes $\tilde{b}_n$ an integer.

3.4. Decoding Buffer State Prediction

To request the bitstream segment of S_n, we choose to optimize the i-th request according to the predicted decoding buffer state at decoding instant $d_{k(i,n)}$, i.e., use this prediction as the $\tilde{b}_n$ in (3). $d_{k(i,n)}$ is defined as the first among the decoding instants at which S_n needs to be decoded and by which the packets responding to the i-th request is more probable to arrive than not, i.e.,

$$k(i,n) = \min\{k : w_k^n \neq 0, F_{RTT}(d_k - r_i - T_{proc}) > 0.5\} \quad (4)$$

We assume that there are N_u packets that have been previously requested but have not yet arrived at the receiver at r_i. The indices of these packets are denoted by $u(q)$, $q = 1, \cdots, N_u$. For each of these packets, the probability of its arrival by a particular decoding instant d_k given it has not yet arrived at r_i can be expressed by:

$$P_a(i,k,u(q)) = Pr\{RTT \leq d_k - r_{\lceil u(q)/N_p \rceil} - T_{proc} \mid RTT > r_i - r_{\lceil u(q)/N_p \rceil} - T_{proc}\} \quad (5)$$

There are 2^{N_u} possible arrival combinations of these packets regarding to d_k, and each results in a possible decoding buffer state. Since packet delay and losses are assumed to be independent across packets, the probability of each possible decoding buffer state is simply the product of that of the outcome of each packet.

Ideally, optimization should be applied for each possible state at $d_{k(i,n)}$, and the request that results in the minimum expected distortion calculated over all possible arrival combinations is chosen. However, this can lead to a long processing time since N_u can become large. Observing that the probability distribution of the possible states is highly skewed, the unlikely states can therefore be neglected. For simplicity, we choose the most probable decoding buffer state at $d_{k(i,n)}$ as the prediction. Consequently, the packet with index $u(q)$ is predicted to arrive by $d_{k(i,n)}$ if and only if $P_a(i, k(i,n), u(q)) > 0.5$.

At each r_i, a *predictive buffer* is created at the receiver. The buffer first duplicates the current state of the decoding buffer as the initial state. To further predict the decoding buffer state for S_n at $d_{k(i,n)}$, the predictive buffer is sequentially updated, in the same way as the decoding buffer, by the packets already in the packet buffer and the not-yet-arrived packets with $P_a(i, k(i,n), u(q)) > 0.5$. Finally, rate-distortion optimized data request in (3) is carried out with $\tilde{b}_n$ being the predictive buffer state.

Furthermore, we make a simplification that the bitstream segment for S_n does not contribute to the distortion reduction in F_k if $\tilde{b}_n$ does not match the actual state b_n at d_k. Therefore, $D_n(R_n)$ in (3a) is further weighted by the probability that the predictive state $\tilde{b}_n$ actually occurs, at each d_k.

4. EXPERIMENTAL RESULTS

Experimental results are shown for the video sequence *Foreman* and *Mobile*, both consisting of 288 frames in QCIF format. The probability density function of the packet delay, given the packet is not lost, is shifted-gamma distributed with shift $\kappa = 10$ ms, mean $\mu = 40$ ms, variance $\sigma^2 = 300$ ms^2, and the packet loss rate is set to 2%, for both the forward and the backward channel. The 4-level Haar wavelet transform is adopted for the temporal DWT, resulting in $N = 288$ temporal subbands. We choose the parameters $V = 5$, $N_{db} = 16$, $N_{pb} = 200$, $N_p = 4$, $T_d = 33$ ms, $T_r = 66$ ms, $T_{proc} = 50$ ms, and only the luminance component is considered throughout the experiments. Two playout delay requirements are considered: $D_{max} = 150$ ms and $D_{max} = 500$ ms. In the proposed scheme, a requesting instant r_i is associated with the 12 frames having decoding instants within a time window $[r_i + D_{max} - 4T_d, r_i + D_{max} + 7T_d]$.

Each parameter of the fitting exponential functions is quantized and encoded with 20 bits, resulting in a 25-byte overhead

per subband that is transmitted at the beginning of the rendering session, along with the initial segment (typically around 10 bytes) of each bitstream. We choose the target forward payload rate, C_f, from 64 up to 448 kbps, corresponding to the average responding packet payload length of 132 to 924 bytes. The maximum packet length is set to 1000 bytes. The payload in the requesting packet is entropy-coded with a rate typically within 1% of C_f [8].

We compare the proposed streaming scheme to a heuristic scheme. In the heuristic scheme, bit allocation among all subbands that minimizes the average distortion in the whole sequence at the given transmission rate is obtained by standard Lagrangian minimization at the sender. The resulting allocation is signalled to the receiver before the playout session. During the playout session, bitstreams are requested to fulfill this allocation in the decoding order of the subbands. Using the 2-level Haar wavelet as an example, a group of 4 frames are temporally decomposed into the LL, H_1, HL, and H_2 subbands. The decoding order in this case is $LL \rightarrow HL \rightarrow H_1 \rightarrow H_2$. Therefore LL is first requested up to the allocated length, followed by the allocated length of HL, ... etc. In addition, a group of 16 subbands in the 4-level Haar case is removed from the consideration of request after every time duration of $16T_d$, even if they have not fulfilled the allocation, to avoid the streaming being throttled by the belated data. The same methods for fragmentation and receiver buffer state prediction as in the proposed scheme are also adopted in the heuristic scheme.

Note that bit allocation in the heuristic scheme is optimal for downloading the video sequence encoded at the given rate before viewing it. However, for the video streaming scenario, it is static whereas the allocation in the proposed scheme can dynamically adapt to the network characteristics and the playout delay requirement. The performance comparison for $D_{max} = 150$ ms is shown as the bottom two curves in each plot in Figure 1. The proposed scheme performs consistently better than the heuristic scheme, for both video sequences, with a PSNR gap of $2.4 \sim 3.1$ dB at the same transmission rate. Correspondingly, to achieve the same quality the proposed scheme can reduce the required transmission rate by $50\% \sim 60\%$ over the heuristic scheme. The performance gap is smaller for $D_{max} = 500$ ms due to the longer delay that allows the heuristic scheme to fulfill its allocation before the reconstruction of the frames. However, the proprosed scheme still outperforms the heuristic scheme by up to 0.5 dB.

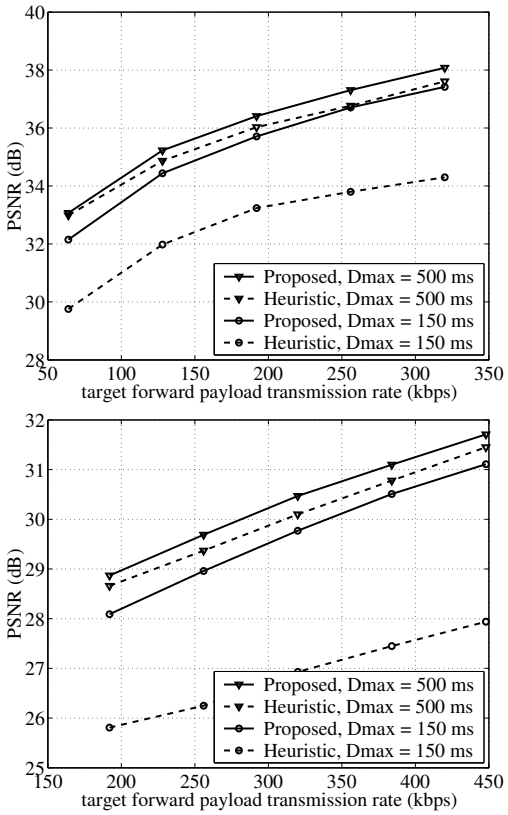

Fig. 1. (top) *Foreman* (bottom) *Mobile*

5. CONCLUSIONS

We propose a rate-distortion optimized framework to stream 3-D wavelet-coded video. Based on the source rate-distortion profiles, desired playout deadline and transmission rate, and network characteristics, the receiver performs rate-distortion optimized data request as a convex optimization process to minimize the distortion in the reconstructed frames. Experiments show that the proposed scheme improves the video quality by up to 3.1 dB over the heuristic scheme at the same rate, or correspondingly reduces the required transmission rate by up to 60% for the same quality.

6. REFERENCES

[1] D. Taubman and A. Zakhor, "Multi-rate 3-D subband coding of video," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 572–588, Sept. 1994.

[1]This work was supported by Grant No. ECS-0225315 of the National Science Foundation and by Philips Corporation.

[2] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 559–571, Sept. 1994.

[3] S. Choi and J. Woods, "Motion compensated 3-D subband coding of video," *IEEE Trans. Image Processing*, vol. 8, pp. 155–167, February 1999.

[4] A. Secker and D. Taubman, "Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting," in *Proc. IEEE Int. Conf. on Image Processing 2001*, Thessaloniki, Greece, Oct. 2001, vol. 2, pp. 1029–1032.

[5] B. Pesquet-Popescu and V. Bottreau, "Three dimensional lifting schemes for motion compensated video compression," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing 2001*, Salt Lake City, UT, USA, May 2001, vol. 3, pp. 1793–1796.

[6] L. Luo, J. Li, S. Li, et al., "Motion compensated lifting wavelet and its application in video coding," in *Proc. IEEE Int. Conf. on Multimedia and Expo 2001*, Tokyo, Japan, Aug. 2001, pp. 481–484.

[7] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *(submitted) IEEE Trans. Multimedia*.

[8] C.-L. Chang and B. Girod, "Rate-distortion optimized interactive streaming for scalable bitstreams of light fields," in *Proc. SPIE Visual Communications and Image Processing 2004*, San Jose, CA, USA, Jan. 2004, pp. 222–233.

[9] C.-L. Chang and B. Girod, "Receiver-based rate-distortion optimized interactive streaming for scalable bitstreams of light fields," in *(to appear) Intl. Conf. on Multimedia and Expo 2004, Taipei, Taiwan, June 2004*.

[10] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on Set Partitioning in Hierarchical Trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, June 1996.