End-to-end Distortion Estimation for RD-based Robust Delivery of Pre-compressed Video

Rui Zhang, Shankar L. Regunathan and Kenneth Rose
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106 [*]

Abstract

Applications where packetized video is streamed over the Internet, must be designed to achieve robustness to packet loss as well as compression efficiency. Whenever possible, the ideal solution to this problem is to jointly optimize the adaptation of the compression and error protection strategies to the network status, so as to minimize the expected end-to-end distortion of reconstructed video at the receiver. However, in the case of pre-compressed video streaming, compression is performed without knowledge of the network condition, and conversely, the delivery is performed without access to the original signal. It is hence difficult for the transmitter to estimate and minimize the end-to-end distortion during delivery. This paper addresses this problem by deriving an algorithm which enables the transmitter, or other intermediate nodes, to estimate the overall end-to-end distortion while delivering pre-compressed video. This estimate fully accounts for the effects of (prior) quantizaton, packet loss and error propagation, as well as error concealment. The accuracy of the estimate is demonstrated by simulation results. The algorithm requires storage of minimal side-information that is computed during compression. The algorithm is of low complexity, and is applicable to virtually all coding techniques, including the standard (predictive) video coders. The paper also discusses the use of this estimate to adapt a variety of packet-loss resilience techniques for pre-compressed video streaming. The considerable potential gains of this approach are illustrated via the example of an FEC-based streaming video system.

1 Introduction

Internet-based packetized video streaming applications have attracted tremendous attention in recent years. The unreliable packet delivery through the Internet requires that video streaming systems provide robustness to loss as well as compression efficiency. While standard source-channel coding algorithms [1] [2] can be used to optimize the delivery of live-content, they are incompatible with applications which stream pre-compressed video. The main difficulty is due to the fact that network conditions are unknown during the compression stage. As an illustration, consider an application that delivers “Video on Demand.” The raw video content is compressed offline, and is stored on the server. Network condition parameters such as bandwidth, packet loss probabilities and delay jitter vary widely based on the characteristics of the available links between the server and the client (receiver). They have significant effects on system performance. Clearly, optimization of the error resilience strategy during delivery has no access to the original video. This represents a major difficulty in estimating and minimizing the end-to-end distortion, which quantifies the difference between the original signal and the decoder reconstructed signal (after loss and error concealment). Further, practical restrictions on server complexity preclude the use of complex algorithms that perform requantization of the source bitstream, or perform other modifications at the source-syntax level. Instead, adaptation should be based on simple transport-level tools, such as Forward Error Correction (FEC) or Automatic Retransmission reQuest (ARQ). Note that similar constraints apply to other streaming video applications such as transcoding at an intermediate node, and Internet multicast.

The problem of robust streaming of pre-compressed video has been addressed in [3] [4] [5] [6]. In [4], it was recognized that the ideal resilience strategy at the server is one which adapts to the actual bandwidth and packet loss statistics of the network in order to minimize the expected end-to-end distortion of reconstructed video at the receiver. A Lagrangian Rate-Distortion (R-D) framework was proposed to achieve the optimal adaptation strategy. But, the practical usefulness of this framework is limited in the absence of a convenient method to compute the overall reconstruction distortion. The task of computing end-to-end dis-

[*] This work is supported in part by the NSF under grants no. MIP-9707764, EIA-9986057 and EIA-0080134, the University of California MICRO Program, Conexant Systems, Inc., Dolby Laboratories, Inc., Lucent Technologies, Inc., Medio Stream, Inc., and Qualcomm, Inc.

tortion is complicated by many inter-related factors. They include (prior) quantization, effective packet loss statistics which is a function of the network condition and the error resilience strategy, and error concealment. Further, the use of inter-frame prediction in video coders results in spatial and temporal error propagation, and hence additional inter-dependencies between packets. The problem of distortion estimation was rendered tractable in prior work by either neglecting the effect of inter-frame error propagation [5] [6], or by ignoring error concealment [4]. However, the consequent inaccuracy in the distortion estimates can result in poor adaptation strategies.

The main contribution of this paper is an efficient algorithm that enables the transmitter to estimate the expected overall end-to-end distortion at the receiver. The algorithm takes into account the effects of quantization, inter-dependencies among packets through prediction and error propagation, and error concealment. The algorithm requires a small amount of side-information that is easily computed during compression, and stored at the server. In addition to its accuracy, the estimate provides another advantage. It is linearly dependent on the packet loss statistics, and thus allows for low-complexity R-D optimization of packet-loss resilience strategies.

The paper is organized as follows: Section 2 introduces notations and derives the decoder distortion estimate. Simulation results demonstrate its accuracy. Section 3 discusses the integration of this estimate within an RD framework for optimizing streaming efficiency and robustness. The potential for substantial performance gains is illustrated using the example of a FEC-based robust delivery system.

2 End-to-end Distortion Estimation for pre-Compressed Video

In this section, we analyze the problem of end-to-end distortion estimation for a system that delivers pre-compressed video. We then derive a first order estimation algorithm as an efficient solution.

2.1 End-to-end Distortion

Without loss of generality, we assume that the compressed video is packetized into independent groups of packets (GOP). The expected distortion of each GOP can be calculated independently as there is no dependency across GOPs. However, packets within one GOP may depend on each other due to prediction. Thus, the distortion for all packets in one GOP must be calculated jointly.

Let there be N source packets per GOP. Let p_i denote the effective packet loss rate (PLR) of packet i. Note that p_i is a function of both the network conditions, and the resilience strategy used for this packet. The resilience strategy could involve retransmission of the packet, or the use of error correction codes. The PLR vector for the entire GOP is given by, $\mathcal{P} = \{p_0, p_1, ..., p_i, ..., p_{N-1}\}$. Since each packet can be either received correctly, or considered as lost, there is a total of 2^N possible events for each GOP. The event vector of the entire GOP is represented by $\mathcal{B}^{(k)} = \{b_0^{(k)}, b_1^{(k)}, ..., b_i^{(k)}, ..., b_{N-1}^{(k)}\}$, where k denotes the index of the event ($k = 1, 2, ..., 2^N$), and binary random variable $b_i^{(k)}$ denotes the status of the ith packet in the kth event. The packet is received correctly if $b_i^{(k)} = 0$, and is lost if $b_i^{(k)} = 1$. The probability of the kth event vector is given by $p^{(k)} = \prod_{i=0}^{N-1} (1-p_i)^{(1-b_i^{(k)})} p_i{}^{b_i^{(k)}}$.

Let f denote the value of some pixel in the original video. Let $\tilde{f}$ denote the corresponding reconstructed pixel at the receiver. Note that $\tilde{f}$ is in fact a *random* variable at the transmitter since it depends on the effects of packet loss, concealment and error propagation which are unknown to the transmitter. However, it is important to note that the decoder reconstruction is completely determined given the event vector of the entire GOP. Thus, the decoder reconstruction under the kth event, $\tilde{f}^{(k)}$, can be *exactly computed*. The end-to-end distortion of this pixel under the kth event is given by $d^{(k)} = (f - \tilde{f}^{(k)})^2$. The overall distortion of the GOP distortion under the kth event is

$$D^{(k)} = \sum_{f \in GOP} d^{(k)}. \qquad (1)$$

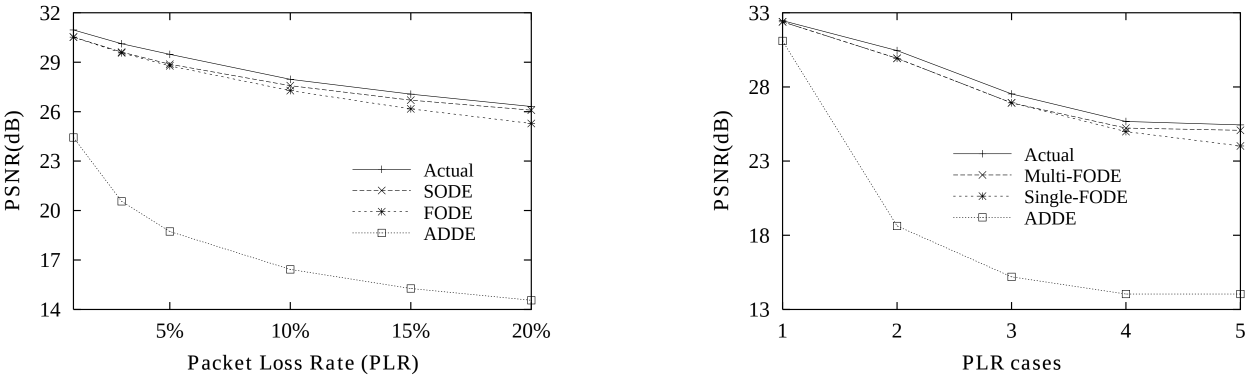
At the compression stage, the encoder can compute $D^{(k)}$ for $k = 1, 2, ..., 2^N$, and store these quantities as side-information at the server.

During delivery, the probability of occurrence of event $\mathcal{B}^k$ is given by $p^{(k)}$. Therefore, the expected overall distortion of the receiver is given by

$$\begin{aligned} E\{D(\mathcal{P})\} &= \sum_{k=1}^{2^N} p^{(k)} D^{(k)} \\ &= \sum_{k=1}^{2^N} (\prod_{i=0}^{N-1} (1-p_i)^{(1-b_i^{(k)})} p_i{}^{b_i^{(k)}}) D^{(k)}. \qquad (2) \end{aligned}$$

Note that this estimate is *exact* (i.e., without approximation). It considers all possible error events, and takes into account the effects of compression, loss, error propagation and error concealment.

In practical applications, this estimate has two major drawbacks. First, 2^N real values (D^k) need to be stored as side-information for each GOP. This imposes a large storage requirement. Second, the expected distortion is a complicated function of the individual packet loss rate as shown in (2). Therefore, the use of this metric to optimize error resilience strategies involves a high computational complexity.



(a) (b)

Figure 1. PSNR vs. packet loss rate for model accuracy. QCIF sequence “carphone”. (a) single-layer bitstream at 32kbps for 10fps. (b) three-layer bitstream at 32/64/96kbps for 10 fps. The packet loss rate for the three layers in (b) are: case 1 (0%, 5%, 10%), case 2: (1%, 3%, 5%), case 3 (3%, 8%, 15%), case 4: (5%, 10%, 95%), and case 5 (5%, 95%, 95%).

2.2 First-order Approximation through Partial Derivatives

The objective of this section is to derive a simple approximation of the end-to-end distortion estimate. At the cost of a slight loss of accuracy, this approximation allows for substantial reduction in the amount of side-information, and computational complexity.

We propose to use the first order Taylor expansion of (2). Assume we expand the current GOP distortion of (2) about a particular *reference PLR vector*, $\bar{\mathcal{P}} = \{\bar{p}_0, \bar{p}_1, ..., \bar{p}_i, ..., \bar{p}_j, ..., \bar{p}_{N-1}\}$. For example, $\bar{\mathcal{P}}$ could correspond to the case when the loss probability is zero for all packets in the GOP. For a PLR vector $\mathcal{P}$ which is close to the reference PLR, it is reasonable to approximate the expected distortion of (2) via the first order Taylor series expansion. Thus, we have

$$\begin{aligned} E\{D(\mathcal{P})\} &\approx E\{D(\bar{\mathcal{P}})\} + \sum_{i=0}^{N-1} \frac{\partial E\{D(\mathcal{P})\}}{\partial p_i}|_{\mathcal{P}=\bar{\mathcal{P}}}(p_i - \bar{p}_i) \\ &= E\{D(\bar{\mathcal{P}})\} + \sum_{i=0}^{N-1} \gamma_i (p_i - \bar{p}_i), \end{aligned} \tag{3}$$

where

$$\gamma_i = \frac{\partial E\{D(\mathcal{P})\}}{\partial p_i}|_{\mathcal{P}=\bar{\mathcal{P}}}, \tag{4}$$

is the partial derivative of the expected distortion with respect to the PLR of packet i. The value of $E\{D(\bar{\mathcal{P}})\}$ is easily pre-computed for any given reference PLR $\bar{\mathcal{P}}$ via (2). Similarly, γ_i may be easily pre-computed for each packet (Due to space constraints, the details are omitted here).

The number of reference PLRs determines the amount of side-information needed for this “First Order Distortion Estimation” (FODE) model. If m reference PLRs are used, we need to store $m(N+1)$ quantities for each GOP, which represents a significant reduction in side-information over the exact approach. This issue is further discussed in the simulation section. Further, note that the expected distortion depends linearly on the PLRs, and all inter-dependencies have been decoupled through the partial differential value γ_i.

2.3 Simulation Results

This subsection demonstrates the accuracy of FODE through simulations. The source video bitstreams were generated by the standard H.263+ codec [8]. We consider both the single layer coding system, and the scalable coding system. The decoder uses the adjacent lower layer reconstruction if any enhancement layer packet is lost, or replaces the lost base layer packet with information in the previous frame. We implemented FODE to pre-calculate the partial derivatives as side information. Using this side information, we estimate the distortion values for different PLR vectors. We compared these estimates to the actual distortion of reconstructed video at the receiver. The actual distortion was averaged over 50 realization of the network under the same PLR conditions. An additional comparison was made to the “Acyclic Dependent Distortion Estimation” (ADDE) proposed in [4] where the effect of error concealment is neglected.

Figure 1 shows the simulation results representing the estimation accuracy under different PLR distribution. Figure 1 (a) gives the results for QCIF sequence “carphone” in a single layer system. For the single layer system, we only use the all-zero reference PLR for the Taylor expansion, $\bar{\mathcal{P}} = \{0, 0, ..., 0, ..., 0\}$. We also simulated the performance of a Second Order Distortion Estimation (SODE). These results demonstrate the high accuracy of FODE in comparison to ADDE. The importance of accounting for the effect of error concealment is obvious. The second order correction of SODE enables slightly better estimates than FODE at large packet loss rates, but requires more side-information and complexity.

Figure 1 (b) presents the results in a three-layer system. For both the single-FODE model where only the all-zero reference PLR is used, and the multi-FODE model where additional reference PLRs are used. These additional reference PLRS are now needed to account for the case where enhancement layer packets are discarded at the transmitter to conserve bits. The reference PLRs used in the multi-FODE model are: $\bar{\mathcal{P}}_0 = \{(0,0,0), ..., (0,0,0), ..., (0,0,0)\}$,$\bar{\mathcal{P}}_1 = \{(0,0,1), ..., (0,0,1), ..., (0,0,1)\}$, and $\bar{\mathcal{P}}_2 = \{(0,1,1), ..., (0,1,1), ..., (0,1,1)\}$. The results demonstrate the accuracy of FODE in scalable coders. Note that the multi-FODE gives better approximation than the single-FODE when the enhancement-layer packets are discarded (as in case 4 and case 5 in Figure 1 (b)). But these gains are achieved at the cost of more side information.

In summary, the simulation results show that FODE is efficient in approximating the expected overall reconstruction distortion at the receiver. While a single reference PLR is sufficient for non-scalable coding systems, multiple PLRs may be needed for scalable coding systems.

3 RD-based Robust Delivery of pre-Compressed Video

In this section, FODE is integrated into the RD framework to optimize error-resilient schemes for delivery of pre-compressed video. The potential performance gains are illustrated using the example of scalable encoder and FEC-based unequal error protection.

3.1 Optimized Delivery Schemes within an RD Framework

Any adaptive error-resilience scheme provides a set of policy choices, $\pi \in \{\pi^{(\mathbf{0})}, \pi^{(\mathbf{1})}, ..., \pi^{(\mathbf{S})}\}$, for each packet. Depending on the resilience scheme, the policy choices could be the number of retransmissions, or the strength of error correction code. The effective loss rate, p_i, for the ith packet, is a function of both the network condition and the policy choice. The cost of the policy choice $c(\pi)$, is usually the total number of bits needed to send the original source packet.

The policy vector for a group of (source) packet (GOP) is defined as $\Pi = \{\pi_0, \pi_1, ..., \pi_i, ..., \pi_{N-1}\}$. The corresponding PLR vector and the cost vector are denoted by $\mathcal{P}(\Pi)$, and $\mathcal{C}(\Pi)$.

The expected end-to-end distortion for a GOP can be estimated using FODE as

$$E\{D(\mathcal{P}(\Pi))\} \approx E\{D(\bar{\mathcal{P}})\} + \sum_{i=0}^{N-1} \gamma_i (p_i(\pi_i) - \bar{p}_i). \quad (5)$$

The total cost for the GOP is given by

$$\mathcal{C}(\Pi) = \sum_{i=0}^{N-1} c_i(\pi_i). \quad (6)$$

The optimal adaptive delivery scheme should then choose the policy that minimizes the expected distortion $E\{D(\mathcal{P}(\Pi))\}$ while satisfying constraint on the cost $\mathcal{C}(\Pi)$. This problem can be recast as an unconstrained minimization of the Lagrangian,

$$\begin{aligned} & E\{D(\mathcal{P}(\Pi))\} + \lambda\mathcal{C}(\Pi) \\ \approx \quad & E\{D(\bar{\mathcal{P}})\} + \sum_{i=0}^{N-1} [\gamma_i (p_i(\pi_i) - \bar{p}_i) + \lambda c_i(\pi_i)]. \end{aligned}$$

Note that the distortion estimate provided by FODE depends linearly on the PLR. Thus, theoretically, the policies can be chosen independently for each packet to minimize the Lagrangian cost, and practically the optimization can be done at any level with any structure at the convenience of the adaptation scheme. This results in low computational complexity of the optimization procedure.

3.2 Simulation Results

For the simulations, we consider a system of layered coding with unequal transport prioritization to demonstrate the superiority of our algorithm. The system consists of a fully standard-compatible layered source coding for pre-compression of the video signal, and unequal error protection through FEC on the packets of different layer at the time of delivery . The systematic Reed-Solomon (RS) code is adopted to generate redundant packets to combat packet loss [2] [5] .

A five-layer bitstream for QCIF sequence "carphone" is generated. Three online delivery schemes are compared. The first is the RD optimized scheme using our multi-FODE model (M-FODE-RD). The second uses only the single-FODE model (S-FODE-RD). Both of them dynamically select the best error protection (n, k) code, given a fixed k, so as to minimize the RD cost for packets in each layer. The third scheme uses fixed unequal error protection (UEP) for each layer, with more protection for lower layers, through

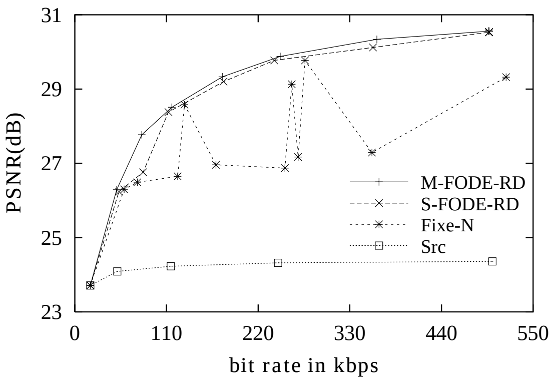


Figure 2. PSNR vs. total bit rate for different delivery schemes. QCIF sequence "carphone", 10 fps, 5-layer bitstream at 16/64/112/240/496kbps.

RS codes (fixed-N). While the first two schemes can adapt to any rate constraint, the fixed-N scheme can be used only for certain target bit rates. The performance of the unprotected source bitstream is also presented as a reference.

The three delivered bitstreams generated by those schemes go through the same time-varying channels with PLR in the range of $1\% \sim 20\%$. Figure 2 shows the decoder distortion for each of them versus the bit rate. The results illustrate that FODE-RD schemes achieve substantial gains while maintaining more flexibility than the fixed-N scheme. Note that multi-FODE-RD scheme yields only small gains over single-FODE-RD scheme. This indicates that single FODE may be sufficient for most practical applications.

4 Conclusion

The estimate of the end-to-end distortion is a fundamental issue in RD-optimized adaptive delivery of pre-compressed video over lossy packet networks. We proposed an algorithm to accurately calculate the overall end-to-end distortion, which takes into account all the effects of the encoder's compression algorithm, the inter-dependencies among packets, the changing network statistics, the delivery schemes and the error concealment used by the decoder. Its accuracy is demonstrated through simulation results. Moreover, it only requires minimal side information. The distortion estimate can be used to optimize various robust adaptation strategies for delivery of pre-compression video. Fairly low complexity in the optimization procedure is achieved due to our linear estimation model. The potential performance gains are illustrated using the example of a delivery system that combines scalable coding with FEC-based error protection.

References

[1] W. Tan and A. Zakhor, "Real-time Internet video using error resilient scalable compression and TCP-friedly transport protocol," *IEEE Transactions on Multimedia*, vol. 1, no. 2, pp. 172-186, June 1999.

[2] M. Gallant and F. Kossentini, "Rate-distortion optimized layered coding with unequal error protection for robust Internet video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 357-372, Mar. 2001.

[3] G. J. Conklin, G. S. Greenbaum, K. O. Lillevold, A. F. Lippman,and Y. A. Reznik, "Video coding for streaming media delivery on the Internet," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 3, pp. 269-81. Mar. 2001.

[4] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," submitted to *IEEE Transactions on Multimedia*, Feb. 2001.

[5] B. Girod, K. Stuhlmuller, M. Link and U. Horn, "Packet loss resilient internet video streaming," *Proceedings of the SPIE, Visual Communications and Image Processing '99*, San Jose, CA, USA, vol. 3653, pp. 833-44. Jan. 1999.

[6] W. Tan and A. Zakhor, "Video multicast using layered FEC and scalable compression," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 373-86, vol. 11, no. 3, Mar. 2001.

[7] ITU-T, Rec. H,263, "Video codeing for low bitrate communications", version 2 (H.263+), Jan. 1998

[8] H.263+ Codec. http://spmg.ece.ubc.ca/