INTERNET VIDEO USING ERROR RESILIENT SCALABLE COMPRESSION AND COOPERATIVE TRANSPORT PROTOCOL

Wai-tian Tan and Avideh Zakhor

Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, CA 94720 USA
e-mail: {dtan avz}@eecs.berkeley.edu

ABSTRACT

We introduce a point to point video transmission scheme over the Internet combining a low-delay TCP-friendly transport protocol in conjunction with a novel, real-time, error-resilient layered compression method. Compressed video is packetized into individually decodable packets that are of equal expected visual importance. As a result, relatively constant video quality can be achieved at the receiver under lossy conditions. The packets can be truncated to meet the time varying bandwidth imposed by the transport protocol. Actual Internet experiments together with simulations are used to evaluate the performance of the combined scheme.

1. INTRODUCTION

With ever growing network resources, video streaming is already an important part of today's Internet applications. However, most video compression methods that are being used for streaming are neither bandwidth-scalable nor error-resilient. This produces a constant volume of inter-dependent packets that are prone to error propagation.

Producing a constant volume of traffic has several disadvantages. First, it would lead to congestive collapse when the aggregate bandwidth of the video traffic exceeds network capacity. Second, it competes unfairly with other adaptive traffic, such as TCP, that reduces transmission rate in face of network congestion. For shared environments like the Internet, it is important that users do not exceed their fair share of resources.

The use of non-error-resilient compression makes it necessary to employ error control mechanisms at the transport level. This typically takes the form of forward error correction (FEC) or retransmission. Retransmission based error control methods fail to be real-time, and effective use of FEC over the Internet requires *a priori* knowledge of the channel conditions.

This work is supported by ONR grant N00011-92-J-1732, ASSERT No N00011-95-1-1163, AFOSR grant F19620-91-1-0359, California State MICRO, Sun Microsystems, Philips, Tekronix, LG Electronics, and Sharp Corporation.

An attractive alternative is to use scalable video compression with feedback rate control whereby transmission sources adjust their rates in response to changing network conditions. This is typically done by measuring packet loss rate or changes in round-trip delay. Such schemes are therefore *reactive* and do not prevent packet loss. As a result, error resilience remains an important issue.

This paper introduces an Internet video transport scheme combining a TCP-friendly transport protocol with a layered compression method that not only does not suffer from error propagation, but also provides (a) constant visual quality under lossy conditions, (b) good compression efficiency, and (c) low complexity encoding and decoding. The combined scheme does not preclude the additional use of FEC or partial retransmission even though they are not considered in this work.

2. ROBUST SCALABLE COMPRESSION

To eliminate error propagation, we need every packet to be individually decodable. One way to achieve this is to employ a forward decomposition of the source material into M *components* and then compress each component independently to form a packet. Each packet can then be decoded to a co-image where the sum of all co-images form the original image.

There are many such decompositions. One example is the polyphase decomposition which takes every M consecutive pixels and distributes one pixel to every component. Each component then would clearly be individually decodable and approximately of equal importance. This scheme suffers from low compression efficiency. Another approach is to use block based coding in the pixel domain. However, when one packet contains all information about a spatial location, its loss will cause all information in that location to be lost. Yet another approach is to use subband decomposition to divide source into subbands that can be compressed independently. However, the DC-subband contains most of the energy for natural images. If each subband goes into a packet, this skewness would cause large variability in decoded picture quality under lossy conditions.

To overcome the problems of the above approaches, we propose a novel packetization scheme for subband decomposition: instead of making each subband a component, we partition each subband into an equal number of coefficient blocks. Each coefficient block in a subband carries information about some localized region in the original frames. The components are then formed by grouping from each subband, equal number of coefficient blocks that correspond to different spatial regions of the source. As an example, Fig. 1 shows the formation of one component out of a total of nine. It would take at least seven packet losses to completely eradicate a particular spatial region.

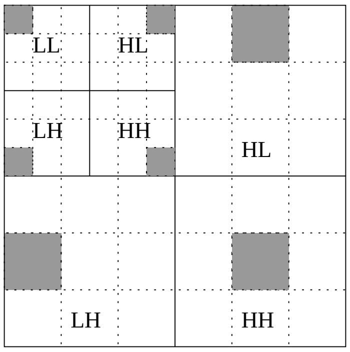

Figure 1: Grouping coefficient blocks from different subbands to form a component.

Each coefficient block is progressively quantized and compressed independent of other blocks using layered block coding [5]. However, subsequent quantizer outputs of the same block are compressed inter-dependently. To achieve error resilience, the compressed quantization layers are packed in a pre-determined order based on the relative importance of the subbands while preserving the dependency between quantization layers. Because the decoder will decode in the same order, the packet length can be truncated to suit any targeted transmission rate.

Fig. 2(a) shows original "Lena" image at 512×512. Five levels of spatial decomposition, using a 5/3-tap biorthogonal filter, are performed on the image to get 16 subbands. Each subband is then divided into 256 coefficient blocks. The largest coefficient block is 16×16 while the smallest is 1×1. We form 256 components and compress each component using layered block coding method described in [5] to get 256 packets which are then subjected to a 22% random packet loss. The image reconstructed from the survived packets is shown in Fig. 2(b). No error concealment has been applied to the image. We see that errors are dispersed over a wide support and while the image is uniformly blurred and the total energy is diminished, all features of the original image are still visible. Furthermore, even though block based coding is employed, there are no sharp discontinuities because data partitioning is performed in the frequency domain instead of the pixel domain.

(a) (b)

Figure 2: Original Lena (a) and Lena at 0.3 bits/pixel and 22% loss (b).

To extend the framework from still image to video, one possible way is to use 2D subband with motion compensation. However, since motion compensation does not perform well when required to produce finely scalable video, a scheme based on 3D subband coding is used. A Haar filter is used to generate temporal subbands. A component then is formed by getting coefficient blocks of different spatial locations from the set of spatio-temporal subbands thus generated.

3. TCP-FRIENDLY PROTOCOLS

Because the Internet is dynamically shared by many users, it is imperative that a transport protocol can share resources fairly with multiple instances of itself and with TCP, the dominant source of Internet traffic. TCP cannot be used directly for video transport because its reliability is achieved at the expense of time varying delay and throughput. Fig. 3 shows the end-to-end delay in seconds when 300 seconds of video material generated at 600 and 630 *kbps* respectively are transmitted from Toronto to Berkeley using TCP at noon time. In both cases, even though the long term average throughput of the TCP connections exceeds the data rate, the end-to-end delay can still be significant.

One way to ensure that a video transport protocol competes fairly with TCP is to use exactly the same rate controller as TCP but without the retransmission part. However, TCP uses a window based flow controller which geometrically reduces the window size on congestion. As a result, there will be periods in which no data could be sent. This produces traffic pattern with abrupt changes, making TCP rate control inap-

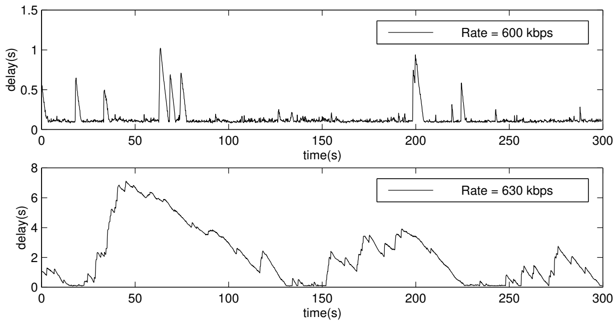

Figure 3: Delay for video transport using TCP from Toronto to Berkeley (noon, May 11, 1998).

plicable for real-time video.

Instead of matching the TCP traffic pattern exactly and instantaneously, a more relaxed form of fairness can be obtained by matching the TCP throughput on a macroscopic scale. Mahdavi and Floyd [2], Mathis *et.al.* [3] for example, have derived expressions relating the average TCP throughput (T) to the packet loss rate (p):

$$T = k \frac{MTU}{RTT \times \sqrt{p}} \tag{1}$$

where RTT is the dynamic estimate of the round-trip time, MTU is the maximum transport unit of the connection, and k is a constant that equals 1.22. It should be noted that p is an estimate of the transmitter rather than the actual packet loss rate seen at the receiver.

Instead of using a window based rate controller as in TCP, our TCP-friendly transport protocol for scalable video takes a rate based approach in which Eqn. 1 is used to provide the instantaneous transmission rate. For every packet received, an acknowledgement will be sent to the transmitter so that RTT and p can be estimated. To give close correspondence with TCP, measurements of RTT and determination of packet losses are done in the same manner as TCP, with packet losses within one RTT counted only once. To avoid excessive fluctuations in transmission rate caused by isolated packet loss, a smoothing time window of 256 RTT is chosen to measure p. A smaller time window has the advantage of faster response to changing network conditions at the expense of higher bandwidth variability under steady state operation.

Fig. 4 shows the throughput of one TCP connection and two instances of our TCP-friendly protocol transmitting simultaneously from Toronto to Berkeley. It is seen that not only do the protocols coexist, they also share bandwidth fairly with each other. Unlike most retransmission based streaming protocol, no buffering is required for our protocol, and end-to-end delay is dominated by the network propagation delay.

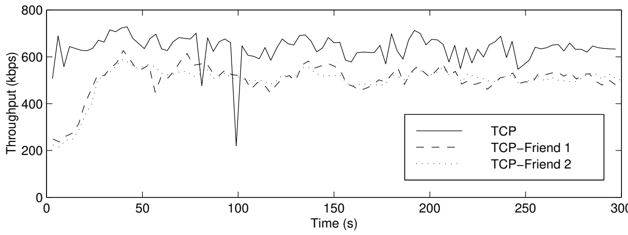

Figure 4: Throughput of TCP and TCP-friendly protocol from Toronto to Berkeley (2 pm, May 8, 1998).

4. RESULTS

All results described in this section are obtained using two levels of temporal and 4 levels of spatial decompositions. Four different schemes are considered: (P) our proposed scheme in Section 2, (M) MPEG-1, (S) 3D subband coding in which each packet contains one subband and, (T) a scalable compression scheme [5] that is similar to scheme (P) except for packetization. Specifically, under scheme (T), the first packet contains the most important information followed by the second packet and so forth. This produces packets that are *linearly* dependent, i.e., for every K frames N packets are produced so that loss of packet i would render packets $i+1$,...,N useless.

Resilience to Loss: Fig. 5 shows MSE as a function of time when video compressed under scheme (T) is transmitted from Toronto to Berkeley. The dynamic range of the plots were intentionally limited to 400 even though the peak MSE were actually in excess of thousands. The top graph shows the video quality when 3 non-adaptive UDP flows of 1 $Mbps$ are transmitted simultaneously. The resulting packet loss causes received video quality to vary wildly, with peak MSE as high as 1439. One possible improvement would be to employ the TCP-friendly protocol of Section 3. The result is shown in the middle graph. The use of rate control greatly reduces packet loss and the overall quality is more constant. However, occasionally an important packet may be lost, causing peak MSE to be 1320 even when most of the packets are actually received. Further improvement can be obtained by applying FEC on top of the TCP-friendly protocol. This is done by replacing less important packets with duplicates of more important packets so as to minimize the expected MSE given the transmission budget and current estimate of packet loss rate. The result is shown in the bottom graph. Even though a lower average distortion is achieved, there are still occasions under which important packets are not received, resulting in peak MSE of 1320. This is partly due to the fact that we have only delayed estimates of the network conditions.

For schemes such as (T) that generate linearly de-

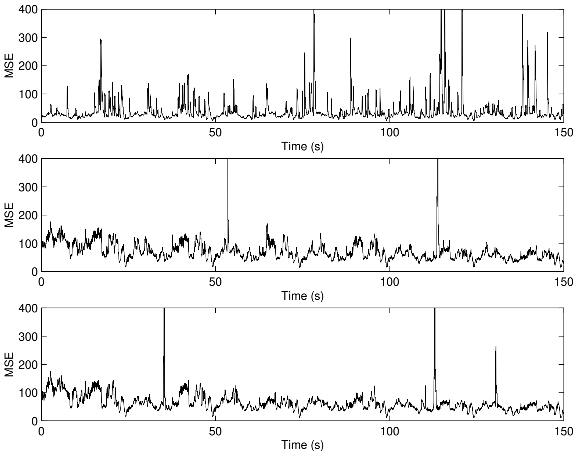

Figure 5: Video quality for Scheme (T) transmitted from Toronto to Berkeley (noon, May 12, 1998). Top is plain UDP, middle uses TCP-friendly rate control, bottom uses FEC and TCP-friendly rate control.

pendent packets, large variability in received quality is expected: assuming independent packet reception rate of p, the probability we decode exactly i packets out of a total of N transmitted packets is $(1-p)p^i$ for $i \neq N$, and p^N for $i = N$, a bimodal distribution that is geometric but with a tall spike at $i = N$. For example, for $N = 20$ and $p = 0.95$, 70% of the time we can either decode all 20 or at most 6 packets, resulting in large variability in the quality of the received video.

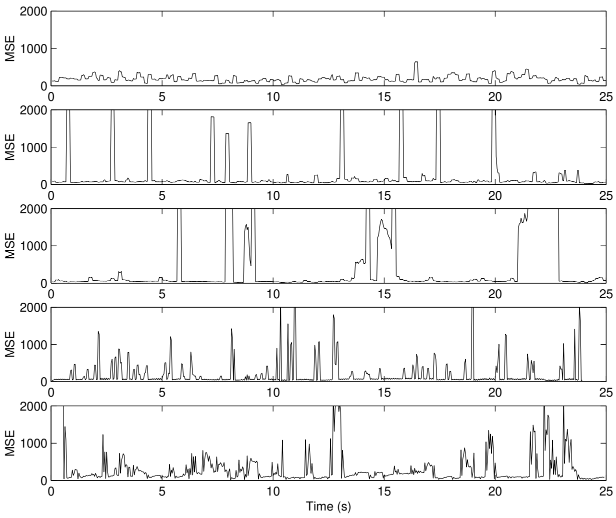

Figure 6: Variability of quality at 5% simulated random packet loss. From top to bottom: (P), (S), (T), (M) with GOP 2, (M) with GOP 15.

Fig. 6 shows the effects of all the schemes considered under 5% simulated random loss. The last 2 plots in Fig. 6 correspond to MPEG-1 simulations. In MPEG-1, a *slice*, which corresponds to a rectangular strip in a picture, is the smallest unit that can be decoded independently. Packetization then is performed so that no slice is split across different packets unless the slice size exceeds the packet size [1]. In the last two plots of Fig. 6, 10 slices are used per frame and it is assumed that headers in the picture, group of pictures, and sequence levels are transmitted error free. During decoding, when a particular region in a frame has no coded information due to packet loss, the corresponding region from the previous frame is copied.

As seen in Fig. 6, only scheme (P) enjoys a uniform high quality of received video. Even though packets under scheme (S) are independent, the skewness in their energy causes large variability in received video quality. Schemes (T) and (M) suffer from error propagation and show even greater variability.

As a direct result of motion compensation in MPEG-1, when information in the reference frame is lost, error propagates to subsequent frames. We see from Fig. 6 that errors have larger "tail" with longer group of pictures. Another point worth noting is that lost slices appear visually as rectangular strips with abruptly discontinuous edges, while packet losses in scheme (P) appear as smoothly blurred patches.

Fig. 7 shows the video quality when a 600 frame sequence is repeatedly transmitted from Toronto to Berkeley 6 times at 12 frames per second during non-busy hour using scheme (P) and our TCP-friendly protocol. Initially the throughput stabilizes at around 700 *kbps* until two more transmissions are started 80 seconds into the experiment. This causes temporary congestion before each instance of the protocol adjusts its transmission rate to around 480 *kbps*. The average MSE at the two bit rates are 14.6 and 20.3 respectively. Since the TCP-friendly protocol is successful in modulating the transmission rate to the available bandwidth of the channel, there is little actual packet loss except when the protocol is reacting to the change in network condition. Given RTT of around 75 ms, the reaction time is roughly 256×0.075 or 19 seconds. Using a smaller window to estimate p reduces the reaction time, but at the cost of more fluctuations in the transmission bit rate.

The experiment is repeated for a single transmission during busy hour, with an average throughput of 520 *kbps*. As seen in Fig. 8, the number of isolated packet losses at the receiver increases significantly. Unnecessary rapid changes in the transmission bit rate are avoided through the use of the 256 RTT smoothing window for p. For example, even though Fig. 8 indicates that the receiver observes packet loss rate as high as 10%, the p estimate at the transmitter varies smoothly between 1.5 to 2.3%. In both experiments,

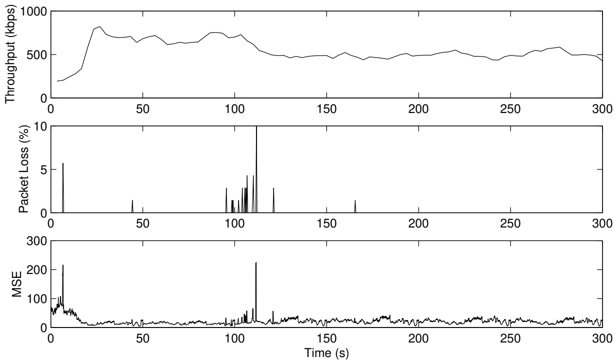

Figure 7: Throughput (top), packet loss rate at receiver (middle) and video quality (bottom) for Scheme (P) with TCP-friendly protocol from Toronto to Berkeley (6 pm, June 23, 1998).

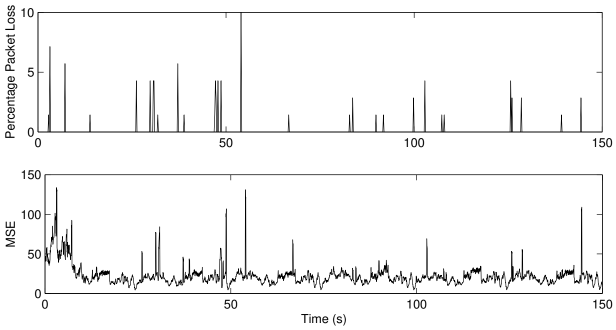

Figure 8: Packet loss rate at receiver (top) and video quality (bottom) for Scheme (P) with TCP-friendly protocol from Toronto to Berkeley (2 pm, May 8, 1998).

simple error concealment is performed on the DC-subband where every missing coefficient is estimated by the average of its surviving neighbors. Even though Fig. 7 and Fig. 8 show strong correlation between packet losses and peaks in MSE, due to the error resilience of scheme (P), high and relatively constant quality of reception is achieved for steady state transmission while acceptable quality is still maintained during the transition period before the TCP-friendly protocol can react to a change in network condition. Comparing the bottom graph of Fig. 8 and the middle graph of Fig. 5, we see that scheme (P) has (a) a lower worst case MSE of 225 versus 1320 for scheme (T) and (b), a lower standard deviation for MSE of 14.5 for (P) versus 67.0 for (T).[1]

Compression Efficiency: The proposed framework described in Section 2 forbids us from exploiting correlation between components. Even though this limits error propagation, there is in general a decrease in compression efficiency. Tab. 1 shows the rate-distortion characteristics of our method as well as that of an MPEG-1 software [4].

Bit Rates (kbps)	500	1000	1500	3000
Mother (P)	34.9	38.8	40.7	44.4
Mother (M)	36.0	38.7	40.7	42.9
Raider (P)	31.7	34.7	36.2	41.2
Raider (M)	30.9	34.1	35.9	38.9

Table 1: Compression Performance for (P) and (M).

We compare PSNR for two sequences: "Raider of the Lost Arc", and "Mother and Daughter", with 600 and 300 frames respectively. The MPEG results are generated using GOP size 4, 1 slice per frame and using exhaustive search. While our method can produce one embedded bit-stream which can be decoded at many different rates, MPEG requires a different bit-stream to be generated for each rate. We see that the two compression methods have comparable rate-distortion performance. Because temporal subband decomposition of (P) is more restrictive than the block based motion compensation of (M), scheme (P) suffers a loss in compression efficiency. This is manifested in the lower compression efficiency as compared to (M) at low bit rates, e.g., 500 $kbps$ and below. At high bit rates such as 3 $Mbps$, (P) typically outperforms (M) because the more efficient residue coding of (P) makes up for the inefficiency of the motion model.

Complexity: For grayscale Ping-pong sequence of size 352×224 on a 170 MHz Ultra-1 workstation, the encoding speeds are given by 31.7 to 21 frames per second in the range of 200 $kbps$ to 2 $Mbps$ respectively. The decoding speeds in the same range varies from 37.9 to 26 frames per second. The reported times exclude disk access and display time [5]. On the same machine MPEG encoding proceeds at 0.4 to 1.6 frames per second using exhaustive and logarithmic search respectively.

5. REFERENCES

[1] D.Hoffman, G.Fernando and V.Goyal. RTP Payload Format for MPEG1/MPEG2 Video. RFC 2038, October 1996.

[2] J.Mahdavi and S.Floyd. TCP-Friendly Unicast Rate-Based Flow Control. *Technical note sent to the end2end-interest mailing list*, January 8, 1997

[3] M.Mathis, J. Semke, J.Mahdavi, T.Ott. The Macroscopic Behavior of the TCP Congestive Avoidance Algorithm. *CCR*, Vol. 27, No. 3, July 1997.

[4] K. Patel, B. Smith, L. Rowe, Performance of a Software MPEG Video Decoder, *Proc. of the ACM Multimedia '93* pp 75-82.

[5] W. Tan, E. Chang, and A. Zakhor. Real Time Software Implementation of Scalable Video Codec, *Proc. ICIP*, Vol. 1, pp 17-20, September 1996.

[1] The DC concealment method used for scheme (P) is inapplicable to (T) where all spatial locations in the DC subband are compressed and transmitted together.