LOSS CONCEALMENTS OF SUBBAND CODED IMAGES FOR REAL-TIME TRANSMISSIONS IN THE INTERNET

Benjamin W. Wah

Department of Electrical and Computer Engineering
and the Coordinated Science Laboratory
University of Illinois, Urbana-Champaign
Urbana, IL 61801, USA
wah@manip.crhc.uiuc.edu

Xiao Su

Media-IXT Group
Inktomi Corporation
75 Hawthorne Street
San Francisco, CA 94105
xiao@inktomi.com

ABSTRACT

Subband-coded images can be transmitted in the Internet using either the TCP or the UDP protocol. Due to the facts that TCP employs congestion control and retransmissions and that UDP and TCP packets are treated differently by routers, TCP takes much longer delays than those of UDP to deliver an image, but packet losses in UDP may lead to poor decoding quality if the image is single-description coded (SDC) and the losses cannot be concealed. In this paper, we study the use of UDP to deliver multi-description coded (MDC) reconstruction-based subband-transformed (ST) images and the reconstruction of missing information at a receiver based on information received. To facilitate recovery from UDP packet losses, we propose a joint sender-receiver approach for designing *optimized reconstruction-based subband transform* (ORB-ST) in MDC. We then carefully evaluate the delay-quality trade-offs between the TCP delivery of SDC images and the UDP and combined TCP/UDP delivery of MDC images. Experimental results show that our proposed ORB-ST performs well in real Internet tests, and UDP and combined TCP/UDP delivery of MDC images provide a range of attractive alternatives to TCP delivery.

1. INTRODUCTION

Quality and delay are two key performance measures to evaluate the delivery of coded images. Previously, high quality in delivery is considered more important because image transfers are not real time in nature and are generally done using a reliable transport protocol like TCP. To transfer images with shorter end-to-end delays, our goal in this paper is to design schemes for reconstructing lost information when image data is subband coded and sent by UDP.

Existing loss concealment schemes are performed in the sender side, or in the receiver side, or in both sides.

Sender-side loss concealments of coded images consist of *layered coding* and *multiple description coding* (MDC) methods.

Layered coding [1] partitions an image into a base layer with visually important image data and a few enhancement layers. In networks with priority support, the base layer is normally assigned a higher priority so that it has a larger chance to be delivered error free. However, layered coding is not applicable in the Internet because the current Internet does not provide priority delivery service for different layers. Moreover, losses cannot be concealed when part of the base layer is lost, resulting in poor decoding quality.

PROC. IEEE INT'L CONF. ON MULTIMEDIA & EXPO, 2002.

In contrast, MDC divides image data into equally important streams. For subband coded images using MDC, scalar quantizers [2] have been designed in order to produce two descriptions using two side-scalar quantizers. These schemes, however, have very complicated encoding and decoding algorithms that make them infeasible for low-delay transmissions.

Receiver-based recovery is usually formulated as heuristic optimizations based on the smoothness assumption of image pixels. Besides being computationally expensive, mistakes in detection of image structures may yield annoying artifacts and blurred edges.

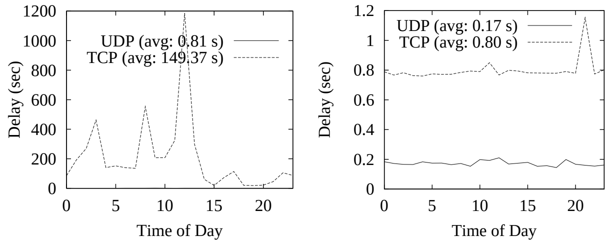
Sender-receiver-based schemes require senders and receivers to cooperate in loss concealments. *Joint source channel coding (JSCC)* [3], minimizes transmission errors by jointly designing the quantizer and the channel coder, according to a given channel-error model and feedbacks from receivers. They are hard to apply in the Internet, since the Internet does not have a well-defined channel model. Another approach based on interleaving with reconstruction [4] is simple and efficient, but may be deficient because it does not consider the reconstruction operation at the receiver side when generating multiple descriptions in the sender side.

In short, existing techniques either rely on the inadequate capability of senders or receivers to do reconstruction, or assume certain transmission channels in designing encoders. None of them considers the reconstruction process performed at receivers.

In this paper, we study a joint sender-receiver-based coding and reconstruction scheme for the delivery of MDC images by UDP. In our system, we *interleave* adjacent pixels into multiple descriptions, decompose each description into segments so that each fits in a packet, and code each segment by a nonredundant error-concealment coding scheme before transmitting the packets.

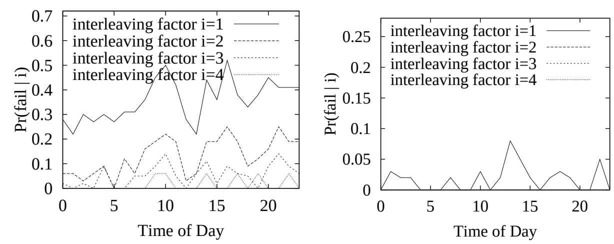
To design MDC at senders, we adopt a joint sender-receiver approach, instead of using previous approaches that design coders independent of reconstruction methods. The coder at a sender applies an *optimized reconstruction-based subband transform* (ORB-ST) that minimizes the reconstruction error, when some of the descriptions are lost and reconstructed using average interpolations from the descriptions received. We have adopted a simple reconstruction algorithm at receivers in order to facilitate fast playback.

Since the proposed MDC can generate coded streams resilient to packet losses, we deliver them using the unreliable but fast UDP or combined TCP/UDP protocols. Our proposed approach can lead to good reconstruction quality with small end-to-end delays but, as expected, degraded decoding quality when compared to the TCP delivery of single-description coded (SDC) images.



a) Urbana-China b) Urbana-California

Figure 1: Round-trip delays of sending 64 UDP packets and the same data in TCP packets encapsulated in UDP ones to the UDP echo port of two remote computers. The experiments were carried out over a 24-hour period on April 8, 2001.



a) Urbana-China b) Urbana-California

Figure 2: $Pr(fail|i)$, probability of bursty losses that cannot be recovered, conditioned on interleaving factor i, at different times on April 8, 2001.

2. INTERNET DELAY AND LOSS BEHAVIOR

We study in this section the end-to-end delays of both TCP and UDP delivery in the Internet and the loss behavior of UDP delivery.

From a site in Urbana, we chose two destination sites in our experiments. The first one is a low-loss connection to Berkeley, and the second, a high-loss connection to China.

Since we have no control of these destination computers, we carried out our experiments by sending packets to the echo port of each destination. To ensure fair delay comparisons between TCP and UDP, we modified the Linux kernel so that TCP echo packets were encapsulated in UDP ones and sent to the UDP echo port of the remote server. In this way, encapsulated TCP packets would be echoed immediately when they were received by the echo server, just as UDP packets. In determining the number of packets sent, we assume that a 512-by-512 image is compressed at an $8:1$ ratio and sent in 512-byte packets, leading to 64 packets sent per image.

Figure 1 shows the end-to-end delays of sending 64 UDP packets and 64 TCP packets encapsulated in UDP packets to the UDP echo port of two remote servers. To avoid overflow of receiver buffers when all 64 packets were dumped to a remote UDP echo port simultaneously, we sent them in three batches, each consisting of 20 packets and separated by 20 ms. The 20-ms delay was the minimum chosen in such a way that longer delays did not lead to lower average loss rate. (Such a choice is, of course, not TCP friendly.) In contrast, the pacing and retransmissions of TCP packets were controlled by the TCP protocol itself.

The graphs in Figure 1 show that the end-to-end response times of UDP delivery have far less variations and are shorter than those of TCP delivery. For example, for transmissions between Urbana and China (Figure 1a), UDP delays range from 0.5 to 1.2 seconds, whereas TCP delays range from 18 to 1186 seconds. In terms of average speed, TCP transfers are one to two orders slower than UDP transfers. The long delay in TCP is attributed to its coarse grained timeouts and congestion avoidance algorithms. In practice, this means that we may need to wait for an average of over two minutes to download an image from China by TCP.

Although UDP delivery is much faster, it suffers from losses that may lead to large degradations in image quality or render images not decode-able. In order to send images using UDP, we need to understand the conditions under which losses can be concealed.

It is well known that interleaving can convert bursty losses to random losses and facilitate recovery by exploiting inherent redundancies in image data. To conceal bursty losses most of the time, it is necessary to choose interleaving factor i so that the probability of packets that are not recoverable using i is small enough.

Let the total number of packets sent be n_p and the interleaving factor be i. Over all the interleaved sets, assume that losses of j consecutive packets, $j \le i$, happen m_j^i times.[1] After deinterleaving, a packet cannot be reconstructed when all other packets in the same interleaved set are lost. Hence, $Pr(fail \mid i)$, the unconditional probability that a packet cannot be reconstructed using interleaving factor i, can be computed as follows:

$$Pr(fail \mid i) = Pr(fail \mid loss, i) \times Pr(loss) = \frac{i \times m_i^i}{n_p}. \quad (1)$$

Figure 2 shows that $Pr(fail \mid i)$ drops quickly with increasing interleaving factor i. In summary, the probability of not able to reconstruct a lost packet is under 5% with an interleaving factor of 2 for connections to California, and with an interleaving factor of 4 to China. Based on the statistics collected, we conclude that UDP delivery can be one to two orders faster than TCP delivery, and that bursty losses in UDP delivery can be concealed by interleaving and by reconstruction using an interleaving factor of four for most Internet destinations.

3. ORB-ST FOR CONCEALING BURSTY LOSSES

Although interleaving and interpolations are effective for concealing bursty losses, simple coding of interleaved streams may not work well because the original coding algorithm may not be the best for reconstructing lost streams. In this section we propose a new optimized reconstruction-based subband transform (ORB-ST) that takes into account the reconstruction process at receivers. A different derivation of an optimized reconstruction-based DCT transform for video coding can be found elsewhere [5].

In the following, we first derive ORB-ST based on partitioning image data into two descriptions. Its extension to four descriptions is omitted due to space limitations [5]. Next, we compare the quality of images transformed by ORB-ST and by the original subband transform (ST) using trace-driven experiments.

Figure 3 shows the basic building blocks in our proposed subband image coding system for two descriptions. It is based on existing state-of-the-art image codecs that consist of several stages: a subband transformation, a quantizer and an optional entropy coder.

Assume that each row of the original image, $\vec{\mathbf{x}}$ of size n, is transformed into $\vec{\mathbf{c}}_0$ of size $\frac{n}{2}$ and $\vec{\mathbf{c}}_1$ of size $\frac{n}{2}$, corresponding to the descriptions of even-numbered and odd-numbered pixels. Here, $\vec{\mathbf{c}}_i, i = 0, 1$, is an interleaved vector of components from $\vec{\mathbf{c}}_0^i$ and $\vec{\mathbf{c}}_1^i$, where $\vec{\mathbf{c}}_j^i$ is the output from subband j, and subbands are ordered from low to high frequency.

[1] Note that the case $j > i$ does not need to be considered because loss concealment is only carried out on each interleaved set and not across interleaving sets.

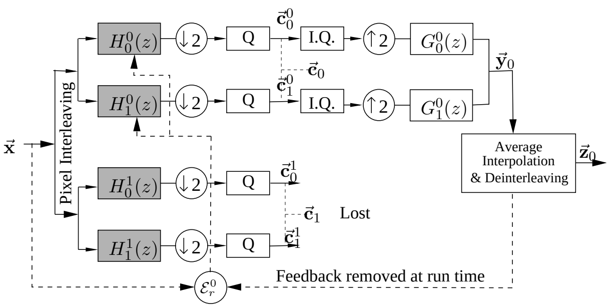


Figure 3: Basic building blocks of a modified codec. (The shaded block is our proposed ORB-ST.)

Our objective is to find $\vec{\mathbf{c}}_0$ and $\vec{\mathbf{c}}_1$ in order to minimize $\mathcal{E}_r$, the reconstruction error between reconstructed output $\vec{\mathbf{z}}_i$ and input $\vec{\mathbf{x}}$. If we consider quantization, the minimization of $\mathcal{E}_r$ becomes an integer optimization problem, because $\vec{\mathbf{c}}_i$ takes integer values. Such optimizations are computationally prohibitive in real time. In the following, we derive an approximate solution that does not take into account quantization effects. Since the derivations are similar, we only show that for $\vec{\mathbf{c}}_0$.

As the synthesis system, consisting of up-sampling, $G_0(z)$, and $G_1(z)$, is equivalent to a linear transform $\hat{\mathsf{G}}$ in spatial domain, output $\vec{\mathbf{y}}_0$, after synthesis filtering, can be calculated as:

$$\vec{\mathbf{y}}_0 = \hat{\mathsf{G}}\ \vec{\mathbf{c}}_0. \tag{2}$$

The set of interpolated pixels, $\vec{\mathbf{z}}_1$, obtained by inserting odd-numbered columns as the average of columns from $\vec{\mathbf{y}}_0$, with the boundary column duplicated, can also be expressed as a linear transform of $\vec{\mathbf{y}}_0$ as follows:

$$\vec{\mathbf{z}}_0 = \mathbf{U}\ \vec{\mathbf{y}}_0 = \mathbf{U}\ \hat{\mathsf{G}}\ \vec{\mathbf{c}}_0. \tag{3}$$

Hence, the distortion between the original and the reconstructed pixels becomes:

$$\mathcal{E}_r^0 = \left\| \mathbf{U}\hat{\mathsf{G}}\vec{\mathbf{c}}_0 - \vec{\mathbf{x}} \right\|^2 = \|\mathsf{P}\vec{\mathbf{c}}_0 - \vec{\mathbf{x}}\|^2. \tag{4}$$

Since the linear system of equations $\mathsf{P}\vec{\mathbf{c}}_0 = \vec{\mathbf{x}}$ is an over-determined one, there exists at least one least-square solution $\vec{\mathbf{c}}_0$ that minimizes (4), according to the theory of linear algebra. Specifically, the solution $\vec{\mathbf{c}}_0$ with the smallest length $|\vec{\mathbf{c}}_0|^2$ can be found by first performing SVD decomposition of matrix P:

$$\mathsf{P} = \mathbf{S}\ [diag(w_j)]\ \mathbf{D}^T, \qquad j = 1, 2, \ldots, \frac{n}{2}, \tag{5}$$

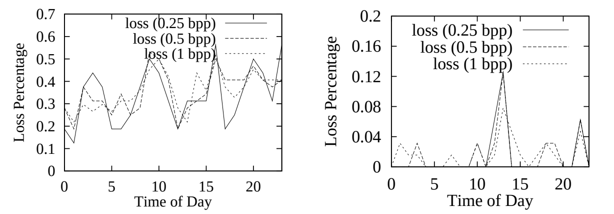
where $\mathbf{S}$ is an $n \times \frac{n}{2}$ column-orthogonal matrix, $[diag(w_j)]$, an $\frac{n}{2} \times \frac{n}{2}$ diagonal matrix with positive or zero elements (singular values), and $\mathbf{D}$, an $\frac{n}{2} \times \frac{n}{2}$ orthogonal matrix. Then the least-square solution can be expressed as:

$$\vec{\mathbf{c}}_0 = \mathbf{D}\ [diag(1/w_j)]\ \mathbf{S}^T\ \vec{\mathbf{x}}. \tag{6}$$

In the above diagonal matrix $[diag(1/w_j)]$, element $1/w_j$ is replaced by zero if w_j is zero. Therefore, ORB-ST is a product of three matrices: $\mathbf{D}$, $[diag(1/w_j)]$, and $\mathbf{S}^T$.

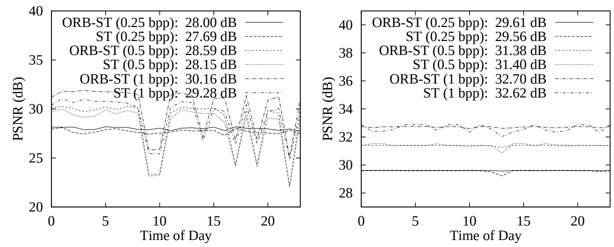
To derive the ORB-ST transform for $\vec{\mathbf{c}}_1$, simply replace $\mathbf{U}$ by the matrix corresponding to the interpolation of odd-numbered columns, and the rest of the steps are similar.

When both $\vec{c}_0$ and $\vec{c}_1$ are available at the receiver, we can apply an inverse transform to achieve perfect reconstruction, where the inverse transform is derived by inverting the matrix that is made up of interleaved row vectors from ORB-ST for $\vec{\mathbf{c}}_0$ and $\vec{\mathbf{c}}_1$. In practice, perfect reconstruction is not possible due to errors caused by truncations of floating point numbers as well as quantization.



a) Urbana-China b) Urbana-California

Figure 4: Loss rates of 16-, 32- and 64-packet transmissions from Urbana to two remote sites.



a) Urbana-China b) Urbana-California

Figure 5: Comparisons of reconstruction quality in transmitting image *peppers* over a 24-hour period when each image was coded at, respectively, 0.25 bpp, 0.5 bpp, and 1 bpp, and placed in 16, 32, and 64 packets for transmission.

To evaluate our proposed scheme, we built a prototype and tested the quality of frames reconstructed by linear interpolations of adjacent pixels received when the original frame was either ST transformed or ORB-ST transformed. For a fair comparison under the same traffic conditions, we did trace-driven simulations by applying reconstructions on the trace of packets received in real Internet transmissions (see Section 2).

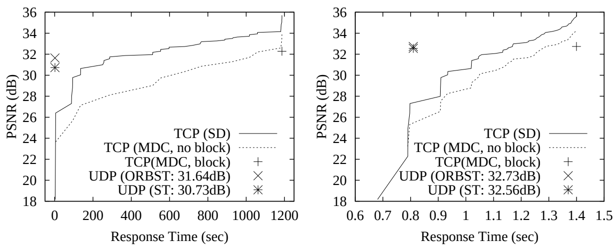
In order to packetize coded descriptions in such a way that each packet can be decoded independently, we first divide an image into equal-size segments, code each using the same bit rate, and place a coded segment into a distinct packet. Since we have used suboptimal strategies in dividing segments into equal sizes and coding each using the same bit rate, we expect lower image quality when compared to that of MDC without segmentation.

Figure 4 plots the loss rates of traces over a 24-hour period when sending 16, 32, and 64 packets to the UDP echo ports of two remote servers. Figure 5 compares the reconstruction quality of sending test image *peppers* using the traces obtained, when the image was coded at, respectively, 0.25 bpp, 0.5 bpp, and 1 bpp and put into 16, 32, and 64 packets for transmission.

For the Urbana-China connection, ORB-ST outperforms ST at all bit rates for both images, with an average of 0.31-0.38 dB gain for the 0.25-bpp case, 0.44-0.48 dB gain for the 0.5-bpp case, and 0.61-0.88 dB gain for the 1-bpp case. Quality gains improve with increasing bit rates when there is less quantization noise. When entire interleaved sets were lost at certain hours, they were simply filled in by the average of image pixels, leading to significant quality degradations (such as hours 9, 11, 17 and 19 at 1 bpp).

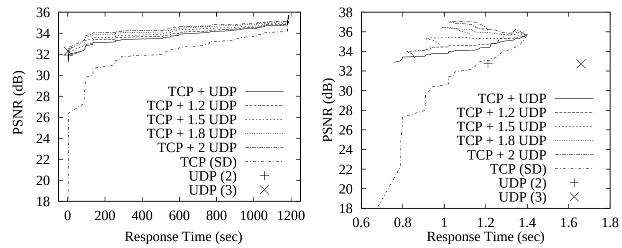
For the Urbana-California connection, the reconstruction quality of the two schemes are comparable. In these two cases, the gain of performing ORB-ST is, in general, not as much as in the Urbana-China connection because the gain is offset by degradations when all the descriptions are received under low loss rates.

These results show that ORB-ST is more suitable for the delivery of images over unreliable channels than the original ST.



a) Urbana-China b) Urbana-California

Figure 6: Delay-quality trade-offs between TCP delivery of transmitting SDC image data *pepper* to two remote sites at 12 noon their local time. (The behavior at other times is similar.)



a) Urbana-China b) Urbana-California

Figure 7: Delay-quality trade-offs of the combined TCP/UDP delivery of SDC/MDC data for *peppers* for the transmissions from Urbana to two remote sites at 12 noon their local time.

4. DELAY-QUALITY TRADE-OFFS

In this section, we evaluate the delay-quality trade-offs between the UDP delivery of MDC images and the TCP delivery of SDC images. Figure 6 shows such trade-offs at 12 noon local time of the remote server using five modes of delivery: a) TCP delivery of SDC image data, b) TCP delivery of MDC data in which the image is not segmented, c) TCP delivery of MDC data in which the image is segmented, d) UDP delivery of MDC ST-coded and segmented image data, and e) UDP delivery of MDC ORB-ST-coded and segmented image data. Results at other times are similar.

In Figure 6, the two curves and one point related to TCP delivery were obtained by assuming that each image was coded in 1 bpp and transmitted in 64 packets. Based on the statistics collected, we calculated the average arrival times of the first i, $i = 1, 2, \ldots, 64$, packets and evaluated the quality of the corresponding packets after decoding them by the SPIHT decoder. The times in each curve include both end-to-end delays and decoding times.

The two points related to UDP delivery were obtained under 1 bpp and included end-to-end delays, decoding time, and reconstruction time when losses happened.

The graphs show that UDP delivery of MDC images is an attractive alternative to TCP delivery of SDC images when an end user only tolerates small delays and when absolute quality is not critical. Further, TCP delivery generally leads to poorer quality using the same amount of time required by UDP delivery.

The graphs also illustrate three factors that cause the degradation in quality by several dBs between TCP delivery of SDC images and UDP delivery of MDC images.

First, MDC alone causes between 1 to 3.5 dB loss in PSNR due to reduced correlations when partitioning an image into multiple descriptions and the suboptimal fixed coding rate for each description. This is illustrated by the difference between the top two curves in each graph.

Second, another 2 to 3.5 dB loss in PSNR is caused by the suboptimal strategies of using fixed-size segments in the segmentation of image data in each description and of using a fixed coding rate for each segment in order for the coded segment to fit in a 512-byte packet (the difference between the point on the right of the dotted line and the cross on the right of each graph).

Third, packet losses and reconstructions in delivering segmented ST-MDC data by UDP lead to further degradations of up to 2 dB.

The delay-quality trade-offs studied previously only show two extreme cases of image transmission, either by TCP or by UDP. A promising hybrid approach is to combine TCP and UDP in order to give better trade-offs. For TCP delivery, quality improves very quickly in the beginning but saturates when more packets are available. Since the first few packets delivered by TCP incur insignificant delays, we can transmit them by TCP and deliver the MDC residuals by UDP. In short, we have:

$$\{x : \text{SDC by TCP}\} \cup \{\alpha(1-x) : \text{MDC by UDP}\},$$

where $0 \leq x \leq 1$ and $\alpha \geq 1$. In this approach, the first $x\%$ of the bit stream is coded by SDC and delivered by TCP, and the rest of the bit stream is coded either redundantly ($\alpha > 1$) or non-redundantly ($\alpha = 1$) by MDC and delivered by UDP.

The combined approach can, to some extent, reduce the three kinds of losses identified earlier, depending on the factor x chosen. Since the first $x\%$ of the bit stream is coded in SDC and transmitted by TCP, it suffers from none of the above three kinds of losses. The larger the x is, the less degradation in quality one has to pay and the longer delay one has to wait. In fact, if both x and α are equal to one, this approach is reduced to pure TCP delivery of SDC images, leading to the best quality and the longest delay.

To evaluate these trade-offs, we compare the following approaches for two chosen sites and image *peppers*: a) TCP delivery of SDC images; b) combined TCP/UDP delivery of SDC and MDC images, with α set to $1, 1.2, 1.5, 1.8$ and 2, respectively; and c) redundant UDP delivery of MDC images by sending y copies of UDP packets that contain MDC data, with y set to 2 or 3.

Figure 7 illustrates the trade-offs in transmitting *peppers* to China and California, respectively. We can see that the quality of the combined approach improves with increasing α, and that the combined approach can generate a range of trade-offs with better quality than pure UDP delivery and with shorter delays than pure TCP delivery. Further, the redundant UDP delivery of MDC images does not appear to be an attractive approach due to it high redundancy, mediocre quality, and long delays.

In summary, the hybrid approach allows users to choose a suitable combination between UDP and TCP deliveries in order to suit their QoS requirements and network resources.

5. REFERENCES

[1] I. Sodagar, H.-J. Lee, P. Hatrack, and Y.-Q. Zhang, "Scalable wavelet coding for synthetic/natural hybrid images," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 9, pp. 244-254, Mar. 1999.

[2] S. D. Servetto, K. Ramchandran, V. A. Vaishampayan, and K. Nahrstedt, "Multiple description wavelet based image coding," *IEEE Trans. on Image Processing*, vol. 9, pp. 813-826, May 2000.

[3] V. A. Vaishampayan and N. Farvardin, "Optimal block cosine transform image coding for noisy channels," *IEEE Trans. on Communications*, vol. 38, pp. 327-336, Mar. 1990.

[4] C. J. Turner and L. L. Peterson, "Image transfer and end-to-end design," in *Proc. SIGCOMM'92*, 1992, pp. 258-268.

[5] X. Su, *Error Concealments for Robust Image and Video Transmissions on the Internet*, Ph.D. Thesis, Dept. of Computer Science, Univ. of Illinois, Urbana, IL, Aug. 2001.