Multiple Descriptions

Source-Channel Coding Methods for Communications

Jelena Kovačević and Vivek K Goyal

Bell Labs, Innovations for Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ 07974, USA

Abstract. We describe multiple description transform coding (MDTC), a joint source-channel coding method for robust data communication. MDTC achieves robustness and graceful degradation in the presence of erasures by representing a source using several descriptions. Then, if one of the descriptions is lost, we can reconstruct from those received. After the theoretical foundation is laid out, applications to robust audio and image transmission are presented.

1 Introduction

In recent years the following problem has received considerable attention: Suppose we are transmitting data over network links of different capacities. Such networks are called heterogeneous networks. A typical scenario might involve a high-capacity network link switching over to a wireless link where packets have to be dropped to accommodate the lower capacity of the latter link. Packets could also be lost in the network due to transmission errors or congestion. If packet retransmission is not an option (for example, due to real-time constraints), one has to devise a way of getting meaningful information to the recipient despite the loss.

This problem finds its natural solution in the so-called multiple-description framework. The idea is to send multiple descriptions of a single source to the receiver. Each of the descriptions can be lost with a certain probability. If all the descriptions are received, we want a high-fidelity estimate of the original data. If only some of the descriptions are received, we want to be able to reconstruct the transmitted data as well as possible. These requirements imply that each of the descriptions should individually be good and thus close to the original data. If this is true, however, the descriptions are very similar so receiving more descriptions will add little extra information. For the descriptions to be collectively good and not add much to the data rate, they must be relatively independent, but such descriptions cannot be individually good. These conflicting requirements will lead to the trade-offs typical in the design of joint source-channel coding systems.

The original multiple description problem with two descriptions and three receivers was posed by Gersho, Witsenhausen, Wolf, Wyner, Ziv and Ozarow at the 1979 IEEE Information Theory Workshop. It can be formulated as the search for achievable quintuples $(R_1, R_2, D_0, D_1, D_2)$ where $R_i, D_i, i = 1, 2$ are the rates and distortions for the ith channel, respectively, and D_0 is the distortion when both descriptions are received. Interestingly, to this day, the problem has been solved only for the memoryless Gaussian source with squared error distortion (Ozarow [1]). El Gamal and Cover [2] construct an achievable rate region for a memoryless source. To show how difficult the problem is, consider, for example, the special case of an i.i.d. binary source with $R_1 = R_2 = 1/2$, $D_0 = 0$ and $D_1 = D_2 = D$. How small can D be? A naive approach of sending alternate source symbols over the two channels results in $D = 0.25$; evaluating the distortion-rate function at rate $1/2$ implies that $D \geq 0.11$. In between these extremes lies the minimum D. In 1980, Wolf, Wyner and Ziv [3] showed that $D \geq 1/6 \approx 0.16$ [3], and the following year Witsenhausen and Wyner [4] improved the lower bound to $D \geq 1/5 = 0.2$. In 1982, El Gamal and Cover [2] gave the upper bound $D \leq (\sqrt{2} - 1)/2 \approx 0.207$. Finally, in 1983, Berger and Zhang [5] closed the gap and showed that $D = (\sqrt{2} - 1)/2$.

Vaishampayan [6] has worked extensively on designing multiple description scalar quantizers as well as on applying the multiple description framework to speech, image and video coding.

We consider multiple description joint source/channel transform coding for erasure channels. This is motivated by the fact that the limitations of separate source and channel coding (spurred by Shannon's famous "separation principle") have lead many researchers to the problem of designing joint source-channel (JSC) codes. An examination of Shannon's result leads to the primary motivating factor for constructing joint source-channel codes: The separation theorem is an asymptotic result which requires infinite block lengths (and hence infinite complexity and delay) at both source coder and channel coder; for a particular finite complexity or delay, one can often do better with a JSC code. JSC codes have also drawn interest for being robust to channel variation.

We propose two methods: In the first, a block of n independent, zero-mean Gaussian variables with different variances are transformed to a block of n transform coefficients in order to create a known statistical correlation between transform coefficients. The transform coefficients from one block are distributed to different packets so in the case of a packet loss, the lost coefficients can be estimated from the received coefficients. The redundancy comes from the relative inefficiency of scalar entropy coding on correlated variables. This method is a generalization of the technique proposed in [7, 8] for two channels. The second method uses a *deterministic* redundancy between descriptions, introduced by representing the source data via a frame. That is, we use a frame expansion from C^n to C^m $(m > n)$ as a computationally simple approach to generalized MD coding [9].

2 Multiple Descriptions Using Statistical Redundancy

In this first method (see [10, 11]), a block of k independent, zero-mean variables with different variances are transformed to a block of k transform coefficients in order to create a known statistical correlation between transform coefficients. The coding of a source vector x proceeds as follows:

1. x is quantized with a uniform scalar quantizer with step size Δ: $x_{q_i} = [x_i]_\Delta$, where $[\,\cdot\,]_\Delta$ denotes rounding to the nearest multiple of Δ.
2. The vector $x_q = [x_{q_1},\ x_{q_2},\ \ldots\ x_{q_k}]^T$ is transformed with an invertible, discrete transform $\hat{T} : \Delta Z^k \to \Delta Z^k$, $y = \hat{T}(x_q)$. The design and implementation of $\hat{T}$ are described below.
3. The components of y are independently entropy coded.
4. If $n < k$, the components of y are grouped to be sent over the n channels.

The discrete transform $\hat{T}$ is related to a continuous transform T through "lifting." Starting with a linear transform T with determinant one, the first step in deriving a discrete version $\hat{T}$ is to factor T into a product of upper and lower triangular matrices with unit diagonals $T = T_1 T_2 \cdots T_m$. The discrete version of the transform is then given by

$$\hat{T}(x_q) \;=\; \left[T_1 \left[T_2 \ldots [T_m x_q]_\Delta\right]_\Delta\right]_\Delta . \tag{1}$$

The lifting structure ensures that the inverse of $\hat{T}$ can be implemented by reversing the calculations in (1):

$$\hat{T}^{-1}(y) \;=\; \left[T_m^{-1} \ldots \left[T_2^{-1} \left[T_1^{-1} y\right]_\Delta\right]_\Delta\right]_\Delta .$$

When all the components of y are received, the reconstruction process is to (exactly) invert the transform $\hat{T}$ to get $\hat{x} = x_q$. The distortion is precisely the quantization error from Step 1. If some components of y are lost, they are estimated from the received components using the statistical correlation introduced by the transform $\hat{T}$. The estimate $\hat{x}$ is then generated by inverting $\hat{T}$. The reader is referred to [10, 11] for the algebraic details.

The optimal design of the transform $\hat{T}$ for Gaussian sources, where arbitrary (unequal, dependent) packet loss probabilities are allowed, is discussed in [10, 11]. Here we consider the simpler case where packet losses are i.i.d. and the transform is implemented as parallel and/or cascade combinations of 2-by-2 transforms. It is shown in [10, 11] that for coding a two-tuple source over two channels, where each is equally like to fail, it is sufficient to consider transforms of the form

$$T_a \;\stackrel{\text{def}}{=}\; \begin{bmatrix} a & 1/(2a) \\ -a & 1/(2a) \end{bmatrix} . \tag{2}$$

We use this as a building block to form larger transforms as in the cascade structures [11]. The cascade structure simplifies the encoding, decoding, and design when compared to using a general $k \times k$ transform. Empirical evidence suggests that for $k = 4$ and considering up to one component erasure, there is no performance penalty in restricting consideration to cascade structures.

3 Multiple Descriptions Using Frames

Robustness to lost packets comes from redundancy in the source representation. In the previous technique, the redundancy is *statistical*: the distribution of one part of the representation is reduced in variance by conditioning on another part. The second method that we consider (see [9]) uses a *deterministic* redundancy between descriptions.

Consider a discrete block code which represents k input symbols through a set of n output symbols such that any k of the n can be used to recover the original k. (For concreteness, this may be a systematic (n, k) Reed-Solomon code over $GF(2^m)$ with $n = 2^m - 1$ [12].) If the k input symbols are quantized transform coefficients, this may be a good way to communicate a k-dimensional source over an erasure channel that erases symbols with probability less than $(n-k)/n$. A problem with this approach is that except in the case that exactly k of n transmitted symbols are received, the channel has not been used efficiently. When more than k symbols are received, those in excess of k provide no information about the source vector; and when less than k symbols are received, it is computationally difficult to use more than just the systematic part of the code.

An alternative to (discrete) block coding was proposed in [9]. A linear transform from R^k to R^n, followed by scalar quantization, is used to generate n descriptions of a k-dimensional source. These n descriptions are such that a good reconstruction can be computed from any k descriptions, but also descriptions beyond the kth are useful and reconstructions from less than k descriptions are easy to compute.

Assume that we have a tight frame $\Phi = \{\varphi_m\}_{k=1}^{n} \subset R^k$ with $\|\varphi_m\| = 1$ for all m and that $y = Fx$, where F is the frame operator associated with Φ (see [13] and references therein for details on frames). This vector passes through the scalar quantizer Q: $\hat{y} = Q(y)$. The entropy-coded components of $\hat{y}$ can each be considered a description of x.

For simplicity, let us assume that Q is a uniform quantizer with step size Δ and that $n < 2k$. If $m \geq k$ of the components of $\hat{y}$ are known to the decoder, then x can be specified to within a cell with diameter approximately equal to Δ and thus is well approximated. Since the constraints on x provided by each description are independent, on average, the diameter is a nonincreasing function of m.

When $m < k$ components of $\hat{y}$ are received, R^k can be partitioned into an m-dimensional subspace and a $(k-m)$-dimensional orthogonal subspace such that the component of x in the first subspace is well specified. With a mild zero-mean condition on the component in the latter space, a reasonable estimate of x is easily computed. For any m, estimating x can be posed as a simple least-squares problem.[1]

Let us analyze the distortion when e of the descriptions are erased. Let E denote the index set of the erasures, that is, suppose $\{\langle x, \varphi_m \rangle\}_{m \in E}$ are erased. If $\Phi' = \Phi \setminus \{\varphi_m\}_{m \in E}$ is a frame, the minimum MSE estimate $\hat{x}$ is obtained with the dual frame of Φ'; otherwise, x can only be estimated to within a subspace and distributional knowledge is needed to get a good estimate.[2] We model the quantizer as an additive white noise source,[3] so $\eta = \hat{y} - Fx$ has independent components and is independent of x with $E|\eta_m|^2 = \sigma^2$.

When there are no erasures, the error between the reconstructed signal $\hat{x}$ and source x is due only to the quantization noise. The MSE in this case is $\mathrm{MSE}_0 = k^2\sigma^2/n$ [13]. Now suppose (renumbering, if necessary) $E = \{1, 2, \ldots, e\}$ and let F_e denote the frame operator associated with Φ'. The MSE can then be written as

$$\begin{aligned} \mathrm{MSE}_e &= \sigma^2 \sum_{m=e+1}^{n} \|(F_e^* F_e)^{-1} \varphi_m\|^2 \\ &= \left[1 - \frac{e}{n} + \frac{k}{n^2} \sum_{i,j=1}^{e} B_{ij}^{(e)} C_{ij}^{(e)} \right] \mathrm{MSE}_0, \end{aligned}$$

where $B^{(e)}$ and $C^{(e)}$ depend on the inner products between erased vectors. A simple special case is when the erased components are pairwise orthogonal. In this case, (3) reduces to $\mathrm{MSE}_e = (1 + e/(n-k))\mathrm{MSE}_0$.

The coding method proposed here could be viewed as a signal-domain alternative to a (discrete) rate-(k/n) block channel code. Though discussion here was limited to the general framework and the distortion with different numbers of erasures, preliminary calculations and simulations indicate potential for the proposed method. Compared to the use of a discrete channel code, the proposed method seems to give better performance at high rates and to be less sensitive to knowledge of the erasure probability. Minimum norm reconstructions can be computed in polynomial time for any e.

4 Applications

In this section, we give examples of the MDTC applied to robust audio as well as robust image transmission systems. They are meant to illustrate the potential of the proposed method; for more details, the reader is referred to [15, 16].

---

[1] For $m \geq k$, a better estimate can be found by exploiting the boundedness of the quantization error [13, 14].

[2] Extensions to where Φ' is not a frame are suggested by [10].

[3] Better reconstructions are possible when the boundedness of the quantization error is exploited [13, 14].

4.1 Multiple Description Source-Channel Coding of Audio

Audio compression uses a digital representation of audio signals to provide maximum signal quality with a given number of bits, delay and cost. Human perception plays a key role in compression of audio material. As a result, recent audio standards work has concentrated on a class of audio coders known as *perceptual coders*. Rather than trying to understand the source, they model the listener and attempt to remove *irrelevant* information contained in the input signal.

We apply the MDTC to a state-of-the-art audio coder developed at Bell Labs, the *Perceptual Audio Coder (PAC)* [17]. Instead of developing a new audio coder from scratch, we implement this new technique in a well-known and fully operational audio coder.

PAC Coder Like most perceptual coders, the PAC combines both source coding techniques to remove signal *redundancy* and perceptual coding techniques to remove signal *irrelevancy*. The PAC divides the input signal into 1024-sample-blocks of data – *frames* – that will be used throughout the encoding process. It consists of five basic parts: The *analysis filter bank* converts the time-domain data to frequency domain. The *perceptual model* computes the frequency-domain threshold of masking both from the time-domain signal and from the output of the analysis filter bank. Depending on the transform that was used previously, each 1024-block is split into a predefined number of groups of bands – *gain factor bands*. Within each factor band, a perceptual threshold value is computed. In the *quantization* process, within each factor band the quantization step sizes are adjusted according to the computed perceptual threshold values in order to meet the noise level requirements. Once a quantized representation that fits with the constraint on the coded signal bit rate has been obtained, *noiseless coding* such as Huffman coding is used to provide an efficient representation of the quantized coefficients. The *frame formatter* forms the bit stream, adding to the coded quantized coefficients the side information needed at the decoder to reconstruct the 1024-sample-block. This block is defined as the *frame*.

At the decoder, all the blocks are inverted and an error mitigation block is added between the inverse quantization and the synthesis filter bank. In this block, lost frames are interpolated based on the previous and following frames.

MD PAC Coder The only difference when compared to the PAC coder is the addition of the MDTC block together with the off-line design.

An *MD transform* block is inserted between the quantizer and the noiseless coder. Within each 1024-sample-block or eight 128-sample-blocks contained in the 1024-sample-unit-block, MDTC is applied to the quantized coefficients (integers) coming out of the quantizer. The transform is applied to pairs of quantized coefficients and produces pairs of *MD-domain quantized coefficients*, using the off-line designed side information. Within each pair, MD-domain quantized coefficients are then assigned to Channel 1 (quantized coefficient with higher variance) or Channel 2 (quantized coefficient with smaller variance)[4]. Side information contains both the way quantized coefficients have to be paired and the parameter of the transform for each pair. Then, the MD-domain quantized coefficients are passed to the noiseless coder.

We insert the inverse MDTC block between the noiseless decoder and the inverse quantizer. Within each 1024-sample-block or eight 128-sample-blocks contained in the 1024-sample-unit, the inverse MDTC function is applied to the MD-domain quantized coefficients (integers) coming out of the noiseless decoder. Then, if both channels are received, we invert the MD transform exactly, recovering perfectly the quantized coefficients. If only one of the channels is lost, we estimate its lost coefficients from their counterparts in the other channel, and invert the MDTC. If both channels are lost, we use the built-in loss mitigation feature of the PAC.

As in the encoder, side information provides the way quantized coefficients have to be paired, the parameter a of the inverse transform for each pair, and the variances to be used in the estimation of lost MD-domain quantized coefficients. Once the MDTC has been inverted according to one of these four strategies, the output quantized coefficients are simply passed to the inverse quantizer.

Since we have blocks of 1024 coefficients and we want to group them into two channels, we need to design the pairing. As described in [11], when there are $2N$ variables and two channels, the optimal pairing consists of pairing the variable with the highest variance with the one with the lowest variance, the second highest variance variable with the second lowest variance one, etc.

Therefore, we first tried pairing in the optimal fashion, that is, across all bands. According to this scheme, we can have either 1024 or 128 variables that we have to pair, leading to either 512 or 64 pairs. Of course, since factor bands may have different quantization steps, this approach implies a rescaling of the domain spanned by the variables prior to the application of MDTC, by multiplying variables by their respective quantization steps.

As will be explained later, the optimal pairing across all bands did not work well. Thus we now take the factor bands into account, and pair variables belonging to the same factor band.

The next step was to design the correlating transform T_a defined by (2). For each pair we obtain the transform parameter a and find both the optimal

[4] If the source is Gaussian, uncorrelated, zero mean and if the quantizer preserves the zero mean, then the outputs of the MDTC block should have the same variance. The reason why this is not the case here is that either the quantizer does not preserve the zero mean or that the source is sufficiently far away from Gaussian. We leave this investigation for future work.

redundancy allocation between pairs and the optimal a for a given set of variances and their pairing. After the application of MDTC, we pass the two channels (the MD-domain quantized coefficients) from the 1024 or 128-sample-block to the noiseless coder of the PAC. We do not entropy code each set separately. We do this for convenience and do not optimize the codebooks for our MD-domain quantized coefficients. Since we lose by doing it, we feel that the comparison between the SD PAC and MD PAC is fair. It is part of future work for us to actually separate the channels and optimize codebooks. From one set of variables, the MDTC scheme produces two distinct channels that have to be sent separately through a network. Since these two channels have to be sent separately, side information of the original frame has to be doubled and sent with each channel.

Experimental Results Here we discuss the experiments we did to compare the MD PAC to the original single description version – SD PAC. Since we were interested in Internet audio applications, we selected a bit rate of 20 kbps. Our experiments with the MD PAC were all done with a small amount of redundancy, $\rho = 0.1$ bits per variable.

EXP 1: The first experiment is to compare SD PAC and MD PAC at the same bit rate when no frames are lost. Since we are introducing redundancy, the MD version should sound slightly worse.

EXP 2: Then, still without losing anything, we increase the bit rate in the MD PAC until we reach the same quality as in SD PAC. Here we can see the price we pay in bits for robustness; these bits are wasted when no data is lost.

EXP 3: Finally, we compare the MD PAC and SD PAC at various loss rates. The SD PAC uses frame interpolation to recover from lost frames. If frame information from only one channel is lost, the MD PAC uses statistical redundancy. If frame information from both channels is lost, that is, we lost the whole frame, then we use the SD PAC error recovery scheme.

In what follows, P_i is the loss rate of Channel i, $i = 1, 2$. We also define the overall loss rate as $P_{tot} = (P_1 + P_2)/2$. For example, if $P_1 = 20\%$, then 20% of half-frames corresponding to Channel 1 are lost.

Pairing Across All Bands When we performed Experiment 1, we were quite disappointed. The quality degradation in the MD PAC was extreme. Here is an explanation why:

After applying MDTC to the quantized coefficients, the MD-domain outputs were simply passed to the original PAC noiseless coder. Since the correlating transforms have been designed to produce two equal-rate outputs, we are introducing nonzero values at the positions where the noiseless coder is expecting zeros. Thus, modifying the input to the noiseless coder in such a way led to ineffective coding, resulting in quality degradation. A solution to this problem would be to design and optimize new entropy coders to be used for the MD-domain quantized coefficients. We leave this for future work.

Pairing Within Factor Bands We now restrict ourselves to pairing variables belonging to the same factor band. Throughout the rest of the paper, we will be pointing the reader to the results of our experiments. The audio files are provided in three formats (*aiff*, *wave* and *next*) and can be accessed on the Web at: `http://cm.bell-labs.com/who/jelena/` under `Interests/MD/AudioDemo/DemoList.html`.

We performed **EXP 1** at 20 kbps. The quality degradation due to the redundancy was very low, though noticeable for expert listeners (listen to Files 1 and 2 under "No losses").

In **EXP 2**, the difference was hardly noticeable (listen to Files 1 and 3 under "No losses"). The price we pay is an extra 2 kbps for the MD PAC.

EXP 3 was performed at various loss rates. We started with $P_1 = 100\%$ and $P_2 = 0\%$, and then $P_1 = 0\%$ and $P_2 = 100\%$. A very annoying high-frequency artifact appeared in the decoded files. The higher the loss rates, the more present this artifact was. It turns out that the high-frequency artifact came from overvaluation of variables within a particular factor band, the one where the variances drop to very low levels (the limit of the spectrum that will be coded at a given bit rate). This overvaluation seems to stem from the huge difference of variances of the variables within this factor band, leading to a very small transform parameter a.

To improve the estimation within this factor band, we first set the variables belonging to this factor band to zero. The artifact disappeared. However, since the resulting decoded files lost their highest frequencies, they sounded quite unnatural (listen to File 3 under "Loss rate = 50%").

Note that above, we simulated only the extreme case when one entire channel is lost. We now discuss the results when losses are spread over both channels. The SD PAC error recovery scheme is particularly effective below $P_{tot} = 10\%$. However, for higher loss rates, the interpolation process becomes less and less sufficient, and gaps appear in the music (listen to File 4 under "Total loss rate = 50%").

On the other hand, the MD PAC estimation process seems to have a better behaviour at high loss rates, typically over 80% of one of the channels (listen to Files 1, 2, 3 and 4 under "Total loss rate = 50%"). Even if the estimation tends to be noisy as we have seen previously, the result is satisfactory. However, for loss rates around 50%, the noisy effects are annoying. This might be due to the fact that we are jumping from the spectrum of a perfectly received frame to one estimated in the case of a loss. Furthermore, it seems that the estimated spectrum is not only noisy but also biased from the original one. Moreover, audio is inherently nonstationary, and we are using only one set of estimated variances for an entire audio file. This leads to the differences between original and estimated spectra. As future work, we will implement an adaptive scheme where the variances are estimated on shorter pieces of music. At low loss rates, the behaviour of the MD PAC coder is effective, since the previously described effect is hardly present (listen to Files 1,2 and

3 under "Total loss rate = 5%"). At the loss rate of $(P_1, P_2) = (20\%, 20\%)$, which means that the percentage of loss of both half-frames is $P_{12} = 4\%$, the quality is satisfactory, even though we are mixing spectra from three sources: received, estimated and interpolated (listen to Files 1,2 and 3 under "Total loss rate = 20%"). However, at the loss rate of $(P_1, P_2) = (40\%, 40\%)$, $P_{12} = 16\%$, the decoded file is loaded with annoying artifacts coming from this mix as well as clipping due to the high P_{12}.

4.2 MDTC of Images

MDTC of Images Using Statistical Redundancy To demonstrate the efficacy of the correlating transform method for image coding, we consider the case of coding for four channels. This method is designed to operate on source vectors with uncorrelated components. We (approximately) obtain such a condition by forming vectors from DCT coefficients separated both in frequency and in space. A straightforward application proceeds in the following steps:

1. An 8-by-8 block DCT of the image is computed.
2. The DCT coefficients are uniformly quantized.
3. Vectors of length 4 are formed from DCT coefficients separated in frequency and space. The spatial separation is maximized, that is, for 512×512 images, the samples that are grouped together are spaced by 256 pixels horizontally and/or vertically.
4. Correlating transforms are applied to each 4-tuple.
5. Entropy coding akin to that of JPEG is applied.

The system design is completed by determining which frequencies are to be grouped together and designing a transform for each group. This can be done based on training data. Even with, say, a Gaussian model for the source data, the transform parameters must be numerically optimized.[5]

We have simulated an abstraction of this system. If we were to use precisely the strategy outlined above, the importance of the DC coefficient would dictate allocating most of the redundancy to the group containing the DC coefficient. Thus for simplicity we assume that the quantized DC coefficient is communicated reliably through some other means. We separate the remaining coefficients into those that are placed in groups of four and those that are sent by one of the four channels only. The optimal allocation of redundancy between groups is difficult, so we allocate approximately the same redundancy to each group. For comparison we consider a baseline system that also communicates the DC coefficient reliably. The AC coefficients for each block are sent over one of the four channels. The rate is estimated by sample scalar entropies.

[5] In the case of pairing transforms as in [8], the optimal pairing and allocation of redundancy between the pairs can be found analytically [11].

Simulation results for the standard 512×512 'Lena' image are given in Fig. 1. As desired, the MD system gives a higher quality image when one of four packets is lost at the expense of worse rate-distortion performance when there are no packet losses.

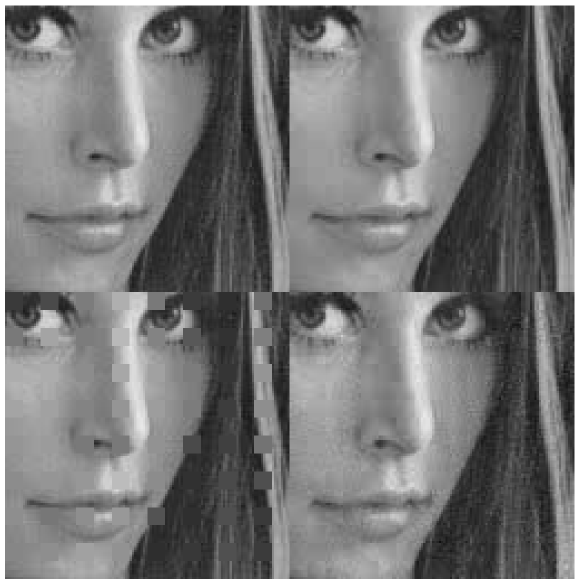

Fig. 1. Results for correlating transform method at 1 bpp. Top row: no packet losses; bottom row: one packet lost. Left column: baseline system; right column: MD system.

The results presented here are only preliminary because we have applied the techniques of [10] without much regard for the the structure and properties of images. The transform design is based on high-rate entropy estimates for uniformly quantized Gaussian random variables. Effects of coarse quantization, dead zone, divergence from Gaussianity, run length coding, and Huffman coding are neglected. Incorporating these will require a refinement of the theory and/or an expansive numerical optimization. Aside from transform optimization, this coder could be improved by using a perceptually tuned quantization matrix as suggested by the JPEG standard. Here we have used a constant quantization matrix for simplicity. With this type of tuning it should be possible to design a system which, say, performs precisely as well as the system in [7] when two or four of four packets arrive, but which performs better when one or three packets arrive.

A full image communication system requires packetization. We have not explicitly considered this, so we do not produce four streams with precisely the same number of bits. The expected number of bits for each stream is equal because of the form of (2). In contrast, with the transforms used in [8] one must multiplex the streams to produce packets of approximately the same size.

MDTC of Images Using Frames As an example, we consider a frame alternative to a $(10, 8)$ block code. For the 10×8 frame operator F we use a matrix corresponding to a length-10 real Discrete Fourier Transform of a length-8 sequence [13]. This can be constructed as $F = [F^{(1)}\ F^{(2)}]$, where

$$F_{ij}^{(1)} = \frac{1}{2} \cos \frac{\pi(i-1)(2j-1)}{10} \quad \text{and}$$

$$F_{ij}^{(2)} = \frac{1}{2} \sin \frac{\pi(i-1)(2j-1)}{10},\ 1 \le i \le 10,\ 1 \le j \le 4.$$

In order to profit from psychovisual tuning, we apply this technique to DCT coefficients and use quantization step sizes as in a typical JPEG coder. The coding proceeds as follows:

1. An 8-by-8 block DCT of the image is computed.
2. Vectors of length 8 are formed from DCT coefficients of like frequency, separated in space.
3. Each length 8 vector is expanded by left-multiplication with F.
4. Each length 10 vector is uniformly quantized with a step size depending on the frequency.

The baseline system against which we compare uses the same quantization step sizes, but quantizes the DCT coefficients directly and then applies a systematic $(10, 8)$ block code which can correct any two erasures. We assume that if there are more than two erasures, only the systematic part of the received data is used. (Maximum likelihood decoding would perform somewhat better, but is complex. In practice, one often discards the entire codeword if there are too many erasures.)

We have simulated the two systems with quantization step sizes conforming to a *quality* setting of 75 in the Independent JPEG Group's software. For the 'Lena' image, this corresponds to a rate of about 0.98 bpp plus 25% channel coding. In order to avoid issues related to the propagation of errors in variable length codes, we consider an abstraction in which sets of coefficients are lost. The alternative would require explicitly forming ten entropy coded packets. The reconstruction for the frame method follows a least-squares strategy. For the baseline system, when eight or more of the ten descriptions arrive, the block code insures that the image is received at full fidelity. The effect of having less than eight packets received is simulated using the following combinatorial result: With $e > n - k$ erasures distributed uniformly in a systematic (n, k) code, the probability that m data symbols are erased is

$$\binom{n}{e}^{-1} \binom{k}{m} \binom{n-k}{e-m} \quad \text{for } e - (n-k) \le m \le \min(e, k).$$

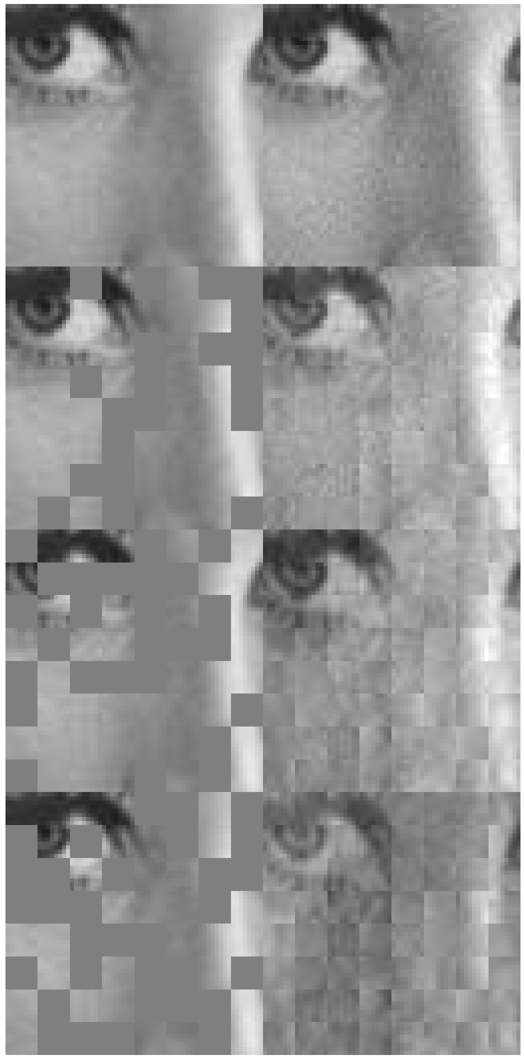

Fig. 2. Results for frame method at 1 bpp. Left column: baseline system; right column: MD system. From top to bottom, number of packets received is 8, 7, 6, and 5.

Results are shown in Fig. 2 for five through eight received packets. As expected, the frame system has better performance when less than eight packets are received. The performance of the MD system degrades gracefully as the number of lost packets increases.

References

1. L. Ozarow, "On a source-coding problem with two channels and three receivers," *Bell Sys. Tech. J.*, vol. 59, pp. 1909–1921, December 1980.
2. A.A. El Gamal and T. M. Cover, "Achievable rates for multiple descriptions," *IEEE Trans. Inform. Th.*, vol. 28, pp. 851–857, November 1982.
3. J. Wolf, A. Wyner, and J. Ziv, "Souce coding for multiple descriptions," *Bell Sys. Tech. J.*, vol. 59, pp. 1417–1426, October 1980.
4. H. Witsenhausen and A. Wyner, "Souce coding for multiple descriptions II: A binary source," *Bell Sys. Tech. J.*, vol. 60, pp. 2281–2292, December 1981.
5. T. Berger and Z. Zhang, "Minimum breakdown degradation in binary source encoding," *IEEE Trans. Inform. Th.*, vol. 29, pp. 807–814, November 1983.
6. V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. Inform. Th.*, vol. 39, pp. 821–834, May 1993.
7. Y. Wang, M. T. Orchard, and A. R. Reibman, "Multiple description image coding for noisy channels by pairing transform coefficients," in *Proc. First IEEE SP Workshop on Multimedia Signal Processing*, (Princeton, NJ), pp. 419–424, June 1997.
8. M. Orchard, Y. Wang, V. Vaishampayan, and A. Reibman, "Redundancy rate-distortion analysis of multiple description coding using pairwise correlating transforms," in *Proc. IEEE Conf. on Image Proc.*, (Santa Barbara, CA), October 1997.
9. V. K Goyal, J. Kovačević, and M. Vetterli, "Multiple description transform coding: Robustness to erasures using tight frame expansions," in *Proc. IEEE Int. Symp. on Inform. Th.*, (Boston, MA), August 1998.
10. V. K Goyal and J. Kovačević, "Optimal multiple description transform coding of Gaussian vectors," in *Proc. Data Compr. Conf.*, (Snowbird, UT), pp. 388–397, March 1998.
11. V. K Goyal and J. Kovačević, "Multiple description source-matched channel coding of a Gaussian source," *IEEE Trans. Inform. Th.*, 1998. To be submitted.
12. S. Lin and D. Costello, *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs, NJ: Prentice-Hall, 1983.
13. V. K Goyal, M. Vetterli, and N. Thao, "Quantized overcomplete expansions in $\mathcal{R}^n$: Analysis, synthesis and algorithms," *IEEE Trans. Inform. Th.*, vol. 44, pp. 16–31, January 1998.
14. S. Rangan and V. K Goyal, "Recursive Consistent Estimation with Bounded Noise," *IEEE Trans. Inform. Th.*, July 1998. Submitted.
15. R. Arean, J. Kovačević, and V. K Goyal, "Multiple description source channel coding of audio," *IEEE Trans. on Speech and Audio Proc.*, July 1998. Submitted.
16. V. K Goyal, J. Kovačević, R. Arean, and M. Vetterli, "Multiple description transform coding of images," in *Proc. IEEE Conf. on Image Proc.*, (Chicago, IL), October 1998.
17. D. Sinha. Technical Report, Bell Labs 1997.