A Corruption Model for Motion Compensated Video Subject to Bit Errors

Gustavo de los Reyes
AT&T Labs
gdelosreyes@att.com

Amy R. Reibman
AT&T Labs-Research
amy@research.att.com

Shih-Fu Chang
Columbia University
sfchang@ctr.columbia.edu

Abstract: We present a model of how corruption propagates in a block-based-encoded video decoder when the bitstream is subjected to bit errors. Our Markov model takes into account both corruption introduced via bit errors and corruption propagated via motion compensation. The model incorporates the impact of spatial and temporal source resilience on limiting the propagation of corruption within a frame and across time. For a given video sequence, the model requires source-specific inputs characterizing motion and spatial energy. It also requires encoder-specific inputs characterizing resilience—slice length and the proportion of I-blocks per frame. One final parameter needed by the model is the average number of bits in a compressed block. We apply the model to the specific example of determining the optimal allocation of bit rate among spatial resilience, temporal resilience, and source rate in a transcoder that injects resilience into the compressed bitstream.

1. Introduction

Designing video coding systems for noisy channels is a difficult problem that has been receiving much attention due to the increase in the use of wireless devices. These efforts have resulted in recent video coding standards for low-bitrate communications that include methods for improving spatial and temporal source resilience in addition to the usual channel coding methods [2]. Spatial and temporal localization techniques limit the propagation of errors within a frame or to subsequent frames. Despite the number of algorithms that have been proposed for improving the resilience of video over noisy channel [4, 6], few analytical models have been proposed to support these and other algorithms. In this paper, we derive a Markov model that describes how corruption, initially caused by bit errors during transport, propagates in videos that are coded using motion compensation. Our model takes into account the effects of spatial and temporal resilience as a means for limiting the propagation of the corruption. Budagavi [1] presented a model for error propagation using a restrictive motion model where errors affect only future blocks in the same spatial location as the original error. In this paper, we allow unrestricted block-based motion compensation as defined in H.263 [2] and similar block-based encoding standards. Hence the model is applicable to a wide range of compressed video sequences.

In addition, little work has been done to combine the various methods of resilience and provide guidance as to how best to combine the methods given a fixed bitrate budget. In [3], we described a transcoder that injects combined spatial and temporal source resilience into the compressed bitstream at the wireless base station or mobile switch. In this paper, we use our Markov model to derive operational rate distortion functions for spatial and temporal source resilience, and use these to determine the optimal bitrate allocation among source rate, spatial resilience, and temporal resilience. We use software simulation to verify the results obtained from our model.

2. Resilience

In our model, we focus on techniques for spatial and temporal localization. These methods prevent signal errors caused by a bit error or packet loss from propagating within a video frame or to subsequent frames. Spatial error propagation can occur when the decoder loses synchronization while decoding the variable length codes. In this paper, we limit spatial error propagation by inserting additional synchronization headers to reduce the number of blocks in a slice. With a shorter slice, the decoder re-synchronizes more quickly, resulting in less lost data. For temporal localization, we transmit more intra-frames (I-frames) or I-blocks. These frames or blocks are coded to be temporally independent of previous frames or blocks and are used as references for subsequent temporal prediction. More frequent I-blocks reduce the duration of error propagation.

The temporal and spatial localization techniques improve resilience by increasing the overall bit-rate. Therefore, we also use rate-reduction techniques to recover some bit-rate to allocate for improving resilience. Rate reduction techniques fall into two general classes: requantizing coefficients and discarding coefficients. In this paper, we take the simple approach of discarding coefficients to reduce the rate in a given frame by using a zonal mask, where the size of the mask is selected to provide the required rate reduction. It is the contention for bit rate between resilience and source rate that leads us to use rate distortion theory as a guide for selecting optimal bit rates as discussed below.

3. Overview of Corruption Propagation Model

In this section we describe the motivation for using a Markov model, and we describe the basic components of the model. Blocks within a frame can be corrupted for two main reasons: bit errors occur during transport, or corrupted information from the previous frame is used during motion compensation. Therefore, the number of corrupted blocks in the current frame only depends on the input errors for the current frame, the motion vectors in the current frame, and the number of corrupted blocks in the previous frame. This fits naturally into the structure of a Markov model. The overall corruption propagation model calculates the number of corrupted blocks in each video frame which is the sum of the number of blocks corrupted due to bit errors and the number of blocks corrupted due to motion compensation minus the intersection between the two types of corrupted blocks. The number of corrupted blocks, X_n, within frame, n, is then:

$$X_n = X_{n,mc} + X_{n,loss} - \left(X_{n,mc} \cap X_{n,loss}\right) \qquad \text{(Eq. 1)}$$

where *mc* indicates corruption due to motion compensation, and *loss* indicates corruption due to bit errors.

For blocks that are lost during transport due to bit errors, $X_{n,loss}$, we derive a statistical model for the expected number of blocks lost as a function of the spatial resilience (slice length) and the bit error rate (BER) in the channel. For blocks that are corrupted during motion compensation, $X_{n,mc}$, we derive a Markov model that includes the probability that a block is corrupted by referencing a block that has previously been corrupted. Within each frame, the probability that a block is corrupted by motion compensation is treated as a Bernoulli process with probability of success, p. This probability depends heavily on the motion in the video and the corruption in the previous frame. Since the corruption in the previous frame includes blocks that were lost to bit errors, the Markov model includes the effects of the spatial resilience model. We will show that this Markov model also includes the effects of temporal resilience.

To fold the spatial model into the Markov model, we simplify Equation 1 and use the expected value of $X_{n,loss}$,. Equation 1 then becomes.

$$X_n = X_{n,mc} + E\left[X_{n,loss}\right] - \left(X_{n,mc} \cap E\left[X_{n,loss}\right]\right)$$

We estimate the intersection as $X_{n,mc} \cap E\left[X_{n,loss}\right] = rE\left[X_{n,loss}\right]$ (Eq. 2)

where $r = \dfrac{X_n}{F}$ and F is the total number of blocks in a frame.

In the next sections we describes the spatial model and the Markov model in more detail.

4. Spatial Resilience Model

In this section we derive a statistical model of the video quality as a function of the spatial resilience (as measured by slice length) and the channel conditions (as measured by mean bit error rate

(BER)). This model takes into account the corruption propagation within a frame caused by bit errors during transport. This spatial resilience model is one part of the overall error propagation model.

4.1 Derivation of Spatial Resilience Model

The important parameters in this model are slice length, BER, and the number of bits per block. The slice length is measured in macroblocks as described in [2]. In this paper, we use the more generic term blocks rather than the specific term macroblocks. However, during implementation, we use the specific definition of macroblocks in [2]. For this analysis, we only use mean BER as the indicator of wireless channel conditions and assume bit errors are independent in time. We assume that once the decoder detects an error, it discards the block containing the error and all subsequent blocks until the next slice header. We further assume that every bit error results in an error that is detected by the decoder. This overestimates the number of lost blocks but does not characterize the distortion caused by errors that cannot be detected from video syntax.

For the spatial resilience model, we use a probabilistic model for the expected block losses. The probability that block *m* and all subsequent blocks in the slice are lost is equal to the probability that no bit errors occur in blocks 1 to (*m*-1) and at least one bit error occurs in block *m*. Therefore, if we assume a constant block length in bits, the expected number of blocks lost per slice is:

$$E\left[\frac{X_{n,loss}}{slice}\right] = \sum_{i=0}^{length}\left\{(length - i)\left[(1-P_e)^{ib+c} - (1-P_e)^{(i+1)b+c}\right]\right\}$$

where $X_{n,loss}$ are the blocks lost, i is the block position,

$length$ is the length of the slice in blocks, (Eq. 3)

P_e is the probability of error (or BER), c is the header length, and

b is the constant length of the block size in bits.

4.2 Effect of Spatial Resilience

The result of the spatial resilience model is shown in Figure 1, which shows how adding more slice headers (reducing the slice length) leads to a lower Mean Square Error (MSE) per video frame, especially at higher BER. This figure also shows strong non-linearities across both slice length and BER. One of our goals is to take advantage of these non-linearities to select an operating point for spatial resilience that produces the best video quality in the presence of bit errors.

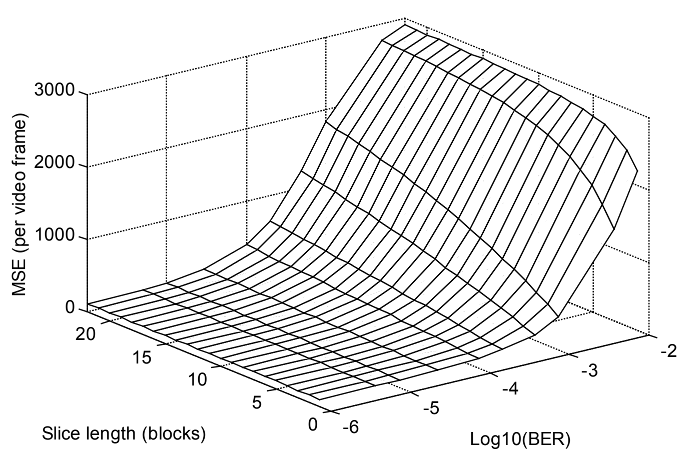

Figure 1. Effect of spatial resilience and wireless channel conditions on block loss.

5. Overall Corruption Propagation Model

In this section we derive the Markov model that describes the propagation of corruption due to motion compensation. Also, we include the results of the spatial model in the motion compensation model and derive an overall corruption-propagation model.

5.1 Corruption due to Motion Compensation

We begin this derivation by defining a transition matrix that governs the propagation of corruption in frame *n+1* during motion compensation given a number of corrupted blocks in frame *n*. As discussed previously, our model treats the probability of corruption due to motion compensation as a Bernoulli process with probability of success, *p*. The motion compensation transition matrix is then defined by a binomial distribution:

$$P(i, j_{mc}) = P\{X_{n+1,mc} = j_{mc} \mid X_n = i\} = \binom{k}{j_{mc}} p^{j_{mc}} q^{k-j_{mc}}$$

where X_n is the number of corrupted blocks in frame n, (Eq. 4)

p is the probability a block is corrupted via motion compensation, $q = 1 - p$,

k is the number of blocks in a frame, and $i, j_{mc} = 0,1,\ldots,k$.

Note that this equation describes how motion compensation propagates corruption from the previous (reference) frame. This equation does not specify how the corruption occurred in the previous frame. That will be described in the next sections.

5.2 Corruption due to Motion Compensation and Bit Errors

Recall that Equation 2 describes how we model corruption due jointly to bit errors, $X_{n,loss}$, and motion compensation, $X_{n,mc}$. By substituting Equation 2 into Equation 4 and solving for X_{n+1}, we derive the transition matrix that governs the overall propagation of errors from frame to frame.

$$P(i, j) = P\{X_{n+1} = j \mid X_n = i\} = \binom{k}{j} p^j q^{k-j}$$

where X_n is the number of corrupted blocks in frame n, (Eq. 5)

p is the probability a block is corrupted via motion compensation, $q = 1 - p$,

k is the number of blocks in a frame, and $i, j = 0,1,\ldots,k$.

Recall that *P(i, j)* is indeed a function of *i* because *p* is a function of the corruption in the reference frame, *n*. We will derive an equation for *p* shortly. Figure 2 shows schematically how the spatial model and the motion-compensated Markov model are used together to describe the overall corruption propagation. The large squares indicate video frames, and the smaller squares indicate lost blocks within the frames. The letters n, M, and L are used to indicate net corrupted blocks, corruption due to motion compensation, and losses due to channel bit errors, respectively. In the initial video frame, losses are introduced due to bit errors. These losses are propagated via motion compensation as described in Equation 4. According to Equation 2, new losses are introduced in the next frame due to bit errors, and intersecting losses are removed. This results in the net corruption due to both bit-error losses and motion compensation. However, by using the transition matrix of Equation 5, the net result from Frame 0 to Frame 1 can be calculated directly. Then the process repeats for the following frame.

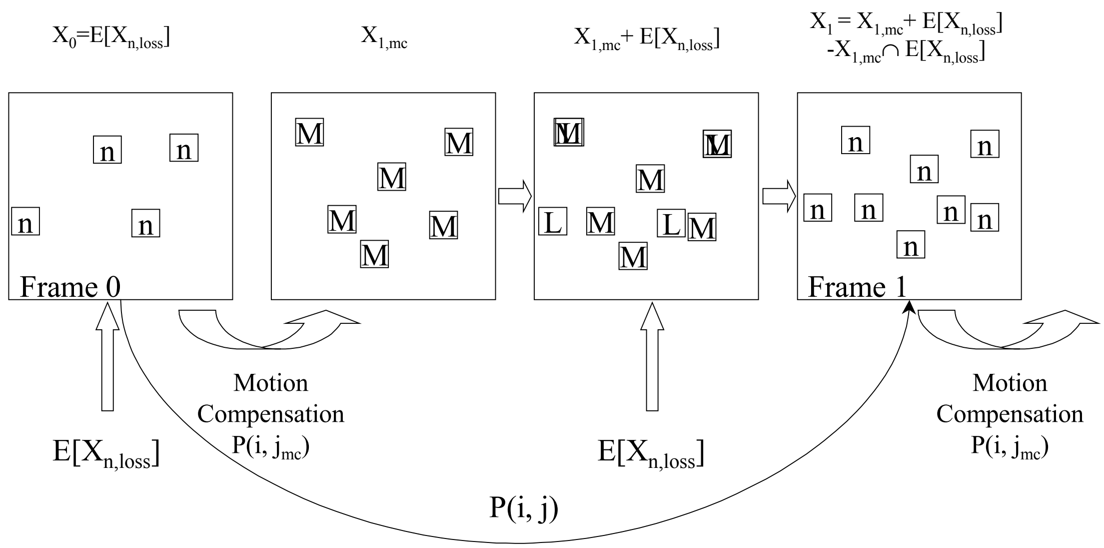

Figure 2. Schematic propagation of corruption via bit errors and motion compensation.

Now that we have the transition matrix, we can determine the expected number of corrupted blocks due to motion compensation and bit errors in each video frame, *n*, from the *n*-step transition probabilities:

$$P^n(i,j) = P\{X_n = j \mid X_0 = i\} \qquad \text{(Eq. 6)}$$

The expected number of corrupted blocks due to motion compensation in each frame in the video is:

$$\begin{aligned} E[X_n \mid X_0 = i] &= \sum_{j=0,1,\ldots,k} jP\{X_n = j \mid X_0 = i\} \\ &= \sum_{j=0,1,\ldots,k} jP^n(i,j) \qquad \text{(Eq. 7)} \end{aligned}$$

where *i* is the number of corrupted blocks in the first frame, 0.

From Equation 7, one can see how the model describes the error propagation frame by frame. The corruption in each frame, *n*, is just a function of the *n*-step transition probabilities for that frame.

Now we determine the probability, *p*, that a block is corrupted through motion compensation so that we can compute the transition matrix, *P(i, j)* of Equation 5. A block in the current frame will be corrupted through motion compensation if it references corrupted information (pixels) in the previous frame. Since we are working with blocks as the unit of measure rather than pixels, this is equivalent to saying that a block in the current frame will be corrupted if it references a corrupted block, and it references corrupted pixels within that corrupted block.

To compute these probabilities, we group the blocks into four categories characterized by their motion vectors (MVs). The first category is for blocks that have MVs that describe no motion. The second category is for blocks that have MVs that describe horizontal or vertical motion. The third category is for blocks that have MVs that describe combined horizontal and vertical motion. The last category is for I-blocks that have no associated MVs. For each category of block, we compute the probability, p_i, that a block of that type is corrupted via motion compensation. The overall probability of corruption for a block given a certain amount of corruption in the previous frame is then:

$$p = \sum_{i=1,2,3,4} p_i \bullet P(block_i) \qquad \text{(Eq. 8)}$$

where i is one of the four possible categories—i = 1 is a block of no motion, i = 2 is a block of horizontal or vertical motion, i = 3 is a block of combined horizontal and vertical motion, and i = 4 is an I-block.

For I-blocks, the probability of an I-block being corrupted by motion compensation is clearly zero. On the other hand, for a block with no motion, the probability of corruption is one if it references a corrupted block. For these blocks, the probability of corruption is simply the probability that a block in the referenced frame is corrupted, which we estimate by the proportion of blocks in the previous frame that are corrupted, r.

Calculating the probability of corruption for the other two categories of blocks is somewhat more involved. These blocks can reference either two or four blocks in the previous frame, and various combinations of corrupted and not corrupted blocks can exist for these referenced blocks. Furthermore, to be corrupted, a block must still reference the corrupted information in a corrupted block. For these two categories of blocks, the probability that a block *references* one or more corrupt blocks, p_{ref}, can be described by a binomial distribution:

$$p_{ref} = \sum_{j=1}^{l} \binom{l}{j} p_{corr}{}^{j} (1 - p_{corr})^{l-j} = 1 - (1 - p_{corr})^{l}$$

where p_{corr} is the probability a block in the reference frame is corrupted, and (Eq. 9)
l is the number of blocks that a current block references: $l = 1$ for no motion,
$l = 2$ for vertical or horizontal motion, and $l = 4$ for combined vertical and horizontal motion.

As before, we estimate the probability that a block in the referenced frame is corrupted by the proportion of blocks in the referenced frame that are corrupted, $p_{corr} = r$.

Thus far, we have estimated the probability that a block in the current frame references a corrupted block. Now we estimate the probability that a block in the current frame references corrupted information given that it references a corrupted block. For each block, we assume a uniform distribution of corruption from minimum to complete corruption. Then, the final equation for the probability that a block is corrupted through motion compensation, p, is now:

$$p = a_1 r P(block_1) + a_2 \left[1 - (1-r)^2\right] P(block_2) + a_3 \left[1 - (1-r)^4\right] P(block_3)$$

and

$$\sum_{i=1,2,3,4} P(block_i) = 1 \qquad \text{(Eq. 10)}$$

where $block_i$ are the categories of blocks defined above, and
$a_1 (=1)$, a_2, and a_3 are the probability of referencing corrupt information in blocks of category $block_1$, $block_2$, and $block_3$, respectively.

5.3 Effect of Temporal Resilience

In this section we show how our model incorporates the effect of using additional I-blocks for temporal resilience. In Equation 10, since the probability of all four categories of blocks must sum to one, as the proportion of I-blocks, $P(block_4)$, increases, the probability of receiving the other blocks, $P(block_1) + P(block_2) + P(block_3)$, decreases. This decreases p, the probability that a block is corrupted via motion compensation. This behavior is shown in Figure 3, which shows the invariant (steady state) distribution of the Markov model with a BER = $5x10^{-4}$, and two different probability of I blocks, *P(block$_4$)*. On average, there are fewer corrupted blocks as *P(block$_4$)* increases.

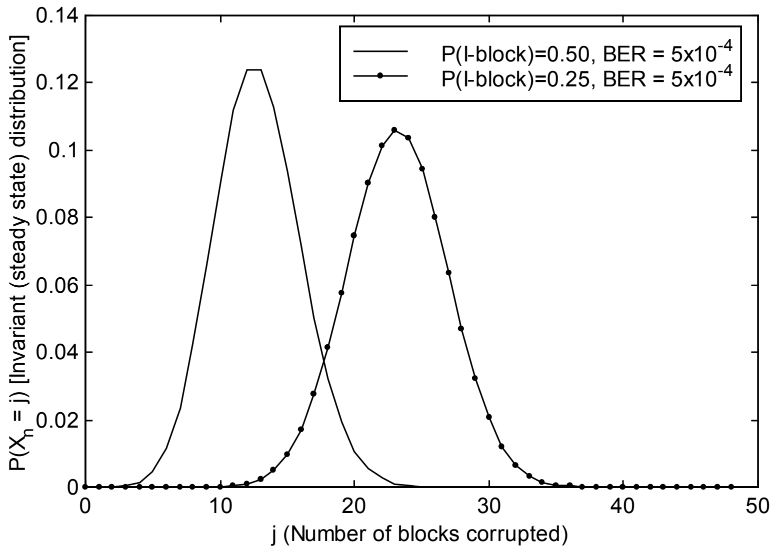

Figure 3. Invariant distribution of Markov model as a function of *P(block$_4$)*=probability of I blocks.

6. Optimal Allocation of Bit Rate

In the rest of the paper we show one way that we use the corruption-propagation models: to optimally allocate resilience. Given that we can allocate bit rate over three sources—source rate, spatial resilience, and temporal resilience—we would like to allocate the bit rate optimally to minimize distortion in the video. Applying rate distortion theory to our resilience problem, we want to find a vector of rates which minimizes an overall distortion measure subject to a rate constraint, *B*. Solving this constrained optimization problem is equivalent to solving the unconstrained optimization problem with an objective function, **H**:

$$\mathbf{H} = D(\mathbf{L}) + \lambda R(\mathbf{L}) \quad \text{(Eq. 11)}$$

where **L** is the vector of resources,
λ is the Lagrangian multiplier,
D is the distortion that results from applying the resources, and
R is the rate that results from applying the resources.

The objective function can be written as:

$$\mathbf{H} = \sum_{i=1}^{N} \left[D_i(L_i) + \lambda R_i(L_i) \right] \quad \text{(Eq. 12)}$$

Solving this set of simultaneous equations leads to:

$$\lambda = \frac{\partial D_1}{\partial R_1} = \cdots = \frac{\partial D_N}{\partial R_N} \quad \text{(Eq. 13)}$$
$$\text{subject to } R(\mathbf{L}) \le B$$

That is, given that there is at least one solution to the problem, the solutions exist at the points where the slopes of the distortion-rate functions $D_i(R_i)$ are equal, and the sum of the rates R_i are less than or equal to the available bit rate. In our case, the vector of resources includes the spatial resilience, the temporal resilience, and the source rate. We have an additional constraint in that the resources vary in discrete amounts. For example, slice lengths vary in integer values from one to 22. Therefore, the bit rates used by the resources also vary in discrete amounts. This discreteness affects the optimal solution. We consider the optimal solution to be the one where the total bit rate comes closest to the available bit rate of the channel. Also, we define the rate distortion functions as operational functions that are generated from the Markov models or from simulations.

7. Transcoder

In [3] we extended the use of transcoding beyond rate reduction [5], to include the addition of resilience information in the bitstream. The transcoder operation is shown in Figure 4. The transcoder decodes the incoming bitstream to the degree required to add resilience. Resilience is added by the transcoder, and, if necessary, the bit-rate is reduced. Then the resilient bitstream is re-quantized (for temporal resilience only) and variable length re-encoded.

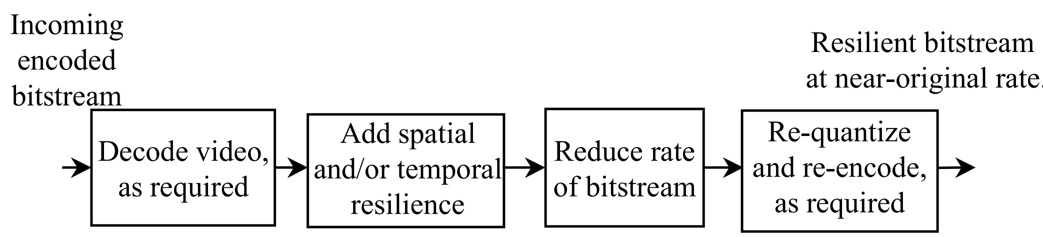

Figure 4. High-level transcoder operation.

The spatial localization technique of adding more slice headers requires only simple bitstream parsing and arithmetic operations. A variable-length decoder parses the bitstream and inserts additional slice start codes where necessary. In addition, motion vectors (MVs) are decoded, a new differential MV is calculated from reference MVs only within the current slice, and the result is variable length encoded. The transcoder requires more complicated processing to increase the temporal resilience by using more frequent I-blocks. The incoming bitstream is fully decoded, including motion compensation, to create the new I-blocks. The coefficients of the new I-blocks are then DCT-coded, re-quantized, scanned, and variable-length encoded. Because the transcoder can modify the resilience to match the prevailing channel conditions, a method to determine the optimal resilience is invaluable.

8. Results

The operational rate distortion function (ORDF) for source rate was generated from simulation, while the ORDF for spatial and temporal resilience were generated from our analytical models. In addition, the models estimate block losses and corruption that must be converted to distortion to produce the ORDF. This conversion is a function of the video content, so we estimated the distortion by simulating losses in the decoded video. For each of the ORDF in the figures below, we curve fitted the data and then differentiated the fit curves to get smooth slopes as a functions of bit rate. The figures below show the original ORDF with the superimposed curve fit on one plot, and the corresponding slope functions on the adjacent plot. Figures 5-7 show the operational rate distortion functions and their slopes at a BER of 5×10^{-4} for source rate, spatial resilience (slice length), and temporal resilience (I-block proportion), respectively. The y-axis for the slope of the spatial resilience ORDF (Figure 6) has been scaled to highlight the area of interest. For source rate, the x-axes are the ratio of the resulting bit rate when coefficients are dropped to the original bit rate with all coefficients. For spatial resilience and temporal resilience, the x-axes are the ratio of the resulting bit rate when resilience is added to the baseline resilience (minus 1).

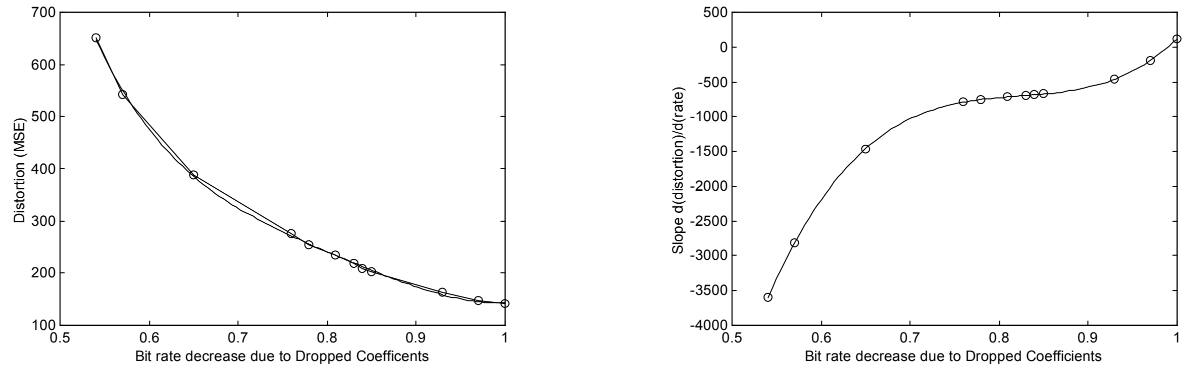

Figure 5. Operational rate distortion function and slope function for source rate (from simulation).

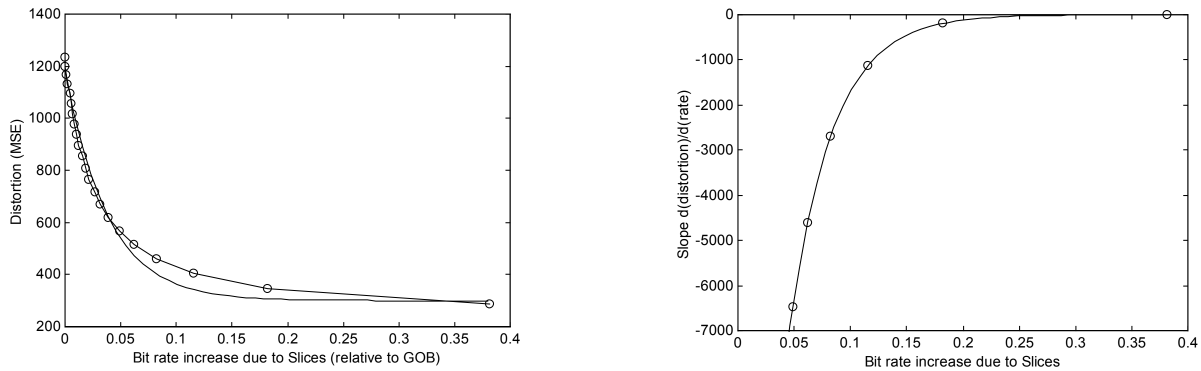

Figure 6. Operational rate distortion function and slope function for spatial resilience (analytical model).

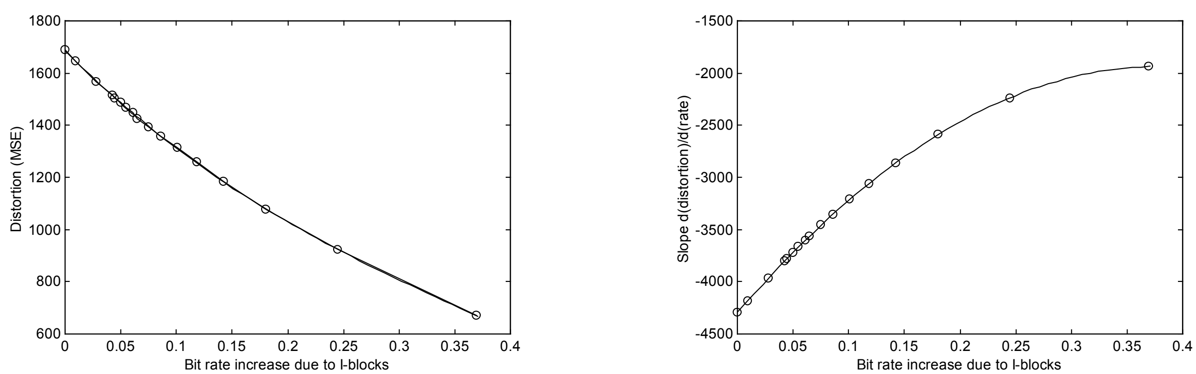

Figure 7. Operational rate distortion function and slope function for temporal resilience (Markov model).

The operating point is selected by trying to match slopes such that the total bit rate is less than or equal to the bit rate of the original video. However, the possible solutions are further constrained because the resources vary in discrete amounts. In addition, the effect of MSE due to source rate is perceptually different than the effect of MSE due to resilience. Decreasing the source rate tends to blur the video, while decreasing spatial and temporal resilience tends to increase the number of severe artifacts. For these severe artifacts, several blocks of the video may change color, or parts of the video become mismatched. Therefore, a certain amount of judgement must be applied when selecting the optimal solution.

In [3] we used a trial-and-error approach to select the operating point, while here we described an optimal method for selecting the operating point. The optimal operating point for a BER of $5x10^{-4}$ corresponds to a slice length of 4 blocks, an I-block proportion equal to each block being refreshed every 5 frames, and a source rate such that a maximum of 11 coefficients are retained in each block. This results in a Peak Signal to Noise Ratio (PSNR) of 18.0 dB. The results presented in [3] for the same BER resulted in a PSNR of 17.4 dB for a slice length of 5 blocks, an I-block proportion equal to each block being refreshed every 10 frames, and a source rate such that a maximum 15 coefficients are retained in each block. The result for the baseline video with nominal resilience is a PSNR of 13.5 dB. For nominal resilience we use a slice length of 22 blocks and a single I-frame followed by all P-frames.

We also computed the optimal resilience for a BER of 10^{-4}. The operating point with the optimal method is a slice length of 10 blocks, an I-block proportion equal to each block being refreshed every 7 frames, and a source rate such that a maximum 16 coefficients are retained in each block. This results in a Peak Signal to Noise Ratio (PSNR) of 21.4 dB. This is slightly less than the PSNR of 21.8 dB presented in [3] for the same BER. However, both methods lead to much better results than the baseline video with nominal resilience and a PSNR of 18.5 dB. Therefore, we can achieve nearly the same results with the optimal approach as compared to our previous trial-and-error approach. This is a great benefit since the optimal approach depends more on analytical models and less on the specific video sequence. This makes the optimal approach better suited to near-real-time calculations that can be used in a resilience transcoder.

9. Summary and Conclusions

We developed an analytical model for the propagation of corruption as a function of spatial and temporal resilience and BER. For a given video sequence, the model requires source-specific inputs characterizing motion and spatial energy. The spatial energy of the video is used to estimate the distortion (MSE) that results from block losses and corruption. For a system implementation, a likely place to estimate this distortion is at a scene change. The energy content per frame between scenes changes will be reasonably steady, and methods exist for determining scene changes in real time. The model also takes as input the encoder-specific parameters of slice length and I-block proportion per frame. We apply the model to the specific example of determining the optimal allocation of bit rate among spatial resilience, temporal resilience, and source rate in a transcoder that injects resilience into the compressed bitstream. The optimal approach leads to nearly the same results than previously reported when using a trial-and-error approach.

10. References

[1] Budagavi, M. and J. D. Gibson. "Error Propagation in Motion Compensated Video over Wireless Channels," IEEE International Conference on Image Processing, October 1997.

[2] Draft ITU-T Recommendation H.263, Video Coding for Low Bitrate Communication, Sept. 1997.

[3] De los Reyes, G., A. R. Reibman, J. C. Chuang, and S.-F. Chang, "Video Transcoding for Resilience in Wireless Channels," IEEE International Conference on Image Processing, October 1998.

[4] Farber. N., E. Steinback, and B. Girod, "Robust H.263 Compatible Video Transmission over Wireless Channels," International Picture Coding Symposium, Melbourne, Australia, March 1996.

[5] Keesman, G., R. Hellinghuizen, F. Hoeksema, G. Heidman, "Transcoding of MPEG Bitstreams," Signal Processing: Image Communications, 8 (1996), pp 481-500.

[6] Liao, J. Y. and J. D. Villasenor, "Adaptive Intra Update for Video Coding over Noisy Channels." IEEE International Conference on Image Processing, September 1996.