Joint Source-Channel Distortion Modeling for MPEG-4 Video

Muhammad Farooq Sabir, *Member, IEEE*, Robert W. Heath, Jr., *Senior Member, IEEE*, and Alan Conrad Bovik, *Fellow, IEEE*

***Abstract*—Multimedia communication has become one of the main applications in commercial wireless systems. Multimedia sources, mainly consisting of digital images and videos, have high bandwidth requirements. Since bandwidth is a valuable resource, it is important that its use should be optimized for image and video communication. Therefore, interest in developing new joint source-channel coding (JSCC) methods for image and video communication is increasing. Design of any JSCC scheme requires an estimate of the distortion at different source coding rates and under different channel conditions. The common approach to obtain this estimate is via simulations or operational rate-distortion curves. These approaches, however, are computationally intensive and, hence, not feasible for real-time coding and transmission applications. A more feasible approach to estimate distortion is to develop models that predict distortion at different source coding rates and under different channel conditions. Based on this idea, we present a distortion model for estimating the distortion due to quantization and channel errors in MPEG-4 compressed video streams at different source coding rates and channel bit error rates. This model takes into account important aspects of video compression such as transform coding, motion compensation, and variable length coding. Results show that our model estimates distortion within 1.5 dB of actual simulation values in terms of peak-signal-to-noise ratio.**

***Index Terms*—Distortion modeling, joint source-channel coding, MPEG-4, quality assessment, video communication.**

I. Introduction

A few years ago, the concept of video communication using cellular phones was considered highly impractical. Video recording and transmission features though are now very common in most consumer cellular phones. Digital videos are coded at high data rates to achieve good quality and, hence, have high bandwidth requirements. In addition, for real-time video communication, low latency is another important requirement. Source coding is commonly used to reduce the data rate of digital videos. Since most of the current video coding standards use lossy source coding methods, distortion is introduced in the coded source. Furthermore, the coded data stream becomes highly sensitive to transmission bit errors due to the presence of entropy coding, differential coding, and motion compensation. These bit errors can introduce large distortions in the transmitted videos. This implies the need for error protection, commonly known as channel coding. Though channel coding protects the coded bitstream from channel errors, it increases the number of bits to be transmitted and, hence, the bandwidth requirement. Thus, source and channel coding present competing objectives of reducing bandwidth while minimizing distortion in the video stream.

Shannon's classical separation theorem [1] implies that source and channel coding can be done separately and sequentially while maintaining optimality. This, however, is only true for asymptotically long block lengths, which is not achievable in practical coding and transmission systems. For this reason, many researchers have argued that to optimize the use of available bandwidth and data rate, while still maintaining very good quality, it is prudent to use joint source-channel coding (JSCC) schemes to transmit digital images and videos [2]–[6]. Significant quality gains can be achieved by jointly optimizing the allocation of source and channel coding bits without any increase in bandwidth. JSCC has gained interest in the research community resulting in a large amount of work being published over the past few years.

A fundamental component of almost all JSCC methods is an estimate of distortion that occurs due to quantization and transmission errors at different source coding rates and channel bit error rates. This distortion estimate can either be computed using simulations and operational rate-distortion curves, or it can be obtained using statistical distortion models. While the simulation and operational rate-distortion based approaches are easier to formulate, estimate distortion with high accuracy, and are the traditional methods to obtain the distortion estimate, they usually are computationally intensive and, hence, cannot be used for real-time applications. Model based approaches on the other hand, are difficult to formulate. However, they are computationally simpler and provide reasonably accurate estimates of distortion. For this reason, model based distortion estimation schemes are better suited for real-time video communication applications. Most of the joint source-channel coding schemes in the literature have focused on simulation and operational rate-distortion based design strategies. Little work has been done in the field of developing distortion models for practical image and video coding standards.

In this paper, we formulate and present a distortion model that predicts the amount of distortion due to quantization and channel errors in Moving Picture Experts Group 4 (MPEG-4)

Manuscript received May 13, 2007; revised August 05, 2008. First published November 25, 2008; current version published December 12, 2008. This work was supported by the Texas Advanced Technology Program Grant 003658-0380-2003. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sohail Dianat.

M. F. Sabir is with the K-WILL Corporation, San Jose, CA 95134 USA (e-mail: mfsabir@ieee.org).

R. W. Heath, Jr., and A. C. Bovik are with the Wireless Networking and Communications Group, Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712-1084 USA (e-mail: rheath@ece.utexas.edu; bovik@ece.utexas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2008.2005819

coded video sequences. To motivate the need for such a distortion model, we discuss a few key JSCC techniques along with their distortion estimation methods for image and video communication in Section I-A. We then discuss the general limitations of these methods in Section I-B, and outline our contributions in Section I-C.

A. Previous Work

In [7], Bystorm and Modestino presented a JSCC method to optimally allocate source and channel coding bits with a fixed constraint on transmission bandwidth for video transmission over an additive white Gaussian noise (AWGN) channel. They used normalized mean-square error as the distortion metric and assumed that the distortion is additive and independent on a frame-by-frame basis. For a given video sequence, universal operational rate-distortion curves were constructed. These curves were used to find the distortion values for the optimization problem. This method required a significant amount of computations for construction of universal curves. Furthermore, these curves were required to be constructed for each video sequence, making this method infeasible for real-time video communications. In [8], Cheung and Zakhor presented a bit allocation method for allocating source and channel bits between the subbands of a scalable video, such that the overall distortion was minimized given the channel conditions and a total bit budget. This method used the mean squared error (MSE) as the distortion metric. MSE values for different source and channel coding configurations were obtained using empirically computed functions.

Kondi *et al.* proposed a JSCC scheme based on universal rate-distortion curves for motion compensated discrete cosine transform (DCT) based signal-to-noise-ratio (SNR) scalable video in [9]. In this method, universal rate-distortion curves were constructed using simulations at different source coding rates. These curves were used to construct operational rate distortion curves, which were then used to determine the value of distortion at different source coding and channel bit error rates. Using these operational rate-distortion curves, the optimal bit allocation strategy and the minimum MSE distortion were obtained. In [10], Zhai *et al.* considered a hybrid JSCC scheme consisting of error resilient source coding, channel coding and error concealment for real-time packetized video transmission. Expected values of the distortion due to source coding and channel errors for each packet were computed recursively, and were then used in the optimization framework to minimize the overall distortion.

In a rate-distortion optimized scheme, Gallant and Kossentini [11] presented an optimal mode-selection method for robust video transmission over the Internet. A statistical model for estimating the concealment distortion for each block by weighting the distortions in the surrounding blocks of the previous frames was developed. This method selected the optimal amount of temporal error resilience to be inserted in the bitstream based upon the decoder concealment method, the channel packet loss rate, and the FEC code rate. Bystorm and Stockhammer [12] constructed rate-distortion surfaces for different video frames using simulations, and used polynomial models to approximate these surfaces. These rate-distortion surfaces were then used with channel coding models to allocate the source and the channel coding rates between the different frames of a video sequence, with the goal of minimizing the overall distortion under a constraint on the total rate.

In another model based approach, He *et al.* [13] presented a method for adaptive mode selection and rate control for video transmission over wireless links. The authors developed a model for estimating distortion due to bit errors in motion compensated video, and then used this model with the source rate-distortion model for adaptive intra mode selection and joint source-channel rate control under time varying channel conditions. Their distortion model estimated distortion due to channel errors recursively by using distortion due to channel errors in the previous frames. In [14], Cheng *et al.* presented an unequal loss protection method for transmission of fine-granular-scalability (FGS) based MPEG-4 coded video sequences. Source distortion was estimated using piecewise linear interpolation of actual rate-distortion points obtained during encoding, and channel distortion was modeled as the reduction in the distortion when more fragments were received. The authors used these models to carry out rate-distortion optimized bit allocation for source and channel coding.

Models for distortion caused by source coding and channel errors for video coding and transmission were presented in [15] by Stuhlmuller *et al.* The distortion-rate characteristics of the source coder, and the distortion due to residual channel errors were modeled empirically using test sequences. These models also captured the effect of interframe error propagation and error concealment. Zhang *et al.* [16] presented a distortion model for estimating the distortion due to quantization, error propagation and error concealment for video coding and transmission over packet switched networks. Their algorithm estimated the pixel distortion in both the intra and the inter coded macroblocks in a recursive manner, by using distortion in previous frames. This model was then used in a rate-distortion framework for automatic mode switching between inter and intra coding of macroblocks.

In [17], Bergeron and Lamy-Bergot proposed a semi-analytical model for estimating distortion due to source and channel errors in H.264/AVC coded bitstreams. This model estimates distortion in Intra and Predicted frame as well as group of pictures (GOPs) and data partitioned GOPs. In this method, effects of error propagation to future frames were considered by estimating distortion in the current frame conditioned upon the case when there are no errors in the previous frames. This model was then used in a joint source-channel coding setup to determine different protection rates for providing unequal error protection to the coded bitstream.

In [18], Dai *et al.* proposed a statistical distortion model for MPEG-4 fine granular scalability (FGS) coded video sequences. They used a mixture Laplacian model to model the tail as well as the sharp peak in the histogram of the DCT residue in FGS mode of MPEG-4 encoding. Based on the mixture Laplacian model, a closed form distortion expression was derived for MPEG-4 FGS coded video sequences. Their results showed that this model predicts distortion with high accuracy. Though this model accurately predicts distortion due to source coding, while also taking

into account bit-plane coding, it does not into account the effects of channel coding and bit errors. This work was extended in [19], and another R-D model was proposed for scalable video coders. This model also predicted distortion with high accuracy.

Distortion modeling is also an important component of JSCC methods for image communication. In [20], Ruf and Modestino proposed a distortion model for discrete wavelet transform (DWT) compressed images. This model was then used for efficient joint allocation of source and channel coding bits. In our previous work in [21], we presented a joint source-channel distortion model for JPEG compressed images. This model estimated the amount of distortion due to quantization and channel errors in different sub-bands of DCT coded JPEG images.

Distortion modeling is also important for other joint design techniques as well, such as joint optimization of source coding parameters and transmission power/energy, unequal power allocation, and multiresolution source-channel coding. In [22], Appadwedula *et al.* derived an expression for the expected value of distortion for a general class of images. Their model estimated distortion using a sum of exponentials. This model was then used to jointly optimize the source coder, the channel coder, and the power consumption. In [23], a power management scheme for wireless video was proposed. Distortion in H.263 video was modeled by taking into account the effects of error propagation and error concealment. Using this distortion model, the bit error rates for different video frames were optimized such that the consumed power was minimized with a constraint on maximum distortion. Some other relevant joint design techniques along with their distortion estimation methods are discussed in [24]–[30]. In Section I-B, we discuss a few limitations of the different existing distortion estimation methods.

B. Limitations of Existing Methods

Though all the above discussed JSCC schemes provide significant quality improvements and coding gains, and their distortion computation methods estimate the expected value of distortion with good accuracy, they have the following limitations.

- Most of the schemes discussed above use operational rate distortion curves to determine distortion values at different source coding rates and channel bit error rates [7], [9]. Construction of these curves requires many simulations for each video sequence or different classes of video sequences. Though these curves predict distortion with high accuracy, they need to be constructed for every video sequence (or classes of sequences) and, hence, are not feasible for real-time video communication systems due to their high computational complexity.
- Some of the empirical models used in these JSCC methods have their parameters specific for a given sequence [15]. For each new sequence, these parameters must be obtained by fitting the model to a subset of measured rate distortion data. This makes the distortion measurement process computationally intensive and, hence, infeasible for real-time video applications.
- Some of the distortion estimation methods [10], [13], [16] measure distortion in the current frame recursively using distortion in previous frames. Though these methods estimate distortion per frame with high accuracy, they do not predict the exact effect of bit errors in a packet on subsequent frames. This is important because an error in a frame is propagated to subsequent frames and, hence, an error in an earlier frame will most likely result in increased overall distortion in the video sequence as compared to an error in a later frame. To model the exact effect of bit errors at a particular point in the bitstream, the distortion model should predict the amount of total distortion introduced in the video sequence, taking into account the effect of distortion propagation to future frames due to error at that particular point in the bitstream. JSCC schemes can be designed more efficiently if the exact effect of errors at any particular location in the data stream on the future frames could be predicted at the time of coding.
- Video sequences coded by any of the current video coding standards are highly sensitive to bit errors. Due to the presence of motion estimation/compensation, predictive and differential coding, and entropy coding, even a single bit error has the potential to introduce large amounts of distortion in current and subsequent video frames. Because of this, video coding standards employ error resilience tools and methods such as data partitioning [31], [32]. The main goal of data partitioning is to separate more important data in a packet (e.g., DC coefficients for I frames and motion vectors for P frames) from less important data (e.g., AC coefficients for I frames and texture data for P frames), so that the more important partition can still be decoded in case there are errors in the less important partition, hence providing a base level of quality. Most of the above mentioned schemes do not take into account data partitioning while modeling distortion, and code motion vectors and residual error (texture) together. Data partitioning is important not only for error resilience, but also because different partitions can be transmitted with unequal error protection since the data in different partitions have unequal importance.

C. Contributions

In this paper, we present a statistical distortion model for predicting the amount of distortion introduced in MPEG-4 coded video streams due to quantization and channel errors when they are transmitted over noisy/fading channels. MPEG-4 error resilient tools such as data partitioning and packetization are used to encode the video into different partitions/layers. The mean squared error is modeled as a function of source coding rate and channel bit error probability for different partitions in I and P frames separately, and an expression for the total distortion is derived. This model takes into account important components of video coding such as motion estimation and compensation, predictive coding, DCT coding, and variable length coding (VLC). The effects of error propagation to subsequent frames due to motion compensation are taken into account, in order to estimate distortion due to errors in I and P frames. Model parameters are computed using a "training" database of video sequences. These parameters are then used in conjunction with the statistics of different "test" sequences to predict the distortion due to quantization and channel errors in the "test" sequences. Results show that our model predicts distortion within 1.5 dB of actual simulation values in terms of peak-signal-to-noise-ratio (PSNR)

over a wide range of source coding rates and channel bit error rates.

In the case of bit errors in any video packet, our distortion model predicts the amount of distortion introduced in the current frame as well as all subsequent frames in the video sequence until the next I frame. Hence, our distortion model approximates the effect of errors in each video packet on the overall reconstructed video quality. Though the distortion expressions are derived explicitly for MPEG-4 video coded streams, this model can be extended to other similar video coding schemes that use transform coding, motion compensation and entropy coding.

Our proposed distortion model is different in many ways from the existing distortion estimation methods used in JSCC. One of the main differences is that our distortion model does not require constructing computationally intensive operational rate-distortion curves. Also, our model does not compute distortion in a recursive manner. Instead, it takes into account the effects of distortion propagation to future frames due to errors in the current frame. The use of different error resilience tools especially data partitioning also makes our model different from other distortion estimation methods that do not employ these tools. Furthermore, since the parameters of our distortion model are derived from a training database of videos, this model can predict the distortion using the source coding rate, the channel bit error rate (BER), and the statistics of the video data in a packet during the coding process in real-time. Hence, it is well suited for real-time video communication applications. Based on these properties of our model, we believe that efficient JSCC schemes can be designed using our distortion model for transmission of MPEG-4 coded video streams over noisy/fading channels.

This paper is organized as follows. We first outline our system model consisting of MPEG-4 video coder and the channel in Section II. In Section III, we describe our assumptions and notation, and derive the MSE expressions representing distortion due to quantization and channel errors for I and P frames. Section IV present our simulation details and results, along with some discussion on the results. We conclude this paper in Section V.

II. System Model

The presence of differential coding, entropy coding, and motion compensation makes the compressed video bitstream highly sensitive to channel errors. A single bit error can not only corrupt many pixels in the current frame but also has the potential to cause severe distortion in subsequent frames. For this reason, different error resilience tools have been introduced in many video coding standards. In this paper, we use the MPEG-4 part 2 (visual) video coding standard with two very important error resilience features: packetization and data partitioning. We explain our source coding model along with these error resilience tools in the following section, followed by a description of our channel model.

A. Source Coding Model

We use MPEG-4 part 2 (Visual) for source coding. We only consider the simple profile since it is the most commonly used profile in the MPEG-4 Visual standard. In MPEG-4 Visual, a video sequence is treated as a collection of one or more "video objects." A video object (VO) is defined as an "area of the video scene that may occupy an arbitrary-shaped region and may exist for an arbitrary length of time" [33]. A "video object plane" (VOP) is defined as an instance of a VO at a particular point in time. In the simple profile of MPEG-4 Visual, only rectangular I-VOP and rectangular P-VOP are considered. We will use "frame" and "VOP" interchangeably in the following sections because they both mean the same thing in the simple profile of MPEG-4 Visual.

A frame of a video sequence coded in Intra mode, without prediction from any other frame (VOP in the case of MPEG-4), is called a rectangular I-VOP. For an I-VOP, the first step in encoding is the transformation of 8×8 blocks of luma and chroma samples using the discrete cosine transform (DCT). After transform coding, the coded coefficients are quantized. These quantized coefficients are then reordered in a zig-zag scan. If the data partitioning mode of operation is used then the DC coefficients are encoded in a separate partition from the AC coefficients. After reordering (and data partitioning, if present), the quantized coefficients are last-run-level [33] coded followed by variable length coding. At the decoder, to reconstruct the I-VOP, variable length and run length decoding are first carried out. The coefficients are then rearranged in their original order, followed by re-scaling. After that, the inverse discrete cosine transform (IDCT) is applied to the 8×8 blocks of coefficients as the final step of decoding. A block diagram of this encoding/decoding process for an I-VOP is shown in Fig. 1(a).

A P-VOP is a rectangular frame that is encoded using inter prediction from a previously encoded I-VOP or P-VOP. In a P-VOP, block based motion compensation is carried out on 16×16 macroblocks. Motion estimation is first performed, and the resulting prediction is subtracted from the current macroblock to construct a macroblock of residual data (also known as texture, residual error, and the motion compensated prediction). After motion compensation, motion vectors are differentially coded followed by variable length coding. The 8×8 blocks of samples in the residual macroblock are first DCT coded, followed by quantization, reordering, run-length encoding and variable length encoding. At the decoder, the texture data is variable length decoded, followed by run length decoding, reordering, rescaling and IDCT. Motion compensated prediction is formed using the decoded motion vectors and the local copy of the decoded reference VOP. This prediction is then combined with the decoded texture data to reconstruct the macroblock. Fig. 1(b) shows the encoding/decoding process of a P-VOP. Note that the macroblocks in a P-VOP can still be coded in Intra mode. This might occur for regions in a frame where there is no match from the previous frame. Common examples of such regions are frame boundaries.

1) Error Resilience: The presence of differential coding, entropy coding and motion vectors make the encoded bitstream highly sensitive to channel bit errors. A single bit error can cause the decoder to lose synchronization in the decoding process. This may result in corruption of all of the macroblocks until the end of the current VOP, causing large amounts of distortion in the decoded VOP. Furthermore, due to the presence of motion compensation, the effects of this error may also propagate

to subsequent frames, introducing significant amounts of distortion. To mitigate the effects of such errors, the MPEG-4 standard includes certain error resilience tools and features. We use two of these tools in our coding process: data partitioning and packetization. Both these error resilience tools are briefly described as follows.

Packetization: As discussed above, a single bit error can cause the decoder to lose synchronization resulting in spatial propagation of errors. To overcome this problem, a resynchronization mechanism is required. There are quite a few different methods to achieve resynchronization in MPEG-4. The most common method is that of packetization. Packetization is an MPEG-4 error resilience tool that attempts to enable resynchronization between the decoder and the bitstream by inserting resynchronization markers at different locations in the bitstream. The encoder divides the frame into different packets, and a resynchronization marker is placed at the beginning of each packet. Packetization can either be performed such that a resynchronization marker is inserted after a fixed number of macroblocks, or after a fixed number of bits. The encoding and decoding processes restart when a resynchronization marker is encountered; i.e., there is no differential, run-length and variable length encoding across the resynchronization markers. Hence, an error does not propagate across the packet boundary, resulting in a significant reduction in the amount of distortion due to spatial error propagation. Note that the resynchronization marker is uniquely decodeable and distinguishable from all possible codewords. The resynchronization marker is followed by header information consisting of the next macroblock number, the quantization parameter and a flag, header extension code (HEC). HEC indicates whether a duplicate of the VOP header is present in the packet. The macroblock number is used for spatial resynchronization of macroblock data, and the quantization parameter is useful for resynchronization of differential and variable length coding. The duplicate of the VOP header is useful to recover a lost VOP header (in case the packet containing the VOP header is lost). The simplified structure of a video packet in data partitioning mode is shown in Fig. 2.

Data Partitioning: In the data partitioning mode, the encoded data within a video packet is divided into two partitions. The main idea is to separate the more important data (DC coefficients, coding mode information, motion vectors) from the less important data (AC coefficients, residual error). For an I-VOP, the first partition contains the coding mode information of all the macroblocks and the DC coefficients of all the blocks in the packet, and the second partition contains the AC coefficients. For a P-VOP, the first partition contains the coding mode information and motion vectors for all the macroblocks, whereas the second partition contains the DCT data (texture, DC and AC coefficients) for all the blocks in the packet. The partitions in a video packet are separated using secondary resynchronization markers. These secondary markers are unique for the first partition and cannot be emulated by data in the first partition, however, they can be emulated by data in the second partition. Note that within a packet, differential coding, VLC and run-length coding are again re-initialized for different partitions. By dividing data into two partitions, the encoder separates the more important data (DC coefficients for I frames and motion vectors for P frames) from the less important data. Hence, in the case of bit errors in the second partition, the first partition can still be correctly decoded, reducing the amount of distortion introduced in the decoded video sequence. Simplified structures of video packets in data partitioning mode for I and P VOPs are shown in Fig. 2(a) and (b), respectively.

2) Error Detection and Concealment: We assume that in the case of bit errors, the source decoder detects the errors and marks the entire partition in the video packet as corrupted. Usually in variable length coding, the code being used is not complete; i.e., not all the possible codewords are legitimate. Hence, when a bit error results in corrupting a codeword such that the resulting sequence is not in the decoding table, the decoder declares an error. It is possible that the corrupted sequence is also legitimate, resulting in the bit error not being detected. However, since our decoder just needs one illegitimate codeword in the entire partition to declare an error, it is most likely that the error will be detected. More discussion on this assumption can be found in our earlier work in [21], where we provided simulation results in support of the error detection assumption.

For an I-VOP, if a bit error occurs in the DC partition of a packet, all the data in that packet is discarded, and the DC and the AC coefficients of all the blocks in the packet are decoded as zeros. When an error occurs in the AC partition (DC partition is error free), only the AC coefficients of all the blocks in the packet are decoded as zeros. For a P-VOP, when an error occurs in the motion vector (MV) partition of a packet, the entire packet is discarded. In this case, our decoder carries out a simple form of error concealment by copying pixel values from the previous frame at the exact spatial location. Though this is a very simple form of error concealment, it still provides much better results as compared to no error concealment. If an error occurs in the texture partition of a packet (MV partition is error free), then this partition is discarded, and, hence, no texture is added to the predicted macroblocks.

B. Channel Model

We derive our distortion model expressions for a binary symmetric channel (BSC) with a given bit error probability. Given the bit error probabilities for any channel (AWGN, Rayleigh fading, etc) and the fact that the probability of making an error from 0 to 1 is the same as that of 1 to 0, that channel can be represented as a BSC. Therefore, the distortion model presented in this paper can be used to find the distortion curves for any channel that can be represented as a BSC, given that the source coding rate and the bit error rate are known. Hence, our distortion model is independent of modulation type and channel coding. We also assume that the adjacent bit errors are independent, which can be achieved with sufficient interleaving. We do not consider any channel coding in deriving the expressions.

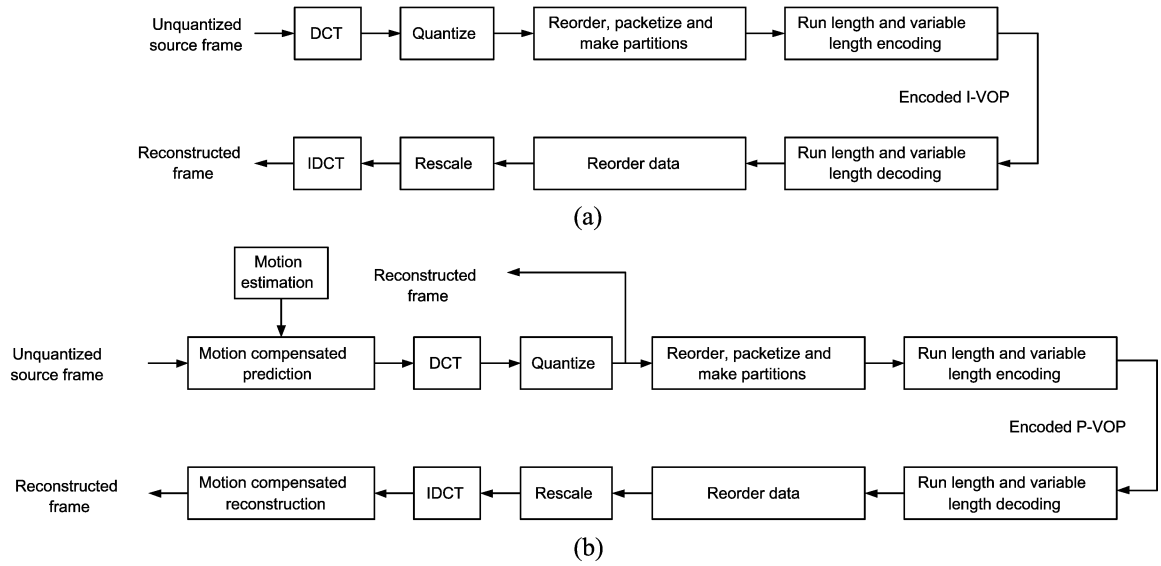


Fig. 1. MPEG-4 encoder and decoder for (a) I-VOP, and (b) P-VOP.

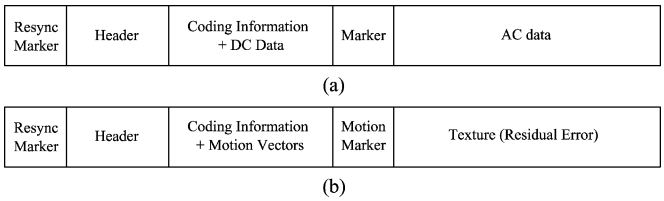


Fig. 2. Simplified MPEG-4 packet structure for (a) I frames and (b) P frames.

III. Distortion Model for MPEG-4

In this section, we derive expressions for estimating distortion due to quantization and channel errors in MPEG-4 coded video. MSE is used as our distortion metric. In the following subsections, we first outline our assumptions and notation, and then derive MSE expressions separately for I and P frames (VOPs).

Assumptions and Notation

The goal of our distortion model is to find MSE expressions for a video sequence as a function of source coding rate and channel bit error probability. It is difficult to model distortion in videos compressed by any of the video coding standards due to the presence of VLC, differential coding, run-length coding, and motion estimation and compensation. As discussed earlier, even single bit errors can have catastrophic effects on a video frame and on many subsequent frames. Therefore, we use the error resilience tools in MPEG-4 as discussed in the previous section, and make certain simplifying assumptions. We assume that headers and markers are transmitted error free separately from the raw data stream, as they constitute only a small portion of the total bitstream, and as they are extremely important for decoding.

We assume that the source decoder detects bit errors, and then discards the entire partition. Furthermore, if bit errors occur in the DC partition of an I frame's video packet, the AC partition is also discarded; and if errors occur in the motion vector partition of a P frame's video packet, then the texture partition is also discarded.

We derive our MSE expressions for a block of T frames, starting with an I frame, and followed by $T-1$ P frames. These frames are numbered from 0 to $T-1$, where 0th frame is the I frame. Since I frames are coded entirely in the DCT domain, we model distortion in I frames in the DCT domain. For P frames, since we have motion vectors in the first partition, we model distortion in the pixel domain. Let J_t be the number of packets in the tth frame $(t = 0 \ldots T-1)$. Also, let $K_{j,t}$ be the number of macroblocks in the jth packet of the tth frame. Each macroblock contains M 8×8 blocks of sample values. These blocks are luma and chroma blocks. Note that M is a constant and depends on the format of the video sequence. For example, for 4:2:0 video format, there are 4 luma and 2 chroma blocks. Hence, $M = 6$ for 4:2:0 video format. We do not use any weighting factors for different luma and chroma blocks. Our notation is outlined in Table I.

A. Distortion Model for I Frames

An I frame's video packet consists of two partitions. The first partition consists of "coding mode" information and coded DC coefficients, while the second partition consists of the remaining 63 coded AC coefficients. We will derive MSE expressions accounting for quantization and channel errors in the DC and the AC partitions of a packet separately, and then combine them to obtain expression for total MSE.

Suppose our I frame consists of J_0 packets, and let the jth packet contain $K_{j,0}$ coded macroblocks, where 0 in the subscript means that it is an I frame. Let $X_{u,m,k,j,0}$, $X^q_{u,m,k,j,0}$, $\widehat{X}^q_{u,m,k,j,0}$ be the unquantized, the quantized and the erroneous DCT coefficients corresponding to the uth subband of the mth block of the kth macroblock in the jth packet of the I frame, and $\xi_{u,m,k,j,0}$ be the corresponding quantization error $\left(X^q_{u,m,k,j,0} = X_{u,m,k,j,0} + \xi_{u,m,k,j,0}\right)$. Also, let $\mu_{X,u,j}$ and $\sigma^2_{X,u,j}$ respectively denote the sample mean and variance of the

TABLE I
NOTATION

Notation	Description
N	The total number of pixels in a video frame.
M	Number of 8×8 blocks in a macroblock. This is a constant for a video sequence.
T	The total number of frames (VOPs) starting from an I frame to (not including) the next I frame.
J_t	Number of video packets in the t^{th} frame ($t = 0...T-1$).
$K_{j,t}$	Number of macroblocks in the j^{th} packet of the t^{th} frame.
$L_{DC}^{j,0}, L_{AC}^{j,0}$	Number of DC and AC bits respectively in the j^{th} packet of the I frame (0 stands for I frame).
$L_{MV}^{j,t}, L_{TX}^{j,t}$	Number of bits in MV and texture partitions in the j^{th} packet of the t^{th} frame, respectively.
$p_e^{j,t}$	Probability of bit error in the j^{th} packet of the t^{th} frame.
$p_{DC}^{j,0}, p_{AC}^{j,0}$	Probability of at least one bit error occurring in the the DC and the AC partitions of the j^{th} packet of the I frame, respectively.
$p_{MV}^{j,t}, p_{TX}^{j,t}$	Probability of at least one bit error occurring in the motion vector and the texture partitions of the j^{th} packet of the t^{th} P frame, respectively.
$p_{DC}^{'}(t-1)$	Probability that the frame at a distance of $(t-1)$ frames from the previous I frame is free from distortion propagation effects due to errors in the DC partitions of the previous I frame's video packets ($t = 1...T-1$).
$p(t)$	Probability of a frame at a distance of t frames from the current frame being affected by an error in the current frame's video packets.
$X_{u,m,k,j,t}, X_{u,m,k,j,t}^{q}$, $\widehat{X}_{u,m,k,j,t}$	u^{th} subband unquantized, quantized and erroneous DCT coefficients respectively in the m^{th} block of the k^{th} macroblock in packet number j of the t^{th} frame.
$V_{i,m,k,j,t}, V_{i,m,k,j,t}^{q}$, $\overline{V}_{i,m,k,j,t}^{q}$	i^{th} unquantized, quantized, and decoded sample values respectively in the m^{th} block of the k^{th} macroblock in packet number j of the t^{th} frame.
$V_{i,m,k,j,t-1}^{q}$	The quantized pixel value in the $(t-1)^{th}$ frame at the exact spatial location as $V_{i,m,k,j,t}^{q}$.
$TX_{i,m,k,j,t}$	Quantized Texture value for the i^{th} pixel in the m^{th} block of the k^{th} macroblock in the j^{th} packet of the t^{th} frame.
$\mu_{X,u,j},\ \sigma_{X,u,j}^2$	Sample mean and variance of the quantized coefficients in the u^{th} subband of all the blocks and the macroblocks in the j^{th} packet of the I frame.
$\mu_{X,\xi,u,j},\ \sigma_{X,\xi,u,j}^2$	Sample mean and variance of the quantization error corresponding to the coefficients in the u^{th} subband of the j^{th} packet of the I frame.
$\mu_{V,j,t},\ \sigma_{V,j,t}^2$	Sample mean and variance for the quantized pixel values in the j^{th} packet of the t^{th} frame.
$\mu_{V,\xi,j,t},\ \sigma_{V,\xi,j,t}^2$	Sample mean and variance of the quantization error for the pixels in the j^{th} packet of the t^{th} frame.
$\mu_{TX,j,t},\ \sigma_{TX,j,t}^2$	Sample mean and variance for the texture values in the j^{th} packet of the t^{th} frame.
$MSE_0^{j,0}$	MSE in the I frame due to the loss of all the DC coefficients in the j^{th} packet of the I frame.
$MSE_{u_1-u_2}^{j,0}$	MSE in the I frame due to the loss of all the coefficients from subbands u_1 to u_2 in the j^{th} packet of the I frame.
$MSE_{DC}^{j,0}$, $MSE_{AC}^{j,0}$	The total MSE per pixel for the block of T frames due to errors in the DC and the AC partitions of the j^{th} packet of the I frame, respectively.
$MSE_{prop}^{j,t}$	MSE in the t^{th} P frame due to an error in the MV partition of the j^{th} packet.
$MSE_{MV}^{j,t}$, $MSE_{TX}^{j,t}$	The total MSE per pixel for the block of T frames due to errors in the motion vector and the texture partitions of the j^{th} packet in the t^{th} P frame, respectively.
MSE_I	MSE per pixel in the block of T frames due to quantization and channel errors in the I frame.
MSE_P^t	MSE per pixel in the block of T frames due to quantization and channel errors in the t^{th} P frame ($t = 1...T-1$).
MSE	Total MSE in the block of T frames due to quantization and channel errors.

quantized DCT coefficients in the uth subband of all the blocks and the macroblocks in the jth packet, and $\mu_{X,\xi,u,j}$ and $\sigma_{X,\xi,u,j}^2$ denote the sample mean and variance of the quantization error, respectively.

Now, let $L_{\mathrm{DC}}^{j,0}$ and $L_{\mathrm{AC}}^{j,0}$ be the number of bits in the DC and the AC partitions of the jth packet, respectively, and $p_e^{j,0}$ be the probability of bit error for the jth packet. Then, the probability that at least one bit error occurs in the DC partition of the jth packet is

$$p_{\mathrm{DC}}^{j,0} = \sum_{i=1}^{L_{\mathrm{DC}}^{j,0}} \left(1 - p_e^{j,0}\right)^{i-1} p_e^{j,0}.$$

In the case of a bit error, all the coefficients in the DC partition will be corrupted. We define MSE in the I frame due to the loss of all the DC coefficients in the jth packet as

$$\mathrm{MSE}_0^{j,0} = \frac{1}{N} \sum_{k=1}^{K_{j,0}} \sum_{m=1}^{M} \left(X_{0,m,k,j,0} - \widehat{X}_{0,m,k,j,0}^{q}\right)^2 \quad (1)$$

where N is the total number of pixels in the frame. We use the number of pixels to define our MSE because we need MSE per frame for our analysis. Note that we do not perform any error concealment on I frames. Hence, the erroneous coefficients are decoded as zeros, $(\widehat{X}_{0,m,k,j,0}^{q} = 0)$. Also, the quantization error and the quantized coefficients are assumed uncorrelated. Hence

$$\mathrm{MSE}_0^{j,0} = \frac{1}{N} \sum_{k=1}^{K_{j,0}} \sum_{m=1}^{M} \left[\left(X_{0,m,k,j,0}^{q}\right)^2 + \left(\xi_{0,m,k,j,0}\right)^2\right]. \quad (2)$$

Since a bit error in the DC partition also results in the AC partition being discarded, the distortion contribution due to the loss of the AC coefficients (there are 63 AC coefficients) is

$$\mathrm{MSE}_{1\text{–}63}^{j,0} = \frac{1}{N} \sum_{u=1}^{63} \sum_{k=1}^{K_{j,0}} \sum_{m=1}^{M} \left(X_{u,m,k,j,0} - \widehat{X}_{u,m,k,j,0}^{q}\right)^2. \quad (3)$$

Similar to the case of the DC coefficients, the erroneous AC coefficients are also decoded as zeros, and the quantized coefficients and quantization error can be assumed to be uncorrelated. Hence, expanding (3) and combining with (2), we get the total MSE, $\mathrm{MSE}_{0\text{–}63}^{j,0}$, in the I frame due to errors in the DC partition of the jth video packet

$$\mathrm{MSE}_{0\text{–}63}^{j,0} = \frac{1}{N} \sum_{u=0}^{63} \sum_{k=1}^{K_{j,0}} \sum_{m=1}^{M} \left[\left(X_{u,m,k,j,0}^{q}\right)^2 + \left(\xi_{u,m,k,j,0}\right)^2\right]. \quad (4)$$

Since

$$\sigma_{X,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,u,j}^2 = \frac{1}{\mathrm{MK}_{j,0}-1} \sum_{k=1}^{K_{j,0}} \sum_{m=1}^{M} \left(X_{u,m,k,j,0}^{q}\right)^2, \text{ and}$$

$$\sigma_{X,\xi,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,\xi,u,j}^2 = \frac{1}{\mathrm{MK}_{j,0}-1} \sum_{k=1}^{K_{j,0}} \sum_{m=1}^{M} \left(\xi_{u,m,k,j,0}\right)^2$$

we can express (4) in terms of the sample mean and variance as (5), shown at the bottom of the page. This is the expression for the MSE in the I frame due to quantization of DC and AC coefficients, and an error in the DC partition of the jth packet. Due to the presence of motion estimation and compensation, the part of this MSE corresponding to the bit error will be propagated to the $T-1$ subsequent P frames. We need to include this propagated distortion in our expression for MSE. Let $p(t)$ be the probability that the frame at a distance of t frames from the current frame is affected by an error in the current frame's video packets. Then, the total MSE (over the block of T frames) due to an error in the DC partition of the jth packet can be written as (6), shown at the bottom of the page.

Now, let us consider the case when the DC partition of a video packet is received error free, but the AC partition has a bit error. Using the same methodology as for the DC partition, the total MSE (over T frames), due to an error in the AC partition of the jth packet can be written as (7), shown at the bottom of the next page.

Combining (6) and (7), adding the quantization error variance for the case when there is no bit error in the entire packet, and summing for all the packets in the I frame, the expected value of MSE (over T frames) due to quantization and bit errors in the DC and the AC partitions of all the packets of the I frame can be written as

$$E(\mathrm{MSE}_I) = \sum_{j=1}^{J_0} \left(\mathrm{MSE}_{\mathrm{DC}}^{j,0} \cdot p_{\mathrm{DC}}^{j,0} + \mathrm{MSE}_{\mathrm{AC}}^{j,0} \cdot \left(1 - p_{\mathrm{DC}}^{j,0}\right)\right.$$

$$\mathrm{MSE}_{0\text{–}63}^{j,0} = \frac{(\mathrm{MK}_{j,0}-1)}{N} \sum_{u=0}^{63} \left(\sigma_{X,u,j}^2 + \sigma_{X,\xi,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\left(\mu_{X,u,j}^2 + \mu_{X,\xi,u,j}^2\right)\right) \quad (5)$$

$$\mathrm{MSE}_{\mathrm{DC}}^{j,0} = \frac{(\mathrm{MK}_{j,0}-1)}{NT} \sum_{u=0}^{63} \left(\left(\sigma_{X,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,u,j}^2\right) \cdot \sum_{t=0}^{T-1} p(t) + \sigma_{X,\xi,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,\xi,u,j}^2\right) \quad (6)$$

$$\cdot p_{\mathrm{AC}}^{j,0} + \sum_{u=0}^{63}\left(\sigma_{X,\xi,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,\xi,u,j}^2\right) p_I^j\Bigg) \quad (8)$$

where

$$p_{\mathrm{AC}}^{j,0} = \sum_{i=1}^{L_{\mathrm{AC}}^{j,0}} \left(1-p_e^{j,0}\right)^{i-1} p_e^{j,0}$$

is the probability that there is at least one bit error in the AC partition of the jth packet of the I frame, and

$$p_I^j = \left(1-p_e^{j,0}\right)^{L_{\mathrm{AC}}^{j,0}+L_{\mathrm{DC}}^{j,0}}$$

is the probability of error-free transmission of the jth packet of the I frame.

B. Distortion Model for P Frames

In the data partitioning mode for P frames, the motion vectors and the texture data are coded separately in different partitions of a video packet. Although texture information is coded as DCT coefficients (all 64 subbands are coded together), we will model distortion in the sample domain to take into account the effects of motion compensation (which is also done in the sample domain). We will use similar notation as for I frames, with slight modifications.

For a P frame, data from the previous frame is used for decoding. If the motion vectors in a video packet are received error free, the pixels are copied from the correct location in the previous frame. However, if there are bit errors in the motion vector partition of a video packet, error concealment is performed as discussed in Section II-A2.

Modeling of distortion is complicated for P frames as compared to I frames because of the presence of prediction from previous frames. Modeling the exact effects of error propagation in frames is practically intractable. For this reason, we will make a few simplifying assumptions to make our modeling process tractable and easier.

We assume that the previous frame is free from distortion due to errors in the DC partition of the I frame. This is because the distortion propagated to P frames due to errors in DC partitions of data packets of the I frame, which is already modeled by (8), will be much higher as compared to the additional distortion caused by erroneous P frames. Therefore, we ignore this additional distortion to keep our analysis simple, and only consider the case when the previous frame is free from distortion due to errors in the DC partitions of I frame's packets.

Now, suppose a bit error occurs in the motion vector partition of the jth packet of the tth P frame. The decoder detects this error, discards the motion and texture data, and conceals the error by copying macroblocks from the $(t-1)$th frame at the exact spatial location. The macroblocks in the previous frame might also be distorted due to distortion propagation from previous P frames. We assume that the distortion introduced due to errors in different P frames is uncorrelated and additive. Let $V_{i,m,k,j,t}$, $V_{i,m,k,j,t}^q$ and $\overline{V}_{i,m,k,j,t}^q$ be the ith unquantized, quantized and decoded sample values in the mth block of the kth macroblock in the jth packet of the tth frame, respectively. Hence, the MSE in the tth P frame due to an error in the MV partition of the jth packet can be written as

$$\mathrm{MSE}_{\mathrm{prop}}^{j,t} = \frac{1}{N} \cdot \sum_{k=1}^{K_{j,t}}\sum_{m=1}^{M}\sum_{i=1}^{64}\left(\left(V_{i,m,k,j,t}^q - \overline{V}_{i,m,k,j,t}^q\right)^2 + \xi_{i,m,k,j,t}^2\right) \quad (9)$$

where $\xi_{i,m,k,j,t}$ is the quantization error (assuming the quantized coefficients and the quantization error are uncorrelated). Now, let $V_{i,m,k,j,t-1}^q$ represent the pixel in the $(t-1)$th frame at the exact spatial location as $V_{i,m,k,j,t}^q$, and $\eta_{i,m,k,j,t-1}$ be the propagated distortion in $V_{i,m,k,j,t-1}^q$ from errors in previous P frames. If $\overline{V}_{i,m,k,j,t}^q$ is erroneous, error concealment is carried out, and we write

$$\overline{V}_{i,m,k,j,t}^q = V_{i,m,k,j,t-1}^q + \eta_{i,m,k,j,t-1}.$$

Based on our assumption, $V_{i,m,k,j,t}^q - V_{i,m,k,j,t-1}^q$ and $\eta_{i,m,k,j,t-1}$ are uncorrelated. Expanding (9) and using this assumption, we get (10), shown at the bottom of the page.

Instead of having a component of distortion from previous frames, we want to configure our distortion expression so that we can predict the effects of distortion propagation to future frames due to errors in the current frame. Therefore, we modify (10) to remove the effects of distortion propagation from previous P frames, and instead modify it to predict the distortion in the future frames due to errors in the current frame. Hence, using our assumption of additivity of distortion due to errors in P frames, the MSE (over the block of T frames) due to errors in the motion vector partition of the jth packet of the tth P frame

$$\mathrm{MSE}_{\mathrm{AC}}^{j,0} = \frac{(\mathrm{MK}_{j,0}-1)}{NT}\left(\sum_{u=1}^{63}\left(\sigma_{X,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,u,j}^2\right)\cdot\sum_{t=0}^{T-1}p(t) + \sum_{u=0}^{63}\left(\sigma_{X,\xi,u,j}^2 + \frac{\mathrm{MK}_{j,0}}{\mathrm{MK}_{j,0}-1}\mu_{X,\xi,u,j}^2\right)\right) \quad (7)$$

$$\mathrm{MSE}_{\mathrm{prop}}^{j,t} = \frac{1}{N}\sum_{k=1}^{K_{j,t}}\sum_{m=1}^{M}\sum_{i=1}^{64}\left(\left(V_{i,m,k,j,t}^q - V_{i,m,k,j,t-1}^q\right)^2 + \eta_{i,m,k,j,t-1}^2 + \xi_{i,m,k,j,t}^2\right) \quad (10)$$

can be written as in (11), shown at the bottom of the page, where $p(n)$ is the probability that the frame at a distance of n frames from the current frame will have distortion due to errors in the current frame. Following similar notation as for the I frame, see the equation at the bottom of the page. Then

$$\mathrm{MSE}_{\mathrm{MV}}^{j,t} = \frac{64MK_{j,t}-1}{NT}\left(\left(\sigma_{V,j,t}^2 + \frac{64\mathrm{MK}_{j,t}}{64\mathrm{MK}_{j,t}-1}\mu_{V,j,t}^2\right) \cdot \sum_{n=1}^{T-t} p(n) + \sigma_{V,\xi,j,t}^2 + \frac{64\mathrm{MK}_{j,t}}{64\mathrm{MK}_{j,t}-1}\mu_{V,\xi,j,t}^2\right). \quad (12)$$

This is the expression for the MSE due to errors in the MV partition of the jth packet of the tth P frame, including the propagation effects of these errors to the subsequent $T-t-1$ P frames.

Now, consider the case when there is an error in the texture partition, but the MV partition is error free. In this case, the texture will be lost, and the predicted pixel values will be displayed. Let $V'^{q}_{i,m,k,j,t}$ be the predicted sample value, $\mathrm{TX}_{i,m,k,j,t} = V^{q}_{i,m,k,j,t} - V'^{q}_{i,m,k,j,t}$ be the quantized texture, and $\sigma^2_{\mathrm{TX},j,t} + (64\mathrm{MK}_{j,t}/64\mathrm{MK}_{j,t}-1)\mu^2_{\mathrm{TX},j,t} = (1/64MK_{j,t}-1)\sum_{k=1}^{K_{j,t}}\sum_{m=1}^{M}\sum_{i=1}^{64}\mathrm{TX}^2_{i,m,k,j,t}$, where $\mu_{\mathrm{TX},j,t}$ and $\sigma^2_{\mathrm{TX},j,t}$ are the sample mean and variance for the quantized texture values in the jth packet of the tth P frame, respectively. Then, the MSE due to an error in the texture partition of the jth packet of the tth P frame can be written as (13), shown at the bottom of the page.

Let $L_{\mathrm{MV}}^{j,t}$ and $L_{\mathrm{TX}}^{j,t}$ be the number of bits in the motion vector and the texture partitions of the jth packet of the tth frame, respectively. Then, the probability of at least one bit error in the MV partition is

$$p_{\mathrm{MV}}^{j,t} = \sum_{i=1}^{L_{\mathrm{MV}}^{j,t}} \left(1-p_e^{j,t}\right)^{i-1} \cdot p_e^{j,t}.$$

Similarly, the probability of at least one bit error in the texture partition is

$$p_{\mathrm{TX}}^{j,t} = \sum_{i=1}^{L_{\mathrm{TX}}^{j,t}} \left(1-p_e^{j,t}\right)^{i-1} \cdot p_e^{j,t}.$$

Also, let $p'_{\mathrm{DC}}(t-1)$ be the probability that the previous frame is free from distortion propagation effects due to errors in the DC partitions of the corresponding I frame's video packets. Then, by combining (12) and (13), and summing for all the packets, we obtain the expected value of MSE (over the block of T video frames) due to quantization and channel errors in the tth P frame: see (14), shown at the bottom of the next page, where $p_P^{j,t} = \left(1-p_e^{j,t}\right)^{L_{\mathrm{MV}}^{j,t}+L_{\mathrm{TX}}^{j,t}}$ is the probability of error free transmission of the jth packet. Note that for the special case where a macroblock is coded in intra mode in a P video packet, we use the same method as described in Section III-B to compute the distortion due to the intra coded macroblock.

C. Total Distortion

The expected value of the total MSE in a block of T video frames is the sum of the MSEs due to individual I and P frames. Using (8) and (14), this can be expressed as

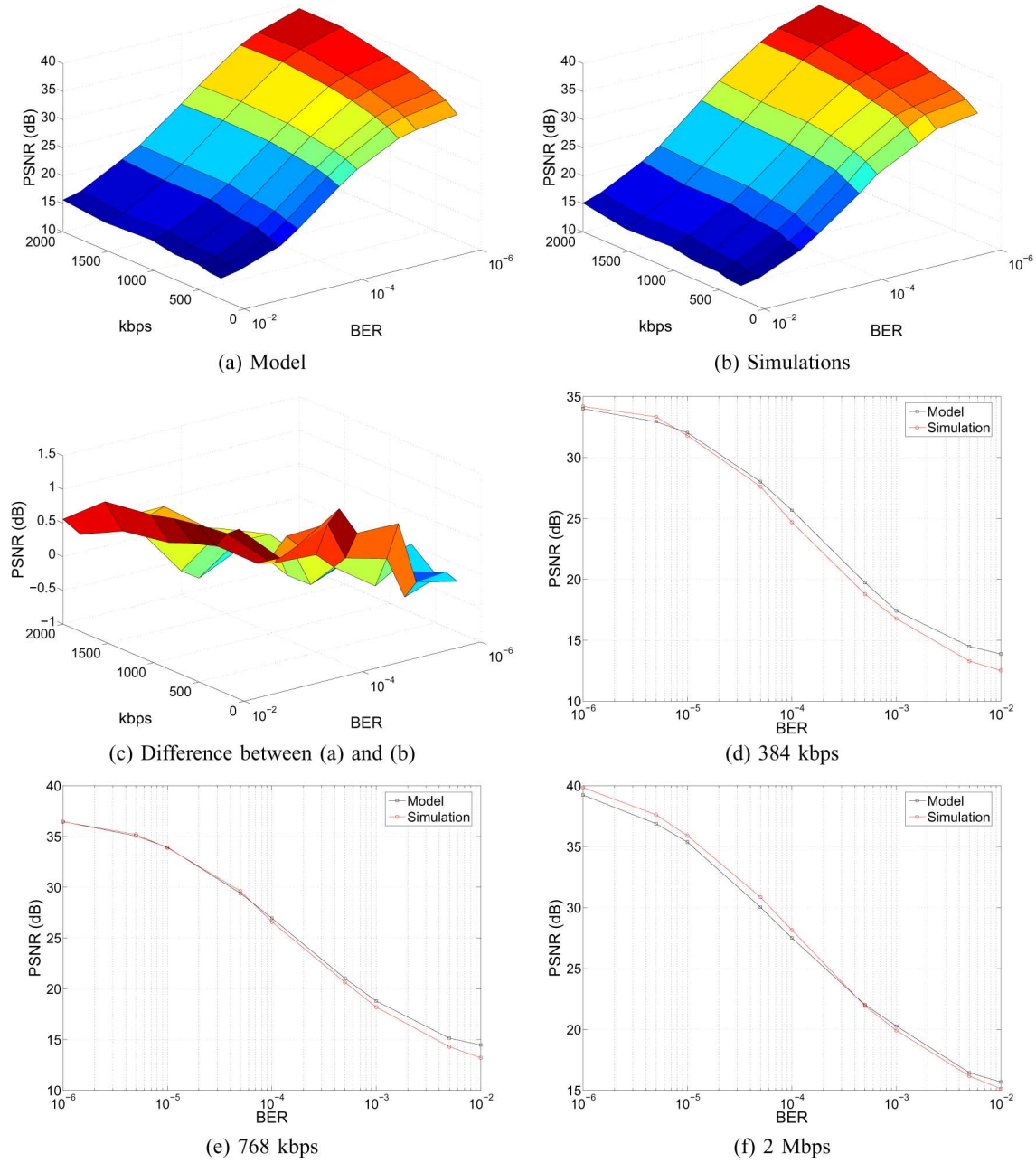
$$E(\mathrm{MSE}) = E(\mathrm{MSE}_I) + \sum_{t=1}^{T-1} E\left(\mathrm{MSE}_P^t\right). \quad (15)$$

$$\mathrm{MSE}_{\mathrm{MV}}^{j,t} = \frac{1}{NT}\sum_{k=1}^{K_{j,t}}\sum_{m=1}^{M}\sum_{i=1}^{64}\left(\left(V^{q}_{i,m,k,j,t} - V^{q}_{i,m,k,j,t-1}\right)^2 \cdot \sum_{n=1}^{T-t} p(n) + \xi^2_{i,m,k,j,t}\right) \quad (11)$$

$$\sigma_{V,j,t}^2 + \frac{64\mathrm{MK}_{j,t}}{64\mathrm{MK}_{j,t}-1}\mu_{V,j,t}^2 = \frac{1}{64MK_{j,t}-1}\sum_{k=1}^{K_{j,t}}\sum_{m=1}^{M}\sum_{i=1}^{64}\left(V^{q}_{i,m,k,j,t} - V^{q}_{i,m,k,j,t-1}\right)^2, \text{ and}$$

$$\sigma_{V,\xi,j,t}^2 + \frac{64\mathrm{MK}_{j,t}}{64\mathrm{MK}_{j,t}-1}\mu_{V,\xi,j,t}^2 = \frac{1}{64MK_{j,t}-1}\sum_{k=1}^{K_{j,t}}\sum_{m=1}^{M}\sum_{i=1}^{64}\left(\xi_{i,m,k,j,t}\right)^2$$

$$\mathrm{MSE}_{\mathrm{TX}}^{j,t} = \frac{64MK_{j,t}-1}{NT}\left(\left(\sigma^2_{\mathrm{TX},j,t} + \frac{64\mathrm{MK}_{j,t}}{64\mathrm{MK}_{j,t}-1}\mu^2_{\mathrm{TX},j,t}\right)\cdot\sum_{n=1}^{T-t}p(n) + \sigma^2_{V,\xi,j,t} + \frac{64\mathrm{MK}_{j,t}}{64\mathrm{MK}_{j,t}-1}\mu^2_{V,\xi,j,t}\right) \quad (13)$$



(a) Model (b) Simulations

(c) Difference between (a) and (b) (d) 384 kbps

(e) 768 kbps (f) 2 Mbps

Fig. 3. PSNR versus BER and kbps curves for the model and the simulations for the "Foreman" video sequence (for all 123 Frames).

The expected value of the MSE for the entire video sequence consisting of multiple blocks of T frames can be computed by taking the average of the MSE values obtained for individual blocks of frames using (15).

IV. Simulations and Results

In this section, we discuss the details of our simulations, and compare our model's prediction of MSE with test simulations.

$$E(\mathrm{MSE}_P^t) = \sum_{j=1}^{J_t} \Bigg(\mathrm{MSE}_{\mathrm{MV}}^{j,t} \cdot p_{\mathrm{MV}}^{j,t} p'_{\mathrm{DC}}(t-1) + \mathrm{MSE}_{\mathrm{TX}}^{j,t} \cdot \left(1 - p_{\mathrm{MV}}^{j,t}\right) p_{\mathrm{TX}}^{j,t} p'_{\mathrm{DC}}(t-1) + \left(\sigma_{V,\xi,j,t}^2 + \frac{64\mathrm{MK}j,t}{64\mathrm{MK}_{j,t} - 1} \mu_{V,\eta,j,t}^2 \right) p_P^{j,t} \Bigg) \tag{14}$$

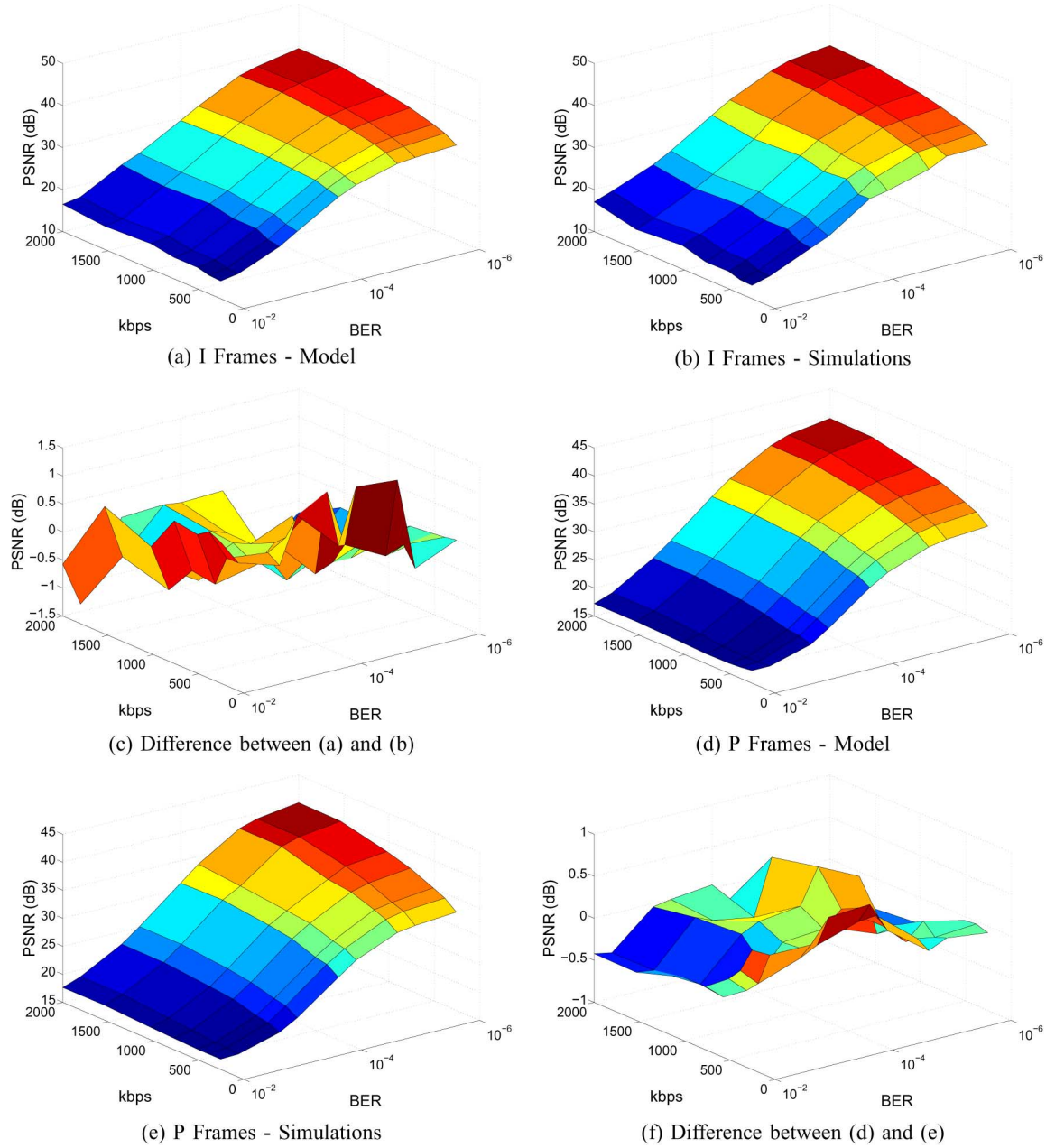


Fig. 4. PSNR versus BER and kbps curves for I and P frames using the model and the simulations for the "Foreman" video sequence.

We convert MSE to PSNR using the simple relation $\mathrm{PSNR} = 10\log_{10}(255^2/\mathrm{MSE})$, since PSNR is a commonly used metric for video and image quality assessment.

A. Model Parameters

In order to predict the MSE using (8), (14), and (15), certain model parameters are needed. These parameters consist of: i) $p(t)$, the probability that the frame at a distance of t frames from the current frame has distortion due to errors in the current frame's video packets, and ii) $p'_{\mathrm{DC}}(t-1)$, the probability that the frame at a distance of $t-1$ frames from the previous I frame is free from the distortion propagation effects due to errors in the DC partitions of the I frame's video packets. We used a training database of 20 352 $\times$ 288 4:2:0 (CIF) format videos with a 25 frames per second frame rate to find these parameters. These parameters were obtained by introducing random bit errors into different frames in the training video sequences, and then computing the distortion propagation to future frames due to these bit errors. For these training simulations, the number of P frames between I frames was varied from 10 to 200. Different source coding rates from 256 kilo bits per second (kbps) to 2 mega bits per second (Mbps), and different packet sizes were used to keep the model parameters as generic as possible.

B. Simulation Details

Two different sets of 20 352 $\times$ 288 4:2:0 (CIF) format videos with a 25 frames per second frame rate were used in the simulations, one for obtaining the model parameters (training) and the other for testing. These model parameters were then used

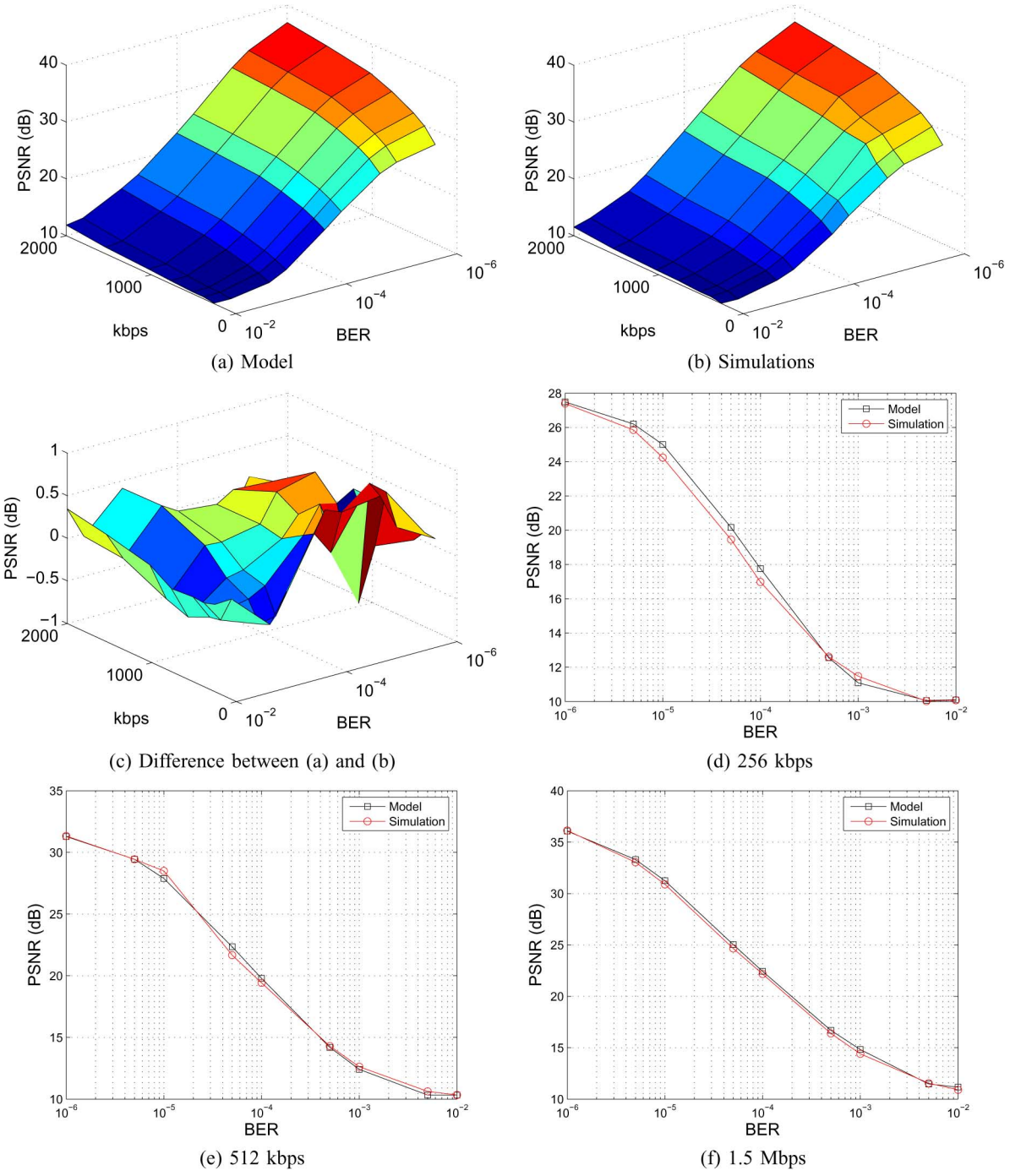


Fig. 5. PSNR versus BER and kbps curves for the model and the simulations for the "Walk" video sequence.

to predict the MSE using our model for different test video sequences at various source coding rates and channel BERs. Note that the means and variances required by our MSE expressions represent the local statistics of the video data in a packet, and are computed during encoding of each packet. These means and variances can be computed either in real-time (for real-time applications) during the encoding process, or they can be computed once for each coded video sequence and stored on file. MSE and PSNR were estimated for the test sequences using our model for source coding rates from 256 kbps to 2 Mbps, and BERs from 10^{-2} to 10^{-6}.

To test the accuracy of our model, we also computed MSE and PSNR values for the test sequences using simulations, by comparing the original video sequences with the quantized and erroneous sequences at various source coding rates and BERs. In these simulations, random bit errors were introduced in the coded bitstreams at the given BERs, and PSNR values were computed for the decoded video sequences. Source coding rate was varied from 256 kbps to 2 Mbps, and BER from 10^{-2} to 10^{-6}. 200 iterations were performed for each source coding rate and BER, and the average PSNR was calculated.

(a) Original Frame

(b) Error free frame at 512 kbps

(c) 512 kbps and 10^{-3} BER. PSNR: 18.55 dB (Model's prediction: 18.36 dB)

(d) 512 kbps and 10^{-4} BER. PSNR: 25.72 dB (Model's prediction: 26.31 dB)

(e) 512 kbps and 10^{-5} BER. PSNR: 31.93 dB (Model's prediction: 32.81 dB)

(f) 512 kbps and 10^{-6} BER. PSNR: 35.19 dB (Model's prediction: 34.91 dB)

Fig. 6. Distortion effects in a P coded frame from the "Foreman" video sequence at different BERs and 512 kbps source coding rate.

C. Results and Discussion

PSNR values were obtained using our model and simulations for different video sequences with different configurations of I and P frames. These PSNR curves are plotted against the source coding rate (in kbps) and the BER. Results for two "test" video sequences, "Foreman" and "Walk" are shown in Figs. 3–7. For both these video sequences, we used a packet size of 2000 bits. Fig. 3 shows the PSNR curves obtained using our model and the test simulations for the Foreman sequence. This video sequence consists of 123 frames with 3 I frames and 120 P frames, with 40 P frames between the I frames. For both the model and the simulations, these PSNR curves were obtained using the average MSE for the entire video sequence. Fig. 3(a) and (b) shows the PSNR curves obtained using our model and the test simulations respectively for the entire video sequence, whereas Fig. 3(c) shows the difference in PSNR between our model's prediction and the test simulations. Fig. 3(d)–(f) shows overlapped slices of Fig. 3(a) and (b) at different source coding rates. Fig. 4 shows the PSNR results for I and P frames separately. Fig. 4(a) and (b) shows the PSNR for I frames using our model and the test simulations respectively, whereas Fig. 4(c) shows their difference. Similarly, Fig. 4(d) and (e) shows the results for our model and the test simulations for P frames, and Fig. 4(f) shows their difference.

The PSNR curves for "Walk" sequence are shown in Fig. 5. For this video sequence, we coded 105 frames, with 5 I frames,

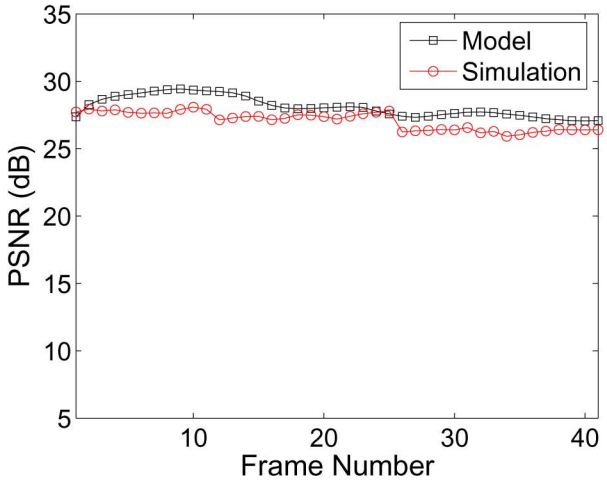


Fig. 7. PSNR predicted by the model as well as that obtained via simulations for the first 41 frames (an I frame and 40 P frames) of the Foreman video sequence at 512 kbps and 10^{-4} BER.

and 20 P frames between the I frames. These PSNR curves were also obtained using the average MSE for the entire video sequence. Fig. 5(a) and (b) shows the model and test simulation results for the entire video sequence, where as Fig. 5(c) is the difference between (a) and (b). Fig. 5(d)–(f) shows the overlapped slices of (a) and (b) at difference source coding rates.

As can be seen from Fig. 3(c), the difference in PSNR between the model and the simulations for the "Forema" video sequence is within 1.5 dB at all source coding rates and bit error rates. Also, as shown in Fig. 4(c) and (f), respectively, the difference in PSNR obtained using the model and the simulations for I frames is again within 1.5 dB, where as for the P frames this difference is within 1 dB at all points. For the "Walk" video sequence, the difference in PSNR between the model and the test simulations is within 1 dB at all points, as shown in Fig. 5(c). Furthermore, Fig. 3(d)–(f) and Fig. 5(d)–(f) also show that the PSNR values obtained using the model and the test simulations are very close at various BERs for different source coding rates. Fig. 6 shows the visual effects of distortion for a P coded frame from the "Foreman" video sequence at 512 kbps and at different bit error rates. The model's predicted PSNR values along with the PSNR values obtained using the simulations are also shown. Again, as can be seen from these PSNR values, our model predicts the distortion with high accuracy. To test how our model predicts distortion on a frame-by-frame basis, we fixed BER to 10^{-4} and source coding rate to 512 kbps, and computed PSNR for the first 41 frames of the Foreman sequence using our model and test simulations. The result of this simulation is shown in Fig. 7. As can be seen from this figure, our model predicts distortion with high accuracy on a frame-by-frame basis as well. Similar PSNR curves were also obtained for 20 other test sequences with different combinations of I and P frames and different packet sizes, however, results for only these two video sequences are shown here due to lack of space.

Considering the different complex components in source coding such as differential coding, VLC, run-length coding and motion compensation and estimation, our model predicts the actual amount of distortion with high accuracy. An important feature of our model is that it can predict the amount of distortion propagated to future frames due to errors at any location in the bitstream. Hence, this model can exactly determine the importance of data in different video packets for the overall video quality. We believe that efficient JSCC schemes can be designed by using this model to determine the distortion due to source coding and bit errors, as well as to determine the importance of different parts of the coded video data. Furthermore, due to the low complexity of this model, it can be used to design real-time JSCC schemes.

V. Conclusion

In this paper, we presented a model for estimating the distortion introduced in the MPEG-4 coded video stream due to quantization and channel errors. This model takes into account the effects of important components of video coding such as motion estimation and compensation, transform coding, and entropy coding, and uses different error resilience tools of the MPEG-4 video coding standard. We derived expressions for predicting the MSE due to quantization and channel bit errors in I and P frames. An important feature of this model is that it predicts the effects of distortion propagation to future frames due to errors at any point in the bitstream. Simulation results show that the PSNR values predicted by our model are accurate within 1.5 dB of the actual PSNR values obtained via simulations. Although this model is fine-tuned for MPEG-4, it can be used for any video coding scheme that uses motion compensation, transform coding and entropy coding, with slight modifications. Since this model predicts distortion with high accuracy and low complexity, it can be used to design efficient joint source-channel coding for real-time video communication applications.

References

[1] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, Jul. 1948.
[2] H. Gharavi and S. M. Alamouti, "Multipriority video transmission for third-generation wireless communication systems," *Proc. IEEE*, vol. 87, no. 10, pp. 1751–1763, Oct. 1999.
[3] R. van Dyck and D. Miller, "Transport of wireless video using separate, concatenated, and joint source-channel coding," *Proc. IEEE*, vol. 87, no. 10, pp. 1734–1750, Oct. 1999.
[4] J. Hagenauer and T. Stockhammer, "Channel coding and transmission aspects for wireless multimedia," *Proc. IEEE*, vol. 87, no. 10, pp. 1164–1777, Oct. 1999.
[5] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.
[6] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting real-time video over the internet: Challenges andapproaches," *Proc. IEEE*, vol. 88, no. 12, pp. 1855–1877, Dec. 2000.
[7] M. Bystrom and J. W. Modestino, "Combined source-channel coding schemes for video transmission over an additive white Gaussian noise channel," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 880–890, Jun. 2000.
[8] G. Cheung and A. Zakhor, "Bit allocation for joint source/channel coding of scalable video," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 340–356, Mar. 2000.
[9] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos, "Joint source-channel coding for motion-compensated dct-based snr scalable video," *IEEE Trans. Image Process.*, vol. 11, no. 11, pp. 1043–1052, Sep. 2002.
[10] F. Zhai, Y. Eisenberg, T. Pappas, R. Berry, and A. Katsaggelos, "Rate-distortion optimized hybrid error control for real-time packetized video transmission," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 40–53, Jan. 2006.

[11] M. Gallant and F. Kossentini, "Rate-distortion optimized layered coding with unequal errorprotection for robust internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 357–372, Mar. 2001.
[12] M. Bystrom and T. Stockhammer, "Dependent source and channel rate allocation for video transmission," *IEEE Trans. Wireless Commun.*, vol. 3, no. 1, pp. 258–268, Jan. 2004.
[13] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 511–523, Jun. 2002.
[14] L. Cheng, W. Zhang, and L. Chen, "Rate-distortion optimized unequal loss protection for FGS compressed video," *IEEE Trans. Broadcasting*, vol. 50, no. 2, pp. 126–131, Jun. 2004.
[15] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1012–1032, Jun. 2000.
[16] R. Zhang, S. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packetloss resilience," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 966–976, Jun. 2000.
[17] C. Bergeron and C. Lamy-Bergot, "Modelling h.264/avc sensivitity for error protection in wireless transmissions," in *Proc. IEEE 8th Workshop Multimedia Signal Processing*, 2006, pp. 302–305.
[18] M. Dai, D. Loguinov, and H. Radha, "Statistical analysis and distortion modeling of MPEG-4 FGS," in *Proc. Int. Conf. Image Processing*, 2003, vol. 3, pp. III-301–4.
[19] M. Dai, D. Loguinov, and H. Radha, "Rate-distortion modeling of scalable video coders," in *Proc. Int. Conf. Image Processing*, 2004, vol. 2, pp. 1093–1096.
[20] M. J. Ruf and J. W. Modestino, "Operational rate-distortion performance for joint source and channel coding of images," *IEEE Trans. Image Process.*, vol. 8, no. 3, pp. 305–320, Mar. 1999.
[21] M. Sabir, H. Sheikh, R. Heath, Jr., and A. Bovik, "A joint source-channel distortion model for JPEG compressed images," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1349–1364, Jun. 2006.
[22] S. Appadwedula, D. Jones, K. Ramchandran, and L. Qian, "Joint source channel matching for a wireless image transmission," in *Proc. IEEE Int. Conf. Image Processing*, 1998, vol. 2, pp. 137–141.
[23] I.-M. Kim and H.-M. Kim, "An optimum power management scheme for wireless video service in cdma systems," *IEEE Trans. Wireless Commun.*, vol. 2, no. 1, pp. 81–91, Jan. 2003.
[24] Q. Zhang, Z. Ji, W. Zhu, and Y.-Q. Zhang, "Power-minimized bit allocation for video communication over wireless channels," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 398–410, Jun. 2002.
[25] H. Yousefi'zadeh, H. Jafarkhani, and M. Moshfeghi, "Power optimization of wireless media systems with space-time block codes," *IEEE Trans. Image Process.*, vol. 13, no. 7, pp. 873–884, Jul. 2004.
[26] H. Zheng and K. Liu, "The subband modulation: a joint power and rate allocation frameworkfor subband image and video transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 5, pp. 823–838, Aug. 1999.
[27] I. Kozintsev and K. Ramchandran, "Robust image transmission over energy-constrained time-varying channels using multiresolution joint source-channel coding," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 1012–26, Apr. 1998.
[28] K. Ramchandran, A. Ortega, K. Uz, and M. Vetterli, "Multiresolution broadcast for digital HDTV using jointsource/channel coding," *IEEE J. Sel. Areas Commun.*, vol. 11, no. 1, pp. 6–23, Jun. 1993.
[29] A. Nosratinia, J. Lu, and B. Aazhang, "Source-channel rate allocation for progressive transmission of images," *IEEE Trans. Commun.*, vol. 51, no. 2, pp. 186–196, Feb. 2003.
[30] Y. Eisenberg, C. Luna, T. Pappas, R. Berry, and A. Katsaggelos, "Joint source coding and transmission power management for energy efficient wireless video communications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 411–424, Jun. 2002.
[31] Y. Wang, S. Wenger, J. Wen, and A. Katsaggelos, "Error resilient video coding techniques," *IEEE Signal Process. Mag.*, vol. 17, no. 4, pp. 61–82, Apr. 2000.
[32] R. Talluri, "Error-resilient video coding in the iso mpeg-4 standard," *IEEE Commun. Mag.*, vol. 36, no. 6, pp. 112–119, Jun. 1998.
[33] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression, Video Coding for Next-generation Multimedia*. New York: Wiley, 2003.

Muhammad Farooq Sabir (S'00–M'07) received the B.S. degree in electronics engineering from the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan, in 1999, and the M.S. and Ph.D. degress in electrical engineering from The University of Texas at Austin, Austin in 2002 and 2006, respectively.

He is currently with K-WILL Corporation, working in the area of image and video quality assessment. His research interests include image and video quality assessment, joint source-channel coding, unequal error protection, multimedia communication, wireless communications, space-time coding, and multiple-input multiple-output systems.

Robert W. Heath, Jr. (S'96–M'01–SM'06) received the B.S. and M.S. degrees from the University of Virginia, Charlottesville, in 1996 and 1997, respectively, and the Ph.D. degree from Stanford University, Stanford, CA, in 2002, all in electrical engineering.

From 1998 to 2001, he was a Senior Member of the Technical Staff then a Senior Consultant at Iospan Wireless, Inc., San Jose, CA where he worked on the design and implementation of the physical and link layers of the first commercial MIMO-OFDM communication system. In 2003, he founded MIMO Wireless, Inc., a consulting company dedicated to the advancement of MIMO technology. Since January 2002, he has been with the Department of Electrical and Computer Engineering, The University of Texas at Austin, where he is currently an Associate Professor and member of the Wireless Networking and Communications Group. His research interests include several aspects of MIMO communication: limited feedback techniques, multihop networking, multiuser MIMO, antenna design, and scheduling algorithms, as well as 60-GHz communication techniques and multimedia signal processing.

Dr. Heath has been an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS and an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He is a member of the Signal Processing for Communications Technical Committee, IEEE Signal Processing Society. He was Technical Co-Chair for the 2007 Fall Vehicular Technology Conference, General Chair of the 2008 Communication Theory Workshop, and co-organizer of the 2009 Signal Processing forWireless Communications Workshop. He is the recipient of the David and Doris Lybarger Endowed Faculty Fellowship in Engineering and a registered Professional Engineer in Texas.

Alan Conrad Bovik (S'80–M'81–SM'89–F'96) received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, in 1980, 1982, and 1984, respectively.

He is currently the Curry/Cullen Trust Endowed Professor at The University of Texas at Austin, where he is the Director of the Laboratory for Image and Video Engineering (LIVE) in the Center for Perceptual Systems. His research interests include image and video processing, computational vision, digital microscopy, and modeling of biological visual perception. He has published over 450 technical articles in these areas and holds two U.S. patents. He is also the author of *The Handbook of Image and Video Processing* (Elsevier, 2005, 2nd ed.) and *Modern Image Quality Assessment* (Morgan & Claypool, 2006).

Dr. Bovik has received a number of major awards from the IEEE Signal Processing Society, including: the Education Award (2007); the Technical Achievement Award (2005); the Distinguished Lecturer Award (2000); and the Meritorious Service Award (1998). He is also a recipient of the Distinguished Alumni Award from the University of Illinois at Urbana-Champaign (2008), the IEEE Third Millennium Medal (2000), and two journal paper awards from the International Pattern Recognition Society (1988 and 1993). He is a Fellow of the Optical Society of America the Society of Photo-Optical and Instrumentation Engineers. He has been involved in numerous professional society activities, including: Board of Governors, IEEE Signal Processing Society, 19961998; Editor-in-Chief, IEEE TRANSACTIONS ON IMAGE PROCESSING, 19962002; Editorial Board, PROCEEDINGS OF THE IEEE, 19982004; Series Editor for Image, Video, and Multimedia Processing, Morgan and Claypool Publishing Company, 2003–present; and Founding General Chairman, First IEEE International Conference on Image Processing, Austin, TX, November 1994. He is a registered Professional Engineer in the State of Texas and is a frequent consultant to legal, industrial, and academic institutions.