

## **Error-Resilience Video Transcoding for Wireless Communications**

Anthony Vetro, Jun Xin, Huifang Sun

TR2005-102 August 2005

### **Abstract**

Video communication through wireless channels is still a challenging problem due to the limitations in bandwidth and the presence of channel errors. Since many video sources are originally coded at a high rate and without considering the different channel conditions that may be encountered later, a means to repurpose this content for delivery over a dynamic wireless channel is needed. Transcoding is typically used to reduce the rate and change the format of the originally encoded video source to match network conditions and terminal capabilities. Given the existence of channel errors that can easily corrupt the video quality, there is also the need to make the bitstream more resilient to transmission errors. In this article, we provide an overview of the error-resilience tools found in today's video coding standards and describe a variety of techniques that may be used to achieve error-resilience video transcoding.

*IEEE Wireless Communications*

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.



# Error-Resilience Video Transcoding for Wireless Communications

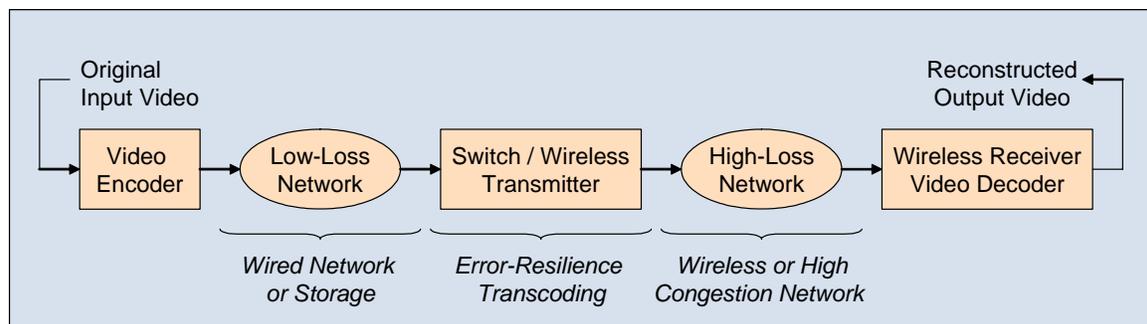
*Anthony Vetro, Jun Xin and Huifang Sun, Mitsubishi Electric Research Labs*

## ABSTRACT

Video communication through wireless channels is still a challenging problem due to the limitations in bandwidth and the presence of channel errors. Since many video sources are originally coded at a high rate and without considering the different channel conditions that may be encountered later, a means to repurpose this content for delivery over a dynamic wireless channel is needed. Transcoding is typically used to reduce the rate and change the format of the originally encoded video source to match network conditions and terminal capabilities. Given the existence of channel errors that can easily corrupt the video quality, there is also the need to make the bitstream more resilient to transmission errors. In this article, we provide an overview of the error-resilience tools found in today's video coding standards and describe a variety of techniques that may be used to achieve error-resilience video transcoding.

## INTRODUCTION

In a typical video distribution scenario as shown in Fig. 1, video content is captured, then immediately compressed and stored on a local network. At this stage, compression efficiency of the video signal is most important as the content is usually encoded with relatively high quality and independent of any actual channel characteristics. We note that the heterogeneity of client networks makes it difficult for the encoder to adaptively encode the video contents to a wide degree of different channel conditions; this is especially true for wireless clients. Subsequently, for transmission over wireless or highly congested networks, the video bitstream first passes through a network node, such as a mobile switch/base-station or proxy server, which performs error-resilience transcoding. In addition to satisfying rate constraints of the network and display or computational requirements of a terminal, the bitstream is transcoded so that an appropriate level of error-resilience is *injected* to the bitstream. The optimal solution in the transcoder is one that yields the highest reconstructed video quality at the receiver. For a general review on additional transcoding techniques, readers are referred to the articles in [1][2].

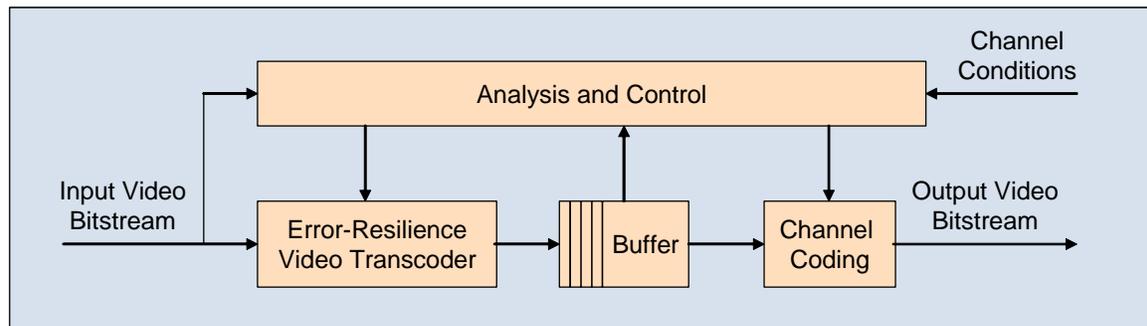


**Figure 1.** Video transmission architecture with error-resilience transcoding.

It should be noted that the process of error-resilience video transcoding is not achieved by the addition of bits into the input bitstream to make the output bitstream more robust to errors. Such an approach is closer to conventional channel coding approaches in which some overhead channel bits are added to the source payload for protection and possible recovery. Rather, for the video source, a variety of strategies exist that affect the bitstream structure at different levels of the stream, e.g. slice versus block level. Among the different techniques are to localize data segments to reduce error propagation, partition the stream so that unequal error protection could be applied, or add redundancy to the stream to enable a more robust means of decoding. In the next section, the specific error-resilience tools provided by the current video coding standards that achieve such features will be reviewed in more detail, where the impact on coding

efficiency and error propagation will be discussed.

Fig. 2 illustrates the high-level operation of a typical error-resilience transcoder. From the source side, characteristics of the video bitstream are usually extracted to understand the structure of the encoded bitstream and to begin building the end-to-end rate-distortion model of the source, while from the network side, characteristics of the channel are obtained. Both the content and channel characteristics as well as the current state of the buffer are used to control the operation of the error-resilience transcoder. Although this article focuses primarily on the robustness that could be achieved using coding tools at the source level, it is also possible to jointly optimize the source and channel coding. It should also be noted that the transcoding of stored video is not necessarily the same as that for live video. For instance, pre-analysis may be performed on stored video to gather useful information that may be used during the transcoding process. In second half of this article, techniques for error-resilience video transcoding will be presented, including some simulation results to demonstrate the effectiveness of this approach for robust video delivery.



**Figure 2.** Error-resilience transcoding of video based on analysis of video bitstream, channel measurement and buffer analysis.

## ERROR-RESILIENCE CODING TOOLS

While coding efficiency is the most important aspect in the design of any video coding scheme, the transmission of compressed video through noisy channels has always been a key consideration. This is evident by the many error-resilience tools that are available in today's video coding standards. In the following, a brief review of these tools is provided; a more comprehensive review of error control and concealment techniques may be found in [3].

Table 1 provides a summary of the tools covered in this article and the key benefits associated with each class of tools. The different strategies, which are not mutually exclusive, that may be employed during coding are:

- Localization – remove the spatial/temporal dependency between segments of the video to reduce error propagation
- Data Partitioning – group coded data according to relative importance to allow for unequal error protection or transport prioritization
- Redundant Coding – code segments of the video signal or syntactic elements of the bitstream with added redundancy to enable robust decoding
- Concealment-Driven – enables improved error concealment after decoding using additional information embedded into the coded stream or by uniquely ordering segments of the video signal

All of these strategies for error-resilience indirectly lead to an increase in the bit-rate and loss of coding efficiency, where the overhead with some is more than others. In the following, we describe each tool in terms of the benefit it provides for error-resilient transmission, as well as its impact on coding efficiency.

Category	Benefit	Tools
Localization	Reduce error propagation	Resynchronization marker Adaptive Intra Refresh Reference Picture Selection Multiple Reference Pictures
Data Partitioning	Enables unequal error protection and transport prioritization	Frequency Coefficients Motion, Header, Texture
Redundant Coding	Enables robust decoding	Reversible Variable Length Coding Multiple Description Coding Redundant Slice
Concealment-Driven	Enables improved error concealment	Concealment Motion Vectors Flexible Macroblock Order

**Table 1.** Benefits of error-resilience tools according to category.

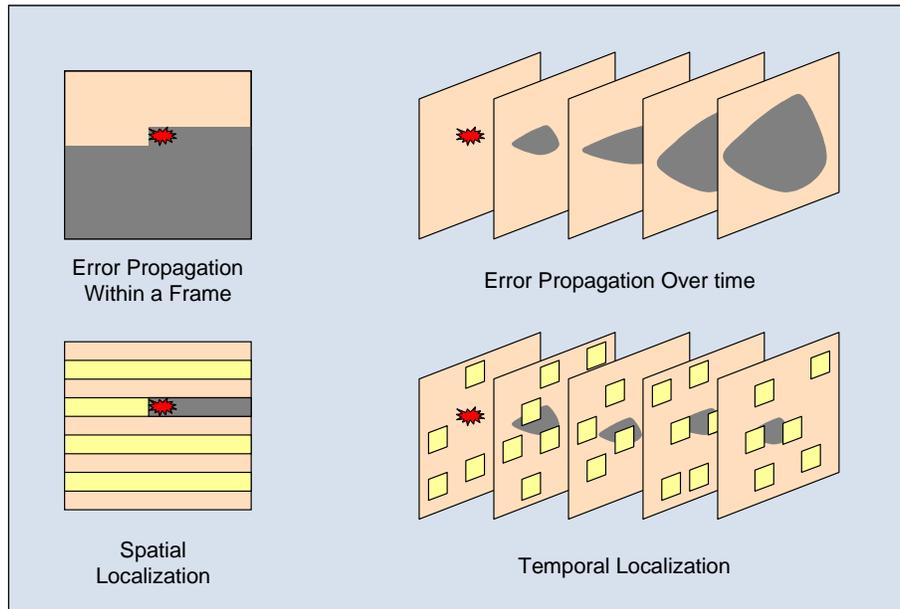
## LOCALIZATION

It is well known that video compression efficiency is achieved by exploiting the redundancy in both the spatial and temporal dimensions of the video. Due to the high correlation within and among neighboring frames, predictive coding schemes are employed to exploit this redundancy. While the predictive coding schemes are able to reach high compression ratios, they are highly susceptible to the propagation of errors. Localization techniques essentially break the predictive coding loop so that if an error does occur, then it is not likely to affect other parts of the video. Obviously, a high degree of localization will lead to lower compression efficiency. There are two methods for localization of errors in a coded video: spatial localization and temporal localization; these methods are illustrated in Fig. 3 and discussed further below.

Spatial localization considers the fact that most video coding schemes make heavy use of variable-length coding to reach high coding performance. In this case, even if one bit is lost or damaged, the entire bitstream may become undecodable due to the loss of synchronization between the decoder and the bitstream. To regain synchronization after a transmission error has been detected, resynchronization markers are added periodically into the bitstream at the boundary of particular macroblocks (MBs) in a frame. This marker would then be followed by essential header information that is necessary to restart the decoding process. When an error occurs, the data between the synchronization point prior to the error and the first point where synchronization is reestablished are typically discarded. For portions of the image that have been discarded, concealment techniques could be used to recover the pixel data, e.g., based on neighboring blocks that have been successfully decoded. For resynchronization markers to be effective in reducing error propagation, all predictions must be contained within the bounds of the marker bits. The restriction on predictions results in lowered compression efficiency. In addition, the inserted resynchronization markers and header information are redundant and they lower the coding efficiency. The spatial localization technique is supported in MPEG-2 and H.264/AVC using slices, and in MPEG-4 using video packets.

While resynchronization marker insertion is suitable to provide a spatial localization of errors, the insertion of intra-coded MBs is used to provide a temporal localization of errors by decreasing the temporal dependency in the coded video sequence. While this is not a specific tool for error-resilience, the technique is widely adopted and recognized as being useful for this purpose. The higher percentage of intra blocks used for coding the video will reduce the coding efficiency, but reduce the impact of error propagation on successively coded frames. In the most extreme case, all blocks in every frame are coded as intra blocks. In this case, there will be no temporal propagation of errors, but a significant increase in bit-rate could be expected. The selection of intra-coded blocks may be cyclic, in which the intra-coded blocks are selected according to a predetermined pattern; the intra-coded blocks may also be randomly chosen or adaptively chosen according to content characteristics.

Another form of temporal localization is reference picture selection, which was introduced in the H.263 and MPEG-4 standards for improved error-resilience. Assuming a feedback-based system, the encoder receives information about corrupt areas of the picture from the decoder, e.g., at the slice level, and then alters its operation by choosing a non-corrupted reference for prediction or applying intra-coding to the current data. In a similar spirit, the support for multiple reference pictures in H.264/AVC could achieve temporal localization as well. We should note that multi-frame prediction is not exclusively an error-resilience tool since it can also be used to improve coding efficiency in error-free environments.



**Figure 3.** Illustration of spatial and temporal localization to minimize error propagation within a frame and over time, respectively. Spatial localization is achieved by means of resynchronization markers, while temporal localization is achieved by means of intra-block coding.

### DATA PARTITIONING

It is well known that every bit in a compressed video bitstream is not of equal importance. Some bits belong to segments defining vital information such as picture types, quantization values, etc. When coded video bit-streams are transported over error-prone channels, errors in such segments cause a much longer lasting and severe degradation on the decoded video than that caused by errors in other segments. Therefore, data partition techniques have been developed to group together coded bits according to their importance to the decoding such that different groups may be more effectively protected using unequal protection techniques. For example, during the bitstream transmission over a single channel system, the more important partitions can be better protected with stronger channel codes than the less important partitions. Alternatively, with a multi-channel system, the more important partitions could be transmitted over the more reliable channel.

In MPEG-2, data partitioning divides the coded bitstream into two parts: a high-priority partition and low-priority partition. The high-priority partition contains picture type, quantization scale, motion vector ranges etc. without which the rest of the bitstream is not decodable. It may also include some macroblock header fields and DCT coefficients. The low-priority partition contains everything else.

In MPEG-4, the data partitioning is achieved by separating the motion and macroblock header information away from the texture information. This approach requires that a second resynchronization marker be inserted between motion and texture information, which may further help localize the error. If the texture information is lost, the motion information may be used to conceal these errors. That is, the texture information is discarded due to the errors, while the motion information is used to motion compensate the previously decoded picture.

## **REDUNDANT CODING**

This category of techniques tries to enhance error resilience by adding redundancy to the coded video. The redundancy may be added explicitly, such as with the Redundant Slices (RSs) tool, or implicitly in the coding scheme, as with Reversible Variable Length Codes (RVLC) and Multiple Description Coding (MD).

RVLC has been developed for the purpose of data recovery at the receiver. Using this tool, the variable length codes are designed so that they can be read both in the forward and reverse directions. This allows the bitstream to be decoded backwards from the next synchronization marker until the point of error. Examples of 3-bit codewords that satisfy this requirement include {111, 101, 010}. It is obvious that this approach will reduce the coding efficiency compared with using normal VLC due to the constraints imposed in constructing the RVLC tables, which is the primary reason we classify the RVLC approach as a redundant coding technique. It also shares the benefit with other tools that robust decoding could be performed. However, since this tool is designed to recover from bit errors, it is not a suitable tool for transmission over packet-erasure channels.

MD coding encodes a source with multiple bitstreams such that a basic-quality reconstruction is achieved if any one of them is correctly received, while enhanced-quality reconstructions are achieved if more than one of them is correctly received. In MD coding, the redundancy may be controlled by the amount of correlation between descriptions. Generally, MD coded video streams are suitable for delivery over multiple independent channels in which the probability of failure over one or more channels is likely.

Redundant slice is a new tool adopted into the H.264/AVC standard [4] that allows for different representations of the same source data to be coded using different encoding parameters. For instance, the primary slice may be coded with a fine quantization, while the redundant slice with a coarse quantization. If the primary slice is received, the redundant slice is discarded, but if the primary slice is lost, the redundant slice would be used to provide a lower level of reconstructed quality. In contrast to MD coding, the two slices together do not provide an improved reconstruction.

## **CONCEALMENT-DRIVEN**

Concealment-driven techniques refer to error-resilience coding tools that help with error concealment at the decoder. While such techniques do add redundancy to the coded bitstream, they are different from the above redundant coding tools in that they aim to improve the concealment of errors after decoding.

Concealment motion vectors are motion vectors that may be carried by intra macroblocks for the purpose of concealing errors. According to the MPEG-2 standard, concealment motion vectors for a macroblock should be appropriate for use in the macroblock that lies vertically below the macroblock in which the concealment motion vector is carried. In other words, when an error occurs in a give macroblock, the concealment motion vector from the macroblock above it could be used to form a prediction from the previous frame.

In the recent H.264/AVC video coding standard, flexible macroblock ordering (FMO) has been adopted to enable improved error concealment. The idea of FMO is to specify a pattern that allocates the macroblocks in a picture to one or several slice groups not in normal scanning order, but in a flexible way. In such a way, the spatially consecutive macroblocks may be assigned to different slice groups. Each slice group is transmitted separately. If a slice group is lost, the image pixels in spatially neighboring macroblocks that belong to other correctly received slice groups can be used for efficient error concealment. The allowed patterns of FMO range from rectangular patterns to regular scattered patterns, such as checkerboards, or completely random scatter patterns. Further information on the use of this tool in different application and delivery environments may be found in [5][6].

## **ERROR-RESILIENCE TRANSCODING TECHNIQUES**

In this section, we provide a brief survey of existing work on error-resilience transcoding. We then describe our own novel approach to the problem, which solves a joint optimization considering inter-frame dependency. Simulation results based on the proposed scheme are also presented.

### **SURVEY OF RELATED WORK**

One of the earliest error-resilience transcoding schemes, which was based on MPEG-2 video, is referred to as Error Resilience Entropy Coding (EREC) [7]. In this method, the incoming bit-stream is reordered without adding redundancy such that longer VLC blocks fill up the spaces left by shorter blocks in a number of VLC blocks that form a fixed-length EREC frame. Such fixed-length EREC frames of VLC codes are then used as synchronization units, where only one EREC frame, rather than all the codes between two synchronization markers, will be dropped should any VLC code in the EREC frame be corrupted due to transmission errors.

Some years later, a rate-distortion framework with analytical models that characterize the error propagation of a corrupted video bit-stream subjected to bit errors was proposed [8]. The models were used to guide the use of spatial and temporal localization tools: synchronization markers and intra-refresh, respectively, to compute the optimal bit-allocation among spatial error-resilience, temporal error-resilience, and the source rate. One drawback of this method is that it is assumed the actual rate-distortion characteristics of video source are known, which makes the optimization difficult to realize practically. Also, the impact of error concealment is not considered.

The work in [9] proposes an error-resilience transcoder for General Packet Radio Service (GPRS) mobile-access networks with the transcoding process performed at a video proxy that can be located at the edge of two or more networks. Two error-resilience tools: the Adaptive Intra Refresh (AIR) and Feedback Control Signaling (FCS) are used adaptively to reduce error effects, while preserving the transmission rate adaptation feature of the video transcoders. In this method, the bit allocation between inserted error resilience and the video source coding is not optimized though.

In [10], optimal error resilience insertion is divided into two sub-problems: optimal mode selection for MB's and optimal resynchronization marker insertion. In [11], a method to recursively compute the expected decoder distortion with pixel-level precision to account for spatial and temporal error propagation in a packet loss environment is proposed for optimal MB coding mode selection. Compared with previous distortion calculation methods, this method has been shown to be quite accurate on an MB level. In both of these methods, inter-frame dependency is not considered and the optimization is only conducted on an MB basis.

In an effort to exploit multiple description coding for error-resilience transcoding, a Multiple-Description FEC (MD-FEC) based scheme, which uses the  $(N, i, N-i+1)$  Reed-Solomon erasure-correction block code to protect the  $i$ -th layer of an  $N$ -layer scalable video was proposed in [12]. The multiple-description packetization method is specially designed to allow the  $i$ -th layer to be decodable when  $i$  or more descriptions arrive at the decoder.

Considering networking-level mechanisms, the scheme in [13] proposes to implement an ARQ proxy at the base station of a wireless communication system for handling ARQ requests and tracking errors to reduce retransmission delays as well as to enhance the error resilience. The ARQ proxy resends important lost packets (e.g., packets with header information and motion vectors) detected through the retransmission requests from wireless client terminals, while dropping less-important packets (e.g., packets carrying DCT coefficients) to satisfy bandwidth constraints. A transcoder is used to compensate for the mismatch error between the front-end video encoder and the client decoders caused by the dropped packets.

### **JOINT OPTIMIZATION CONSIDERING INTER-FRAME DEPENDENCY**

In this section, we address the joint optimization problem between video bit-rate reduction and error resilience insertion, considering inter-frame dependency in a Group-of-Pictures (GOP). We address the optimal bit allocation between video source rate and the error resilience insertion to achieve the best possible video quality under a given bit-rate constraint and channel condition.

More specifically, we aim to minimize the end-to-end distortion of a coded video bitstream subject to rate constraints, where the overall rate budget is allocated among different components that contribute to

the rate. Let  $K$  denote the number of components. Here, three distinct components are considered: video source, intra refresh in inter-coded frames and resynchronization marker insertion. The problem can then be solved as a constrained minimization problem, for which a Lagrangian optimization approach is taken to minimize:

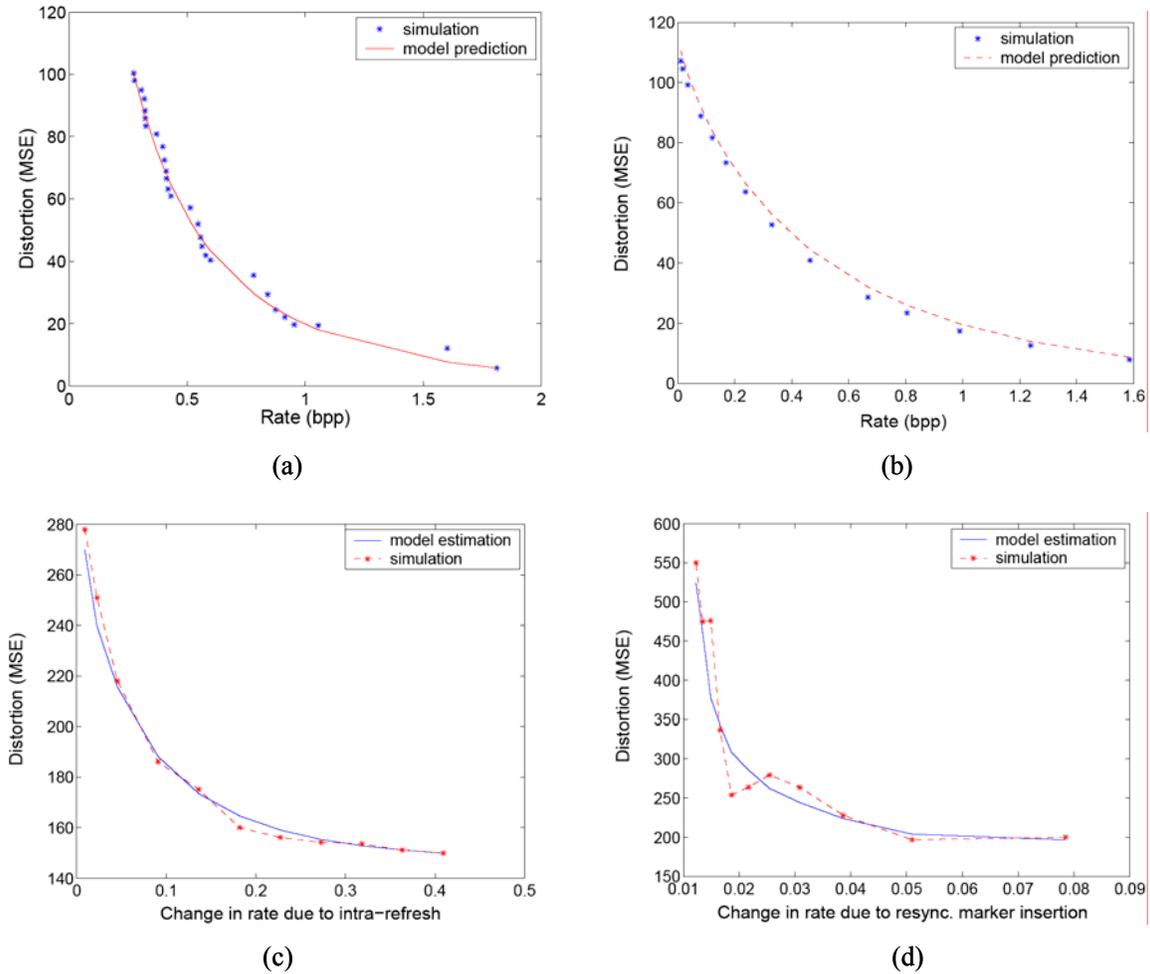
$$\sum_{k=1}^K d_k(\omega_k) + \lambda \sum_{k=1}^K r_k(\omega_k)$$

where  $d_k$  and  $r_k$  are the distortion and rate of each component, respectively,  $\lambda$  is the Lagrangian multiplier, and  $\omega_k$  are the specific parameters used in the allocation, i.e., quantization parameters, intra refresh rate and frequency of synchronization marker insertion. To solve this problem, a bisection algorithm can be used to obtain the optimal  $\lambda$ , but this is computationally expensive. Furthermore, obtaining accurate rate-distortion (R-D) sample points is still an issue.

In order to solve the above problem in a computationally efficient way, we establish R-D models to characterize each transcoding component: video source requantization, resynchronization marker insertion, and intra refresh. The proposed models are novel in that (a) inter-frame dependency is accounted for in both the video source model and error resilience model, and (b) the error resilience model factors in the impact of applying error concealment after decoding. It should be noted that the error-resilience model is designed to account for the different types of errors that may occur in a video frame as well as the types of concealment that are applied. For instance, we model the errors of lost intra-coded blocks that are concealed by spatial concealment techniques, lost inter-coded blocks that are temporally concealed, lost inter-coded blocks that are temporally concealed using a corrupt reference, and inter-coded blocks that have not been lost, but reference a corrupt block. Based on channel statistics and information about blocks in the reference frames, we estimate the expected number of occurrences for different types of block errors and approximate the average distortion for each block error assuming simple spatial and temporal concealment techniques. Of course, more sophisticated error concealment techniques may be applied in this modeling framework. However, assuming a simple error concealment technique in the model has the advantage of providing a worst-case analysis, which is useful when the actual error concealment technique in the receiver is unknown. Further details on these models may be found in [14].

To demonstrate the accuracy of the proposed models, we report the results of several experiments. First, to test the accuracy of the video source requantization model, we encode the Foreman sequence with a GOP size of 12 and with quantization scales  $Q_I = Q_P = 3$ , and then requantize the sequence using arbitrarily selected quantization scales  $Q_I = 8$ ,  $Q_P$  (frames #1 – #10) = 16 and  $Q_P$  (frame #11)  $\in [4, 31]$ . The R-D plots of the requantized I-frame and the last P-frame in the GOP are compared with the model estimates in Fig. 4(a) and (b), respectively. Note that since we account for inter-frame dependency in our model, the model estimates for the last P-frame rely on accurate estimates for all preceding frames. Next, to test the accuracy of the error-resilience models for intra refresh and resynchronization marker insertion, we transcode the Foreman sequence with different settings and simulate a channel with BER =  $10^{-4}$ . Fig. 4(c) shows the model performance as a function of changes in the intra-refresh rate, where only the intra refresh rate varies from 2% to 90%. Fig. 4(d) shows the model performance for resynchronization marker insertion as a function of marker spacing (or video packet length), where only the marker spacing varies from 130 bits to 1300 bits. It can be seen in both cases that the proposed models accurately predict the measured distortion.

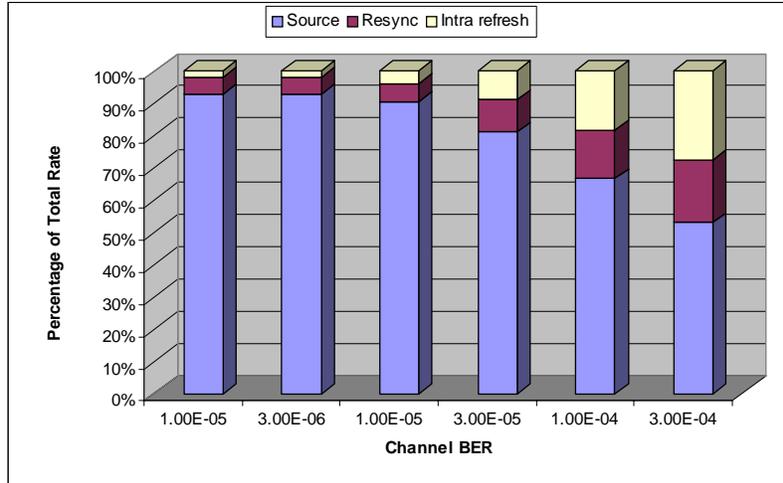
To enable real-time implementation for bit allocation, a technique to determine a sub-optimal operating point has also been proposed [14]. We refer to this technique as an R-D derivative equalization scheme. This scheme is based on the fact that optimal bit allocation is achieved at the point where the slopes of the R-D function for each component are equal [15]. Therefore, if we start from an operation point close to an optimal point, our objective is to continually adjust the operating point in the direction of the optimal point. From the plots in Fig. 4, it is clear that the R-D functions of all components are convex and such an approach would be valid. To achieve this, we assume a short initial delay and initialize the process by carrying out the optimal scheme for the first GOP. Then, for subsequent GOP's, we compare the local derivatives of each R-D curve and adjust the bits allocated to each component accordingly. Given that the rate budget has not changed, we have deduced that reallocating a change in rate,  $\Delta R$ , from the component with the smallest absolute derivative value to the component with the largest absolute derivative value is a close approximation to the optimal solution. This can be shown formally, as well as a means to determine the optimal value of  $\Delta R$ .



**Figure 4.** Illustration of R-D model predictions compared to simulated data. (a) model for requantization of I-frame, (b) model for requantization of P-frame, (c) model for intra-refresh, (d) model for resynchronization marker insertion.

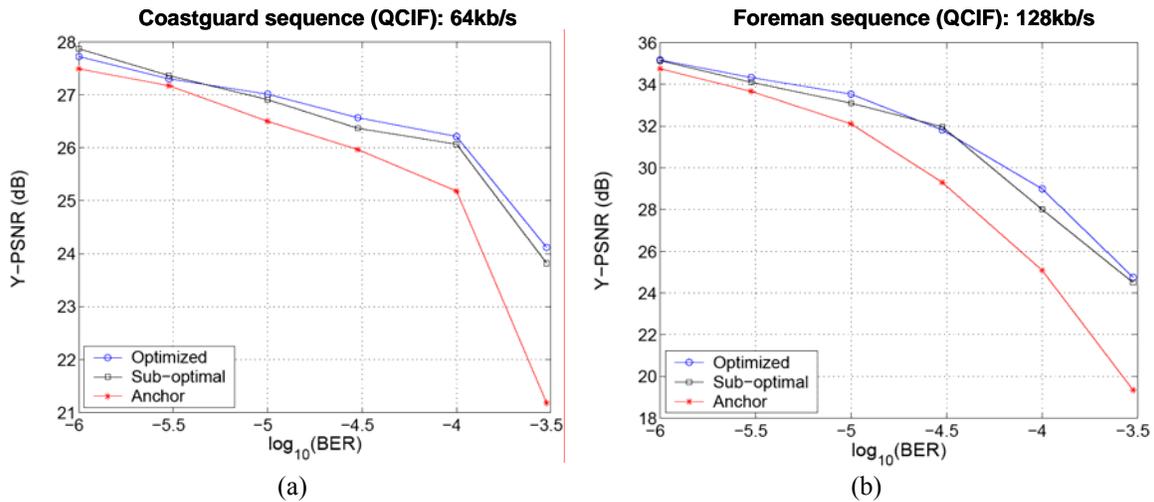
Further experiments are conducted using 300 frames of the Foreman and Coastguard test sequences in QCIF format. The sequences were sub-sampled to 10Hz and coded at a bit-rate of 384 kb/s in the MPEG-2 format. The GOP size was set to 10 frames and B-frames were not included. We then transcoded these video bitstreams to lower bit-rate MPEG-4 streams, while keeping the output frame rate the same.

A typical plot showing the bit allocation at different channel bit-error rates (BER) is illustrated in Fig. 5. In this example, the Foreman bitstream is transcoded down to a rate of 64 kb/s using the optimal bit allocation method with proposed R-D models. It is clear that as the channel becomes worse, a higher rate is allocated to the error-resilience components.



**Figure 5.** Optimal bit allocation for the Foreman bitstream transcoded to 64 kb/s.

For comparison purposes, we simulate an anchor bit allocation scheme that inserts a resynchronization marker every 11 MBs and uses a fixed intra refresh rate of 20% with cyclical refresh pattern. Figs 6(a) and 6(b) show plots of the average PSNR of the reconstructed video (luminance component only) under different BER settings and simulation conditions. In these plots, the sub-optimal scheme achieves similar PSNR performance as the optimal scheme, and both schemes outperform the anchor with a maximum gain of 5dB.



**Figure 6.** Comparison of average PSNR for reconstructed video based on optimized, sub-optimal and anchor bit allocation methods. The transcoded bitstreams are transmitted over simulated channels with different BER. (a) Coastguard bitstream transcoded from 384 kb/s to 64 kb/s; (b) Foreman bitstream transcoded from 384 kb/s to 128 kb/s.

## CONCLUDING REMARKS

Error-resilience transcoding is a key technology that enables robust streaming of stored video content over noisy channels. It is particularly useful when content has been produced independent of the transmission network and/or under dynamically changing network conditions. This article has reviewed a number of error-resilience coding tools, most of which could be found in the latest video coding standards. Additionally, we have outlined a number of error-resilience transcoding techniques and described in some detail our own novel approach to the problem that focuses on the use of localization methods to reduce error propagation. Simulation results show that significant gains up to 5dB could be expected under poor channel conditions.

While there has been a fair amount of work in the area of error-resilience transcoding, there seem to be several promising directions for further exploration that aim to maximize reconstructed video quality. For one, modeling error propagation for the most recent and emerging video coding formats pose several interesting challenges. While H.264/AVC offers superior compression efficiency compared to previous video coding standards, its prediction model is much more complex. Similarly, the latest wavelet-based scalable video coding formats have a more complex spatio-temporal dependency [16]. To achieve optimal or near-optimal bit allocation in practical implementations, accurate low-complexity models that characterize the performance in error-prone transmission environments for these formats are required. It is also worth noting that many of the new coding tools adopted for H.264/AVC have yet to be explored in the context of error-resilience transcoding. Finally, combining channel coding techniques and unique aspects of different networking environments with error-resilience transcoding of the source remains an open research problem.

## REFERENCES

- [1] A. Vetro, C. Christopoulos and H. Sun, "An overview of video transcoding architectures and techniques," *IEEE Signal Processing Magazine*, vol. 20, no. 2, pp. 18-29, Mar 2003.
- [2] J. Xin, C.W. Lin and M.T. Sun, "Digital Video Transcoding," *Proc. IEEE, Special Issue on Advances in Video Coding and Delivery*, Dec 2004 (to appear).
- [3] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, pp. 974-997, May 1998.
- [4] ITU-T Rec. H.264 | ISO/IEC 14496-10, "Advanced Video Coding," 2003.
- [5] S. Wegner, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645-656, July 2003.
- [6] T. Stockhammer, M. Hannuksela and T. Wiegand, "H.264/AVC in Wireless Environments," vol. 13, no. 7, pp. 657-673, July 2003.
- [7] R. Swann, and N. Kingsbury, "Transcoding of MPEG-II for enhanced resilience to transmission errors," *Proc. IEEE Int'l Conf. Image Processing*, vol. 2, pp. 813-816, Oct 1996.
- [8] G. de los Reyes, A. R. Reibman, S.-F. Chang, and J. C.-I. Chuang, "Error-resilient transcoding for video over wireless channels," *IEEE J. Selected Areas Commun.*, vol. 18, no. 6, pp. 1063-1074, June 2000.
- [9] S. Dogan, A. Cellatoglu, M. Uyguroglu, A. H. Sadka, and A. M. Kondoz, "Error-resilient video transcoding for robust internetwork communications using GPRS," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 453-564, June 2002.
- [10] G. Cote, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE J. Selected Areas Commun.*, vol. 18, no. 6, pp. 952-965, 2000.
- [11] R. Zhang, S.L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Selected Areas Commun.*, vol. 18, no. 6, pp. 966-976, 2000.
- [12] R. Puri, K.-W. Lee, K. Ramchandran, and V. Bhargavan, "An integrated source transcoding and congestion control paradigm for video streaming in the Internet," *IEEE Trans. multimedia*, vol. 3, no. 1, pp. 18-32, Mar. 2001.
- [13] T.-C. Wang, H.-C. Fang, and L.-G. Chen, "Low delay and error robust wireless video transmission for video communication," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 12, pp. 1049-1058, Dec. 2002.
- [14] M. Xia, A. Vetro, B. Liu, and H. Sun, "Rate-distortion optimized bit allocation for error resilient video transcoding," *Proc. IEEE Int. Symp. Circuits and Systems*, Vancouver, BC, May 2004.
- [15] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 1991.
- [16] J. Reichel, H. Schwarz and M. Wien, "Working Draft 1.0 of 14496-10:200x/Amd.1 Scalable Video Coding," ISO/IEC JTC1/SC29/WG11 Doc. N6901, Jan. 2005.