Error-Resilient Video Transmission Using Long-Term Memory Motion-Compensated Prediction

Thomas Wiegand, **Niko Färber**, *Member, IEEE*, **Klaus Stuhlmüller**, and
Telecommunications Laboratory
University of Erlangen-Nuremberg
Erlangen, Germany
[wiegand,faerber,stuhl]@lnt.de

Bernd Girod, *Fellow, IEEE*
Information Systems Laboratory
Stanford University
Stanford, CA, USA
girod@ee.stanford.edu

Abstract— **Long-term memory prediction extends the spatial displacement vector utilized in hybrid video coding by a variable time delay permitting the use of more than one reference frame for motion compensation. This extension leads to improved rate-distortion performance. However, motion compensation in combination with transmission errors leads to temporal error propagation that occurs when the reference frames at coder and decoder differ. In this paper, we present a framework that incorporates an estimated error into rate-constrained motion estimation and mode decision. Experimental results with a Rayleigh fading channel show that long-term memory prediction significantly outperforms the single-frame prediction H.263-based anchor. When a feedback channel is available, the decoder can inform the encoder about successful or unsuccessful transmission events by sending positive (ACK) or negative (NACK) acknowledgments. This information is utilized for updating the error estimates at the encoder. Similar concepts such as the ACK and NACK mode known from the H.263 standard are unified into a general framework providing superior transmission performance.**

I. Introduction

With continuously dropping costs of semiconductors, we might soon be able to considerably increase the memory in video codecs. Algorithms to take advantage of such large memory capacities, however, are in their infancy today. This has been the motivation for our research into long-term memory motion-compensated prediction (MCP). The efficiency of long-term memory MCP as an approach to improve coding performance has been demonstrated in [1]. The ITU-T has decided to adopt this feature as an Annex to the H.263 standard. In this paper, we show that the idea can also be applied to the transmission of coded video over noisy channels with the aim of improved rate-distortion performance.

In recent years, several standards such as H.261, H.263, MPEG-1, and MPEG-2 have been introduced which mainly address the compression of video data for digital storage and communication services. H.263 [2] as well as the other standards utilize hybrid video coding schemes which consist of block-based MCP and DCT-based quantization of the prediction error. Also the future MPEG-4 standard [3] will follow a similar video coding approach, but targeting different applications than H.263.

The H.263-compressed video signal is extremely vulnerable to transmission errors. In INTER mode, i.e., when MCP is utilized, the loss of information in one frame has considerable impact on the quality of the following frames. As a result, temporal error propagation is a typical transmission error effect for predictive coding. Because errors remain visible for a longer period of time the resulting artifacts are particularly annoying to end users. To some extent, the impairment caused by transmission errors decays over time due to leakage in the prediction loop. However, the leakage in standardized video decoders like H.263 is not very strong, and quick recovery can only be achieved when image regions are encoded in INTRA mode, i.e., without reference to a previous frame. The INTRA mode, however, is not selected very frequently during normal encoding. In particular, completely INTRA coded key frames are not usually inserted in real-time encoded video as is done for storage or broadcast applications. Instead, only single MBs are encoded in INTRA mode for image regions that cannot be predicted efficiently.

The Error Tracking approach [4], [5], [6], [7] utilizes the INTRA mode to stop inter-frame error propagation but limits its use to severely impaired image regions only. During error-free transmission, the more effective INTER mode is utilized, and the system therefore adapts to varying channel conditions. Note that this approach requires that the encoder has knowledge of the location and extent of erroneous image regions at the decoder. This can be achieved by utilizing a feedback channel from the receiver to the transmitter. The feedback channel is used to send Negative Acknowledgments (NACKs) back to the encoder. NACKs report the temporal and spatial location of image content that could not be decoded successfully and had to be concealed. Based on the information of a NACK, the encoder can reconstruct the resulting error distribution in the current frame, i.e., *track* the error from the original occurrence to the current frame. Then, the impaired MBs are determined and error propagation can be terminated by INTRA coding these MBs.

In this paper, we extend the Error Tracking approach to cases when the encoder has no knowledge about the actual occurrence of errors. In this situation the selection of INTRA coded MBs can be done either randomly or preferably in a certain update pattern. For example, Zhu [8] has investigated update patterns of different shape, such as 9 randomly distributed macroblocks, 1×9, or 3×3 groups of macroblocks. Though the shape of different patterns slightly influences the performance, the selection of the correct INTRA percentage has a significantly higher influence. In [9] and [10] it has been shown that it is advantageous to consider the image content when deciding on the frequency of INTRA coding. For example, image regions that cannot be concealed very well should be refreshed more often, whereas no INTRA coding is necessary for completely static background. In [11], [12], an analytical framework is presented on how to optimize the INTRA refresh rate. In [13], a trellis is used to estimate the concealment quality to introduce a bias into the macroblock mode decision towards INTRA coding.

Similar to the Error Tracking approach, the Reference Picture Selection (RPS) mode of H.263+ also relies upon a feedback channel to efficiently stop error propagation after transmission errors. This mode is described in Annex N of H.263+, and is based on the NEWPRED approach that was suggested in [14]. A similar proposal to NEWPRED has been submitted to the MPEG-4 standardization group [15]. Instead of using the INTRA coding of macroblocks, the RPS mode allows the encoder to select one of several previously decoded frames as a reference picture for prediction. In order to stop error propagation while maintaining the best coding efficiency, the available feedback information can be used to select the most recent error-free frame.

Note that also erroneous frames could be used for prediction, if the concealment strategy at the decoder were standardized. In this case, the encoder could exactly reconstruct the erroneous reference frames at the decoder based on ACK and NACK information. Because of the lack of a standardized concealment strategy and the involved increase in complexity, this approach is not considered in the description of Annex N. Instead, it is assumed that only error-free frames are selected as a reference. However, for very noisy transmission channels, it can be difficult to transmit complete frames without any errors. In this case, the most recent error-free frame can be very old and hence ineffective for MCP. Therefore, the Independent Segment Decoding (ISD) mode as described in Annex R of H.263 has been specified. The ISD mode was suggested in [16]. In the ISD mode, the video sequence is partitioned into sub-videos that can be decoded independently from each other. A popular choice is to use a group of blocks (GOB) as a sub-video. In a QCIF frame, a GOB usually consists of a row of 11 macroblocks [2]. The ISD mode significantly reduces the coding efficiency of motion compensation, particularly for vertical motion, since image content outside the current GOB must not be used for prediction. In this paper, we will not use the ISD mode. Rather, we specify a simple concealment algorithm that is known to encoder and decoder and incorporate it into the encoding algorithms.

Reference Picture Selection can be operated in two different modes, ACK and NACK mode. In the ACK mode case, correctly received image content is acknowledged and the encoder only uses acknowledged image content as a reference. If the round trip delay is greater than the encoded picture interval, the encoder has to use a reference frame further back in time. This results in decreased coding performance for error-free transmission. In the case of transmission errors, however, only small fluctuations in picture quality occur. The second mode is called NACK mode. In this mode only erroneously received image content is signaled by sending negative acknowledgments. During error-free transmission, the operation of the encoder is not altered and the previously decoded image content is used as a reference. Both modes can also be combined to obtain increased performance as demonstrated in [17], [18].

In [19], multiple reference frames have been proposed for increasing the robustness of video codecs. Error propagation is modeled using a Markov approach which is used to modify the selection of the picture reference parameter using a strategy called random lag selection. However, the modifications to the coder control are heuristic. Moreover, the actual concealment distortion, the motion vector estimation and the macroblock mode decision are not considered. In this paper, we propose rate-constrained long-term memory prediction for efficient transmission of coded video over noisy channels. For that, the coder control takes into account the rate-distortion trade-off achievable for the video sequence given the decoder as well as the transmission errors introduced by the channel. We present a framework that unifies concepts such as Error Tracking, ACK, and NACK mode. Furthermore, the proposed framework can also be used to increase the robustness against transmission errors when no feedback is available. This paper is organized as follows. In Section II, the video codec is described. Given the constraints of the video codec, the proposed coder control is described in Section III. Section IV presents experimental results that evaluate the new approach.

II. The Video Codec

The video codec employed in this paper is based on the H.263 standard. Our motivation for that choice is (*i*) the algorithm is well defined [2], (*ii*) the test model of the H.263 standard, TMN-10, can be used as a reference for comparison, and (*iii*) the H.263 Recommendation specifies a state-of-the-art video coding algorithm. In the following, we will describe the extensions made to the H.263+ video codec.

A. Long-Term Memory Motion-Compensated Prediction

Long-term memory MCP extends the motion vector utilized in hybrid video coding by a variable time delay permitting the use of several decoded frames instead of only the previously decoded one for block-based motion compensation. The frames inside the long-term memory which is simultaneously built at encoder and decoder are addressed by a combination of the codes for the spatial displacement vector and the variable time delay. Hence, the transmission of the variable time delay potentially increases the bit-rate which has to be justified by improved MCP. This trade-off limits the efficiency of the proposed approach.

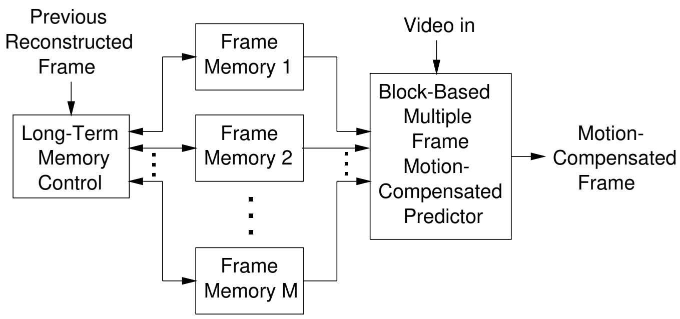


Fig. 1. Long-Term Memory Motion-Compensated Predictor.

The architecture of the long-term memory predictor is depicted in Fig. 1. This figure shows an inter-frame predictor with several frame memories (M, $M \geq 1$) that are arranged using the memory control. The memory control may operate in several modes. In this paper, a sliding time window is accommodated by the memory control unit as depicted in Fig. 1. For that, past decoded and reconstructed frames starting with the immediately preceding one and ending with the frame which has been decoded M frame intervals before are collected in the frame memories 1 to M.

Incorporating long-term memory prediction into an H.263-based video codec requires changes to the syntax. The H.263 syntax is modified by extending the inter-prediction macroblock modes in order to enable multi-frame MCP. More precisely, the inter-prediction macroblock modes INTER and UNCODED are extended by one code word representing the picture reference parameter for the entire macroblock. The INTER-4V macroblock mode utilizes four picture reference parameters each associated with one of the four 8×8 block motion vectors.

The variable length code (VLC) table for the picture reference parameter is specified in Tab. I. The VLC in Tab. I is a regularly constructed reversible table. The binary number "x_0x_1..." in the first column plus the added decimal number denote the picture reference parameter which is given in brackets. The code is constructed such that the binary number is interleaved with bits that indicate if the code continues or ends. For example, 1 bit is assigned to the picture reference parameter with index 0, 3 bits to the picture reference parameters with indexes 1 and 2 etc.

TABLE I

VLCs for picture reference parameter. The x_i are binary numbers.

Picture Reference Parameter Δ	Number of Bits	Code
0 (0)	1	1
"x_0"+1 (1,2)	3	$0x_00$
"x_0x_1"+3 (3...6)	5	$0x_11x_00$
"$x_0x_1x_2$"+7 (7...14)	7	$0x_21x_11x_00$
$\vdots$	$\vdots$	$\vdots$

B. Resynchronization after Errors

Because the multiplexed video bit-stream consists of VLC words, a single bit error may cause a loss of synchronization and a series of erroneous code words at the decoder. The common solution to this problem is to insert unique synchronization code words into the bit-stream in regular intervals, usually followed by a block of "header" bits. The H.263 standard supports optional GOB-headers as resynchronization points which are also used throughout this paper. As mentioned above, a GOB in QCIF format usually consists of 11 macroblocks that are arranged in one row. Because all information within a correctly decoded GOB can be used independently from previous information in the same frame, the GOB is often used as the basic unit for decoding. Hence, if a transmission error is detected, the GOB is discarded entirely.

C. Error Concealment

The severeness of the error caused by discarded GOBs can be reduced if error concealment techniques are employed to hide visible distortion as well as possible. In our simulation environment, we employ the simple and most common approach called *previous frame concealment*, i.e., the corrupted image content is replaced by corresponding pixels from the previous frame. This is conducted by setting the macroblocks in the discarded GOB to the UNCODED mode. The concealment scheme can be applied simultaneously at decoder and encoder yielding the same result at both ends. This simple approach yields good concealment results for sequences with little motion [20]. However, severe distortions may be introduced for image regions containing heavy motion.

III. Coder Control

Given the decoder as described in the previous section, the task of the coder control is to determine the coding parameters that generate a bit-stream which optimizes reconstruction quality at the decoder. The reconstruction quality in this paper is measured as mean squared error. For that, the coder control has to take into account the rate-distortion trade-off achievable for the video sequence given the decoder as well as the transmission errors introduced by the channel. Due to the probabilistic nature of the channel, one has to consider expected distortion measures. For presentation purposes, the results for several channel realizations should be averaged.

A. Rate-Constrained Coder Control

The coder control employed for the proposed scheme mainly follows the specifications of TMN-10 [21], the test model for the H.263 standard specifying a recommended coder control. TMN-10 has been proposed to the ITU-T in [22]. We use TMN-10 because the optimization method in [22] has been proven efficient while requiring a reasonable amount of computational complexity. A further motivation is the use of the TMN-10 coder as an anchor for comparisons. In the following, we briefly describe the TMN-10 scheme and explain the extensions to long-term memory motion-compensated prediction.

The problem of optimum bit allocation to the motion vectors and the residual coding in any hybrid video coder is a non-separable problem requiring a high amount of computation. To circumvent this joint optimization, we split the problem into two parts: motion estimation and mode decision. Motion estimation determines the motion vector and the picture reference parameter to provide the motion-compensated signal. Mode decision determines the use of the macroblock mode which includes the MCP parameters, the DCT coefficients and coder control information. Motion estimation and mode decision are conducted for each macroblock given the decisions made for past macroblocks.

Our block-based motion estimation proceeds over all reference frames in the long-term memory buffer. For each block, a Lagrangian cost function is minimized [23], [24] that is given by

$$D_{DFD}(\mathbf{v}, \Delta) + \lambda_{MOTION} R_{MOTION}(\mathbf{v}, \Delta), \qquad (1)$$

where the distortion of the displaced frame difference (DFD) is measured as the sum of the squared differences (SSD)

$$D_{DFD} = \sum_{x,y \in \mathcal{B}} (o[x,y] - s_{\Delta}[x+v_x, y+v_y])^2, \qquad (2)$$

for all pixels in a block $\mathcal{B}$ in the original frame o and the reconstructed frame s_{Δ} that was decoded Δ frame intervals in the past. The rate term R_{MOTION} is associated with the motion vector $\mathbf{v}$ and the picture reference parameter Δ. The motion vector $\mathbf{v}$ is entropy-coded according to the H.263 specification and the picture reference Δ is signaled using Tab. I. The motion search covers all frames and a range of ± 16 pixels horizontally and vertically.

Given the motion vectors and picture reference parameters, the macroblock modes are chosen. Again, we employ a rate-constrained decision scheme where a Lagrangian cost function is minimized for each macroblock [25]

$$D_{REC}(h, \mathbf{v}, \Delta, c) + \lambda_{MODE} R_{REC}(h, \mathbf{v}, \Delta, c). \qquad (3)$$

Here, the distortion after reconstruction D_{REC} measured as SSD is weighted against bit-rate R_{REC} using the Lagrange multiplier λ_{MODE}. The corresponding rate term is given by the total bit-rate R_{REC} that is needed to transmit and reconstruct a particular macroblock mode, including the macroblock header h, motion information including $\mathbf{v}$ and Δ as well as DCT coefficients c. The mode decision determines whether to code each macroblock using the H.263 modes INTER, UNCODED, INTER-4V, and INTRA [2].

Following [26], the Lagrange multiplier for the mode decision is chosen as

$$\lambda_{MODE} = 0.85 \cdot Q^2, \qquad (4)$$

with Q being the DCT quantizer value, i.e., half the quantizer step size [2]. The Lagrange multiplier used in the motion estimation is chosen as $\lambda_{MOTION} = \lambda_{MODE}$.

B. Incorporating Inter-Frame Error Propagation

When an error occurs, complete GOBs are lost and concealed using previous frame concealment as described above. Hence, the reconstructed frames at encoder and decoder differ. Referencing this image content for MCP leads to inter-frame error propagation.

A common approach to stop inter-frame error propagation is to send INTRA macroblocks. However, INTRA coding of picture content is usually less efficient than INTER coding. Hence, the number of INTRA macroblocks must be balanced against the amount of propagating errors. Another important issue in this context is how the errors are spread via motion compensation. By motion-compensating erroneous image content as prediction into the current frame, the error energy introduced is copied and filtered.

We model the error that may occur with probability p using an error tracking approach similar to the one proposed in [4], [5]. In addition to the scheme in [4], where the macroblock mode decision between INTRA and INTER is controlled, we also utilize error tracking to modify motion estimation. Also, error tracking is applied on a pixel basis in contrast to [4] where only entire macroblocks are considered. Another difference is that we also consider the case where we do not know where an error occurred, i.e., when no feedback is available. Therefore, we have to consider the various combinations of possible error events.

For simplicity, let us assume that each frame is transmitted as one packet and each packet is lost with probability p or correctly received with probability $q = 1 - p$. Further, assume inter-frame coding where the current frame references the immediately preceding frame ($M = 1$). Hence, if the immediately preceding frame is lost, an error is introduced that propagates to the current frame. Figure 2 illustrates the combination of possible error events.

Let us assume, that we are currently coding the frame at time instant k that references frame $k-1$. We want to estimate the average errors that have accumulated in frame $k-1$ to incorporate these measures into the coder control. For that we also have to consider older frames than frame $k-1$ due to inter-frame error propagation. For the sake of simplicity, we assume that the frame at time instant $k-4$ is correctly decoded. In the next frame at time

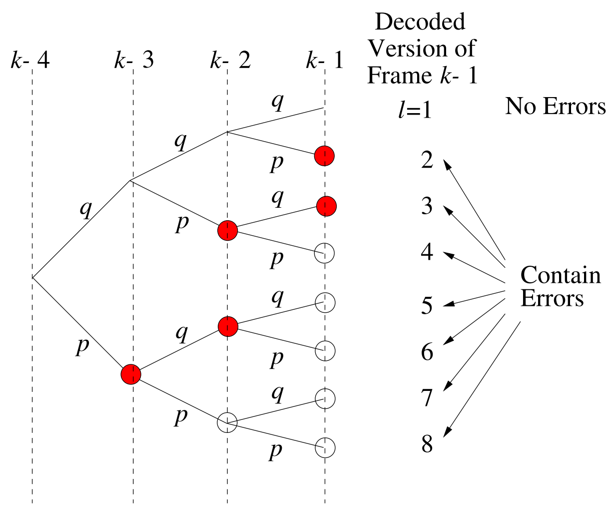


Fig. 2. Binary tree of possible error events. Each node of the tree corresponds to a decoded version of a video frame. The nodes labeled with a circle are those that contain transmission errors. The shaded circles correspond to the error cases considered.

instant $k-3$, reference is made to frame $k-4$ using motion compensation. The image content at time instant $k-3$ is either decoded erroneously with probability p or correctly decoded with probability $q = 1-p$. Hence, the two nodes at time instant $k-3$ correspond to two decoded versions of that video frame. The decoding of image content in the frame at time instant $k-2$ which references frame $k-3$ results in 4 combinations of possible outcomes while image content in the frame at time instant $k-1$ can be decoded in 8 different ways. It is easy to conclude, that any succeeding frame doubles the number of possibilities of the decoding result. Hence, modeling all these branches of the event tree would very quickly be intractable since $L = 2^k$ combinations would have to be computed for a frame that is k time instants decoded after the first frame.

If long-term memory prediction is utilized, the number of branches leaving a node varies since frames other than just the prior decoded frame can also be referenced. Moreover, time instants do not correspond to levels of the tree anymore. On the other hand, if the transmission scheme packetizes a frame into more than one unit that can be correctly received or lost, the tree structure also becomes more complicated in that the number of branches leaving a node also increases.

The distortion that we expect by INTER coding is given by the weighted summation of the distortion between the original frame and each reconstructed version of the reference frame. More precisely, assume that at time instant k a number of L different outcomes of decoding are possible. The expected distortion for a motion vector $\mathbf{v} = (v_x, v_y)$ and picture reference parameter Δ is computed as

$$\begin{aligned} D &= \sum_{l=1}^{L} p_l D_l \\ &= \sum_{l=1}^{L} p_l \sum_{x,y\in\mathcal{B}} (o[x,y] - s_\Delta^l[x+v_x, y+v_y])^2, \end{aligned} \quad (5)$$

with $\mathcal{B}$ being the set of pixels considered, s_Δ^l being the lth version of the reconstructed video signal, and p_l being the probability that version s_Δ^l is the actual decoded one. Note that we have to independently evaluate L distortion terms for each point in the search area. This is computationally very demanding especially when used for motion estimation. Hence, we express the reference frame in the lth decoding branch using the correctly decoded reference frame $s_\Delta[x,y] = s_\Delta^1[x,y]$ plus the remaining error $\varepsilon_\Delta^l[x,y]$,

$$s_\Delta^l[x,y] = s_\Delta[x,y] + \varepsilon_\Delta^l[x,y], \quad (6)$$

The distortion term D_l is approximated by

$$\begin{aligned} D_l &= \sum_{x,y\in\mathcal{B}} (o[x,y] - s_\Delta^l[x+v_x, y+v_y])^2 \\ &= \sum_{x,y\in\mathcal{B}} (o[x,y] - s_\Delta[x+v_x, y+v_y] - \varepsilon_\Delta^l[x+v_x, y+v_y])^2 \\ &\approx \sum_{x,y\in\mathcal{B}} (o[x,y] - s_\Delta[x+v_x, y+v_y])^2 + (\varepsilon_\Delta^l[x+v_x, y+v_y])^2 \end{aligned} \quad (7)$$

where we neglect the cross terms $o[x,y] \cdot \varepsilon_\Delta^l[x,y]$ and $s_\Delta[x,y] \cdot \varepsilon_\Delta^l[x,y]$ since we assume $o[x,y]$ and $s_\Delta[x,y]$ to be uncorrelated from $\varepsilon_\Delta^l[x,y]$ and $\varepsilon_\Delta^l[x,y]$ to have zero mean value. The overall distortion term is modified to

$$\begin{aligned} D = & \sum_{x,y\in\mathcal{B}} (o[x,y] - s_\Delta[x+v_x, y+v_y])^2 \\ & + \sum_{l=2}^{L} p_l \sum_{x,y\in\mathcal{B}} (\varepsilon_\Delta^l[x+v_x, y+v_y])^2 \end{aligned} \quad (8)$$

Note that the first term corresponds to the distortion term usually computed in motion estimation routines (D_{DFD}). The second term represents the error energy caused by transmission errors. The values of the second term can be efficiently precomputed utilizing an algorithm similar to the one proposed in [27]. Nevertheless, the computational burden is still very high because of the large number of combinations involved. Hence, we restrict the number of possibilities of errors to two cases

1 the referenced image content is in error and concealed, (branch $l = 2$ in Fig. 2)
2 the referenced image content has been correctly decoded but references concealed image content (branch $l = 3$ in Fig. 2).

In Fig. 2, each node of the tree corresponds to a decoded version of a video frame. The nodes labeled with a circle are those that contain transmission errors. Our approximation incorporates only those cases with shaded circles. This approximation is justified by assuming p to be very small and two error events in a row to be very unlikely. Other decoded versions s_l can be neglected if an error has occurred several frames in the past and is then several times motion-compensated. Here, we assume that the error is filtered and somewhat reduced. Nevertheless, our assumptions may not hold for some cases. We will elaborate on possible shortcomings of this approximation in the section on experimental results when comparing to more accurate error modeling.

In this work, the error modeling is incorporated into motion estimation and mode decision as follows. The Lagrangian cost term of the minimization routine for motion estimation is modified in that another distortion term is added that incorporates the energy in the case of transmission errors. Hence, (1) is modified to

$$D_{DFD}(\mathbf{v},\Delta)+\kappa D_{ERR}(\mathbf{v},\Delta)+\lambda_{MOTION}R_{MOTION}(\mathbf{v},\Delta), \quad (9)$$

with $D_{ERR}(\mathbf{v},\Delta)$ being

$$D_{ERR}(\mathbf{v},\Delta)=\sum_{l=2}^{3}\sum_{x,y\in\mathcal{B}}(\varepsilon^{l}_{\Delta}[x+v_x,y+v_y])^2 \quad (10)$$

and κ being a weighting term. This weighting term is used as a free parameter in the simulation discussed below and is necessary because of the following reasons. First, it provides a means to adapt to the given channel conditions. Note that in contrast to (8), no error probability is included in (9). Hence, κ is used to scale the estimated error energy D_{ERR} according to the effective loss probability. Second, it is actually necessary to consider not only the error that is introduced in the current frame but also the propagation of errors in future frames. Both effects are difficult to capture and we therefore vary κ in an appropriate range. For a practical system it would be necessary to set κ correctly during encoding. Because we are mainly interested in performance bounds, we use κ as a free parameter and pick the value that results in optimal overall performance (as given in maximum decoder PSNR in this paper). Also note that κ actually would have to be adapted on a macroblock basis for optimum performance. For simplicity, however, we use a fixed value of κ for a given sequence and channel.

The Lagrangian costs for the INTER modes in (3) are modified to

$$D_{REC}(h,\mathbf{v},\Delta,c)+\kappa D_{ERR}(\mathbf{v},\Delta)+\lambda_{MODE}R_{REC}(h,\mathbf{v},\Delta,c). \quad (11)$$

while the Lagrangian costs for INTRA modes remain unchanged as in (3). Note that INTRA coding terminates branches of the tree, since in the case of correctly decoded image content, inter-frame error propagation is stopped. However, the bits for the INTRA code itself could be in error. Therefore, one impact of the error modeling when incorporated into an H.263 and long-term memory codec is that the number of macroblocks coded in INTRA should be increased.

For the long-term memory codec, the frame selection is also affected since the error modeling is incorporated into motion estimation. This effect is very strong in the case of feedback, since if a feedback message is received for a frame transmitted, the exact decoded version of that feedback frame can be reconstructed at the encoder. Note that feedback is provided about correct as well as erroneous macroblocks (ACK+NACK). In this work, we assume that feedback messages are transmitted without error and that the exact concealment method is known to encoder and decoder. Using the exact reconstruction of the decoded frame at the encoder, the succeeding frames are decoded and the inter-frame error propagation, in the case of an error, is therefore tracked exactly. Note that this decoding is only necessary for concealed image content and macroblocks referencing those concealed pixels. A similar idea has been exploited in [28]. The error modeling is updated in that, $D_{ERR}(\mathbf{v},\Delta)$ is set to 0 for the feedback frame and an update is made for all depending frames in those parts of the image affected by a propagated error.

In order to transmit the picture reference parameters with the smallest possible average bit-rate given the VLC in Tab. I, the picture reference parameters are sorted in descending order of their frequency. The result of the sorting is then transmitted to the decoder by sending the number of the picture reference parameters in descending order of their frequency. However, the bit-rate needed for this transmission may be prohibitive when a large number of reference frames is used. Hence, only the most likely K picture reference parameters are transmitted. The remaining picture reference parameters are left in their original order and are appended to the K specified parameters. Typically, the number K is chosen as 3.

IV. Experiments

Before presenting results for the proposed framework, we describe the simulation environment that is used for their evaluation. We follow the basic block diagram of a video transmission system as illustrated in Fig. 3 and describe the individual parts. For the channel model, modulation

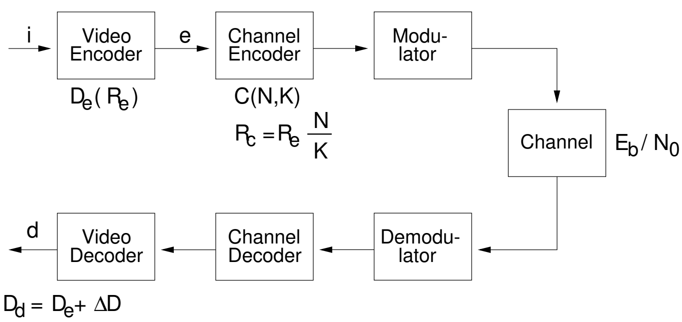


Fig. 3. Basic components of a video transmission system.

scheme, and channel codec, we use standard components rather than advanced techniques that reflect the current state of research. This is justified by our focus on video coding and by the fact that the selected standard components are well suited to illustrate the basic problems and trade-offs. Therefore, the described scenario should be considered as an example that is used for illustration, rather than a proposal for an optimized transmission scheme.

A. Channel Model and Modulation

Our simulations are based on bit error sequences that are used within ITU-T Study Group 16 for the evaluation of current and future error resilience techniques in H.263. The sequences are generated assuming Rayleigh fading with different amounts of channel interference, characterized by ratios of bit-energy to noise-spectral-energy (E_b/N_0) in the range of 14 to 30 dB.

The correlation of fading amplitudes is modeled according to the widely accepted model by Jakes [29]. In this model, the correlation depends significantly on the Doppler frequency f_D, which is equal to the mobile velocity divided by the carrier wavelength. For a given carrier frequency, the correlation increases with decreasing mobile velocity, such that slowly moving terminals encounter longer burst errors. For more information on this very common channel model, see [29], [30].

The modulation scheme and relevant parameters such as carrier frequency and modulation interval are roughly related to the ETSI standard DECT (Digital Enhanced Cordless Telecommunications). Though DECT is originally intended for cordless telephony, it provides a wide range of services for cordless personal communications which makes it very attractive for mobile multimedia applications [31] [32]. Similar to DECT, we use BPSK for modulation and a carrier frequency of $f_c = 1900$ MHz. For moderate speeds a typical Doppler frequency is $f_D = 62$ Hz, which will be used throughout the simulations in the remainder of this paper. According to the double slot format of DECT, we assume a total bit-rate of $R_c = 80$ kbps that is available for both source and channel coding. For simplicity we do not assume any TDMA structure and use a symbol interval of $T_s = 1/80$ ms. The resulting bit error sequences exhibit severe burst errors that limit the effective use of forward error correction. Therefore, even at low channel code rates, residual errors cannot be avoided completely by the channel codec and have to be processed by the video decoder.

B. Channel Codec

For channel coding we use a forward error correction (FEC) scheme that is based on Reed-Solomon (RS) codes [33]. For symbols composed of m bits, the encoder for an RS(N,K) code groups the incoming data stream into blocks of K information symbols (Km bits) and appends $N - K$ parity symbols to each block. Hence, the transmitted output block contains N symbols and each block is treated independently by the channel codec.

The bit allocation between source and channel coding can be described by the *code rate* r, which is defined as $r = K/N$. For RS codes operating on m-bit symbols, the maximum block length is $N_{max} = 2^m - 1$. By using *shortened* RS codes, any smaller value for N can be selected, which provides a great flexibility in system design. As RS codes operating on 8-bit symbols are very popular and powerful, we use $m = 8$ in our simulations and a packet size of $N = 88$ byte.

An RS(N,K) decoder can correct any pattern of bit errors resulting in less than $E < (N-K)/2$ symbols in error. In other words, for every two additional parity symbols, an additional symbol error can be corrected. If more than E symbol errors are contained in a block, the RS decoder fails and indicates an erroneous block to the video decoder. The probability that a block cannot be corrected is usually described by the *residual word error rate* (RWER). In general, the RWER decreases with increasing K and/or with increasing E_b/N_0. For simplicity, we ignore the occurrence of undetected errors, whose probabilities are usually very small compared to the RWER. This is also justified by the fact, that the video decoder itself usually has some error detection capability due to syntax violations that can be exploited.

For the described channel code, modulation scheme, and channel model, this relationship is summarized in Fig. 4 which illustrates the RWER for the values of E_b/N_0 and K that are used in the simulations. As can be seen, the

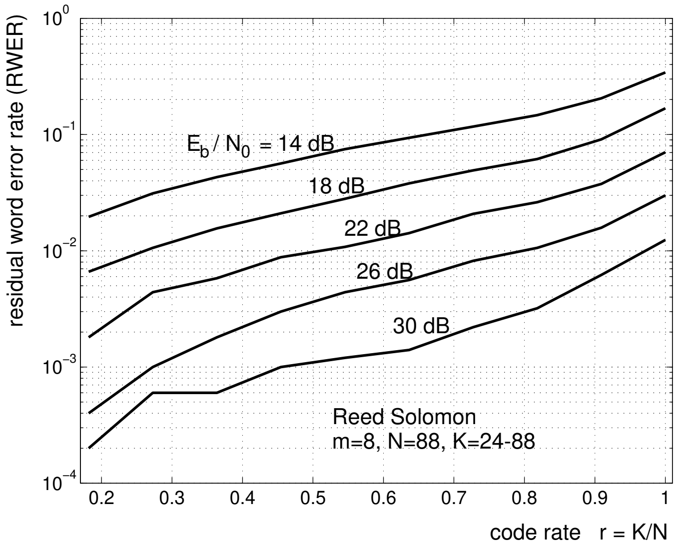
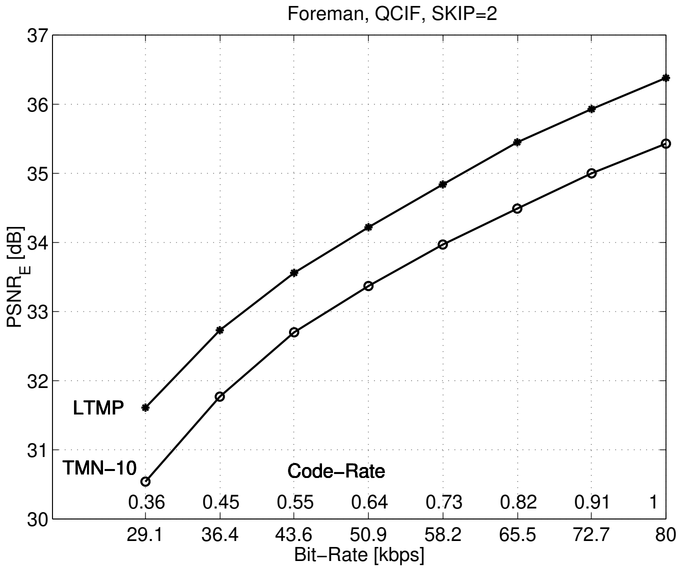


Fig. 4. Residual word error rate (RWER) after channel coding for a Rayleigh fading channel (Doppler frequency $f_D = 2.5$ Hz) with Gaussian Frequency Shift Keying.

RWER for a given value of E_b/N_0 can be reduced by approximately one order of magnitude by varying the code rate in the illustrated range. Though we will see that this reduction is already very helpful for video transmission, the observed gain in RWER is actually very moderate due to the bursty nature of the wireless channel. For channels without memory, such as the additive white Gaussian noise (AWGN) channel, the same reduction in r would provide a significantly higher reduction in RWER. For the AWGN channel it is possible to achieve very high reliability (RWER $< 10^{-6}$) with very little parity-check information and resilience techniques in the video codec would hardly be necessary [12]. For the mobile fading channel, however, the effective use of FEC is limited and the use of error resilience techniques in the source codec is very important.

By increasing the redundancy of the channel code, the available bit-rate for the source coder is reduced. Figure 5 shows rate distortion plots obtained from coding experiments with the QCIF sequences *Foreman* as well as *Mother and Daughter*.

Both coders are run with a rate-control enforcing a fixed number of bits per frame when coding 210 frames of video sampled with 8.33 frames/s. The first 10 frames were ex-



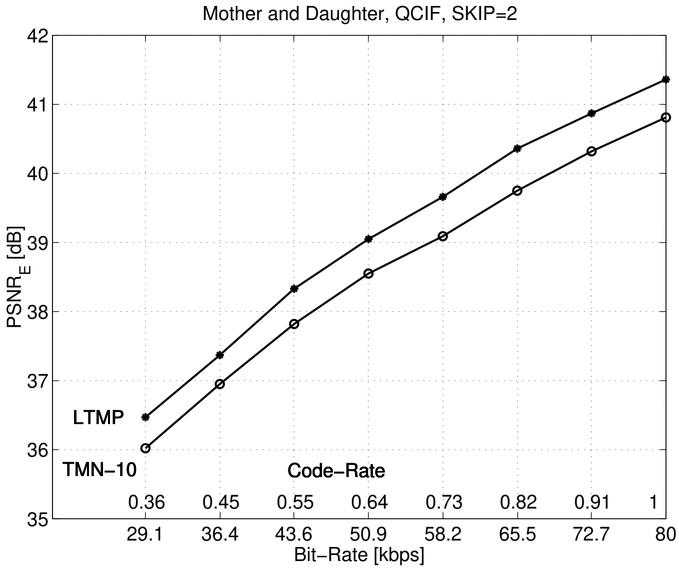


Fig. 5. Average PSNR vs. average bit-rate or code-rate for the sequences *Foreman* (top) and *Mother and Daughter* (bottom). The two curves relate to the two codecs compared: (*i*) TMN-10, the test model of the H.263 standard with Annexes D, F, I, J, T enabled; (*ii*) LTMP: the long-term memory prediction coder with 10 frames also utilizing Annexes D, F, I, J, T.

cluded from the measurements to avoid the transition phase at the beginning of the sequence since we employ long-term memory prediction with 10 reference frames. The two plots in Fig. 5 illustrate various aspects. The PSNR values differ about 5 dB comparing the lowest bit-rate point to the highest bit-rate for both sequences and both codecs. Further, the level of the PSNR values is about 5 dB higher for the sequence *Mother and Daughter* compared to the sequence *Foreman*. This is because, the *Foreman* sequence shows much more motion and high frequency content than the sequence *Mother and Daughter*. The two sequences are chosen as test sequences throughout the paper because we consider them extreme cases in the spectrum of low bit-rate video applications. Finally, the improved coding performance of long-term memory prediction is demonstrated. The two curves in each of the two plots compare

- **TMN-10**: the test model of the H.263 standard with Annexes D, F, I, J, T enabled
- **LTMP**: the long-term memory motion-compensated prediction coder with 10 frames also utilizing Annexes D, F, I, J, T.

The bit-rate savings obtained by the long-term memory codec against TMN-10 are 18 % for the sequence *Foreman* when measuring at equal PSNR of 34 dB and 12 % for the sequence *Mother and Daughter* when measuring at 39 dB. Although not shown here, extending the long-term memory to 50 frames yields a 50 % increase in bit-rate savings [1].

C. *Experimental Results without Feedback*

The first set of simulation results will be presented for the case when there is no feedback available. For that, we compare the TMN-10 coder with the LTMP coder when employing the error modeling approach and various code-rates.

In Fig. 6, the average PSNR measured at the encoder ($\mathrm{PSNR_E}$) is depicted versus various code-rates for the sequence *Foreman*. The upper plot corresponds to TMN-10 while the lower one shows results from the LTMP coder. Both coders are operated under similar conditions as for Fig. 5. The various curves correspond to values of κ, the weight of the modeled distortion in case of an error. The case $\kappa = 0$ is the same as for the curves plotted in Fig. 5. Comparing to this case, a significant degradation in terms of coding efficiency can be observed with increasing values of κ for both coders. This performance loss is explained by the additional cost term in (9) and (11) resulting in increased amounts of INTRA-coded macroblocks and modified motion vectors and, for the LTMP coder, picture reference parameters.

Figure 7 shows the corresponding average PSNR values measured at the decoder ($\mathrm{PSNR_D}$). Each bit-stream is transmitted to the decoder via the error-prone channel. This experiment is repeated 30 times using shifted versions of the bit error sequence that corresponds to the fading channel generated with noise-spectral-energy $E_b/N_0 = 22$ dB under the conditions described above. Again, the upper plot shows TMN-10 results while the lower one depicts results from the LTMP codec both incorporating error modeling. Obviously, the sacrifice at the encoder side pays off at the decoder side. In other words, the weighting factor κ can be used to trade-off coding efficiency and error resilience.

Although the optimum κ generally increases with the code-rate and hence with RWER, there is no direct relationship between κ and RWER. To some extend, this results from the fact that κ and RWER describe the loss of probability of different entities, i.e., MBs and blocks. Further, the described simplifications for the error modeling make it difficult to provide an exact mapping of RWER to κ. Nevertheless, such a mapping is important in practice

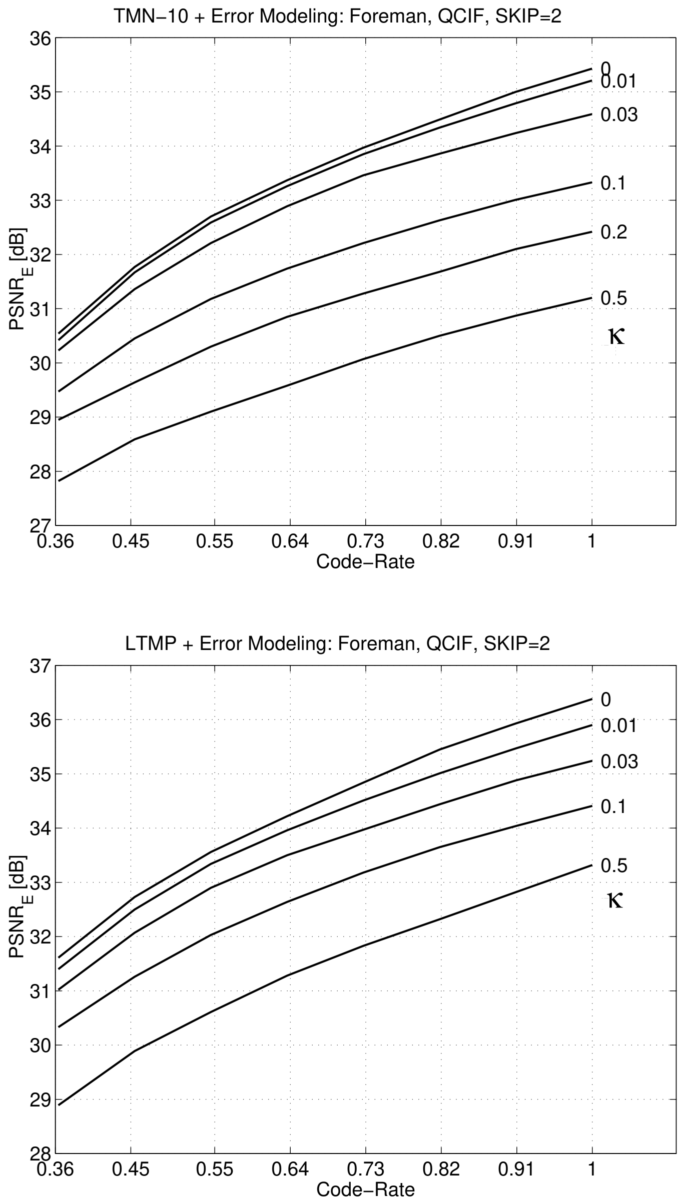


Fig. 6. Average encoder PSNR vs. code-rate for the sequence *Foreman* when running the TMN-10 coder (top) and the LTMP coder (bottom). By increasing κ, the estimated error energy is amplified resulting in reduced coding performance.

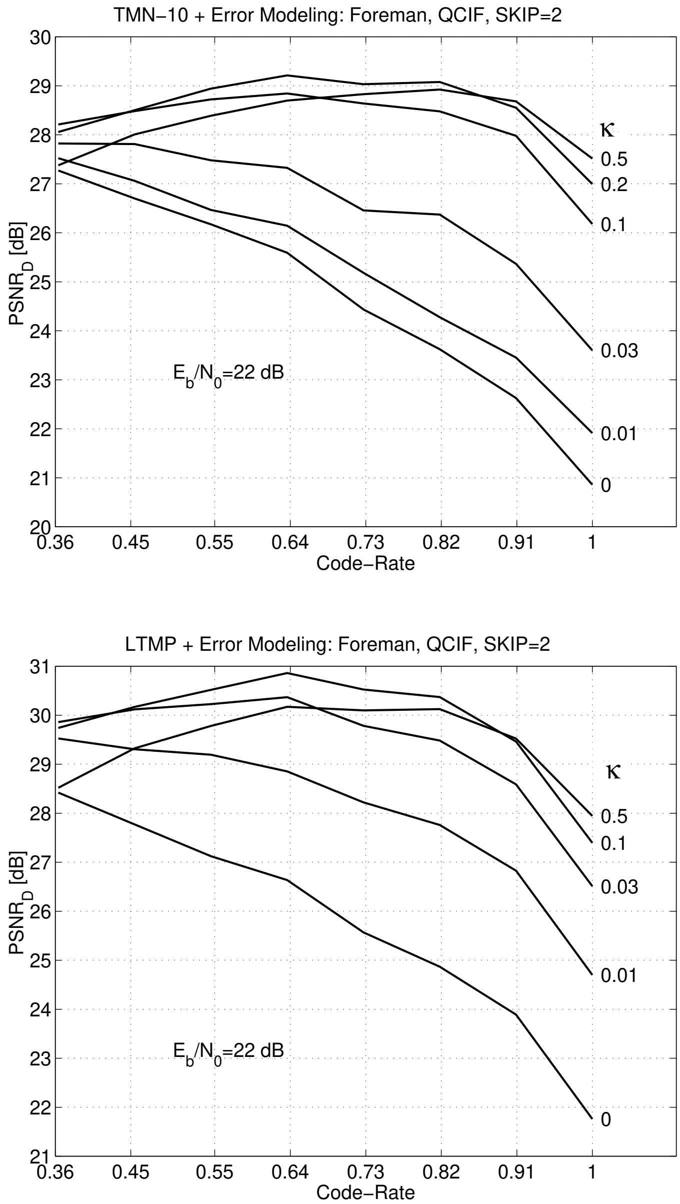


Fig. 7. Average decoder PSNR vs. code-rate for the sequence *Foreman* when running the TMN-10 coder (top) and the LTMP coder (bottom).

to operate the codec at the optimal point and is the subject of future research. On the other hand, code-rate and κ value can be traded off against each other over a wide range leading to a plateau of similar values of decoder PSNR for various selected pairs of code-rate and κ. This feature is especially important when there is no feedback available about the channel status.

The trade-off between code-rate and error modeling for various channel conditions is illustrated in Fig. 8. Average decoder PSNR is shown versus various levels of channel interference expressed by E_b/N_0. The plot is obtained by running the LTMP codec with a fixed value of $\kappa = 0.1$. The various curves relate to 8 code-rates that are equidistantly spaced in the range $32/88 \ldots 1$. Obviously, avoiding channel coding entirely is not advantageous if there are single bit errors as is the case in our simulations. The optimum coding redundancy, of course, depends on the quality of the channel. Nevertheless, the curves corresponding to medium code-rates are close to the maximum of the achievable PSNR values indicating that the choice of the code-rate is not that critical when combined with our error modeling approach. For example, a code rate of 0.55 provides reasonable performance for the whole range of E_b/N_0 as illustrated by the bold curve in Fig. 8.

In Fig. 9, we compare the best performance in terms of maximum decoder PSNR achievable when varying over code-rate as well as κ. For that, we have conducted several simulations to sample the parameter space.

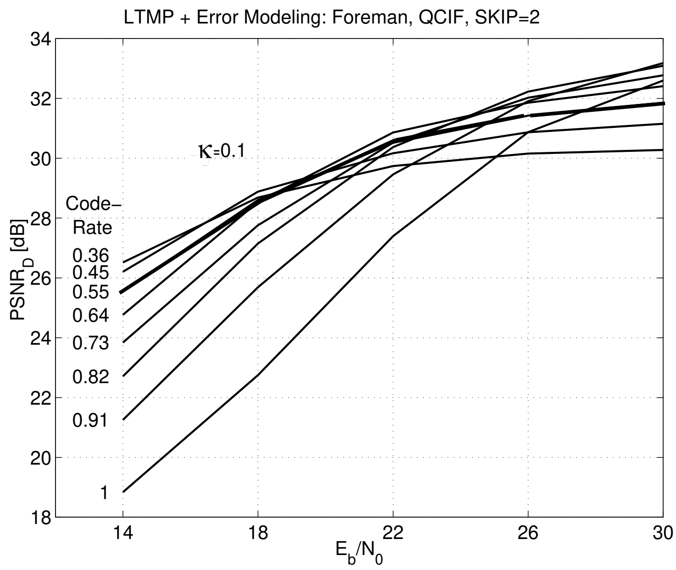


Fig. 8. Average decoder PSNR vs. E_b/N_0 for the sequence *Foreman* when running the LTMP coder for a fixed error weighting but various code-rates.

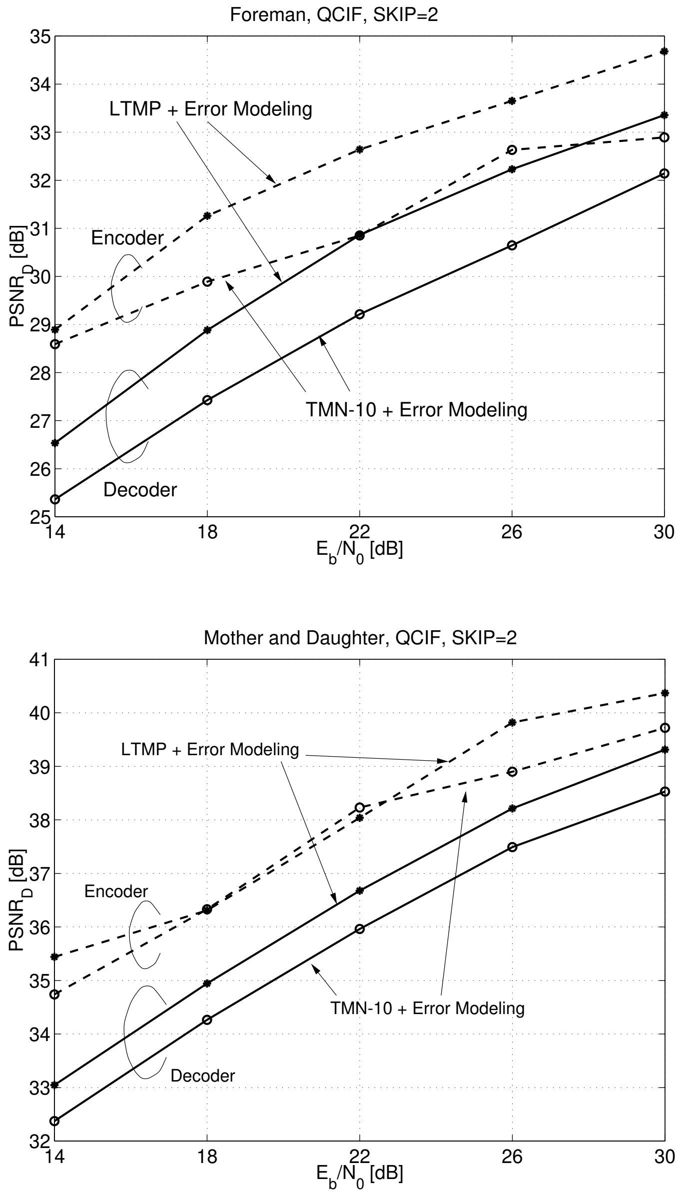


Fig. 9. Average decoder PSNR vs. E_b/N_0 for the sequences *Foreman* (top) and *Mother and Daughter* (bottom) for the optimal code-rate and error model parameter κ without feedback.

More precisely, the code-rate is varied over 8 values that are equidistantly spaced in the range $32/88\ldots1$. The error modeling weight κ is varied over values $\{0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.10, 0.20, 0.30, 0.40, 0.50\}$. For each of these 88 pairs of code-rate and κ, a bit-stream is encoded using the TMN-10 as well as the LTMP coder. Each bit-stream is transmitted to the decoder via error-prone channels. This experiment is repeated 30 times for each channel using shifted versions of the bit error sequences that correspond to $E_b/N_0 = \{14, 18, 22, 26, 30\}$. The dashed lines show encoder PSNR that corresponds to the maximum average PSNR measured at the decoder which is depicted with solid lines. Evaluating decoder PSNR, the LTMP coder outperforms the TMN-10 coder for all channel conditions. For example, the PSNR gain obtained for the sequence *Foreman* is 1.8 dB at $E_b/N_0 = 22$ dB. Correspondingly, a saving of 4 dB in terms of power efficiency is obtained.

Finally, we want to get an indication for the validity of our approximation for the divergence between encoder and decoder. As mentioned before, we only simulate a subset of the entire error event tree in Fig. 2. Hence, we only use an approximation of the expected divergence between encoder and decoder and therefore must obtain suboptimal results. Note that there has been a proposal for a recursive algorithm to model the expected divergence between encoder and decoder [34]. The recursive algorithm in [34] is accurate and has low computational complexity if there is no spatial filtering applied in the video codec. However, if there is spatial filtering as for half-pel accurate motion compensation [2], overlapped block motion compensation (Annex F of H.263) or deblocking filtering (Annex J of H.263) the algorithm has to be extended significantly increasing its complexity.

The main focus of this paper, however, is not the error modeling scheme. Rather, we want to present the error resilience characteristics of long-term memory prediction in contrast to single-frame prediction. A more accurate error modeling for our simulation scenario is given by the average divergence between encoder and decoder for those 30 simulations for which the performance evaluations are conducted. Note that this is a too optimistic anchor, since this scenario can not be realized in practice. Nevertheless, within our simulation framework, it provides us with a baseline to which we can compare. The more accurate error modeling is realized at the encoder by locally decoding the received bit-streams that are transmitted over the 30 channel realizations. Having the 30 decoding results available at the encoder, the squared difference to the correctly decoded picture is computed and averaged over all 30 cases. This squared difference is employed as $D_{ERR}(\mathbf{v}, \Delta)$ with $\kappa = 1$ in the optimization criteria for the motion estimation (9) and mode decision (11).

Figure 10 compares the loss of our approximate error modeling scheme to the more accurate error modeling as described above. The curves for the approximate error modeling are identical to those in Fig. 9. The more accurate error modeling provides improved performance in terms of decoder PSNR. Nevertheless, the maximum gains for optimum error modeling in comparison to our approximation are less than 1 dB for large E_b/N_0 values indicating the validity of our approximation.

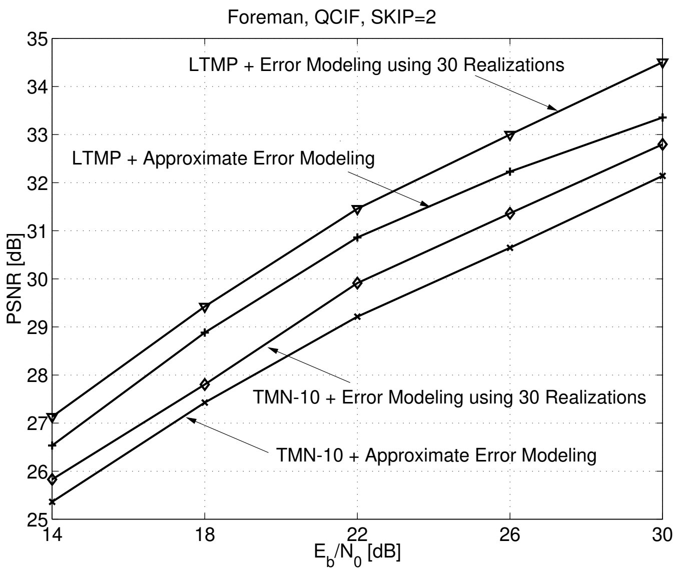


Fig. 10. Average decoder PSNR vs. E_b/N_0 for the sequence *Foreman* (top) when comparing the results of optimal code-rate and error model parameter κ for our approximate error modeling to more accurate error modeling.

D. Experimental Results with Feedback

In the following set of experiments, a feedback channel is utilized. Such a feedback channel indicates which parts of the bit-stream were received intact and/or which parts of the video signal could not be decoded and had to be concealed. The decoder sends a negative acknowledgment (NACK) for an erroneously received macroblock and a positive acknowledgment (ACK) for a correctly received macroblock. In our simulations, we assume that the feedback channel is error-free. The round trip delay is assumed to be approximately 250 ms, such that feedback is received 3 frames after their encoding.

In Fig. 11, we compare three feedback handling strategies for the sequence *Foreman*. The simulation conditions are similar to the previous results. The upper plot shows results obtained with the TMN-10 codec while the lower plot shows results form the LTMP codec. Note that the feedback handling depends on the video codec used.

For the TMN-10 codec, the three feedback schemes are realized as

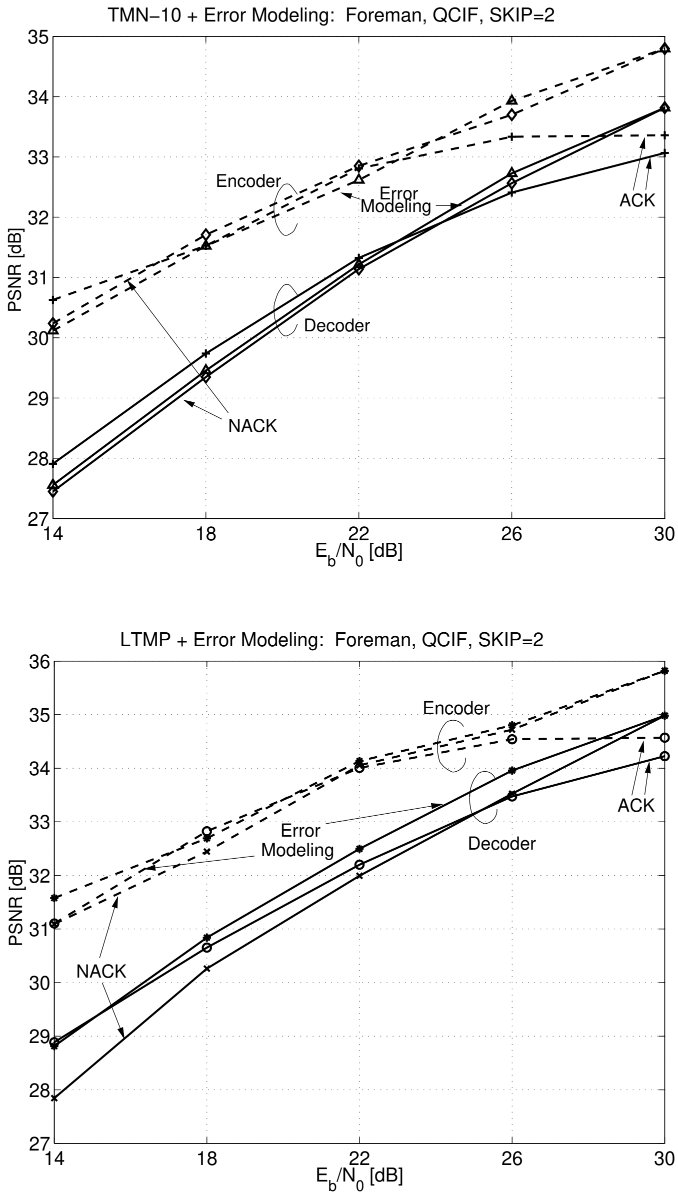


Fig. 11. Average PSNR vs. E_b/N_0 for the sequence *Foreman* running the TMN-10 coder (top) and the LTMP coder (bottom) when feedback is utilized. The various curves correspond to decoder and encoder PSNR for three different feedback handling strategies.

- **ACK mode**: MCP is conducted by referencing the most recent image for which feedback is available,
- **NACK mode**: MCP is conducted as usual ($\kappa = 0$) referencing the most recently decoded frame; only in case of an error indication via feedback, the image is referenced for which that feedback is received after error concealment,
- **Error Modeling**: as the NACK mode, but motion estimation and mode decision are modified by the error modeling term via setting $\kappa > 0$.

Again, the parameter space of code-rates and κ values is sampled so as to obtain maximum decoder PSNR for the various channel conditions given by $E_b/N_0 = 30, 26, 22, 18, 14$. The space of code-rates is sampled so as to show optimum performance. Evaluating decoder PSNR in the upper plot of Fig. 11, the differences between the three schemes are rather small. At $E_b/N_0 = 30$dB, the NACK mode and the error modeling work best, while at $E_b/N_0 = 26$dB the error modeling is slightly better. For lower E_b/N_0 values, the ACK mode outperforms the other two schemes.

For the LTMP codec, the three strategies are implemented as follows.

- **ACK mode**: MCP is conducted by referencing the 10 most recent images for which feedback is available,
- **NACK mode**: MCP is conducted by referencing the most recent 10 decoded frames ($\kappa = 0$), when an error is indicated via feedback from the decoder, the depending frames are decoded again after error concealment in the feedback frame,
- **Error Modeling**: as the NACK mode, but motion estimation and mode decision are modified by the error modeling term via setting $\kappa > 0$.

The remaining simulation conditions regarding κ and code-rates are the same as for the TMN-10 codec. Here, the differences in terms of decoder PSNR are more visible distinguishing the three concepts. The error modeling approach is superior or achieves similar performance comparing it to the ACK or NACK mode. This is because the ACK or NACK mode in the LTMP codec are special cases of the error modeling approach. The ACK mode is incorporated via large values of κ. Then, reference frames for which no feedback is available are completely avoided since for reference frames with feedback, the term D_{ERR} in (9) and (11) is set to 0. On the other hand, the NACK mode is incorporated by simply setting $\kappa = 0$. Hence, it is not surprising that in the case when ACK and NACK mode perform similarly "well" or rather "badly", the error modeling approach provides the largest benefit. This can be seen for the results obtained at $E_b/N_0 = 26$ dB. The error modeling approach releases the structural constraints of ACK and NACK mode and thus provides improved overall performance.

Finally, in Fig. 12, the gains achievable with the LTMP codec over the TMN-10 codec are depicted for the feedback case. We also show results for the case without feedback to illustrate the error mitigation by feedback. Figure 12 shows comparisons for the sequences *Foreman* (top) and *Mother and Daughter* (bottom). The decoder PSNR curves related to the case without feedback are the same as in Fig. 9. For the feedback case, the optimum performance points in terms of decoder PSNR are taken from Fig. 11 for the *Foreman* sequence. The points for the sequence *Mother and Daughter* are generated in a similar manner. For the *Foreman* sequence, the error mitigation by feedback in terms of average decoder PSNR is between 1.8 dB at $E_b/N_0 = 30$ dB and 2.5 dB at $E_b/N_0 = 14$. In the feedback case, the LTMP coder provides a PSNR gain of 1.2 dB compared to the TMN-10 coder. This decoder PSNR gain

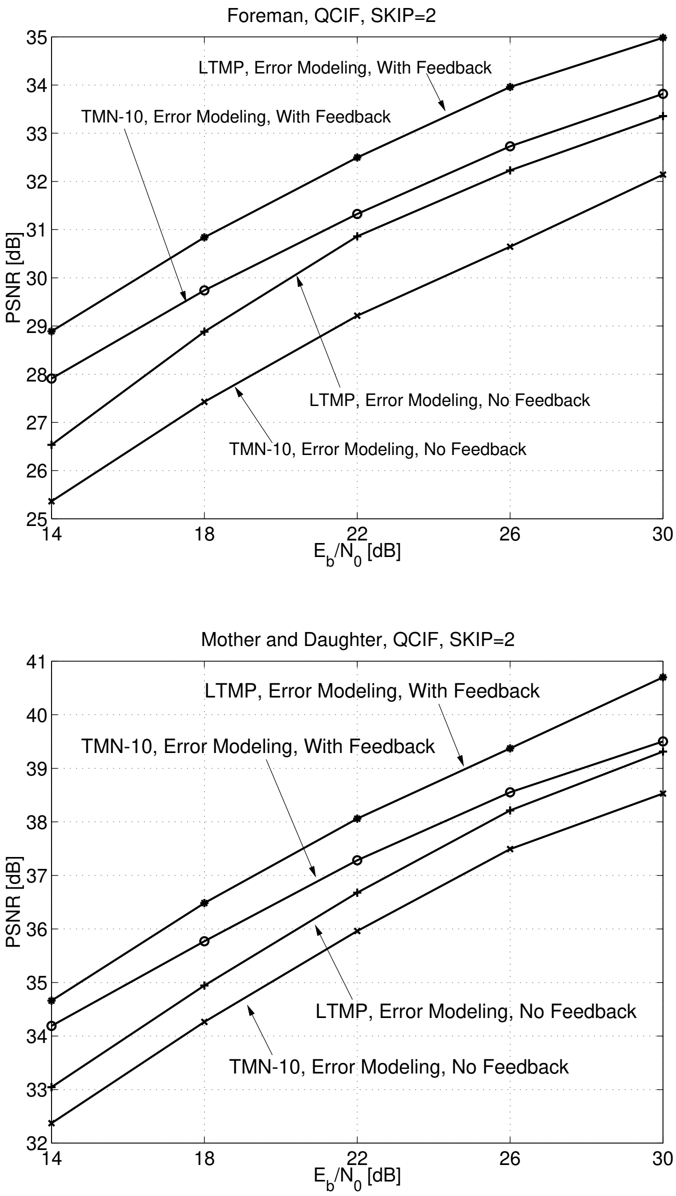


Fig. 12. Average PSNR vs. E_b/N_0 for the sequences *Foreman* (top) and *Mother and Daughter* (bottom) for the optimal feedback strategy and without feedback.

corresponds to a saving in terms of power efficiency between 3.8 dB at $E_b/N_0 = 30$ dB and 2.5 dB at $E_b/N_0 = 14$. The LTMP coder without feedback performs close to the TMN-10 coder with feedback for $E_b/N_0 \geq 22$ dB.

V. Concluding Remarks

In this paper we propose long-term memory prediction for efficient transmission of coded video over noisy channels. For that, the coder control takes into account the rate-distortion trade-off achievable for the video sequence given the decoder as well as the transmission errors introduced by the channel. Due to the probabilistic nature of the channel, one has to consider expected distortion measures. For simulation purposes, the results for several channel realizations are averaged. The proposed framework can be used to increase the robustness against transmission errors when no feedback is available. In experiments incorporating Rayleigh fading channels, the PSNR gain at the decoder obtained for the sequence *Foreman* is 1.8 dB at $E_b/N_0 = 22$ dB.

When a feedback channel is available, the decoder can inform the encoder about successful or unsuccessful transmission events by sending positive (ACK) or negative (NACK) acknowledgments. Upon receipt of feedback, various strategies are known in literature including Error Tracking, ACK and NACK mode. The presented framework unifies these concepts and achieves a trade-off between them by adjusting a simple parameter. Hence, it is not surprising that in the case when ACK and NACK mode perform similarly "well or rather "badly", the error modeling approach provides the largest benefit. The PSNR gain by the long-term memory scheme compared to single-frame prediction is up to 1.2 dB. The error modeling approach releases the structural constraints of ACK and NACK mode an thus provides improved overall performance.

The ITU-T has decided to adopt long-term memory prediction as an Annex to the H.263 standard. In this paper, we have demonstrated that this concept can be employed to the transmission of coded video over noisy channels with improved performance.

References

[1] T. Wiegand, X. Zhang, and B. Girod, "Long-Term Memory Motion-Compensated Prediction", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 70–84, Feb. 1999.

[2] ITU-T Recommendation H.263 Version 2 (H.263+), "Video Coding for Low Bitrate Communication", Jan. 1998.

[3] ISO/IEC JTC1/SC29/WG11, "MPEG-4 Video Verification Model", Draft, July 1997.

[4] N. Färber, E. Steinbach, and B. Girod, "Robust H.263 Compatible Video Transmission Over Wireless Channels", in *Proceedings of the Picture Coding Symposium*, 1996, pp. 575–578.

[5] E. Steinbach, N. Färber, and B. Girod, "Standard Compatible Extension of H.263 for Robust Video Transmission in Mobile Environments", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 6, pp. 872–881, Dec. 1997.

[6] B. Girod and N. Färber, "Feedback-Based Error Control for Mobile Video Transmission", *Proceedings of the IEEE*, vol. 97, no. 10, pp. 1707–1723, Oct. 1999.

[7] N. Färber, B. Girod, and J. Villasenor, "Extensions of the ITU-T Recommendation H.324 for Error-Resilient Video Transmission", *IEEE Communications Magazine*, vol. 36, no. 6, pp. 120–128, June 1998.

[8] Q. F. Zhu and L. Kerofsky, "Joint Source Coding, Transport Processing, and Error Concealment for H.323-Based Packet Video", in *Proceedings of the SPIE Conference on Visual Communications and Image Processing*, San Jose, CA, USA, Jan. 1999, pp. 52–62.

[9] P. Haskell and D. Messerschmitt, "Resynchronization of Motion-Compensated Video Affected by ATM Cell Loss", in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1992, vol. 3, pp. 545–548.

[10] J. Liao and J. Villasenor, "Adaptive Intra Update for Video Coding over Noisy Channels", in *Proceedings of the IEEE International Conference on Image Processing*, Lausanne, Switzerland, Oct. 1996, pp. 763–766.

[11] N. Färber, K. W. Stuhlmüller, and B. Girod, "Analysis of Error Propagation in Hybrid Video Coding with Application to Error Resilience", in *Proceedings of the IEEE International Conference on Image Processing*, Kobe, Japan, Oct. 1999.

[12] K. W. Stuhlmüller, N. Färber, M. Link, and B. Girod, "Analysis of Video Transmission over Lossy Channels", *IEEE Journal on Selected Areas in Communications*, June 2000, to appear in this issue.

[13] R. O. Hinds, T.N. Pappas, and J. S. Lim, "Joint Block-Based Video Source/Channel Coding for Packet-Switched Networks", in *Proceedings of the SPIE Conference on Visual Communications and Image Processing*, San Jose, USA, Jan. 1998, pp. 124–133.

[14] ITU-T, SG15/WP15/1, LBC-95-033, Telenor R&D, "An Error Resilience Method Based on Back Channel Signaling and FEC", Also submitted to ISO/IEC JTC1/SC29/WG11 as contribution MPEG96/M0616, Jan. 1996.

[15] ISO/IEC JTC1/SC29/WG11 MPEG96/M0768, Iterated Systems Inc., "An Error Recovery Strategy for Videophone Applications", Mar. 1996.

[16] ITU-T, SG15/WP15/1, LBC-95-309, National Semiconductor Corporation, "Sub-Videos with Retransmission and Intra-Refreshing in Mobile/Wireless Environments", Oct. 1995.

[17] S. Fukunaga, T. Nakai, and H. Inoue, "Error-Resilient Video Coding by Dynamic Replacing of Reference Pictures", in *GLOBECOM'96*, Nov. 1996.

[18] Y. Tomita, T. Kimura, and T. Ichikawa, "Error Resilient Modified Inter-Frame Coding System for Limited Reference Picture Memories", in *Proceedings of the Picture Coding Symposium*, Berlin, Germany, Sept. 1997, pp. 743–748.

[19] M. Budagavi and J. D. Gibson, "Error Propagation in Motion Compensated Video over Wireless Channels", in *Proceedings of the IEEE International Conference on Image Processing*, Santa Barbara, USA, Oct. 1997, vol. 2, pp. 89–92.

[20] C. Chen, "Error Detection and Concealment with an Unsupervised MPEG2 Video Decoder", *Journal of Visual Communication and Image Representation*, vol. 6, no. 3, pp. 265–278, Sept. 1995.

[21] ITU-T/SG16/Q15-D-65, "Video Codec Test Model, Near Term, Version 10 (TMN-10), Draft 1", Download via anonymous ftp to: standard.pictel.com/video-site/9804_Tam/q15d65.doc, Apr. 1998.

[22] ITU-T/SG16/Q15-D-13, T. Wiegand and B. Andrews, "An Improved H.263-Codec Using Rate-Distortion Optimization", Download via anonymous ftp to: standard.pictel.com/video-site/9804_Tam/q15d13.doc, Apr. 1998.

[23] G. J. Sullivan and R. L. Baker, "Rate-Distortion Optimized Motion Compensation for Video Compression Using Fixed or Variable Size Blocks", in *Proc. GLOBECOM'91*, 1991, pp. 85–90.

[24] B. Girod, "Rate-Constrained Motion Estimation", in *Proceedings of the SPIE Conference on Visual Communications and Image Processing*, Chicago, USA, Sept. 1994, pp. 1026–1034.

[25] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-Distortion Optimized Mode Selection for Very Low Bit Rate Video Coding and the Emerging H.263 Standard", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 2, pp. 182–190, Apr. 1996.

[26] G. J. Sullivan and T. Wiegand, "Rate-Distortion Optimization for Video Compression", *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74–90, Nov. 1998.

[27] W. Li and E. Salari, "Successive Elimination Algorithm for Motion Estimation", *IEEE Transactions on Image Processing*, vol. 4, no. 1, pp. 105–107, Jan. 1995.

[28] M. Ghanbari and T. K. B. Lee, "Use of 'Late Cells' for ATM Video Enhancement", in *Packet Video Workshop*, Portland, Oregon, USA, Sept. 1994, pp. I2.1–I2.4.

[29] W. C. Jakes, *Microwave Mobile Radio Reception*, Wiley, New York, USA, 1974.

[30] B. Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems, Part I: Characterization", *IEEE Communications Magazine*, vol. 35, pp. 90–100, Sept. 1997.

[31] J. E. Padgett, C. Günther, and T. Hattori, "Overview of Wireless Personal Communications", *IEEE Communications Magazine*, vol. 33, no. 1, pp. 28–41, Jan. 1995.

[32] P. Wong and D. Britland, *Mobile Data Communications Systems*, Artech House, Norwood, MA, USA, 1995.

[33] S. B. Wicker, *Error Control Systems*, Prentice Hall, Englewood Cliff, NJ, USA, 1995.

[34] R. Zhang, S. L. Regunathan, and K. Rose, "Video Coding with Optimal Inter/Intra Mode Switching for Packet Loss Resilience", *IEEE Journal on Selected Areas in Communications*, June 2000, to appear in this issue.

Biography: Thomas Wiegand

Thomas Wiegand received the Dipl.-Ing. degree in Electrical Engineering from the Technical University of Hamburg-Harburg, Germany, in 1995. Since then he has been working towards the Dr.-Ing. degree at the University of Erlangen-Nuremberg, Germany. From 1993 to 1994, he was a Visiting Researcher at Kobe University, Japan. In 1995, he was a Visiting Scholar at the University of California at Santa Barbara, USA, where he started research on video compression and transmission. Since then he has published several conference and journal papers on the subject and has contributed successfully to the ITU-T/SG16/Q15 standardization efforts. From 1997 to 1998, he has been a Visiting Researcher at Stanford University, USA, and served as a consultant to 8x8, Inc., Santa Clara, CA, USA. In cooperation with 8x8, he holds a US patent in the area of video compression. His research interests include image communication, video signal processing and compression, as well as computer graphics.

Biography: Niko Färber

Niko Färber (M'99) received a Diplom-Ingenieur degree (Dipl.-Ing) in electrical engineering from the Technical University of Munich, Germany, in 1993. He was then with the research laboratory Mannesmann Pilotentwicklung where he developed system components for satellite based vehicular navigation. In October 1993 he joined the Telecommunications Laboratory at the University of Erlangen-Nuremberg, Germany, and is now a member of the Image Communication Group. He started his research on robust video transmission as a Visiting Scientist at the Image Processing Laboratory of University of California, Los Angeles, CA, USA. Since then he has published several conference and journal papers on the subject and has contributed successfully to the ITU-T Study Group 16 efforts for robust extensions of the H.263 standard. His doctoral thesis is entitled "Feedback-Based Error Control for Robust Video Transmission" and is supervised by Prof. B. Girod. He has also served as the Publicity Vice Chair for ICASSP-97 in Munich, Germany, and performed research for 8x8, Inc., Santa Clara, CA, USA, and RealNetworks, Inc., Seattle, WA, USA. In cooperation with RealNetworks, he is holding a US patent in the area of scalable video coding. His current research project is on scalable video streaming over UMTS, which is conducted with Ericsson Eurolab, Herzogenrath, Germany.

Biography: Klaus Stuhlmüller

Klaus Stuhlmüller received his diploma in Electrical Engineering from University Erlangen-Nürnberg in 1994. From 1994 to 1996 he was at the Fraunhofer Institute for Applied Electronics. Since 1996 he is working as a research assistant at the Telecommunications Laboratory, University Erlangen-Nürnberg. He received the Young Investigator Award of the SPIE Visual Communication and Image Processing Conference in 1996. His research interests are in very low bit rate video coding, object based video coding, real–time video streaming, robust video transmission and modeling of video transmission systems.

Biography: Bernd Girod

Bernd Girod (F'98) is Professor of Electrical Engineering in the Information Systems Laboratory of Stanford University, California.

He received his M. S. degree in Electrical Engineering from Georgia Institute of Technology, in 1980 and his Doctoral degree "with highest honours" from University of Hannover, Germany, in 1987. Until 1987 he was a member of the research staff at the Institut fr Theoretische Nachrichtentechnik und Informationsverarbeitung, University of Hannover, working on moving image coding, human visual perception, and information theory. In 1988, he joined Massachusetts Institute of Technology, Cambridge, MA, USA, first as a Visiting Scientist with the Research Laboratory of Electronics, then as an Assistant Professor of Media Technology at the Media Laboratory. From 1990 to 1993, he was Professor of Computer Graphics and Technical Director of the Academy of Media Arts in Cologne, Germany, jointly appointed with the Computer Science Section of Cologne University. He was a Visiting Adjunct Professor with the Digital Signal Processing Group at Georgia Institute of Technology, Atlanta, GA, USA, in 1993. From 1993 until 1999, he was Chaired Professor of Electrical Engineering/Telecommunications at University of Erlangen-Nuremberg, Germany, and the Head of the Telecommunications Institute I, co-directing the Telecommunications Laboratory. He has served as the Chairman of the Electrical Engineering Department from 1995 to 1997, and as Director of the Center of Excellence "3-D Image Analysis and Synthesis" from 1995-1999. He has been a Visiting Professor with the Information Systems Laboratory of Stanford University, Stanford, CA, during the 1997/98 academic year. His research interests include image communication, video signal compression, human and machine vision, computer graphics and animation, as well as interactive media. For several years, he has served as a consultant to government agencies and companies, with special emphasis on start-up ventures. He has been a founder and Chief Scientist of Vivo Software, Inc., Waltham, MA (1993-98), Chief Scientist of RealNetworks, Inc., Seattle, WA, (since 1998), and a board member of 8x8, Inc., Santa Clara, CA (since 1996).

Prof. Girod has authored or co-authored one major textbook and over 200 book chapters, journal articles and conference papers in his field, and he holds several international patents. He has served as Associate Editor for the "IEEE Transactions on Image Processing" from 1991 to 1995, as Reviewing Editor and Area Editor for the "IEEE Transactions on Communications" since 1995, as Member of the the Editorial Board of "Visual Communication and Image Representation" from 1993 to 1996, and member of the Editorial Board of "Computers and Graphics" from 1992 to 1999. He is additionally a member of the Editorial Boards of the journals "EURASIP Signal Processing," "EURASIP Signal Processing: Image Communication," the "IEEE Signal Processing Magazine," and the "ACM Mobile Computing and Communication Review." He has chaired the 1990 SPIE conference on "Sensing and Reconstruction of Three-Dimensional Objects and Scenes" in Santa Clara, California, and the German Multimedia Conferences in Munich in 1993 and 1994, and has served as Tutorial Chair of ICASSP-97 in Munich and as General Chair of the 1998 IEEE Image and Multidimensional Signal Processing Workshop in Alpbach, Austria. He will be the Tutorial Chair of ICIP-2000 in Vancouver and the General Chair of the Visual Communication and Image Processing Conference in San Jose, CA, in 2001.

Prof. Girod has been a member of the IEEE Image and Multidimensional Signal Processing Committee from 1989 to 1997 and was elected Fellow of the IEEE in 1998 'for his contributions to the theory and practice of video communications.'