H.264/AVC DATA PARTITIONING FOR MOBILE VIDEO COMMUNICATION

Thomas Stockhammer[*]

Institute for Communications Engineering (LNT)
Munich University of Technology (TUM)
80290 Munich, Germany

Maja Bystrom[†]

ECE Department
Boston University
Boston, MA 02215, USA

ABSTRACT

In this work we compare non-scalable video coding with data partitioning using H.264/AVC under similar application and channel constraints for conversational applications over mobile channels. For both systems optimized rate allocation and network feedback has been applied. From the experimental results it is observed that based on the average PSNR the non-scalable system outperforms the data partitioning system. However, with the data partitioning system the percentage of entirely lost frames can be lowered, and the probability of poor quality decoded video can be reduced.

1. INTRODUCTION

Applications such as video telephony, video conferencing, multimedia streaming, or multimedia messaging to wireless clients will be important features in emerging 2.5G, 3G, and future mobile systems and may be a key factor to their success. The challenges involved in wireless video are manifold, e.g., to specify proper video coding techniques, to design networks appropriately, to apply suitable error protection and transmission schemes, as well as to limit encoding and decoding complexity. Conventional video coding and transmission systems based on H.264 [1] usually encode and transmit frames sequentially, each in one transmission packet. However, this single layer system exhibits the same drawback as any other non-scalable system, namely, that either the entire frame is decoded or is lost. Several methods to combat this problem have been proposed for H.264/AVC [2, 3], e.g., slice structured coding or even more advanced concepts such as flexible macroblock ordering. These methods might be useful in systems where packets are lost with equal probability such as in case of best-effort Internet or wireless systems without prioritization. However, the limited intra frame prediction, increased packet overhead, and possibly annoying subjective results due to block artifacts, limit the applicability of these methods [3]. In contrast, if the underlying network can support different priorities, quality–scalable source coding can provide performance gains by assigning higher priority to more important layers. Although scalable video coding methods usually provide high flexibility, they also suffer from reduced coding efficiency at least to date. For example, in [4] it was shown that a system applying progressively coded sources and advanced unequal error protection (UEP) cannot provide any gains when compared to the single layer performance with equal error protection (EEP), mainly due to the fact that the performance of the applied fine granular scalable video codec is inferior to the single layer H.264 performance.

[*]e-mail: stockhammer@ei.tum.de, Tel: +49 89 28923474
[†]e-mail: bystrom@bu.edu, Tel.+1 617 353 6521

Therefore, in this work we are interested in a scalable coder which performs as well as H.264 single layer codec in terms of rate-distortion performance to be used for mobile conversational video applications. Data partitioning in H.264 provides both properties, almost identical rate-distortion performance as well as at least some degree of scalability. Therefore, we will present the transmission system under consideration, discuss an optimized rate–allocation process, and finally present experimental results comparing a single layer system with EEP with data partitioning and UEP.

2. SYSTEM OVERVIEW

2.1. Channel Coding for Mobile Channels

Mobile channels are usually constrained in transmit power and available transmission bandwidth. In addition, highly time-varying behavior in terms of receive power due to short-term fading effects and interference is experienced. In [5] a block-fading additive white Gaussian noise (BF-AWGN) channel with perfect channel state information at the receiver is introduced as an appropriate model for many mobile channels. In the remainder of this work we assume that the transmitter has knowledge of the channel statistics, i.e., the distribution of the channel gain, and the average signal-to-noise ratio (SNR). The propagation channel is assumed to be slowly time-varying and frequency-flat for each time slot. In particular, the channel gain is assumed to be constant over the entire radio slot and iid Rayleigh. We assume that we can access $f_s = 40$ radio slots per second, each with N_s binary channel symbols resulting from a binary modulation scheme. We assume that the resulting total bit-rate $r_t = f_s \cdot N_s$ is variable from 64 kbps to 160 kbps by changing the length of N_s. This can be motivated by common multi-slot extensions in 2.5G systems such as GPRS [6].

Typically, interleaving is performed after channel coding; this spreads the channel encoded word over S radio slots, reducing the variability of the channel, but introducing additional delay. For our application we propose a flexible channel code providing different channel coding rates. For example, codes such as rate compatible punctured convolutional or Turbo codes could be used (see [7] or [8]). However, we apply high memory punctured convolutional codes (memory 96, mother code rate $1/7$, puncturing period 32) with the Far End Error Decoder (FEED) as presented in [9] for scalable image transmission and in [10] for progressive texture video coding due to the flexibility and the excellent performance. Then, channel coding rates r can be selected from the discrete set $\mathcal{R}$ containing the discrete channel coding rates $32/(32+j)$ with $j = 0, 1, 2, \ldots, 214$. This constraint results from the the mother code and the puncturing period of our convolutional code.

0-7803-8554-3/04/$20.00 ©2004 IEEE.

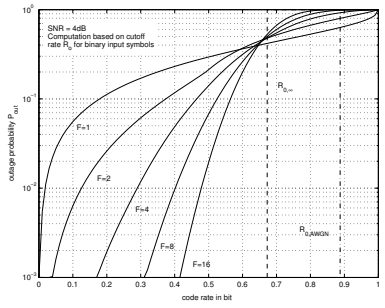


Fig. 1. Outage probability $p_{\text{out}}(r)$ versus channel coding rate r for different interleaving depths S.

Sequential channel decoding with inherent error detection [9] maps the wireless channel into a perfect packet erasure channel. The packet loss probability depends on the applied channel coding rate r according to the outage probability $p_{\text{out}}(r)$ as well as the interleaver depth S. In [10] it was shown that the performance of high-memory convolutional codes in combination with sequential decoding can be well modelled by bounding techniques based on the cutoff rate. For reasons of conciseness we do not address any channel coding simulation in this work. The effects of channel coding and interleaving based on the cutoff rate bounds over different numbers of radio slots are shown in Figure 1: For an average SNR of 4 dB, only one radio slot, $S = 1$, and channel coding rate 0.5, the outage probability p_{out} is 0.3, whereas for increased number of radio slots $S = 4$ and $S = 16$, the variance of the channel can be reduced and the outage probability decreases at the expense of additional delay. Current systems typically apply $S = 4$ as a compromise between variability and delay resulting in a still significantly varying outage probability over the applied channel coding rate r. Since the channel coding rate r is directly proportional to the maximum supported application bitrate, a trade-off between residual losses and this bitrate is necessary.

2.2. System Description

The system presented and investigated below relies on the fact that not all data is lost for poor channels; instead, the most important can be decoded and displayed. This requires that the most important data being protected more strongly against severe channel impairments than would be the less important data; resulting in the typical unequal error protection schemes.

H.264 supports the partitioning of I-slices in two partitions, P-slices in three partitions, and B-slices in a single partition. We focus on P-slices, as they can be viewed as a superset of the other slice types. The partitioning is organized such that different syntax elements are assigned to different partitions. Without going into details, partition A contains all control and header information as well as any data related to the motion compensation process. Whereas data partitioning in previous standards such as MPEG-4 and H.263 version 2 only distinguishes two partitions, in H.264 the second partition is further split into intra-related information assigned to partition B and inter-related information assigned to partition C. The main reason for this is that it was recognized that in error prone environments more frequent intra information is necessary to limit error propagation and that this information is in general more important - especially for the subjective quality - than the inter information. For more details on data partitioning we refer the reader to [2].

The system is presented in detail in Figure 2. The video encoding process is commonly based on a sequential encoding of frames $n = 1, \ldots, N$ with syntax elements being distributed among the partitions. Each partition is separately entropy coded resulting in generally three partitions with partition sizes $\mathbf{b}(q) \triangleq \{b_A(q), b_B(q), b_C(q)\}$ where the partition size depends on the applied quantization parameter q. In H.264 each partition is basically transmitted in a separate Network Abstraction Layer (NAL) unit, but can be concatenated using compound packets as specified in the draft RTP payload specification for H.264 video resulting in a partly "embedded" bit–stream for each video frame. After appropriate specification of channel coding rates $\mathbf{r} \triangleq \{r_A, r_B, r_C\} \in \mathcal{R}^3$ to be discussed in detail later, the encoded video frame is now unequally protected, interleaved, and transmitted over the wireless channel. The receiver decodes the received channel code word using the FEED [9] algorithm, which allows inherent error detection and shortening of code words. Then, bit-stream is depacketized such that only correct partitions are forwarded to the decoder.

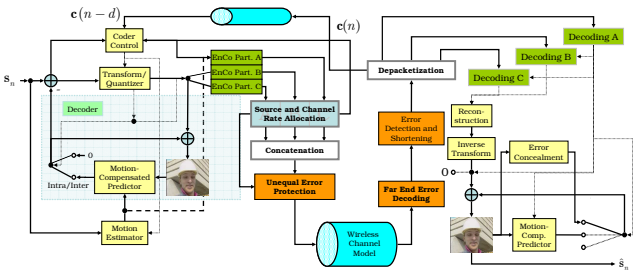


Fig. 2. Mobile Conversational System with Unequal Error Protection and Data Partitioning

The decoder processes the received data partitions as follows: In case of the loss of partition A regular error concealment is carried out, otherwise the header and motion information is decoded and used in the motion compensation process. If partition C is available the displaced frame difference is reconstructed and added to the motion compensated frame. If partition B is also available the intra macroblocks are added to the reconstructed frame. However, if partition B is lost, temporal error concealment is applied. A more advanced error concealment strategy for partition B losses is currently under investigation.

The loss of partitions or entire frames and the subsequent concealment not only causes a display problem for missing frame, but a mismatch in the reference buffers at the encoder and the decoder results in the well-known problem of spatio-temporal error propagation. Whereas in unidirectional video usually only frequent intra information in form of intra frames or intra macroblocks can reduce error propagation [11], in conversational applications the commonly existing backward channel can be used to report a d-frame delayed version of the observed channel behavior, $\mathbf{c}(n-d)$, at the decoder to the encoder [12]. There it is used to reconstruct the same reference frame in the encoder. Implementation details of this information transfer are currently under investigation. We assume that back-channel messages, partition sizes, and applied channel coding rates are available to the receiver.

2.3. Encoder Control and Rate Allocation

To perform well the encoder has to make appropriate decisions under the specified system constraints. This is especially important

as we aim to compare different systems. Therefore, we select local coding options such as macroblock modes and motion vectors using rate-distortion optimized methods [13]. For each video frame to be encoded and transmitted we now have to decide which QP q to be used and which channel coding rates $\mathbf{r} = \{r_A, r_B, r_C\}$ to be applied to the different partitions constrained by the a total bitrate as

$$N_c(q, \mathbf{r}) \triangleq \frac{b_A(q)}{r_A} + \frac{b_B(q)}{r_B} + \frac{b_C(q)}{r_C} \leq N_t. \quad (1)$$

where N_t denotes the total number of bits available for a certain video frame. Due to the stringent delay constraints and assuming a constant frame rate f_r, we assume that for each frame a total constant number of $N_t = r_t/f_r$ bits is accessible to be used for source and channel coding. Note that the size of the partitions $\mathbf{b}(q)$ is directly controlled by the applied QP, q. Due to discrete coding rates specified by the puncturing patterns, signalling overhead, as well as termination overhead, the true N_c is slightly different than presented. However, although this is handled by the implementation, we use this simple model in the discussion for sake of clarity.

The measure of interest defined in the optimization process is the expected distortion $\bar{D}_q(\mathbf{r})$ which can be estimated using well-known equations for scalable systems as

$$\begin{aligned}\bar{D}_q(\mathbf{r}) \quad = \quad & p_0(\mathbf{r}) \cdot D_0 + p_A(\mathbf{r}) \cdot D_A(q) + p_{A,C}(\mathbf{r}) \cdot D_{A,C}(q) \\ & + p_{B,C}(\mathbf{r}) \cdot D_{B,C}(q) + p_{A,B,C}(\mathbf{r}) \cdot D_{A,B,C}(q) \quad (2)\end{aligned}$$

where the index i for both event probability p_i and distortion D_i denotes the correctly decoded partitions. Note that some unreasonable terms are excluded. For example decoding partition B or C without partition A cannot decrease the distortion. The encoder is able to estimate the distortions D_i by applying the appropriate error concealment as discussed previously. Note also that these distortion terms depend on q except for D_0, the distortion in the case of loss of the entire frame.

The probability that a certain event occurs clearly depends on the applied channel coding rates r_i for different partitions and the resulting outage probabilities $p_{\text{out}}(r_i)$. Due to the dependency of partitions B and C on A, it is obvious that the only reasonable channel code rate vectors, $\mathbf{r}$, must fulfill $r_A \leq r_B$ and $r_A \leq r_C$. Although we have supposed that in general partition B is more important than partition C, it is not obvious that this is always the case, since partition B and C can be decoded independently. Therefore, we consider two cases, namely $r_B \leq r_C$ and $r_B > r_C$. Due to the interleaving and the access to the same channel realization for all channel encoded partitions it is obvious that if a partition with channel coding rate r_i cannot be decoded, any partition with $r_j \geq r_i$ cannot be decoded either. With these preliminaries, the event probabilities result in

	$r_B \leq r_C$	$r_B > r_C$
$p_0(\mathbf{r})$	$p_{\text{out}}(r_A)$	$p_{\text{out}}(r_A)$
$p_A(\mathbf{r})$	$p_{\text{out}}(r_B) - p_{\text{out}}(r_A)$	$p_{\text{out}}(r_C) - p_{\text{out}}(r_A)$
$p_{A,B}(\mathbf{r})$	0	$p_{\text{out}}(r_B) - p_{\text{out}}(r_C)$
$p_{A,C}(\mathbf{r})$	$p_{\text{out}}(r_C) - p_{\text{out}}(r_B)$	0
$p_{A,B,C}(\mathbf{r})$	$1 - p_{\text{out}}(r_C)$	$1 - p_{\text{out}}(r_B)$.

(3)

The encoder then selects the QP q_{opt} and the channel coding rates $\mathbf{r}_{\text{opt}}$ such that the expected distortion is minimized, i.e.,

$$\{q_{\text{opt}}, \mathbf{r}_{\text{opt}}\} = \arg \min_{\{q \in \mathcal{Q}, \mathbf{r} \in \mathcal{R}^3\}} \bar{D}_q(\mathbf{r}) \quad (4)$$

subject to $N_c(q_{\text{opt}}, \mathbf{r}_{\text{opt}}) \leq N_t$.

Although the search may be reduced to a linear or quadratic complexity order, a brute force search strategy is feasible by applying some properties to speed up the search. Basically, for each $q \in \mathcal{Q}$, the optimal combination of channel coding rate $\mathbf{r}_{\text{opt}}(q)$ is sought, excluding impossible combinations. In addition, it is assumed that $\bar{D}_q(\mathbf{r}_{\text{opt}}(q))$ has only one global minimum over q, and, that for given r_A and r_B the smallest $r_C \in \mathcal{R}$ is used which fulfills the rate constraint in (1). Though the complexity of this optimization process is managable, we currently investigate less complex optimization schemes including rate control.

3. EXPERIMENTAL RESULTS

3.1. Simulation Environment

In this work we have used a modified version of the test model coder JM1.7, which is provided by the of the Joint Video Team (JVT). Although JM1.7 is not the latest available test software, the coding efficiency is very close to the latest draft software available with respect to compression efficiency [1]. JM1.7 is chosen because it supports data partitioning, error resilience, and feedback methods with multiple reference frames. Specifically, no slices have been used, for entropy coding UVLC was applied, and constrained intra prediction has been turned on. We compare the data partitioning system to a system with just a single packet per video frame encoded as a single slice packet. Obviously, only one channel coding rate is applied over the packet resulting in an equal error protection (EEP) approach. This system can be derived as a subset of the data partitioning system assuming that the entire data is transmitted in partition A, whereas partition B and C do not contain any data.

For both systems, the first 10 seconds of QCIF sequences, Foreman and Carphone, have been encoded at a frame rate $f_r = 10$ Hz applying an IPPP... structure. To obtain sufficient statistics, for each experiment the sequence has been repeated at least 60 times resulting in 6000 encoded, transmitted, and decoded video frames. The rate allocation process aimed to maximize the expected PSNR. The reported average PSNR is the arithmetic mean over the PSNR of each decoded or concealed frame. The bitrate reflects the overall bitrate r_t including channel coding and source bitrate.

3.2. Simulation Results

In Figure 3 the average PSNR versus total bit-rate for the Carphone sequence with five reference frames, single layer codec with EEP, data partitioning with UEP, and different feedback delays are shown. For all cases, performance increases with bitrate, and decreases with feedback delay. For example, about 30% additional bit-rate is necessary to obtain the same performance with feedback delay 1 as with delay 0. However, most interesting is obviously the comparison of the single layer system with the data partitioning system. It can be observed that data partitioning in this environment cannot provide any gains in terms of average PSNR when compared to the single layer system. For low bitrates, the single layer system is even slightly superior than data partitioning. Almost identical results have been obtained for different sequences such as Foreman.

To understand this behavior, in Figure 4 the cumulative distribution of the PSNR is plotted for five reference frames, feedback delay 1, and $r_t = 128$ kbit/s for both systems and two sequences. Both sequences exhibit the same characteristics, with

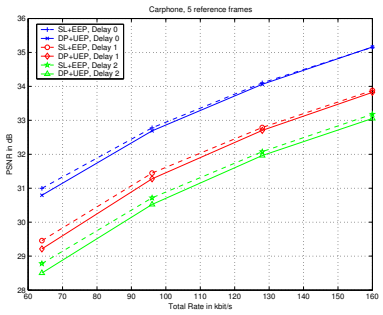
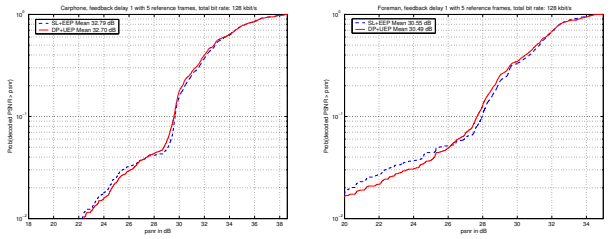


Fig. 3. Average PSNR over total bit-rate for Carphone sequence with five reference frames, single layer codec with EEP, data partitioning with UEP, and different feedback delays.

Foreman emphasizing this more: While there is higher probability of obtaining higher PSNR in the single-layer case, there is a lower probability of poor decoded quality in the data partitioning scheme. That is, on average we expect the data-partitioning scheme to have lower quality, but the single-layer case can result in very poor video quality more often.

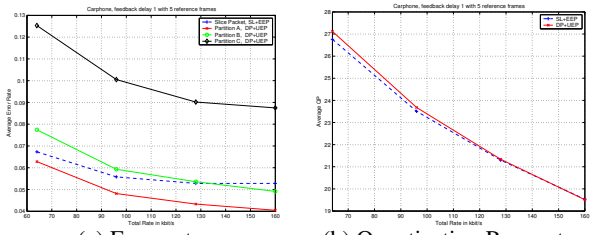


(a) Carphone (b) Foreman

Fig. 4. Cumulative distribution of PSNR for five reference frames, feedback delay 1, and $r_t = 128$ kbit/s.

The reasons for these two phenomena are investigated in Figure 5 in which the error rate and the average QP are plotted versus the total bit-rate. According to Figure 5(a), the error rate of data partition A is lower than for a single slice packet in case of single layer transmission. Therefore, we obviously avoid very poor images, i.e., images skipped due to previous-frame concealment. The channel coding rate assigned to partition C is high, resulting high loss probabilities, whereas the partition B's loss rate is in between that of A and C. The high loss rate of partition C obviously degrades the performance of data partitioning compared with the single layer system. In addition, since the rate allocation algorithm assigns stronger protection to the B-partition than to the C-partition, the separation of intra and inter information in H.264 data partitioning seems to be reasonable.

The second phenomenon, that is, that the single layer case will typically have higher PSNR, is becomes obvious from Figure 5(b): The average QP is slightly higher in case of data partitioning than for the single layer case resulting in lower encoding PSNR. The lower QP in case of data partitioning has mainly two reasons: First, a slightly higher overhead is necessary for partition header signalling, etc. In addition, in the case of data partitioning, using error concealed frames in the encoder prediction results in poor reference signals yielding a higher bitrate for the same QP. Similar effects have previously been recognized for slice structured coding in wireless environments [3].



(a) Error rate (b) Quantization Parameter

Fig. 5. Average error rate and average QP over total bit-rate for five reference frames, feedback delay 1, Carphone.

4. CONCLUSIONS

We have compared non-scalable video coding with data partitioning using H.264/AVC under similar application and channel constraints for conversational applications over wireless channels. For both systems optimized rate allocation and network feedback has been applied. From the experimental results it is observed that, based on the average PSNR, the non-scalable system outperforms the data partitioning system. However, with the data partitioning system the percentage of entirely lost frames can be lowered, and, the probability of decoding poor video quality is reduced. Further investigations are necessary for systems with data partitioning, but without feedback. In these cases error propagation can generally not be avoided and more frequent intra updates are necessary resulting in larger B-partitions with even higher importance.

5. REFERENCES

[1] A. Luthra, G.J. Sullivan, and T. Wiegand, Eds., *Special Issue on the H.264/AVC Video Coding Standard*, vol. 13, July 2003.

[2] S. Wenger, "H.264/AVC over IP," *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645–656, July 2003.

[3] T. Stockhammer, M.M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 657–673, July 2003.

[4] T. Stockhammer, "Is fine–granular scalable video coding beneficial for wireless video applications?," in *IEEE ICME*, Baltimore, MD, USA, July 2003.

[5] E. Biglieri, J. Proakis, and S. Shamai (Shitz), "Fading channels: Information-theoretic and communication aspects," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2619–2692, Oct. 1998.

[6] B. Walke and G. Brasche, "Concepts, services, and protocols of the new GSM phase 2+ general packet radio service," *IEEE Communications Magazine*, pp. 94–104, Aug. 1997.

[7] J. Hagenauer, "Rate–compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Comm.*, vol. 36, no. 4, pp. 389–400, Apr. 1988.

[8] D.N. Rowitch and L.B. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatable punctured turbo (rcpt) codes," *IEEE Trans. Comm.*, vol. 48, no. 6, pp. 948–959, June 2000.

[9] C. Weiß, T. Stockhammer, and J. Hagenauer, "The far end error decoder with application to image transmission," in *Proc. IEEE Globecom*, San Antonio, TX, USA, Nov. 2001.

[10] T. Stockhammer, H. Jenkač, and C. Weiß, "Feedback and error protection strategies for wireless video transmission," *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 465–482, July 2002.

[11] R. Zhang, S.L. Regunthan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 966–976, June 2000.

[12] B. Girod and N. Färber, "Feedback-based error control for mobile video transmission," *Proceeding of the IEEE*, vol. 97, pp. 1707–1723, Oct. 1999.

[13] G.J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.