# Introduction to the Special Issue on the H.264/AVC Video Coding Standard

OVER THE LAST one and a half decades, digital video and audio compression technologies have fundamentally changed the way we create, communicate, and consume audio-visual information. They have not only transformed existing applications and services like the distribution of entertainment video to the home but also spawned brand new industries and services like video-conferencing, direct-to-home satellite distribution, digital video recording, video-on-demand services, high-definition TV, video on mobile devices, streaming video, etc.

Since the early 1990s, when the technology was in its infancy, international video coding standards such as H.261, MPEG-1, MPEG-2/H.262, H.263, and MPEG-4 (Part 2) have been powerful engines behind the commercial success of digital video compression. They have played pivotal roles in establishing the technology via providing interoperability among products developed by different manufacturers and at the same time allowing flexibility for ingenuity in optimizing and molding the technology to fit a given application and making the cost-performance tradeoffs well suited to particular requirements. They have provided much needed assurance to the content creators that their content will run everywhere and that they do not have to create and manage multiple copies of the same content to match the products of different manufacturers. They have permitted the economy of scale to allow steep reduction in cost for masses to be able to afford the technology. They have nurtured open interactions among experts from different companies to spurt exponential growth in innovation to keep pace with the implementation technology and the needs of the applications.

The subject of this Special Issue is the newest entry in the line of international video coding standards, referred to formally as ITU-T Recommendation H.264 and ISO/IEC MPEG-4 (Part 10) Advanced Video Coding (or referred to in short as H.264/AVC). H.264/AVC is the powerful and state-of-the-art video compression standard recently developed by the ITU-T/ISO/IEC Joint Video Team (JVT) consisting of experts from ITU-T's Video Coding Experts Group (VCEG) and ISO/IEC's Moving Picture Experts Group (MPEG). As has been the case with past video coding standards, its design represents a delicate balance between coding gain, implementation complexity, and costs based on state of VLSI (ASICs and Microprocessors) design technology. It has brought the coding algorithm technology in synch with the VLSI technology that is available today or will be available in near future. Some algorithmic features that could not be considered in the past due to implementation complexity could now be accepted. New techniques developed since the time of the prior designs could be incorporated. Some features that provided some increase in coding gain but were too expensive to implement were not included. In the process, the H.264/AVC design emerged with an improvement in coding efficiency typically by a factor of two over MPEG-2—the most widely used video coding standard today—while keeping the cost within the acceptable range. Similar quality gains were shown relative to other prior designs for a very broad range of applications. Above all, this project has proven that the predictions of the demise of video compression algorithm innovations were little premature and that they are continuing to flourish vigorously.

This Special Issue opens with an overview paper about the H.264/AVC standard by Wiegand *et al.* As the title suggests, it provides a concise description of the most important tools used in the standard.

We then dive deeper into the algorithm. We start with two papers on inter-picture prediction and processing. Wedi and Musmann provide good insight into the workings of fractional-sample motion-compensated prediction in their paper "Motion and Aliasing Compensated Prediction for Hybrid Video Coding". Flierl and Girod describe how and why B-pictures work and some ways the B picture concept has been extended in H.264/AVC in their paper "Generalized B Pictures and the Draft H.264/AVC Video Compression Standard".

The next two papers focus on the transforms used and considered in the standardization process. The first of these, "Low-Complexity Transform and Quantization in H.264/AVC" by Malvar *et al.*, describes the integer DCT-like transform and quantization scaling methods adopted in the standard. This transform was designed with particular attention paid to the fact that real-number-based transform specification causes mismatches between encoders and decoders, and can be expensive to implement in products that do not have high end microprocessor or ASIC. The second paper, "Variable Block Size Transforms for H.264/AVC" by Wien, describes the effort spent by JVT on adapting the transform size to the motion estimation block size. This technology was considered to be very interesting but not fully mature or sufficiently clearly beneficial to coding efficiency, and a bit too difficult to implement for inclusion in the standard at this stage.

Deblocking filtering has been one of the centers of great discussions among the experts and has divided the video compression community for a long time. In H.264/AVC, years of past experience in this field have culminated into the development of a new and powerful design. In their paper "Adaptive Deblocking Filter" List *et al.* explain how the filter adapts itself based on the quantization parameter, different compression modes, and motion in a scene to be able to differentiate compression artifacts from the content of the scene. The result has been a filter design that has reasonable implementation complexity and provides a substantial benefit to both subjective and objective video quality.

Digital Object Identifier 10.1109/TCSVT.2003.815169

1051-8215/03$17.00 © 2003 IEEE

# H.264/AVC in Wireless Environments

Thomas Stockhammer, Miska M. Hannuksela, and Thomas Wiegand

***Abstract*—Video transmission in wireless environments is a challenging task calling for high-compression efficiency as well as a network friendly design. Both have been major goals of the H.264/AVC standardization effort addressing "conversational" (i.e., video telephony) and "nonconversational" (i.e., storage, broadcast, or streaming) applications. The video compression performance of the H.264/AVC video coding layer typically provides a significant improvement. The network-friendly design goal of H.264/AVC is addressed via the network abstraction layer that has been developed to transport the coded video data over any existing and future networks including wireless systems. The main objective of this paper is to provide an overview over the tools which are likely to be used in wireless environments and discusses the most challenging application, wireless conversational services in greater detail. Appropriate justifications for the application of different tools based on experimental results are presented.**

***Index Terms*—Error concealment, error-resilient video coding, H.264/AVC, multiple reference frames, rate-distortion optimization, video coding standards, wireless video transmission.**

## I. Introduction

SINCE 1997, the ITU-T's Video Coding Experts Group (VCEG) has been working on a new video coding standard with the internal denomination H.26L. In late 2001, the Moving Picture Expert Group (MPEG) and VCEG decided to work together as a Joint Video Team (JVT), and to create a single technical design called H.264/AVC for a forthcoming ITU-T Recommendation H.264/AVC and for a new part of the MPEG-4 standard called AVC [1][1], [2]. Since the meeting in November 2002, the technical specification is frozen and the standard text and software have been finalized. The primary goals of H.264/AVC are *improved coding efficiency* and *improved network adaptation*. The syntax of H.264/AVC typically permits a significant reduction in bit rate [3] compared to all previous standards such as ITU-T Rec. H.263 [4] and ISO/IEC JTC 1 MPEG-4 [5] at the same quality level.

The demand for fast and location-independent access to multimedia services offered on today's Internet is steadily increasing. Hence, most current and future cellular networks, like GSM-GPRS, UMTS, or CDMA-2000, contain a variety of packet-oriented transmission modes allowing transport of practically any type of IP-based traffic to and from mobile terminals, thus providing users with a simple and flexible transport interface. The third generation partnership project (3GPP) has selected several multimedia codecs for the inclusion into its multimedia specifications [6]. To provide basic video service in the first release of the 3G wireless systems, the well-established and almost identical baseline H.263 and the MPEG-4 visual simple profile have been integrated. The choice was based on the manageable complexity of the encoding and decoding process, as well as on the maturity and simplicity of the design.

However, due to the likely business models in emerging wireless systems in which the end-user's costs are proportional to the transmitted data volume and also due to limited resources bandwidth and transmission power, compression efficiency is the main target for wireless video and multimedia applications. This makes H.264/AVC coding an attractive candidate for all wireless applications including multimedia messaging services (MMS), packet-switched streaming services (PSS), and conversational applications. However, to allow transmission in different environments, not only is coding efficiency relevant, but also seamless and easy integration of the coded video into all current and possible future protocol and multiplex architectures. In addition, for conversational applications the video codec's support of enhanced error-resilience features is of major importance. This has also been taken into account in the standardization of this codec.

This paper is structured as follows. Section II introduces applications and transmission characteristics for wireless video applications. The transport of H.264/AVC video is briefly discussed and common test conditions for mobile video transmission are presented. Section III provides an overview over the H.264/AVC video coding standard from the perspective of wireless video applications. We categorize features according to their applicability in different video services. Section IV discusses the most challenging application in terms of delay constraints and error resilience, namely wireless conversational applications. A system description and problem formulation is followed by providing several alternatives on the system design using H.264/AVC as well as the combination of several modes. Section V provides experimental results for selected system concepts based on the common test conditions.

## II. Video in Mobile Networks

### A. Overview: Applications and Constraints

Video transmission for mobile terminals is likely to be a major application in emerging 3G systems and may be a key factor in their success. The video-capable display on mobile devices paves the road to several new applications. Three

Manuscript received April 9, 2002; revised May 10, 2003.

T. Stockhammer is with the Institute for Communications Engineering (LNT), Munich University of Technology (TUM), 80290 Munich, Germany (e-mail: stockhammer@ei.tum.de).

M. M. Hannuksela is with the Nokia Corporation, 33721 Tampere, Finland (e-mail: miska.hannuksela@nokia.com).

T. Wiegand is with the Fraunhofer-Institute for Telecommunications—Heinrich-Hertz-Institute Einsteinufer 37, 10587 Berlin, Germany (e-mail: wiegand@hhi.de).

Digital Object Identifier 10.1109/TCSVT.2003.815167

[1]All referenced standard documents can be accessed via anonymous ftp at ftp://standard.pictel.com/video_site, ftp://ftp.imtc-files.org/jvt-experts, ftp://ftp.ietf.org/, or ftp://www.3gpp.org/Specs/archive.

1051-8215/03$17.00 © 2003 IEEE

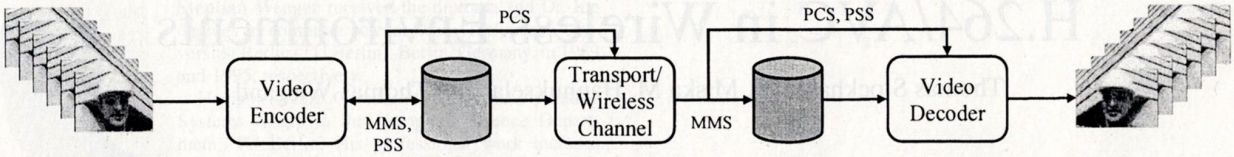



Fig. 1. Wireless video application MMS, PSS, and PCS: differentiation by real-time or offline processing for encoding, transmission, and decoding.

major service categories were identified in the H.264/AVC standardization process [7].

1) Circuit-switched [8] and packet-switched conversational services (PCS) [9] for video telephony and conferencing.
2) Live or pre-recorded video packet-switched streaming services (PSS) [10].
3) Video in multimedia messaging services (MMS) [11].

Although new services such as multimedia broadcast/multicast services (MBMS) [12] are planned for future releases of wireless networks, we restrict ourselves to single receiver applications. Mobile devices are hand-held and constrained in processing power and storage capacity. Therefore, a mobile video codec design must minimize terminal complexity while remaining consistent with the efficiency and robustness goals of the design. As complexity issues are discussed elsewhere in this issue [13], [14], we restrict ourselves to transmission constraints and properties.

The transmission requirements for the three identified applications can be distinguished with respect to requested data rate, the maximum allowed end-to-end delay, and the maximum delay jitter. This results in different system architectures for each of these applications. A simplified illustration is provided in Fig. 1. As MMS does not include any real-time constraints, encoding, transport, and decoding are completely separated. The recorded video signal is offline encoded and locally stored. The transmission is started using the stored signal at any time. The decoding process at the receiver is in general not started until the completion of the download. In PSS applications, the user typically requests pre-coded sequences, which are stored at a server. Whereas encoding and transmission are separated, decoding and display is started during transmission to minimize the initial delay and memory usage in mobile devices. Finally, in conversational services, the end-to-end delay has to be minimized to avoid any perceptual disturbances and to maintain synchronicity of audio and video. Therefore, encoding, transmission and decoding is performed simultaneously in real-time and, moreover, in both directions. These different ancillary conditions permit and require different strategies in encoding, transport, decoding, as well as in the underlying network and control architecture.

In general, the available bandwidth and therefore the bit-rate over the radio link are limited and the costs for a user are expected to be proportional to the reserved bit rate or the number of transmitted bits over the radio link. Thus, low bit rates are likely to be typical, and *compression efficiency* is the main requirement for a video coding standard to be successful in a mobile environment. This makes H.264/AVC a prime candidate for the use in wireless systems, because of its superior compression efficiency [3].

In addition, the mobile environment is characterized by harsh transmission conditions in terms of attenuation, shadowing, fading, and multi-user interference, which result in time- and location-varying channel conditions. The frequency of the channel variations highly depends on the environment, the user topology, the velocity of the mobile user, and the carrier frequency of the signal. For sufficiently long code words averaging over channel statistics is possible and transmission strategies can be used that are based on the long-term averages of fading states and the ergodic behavior of the channel. Many highly sophisticated radio link features such as broadband access, diversity techniques, space-time coding, multiple antenna systems, fast power control, interleaving, and forward error correction (FEC) by Turbo codes are used in 3G systems to reduce variations in channel conditions. However, only for fast-moving users and relatively large tolerated maximum delay can these advanced techniques provide a negligible bit error and radio block loss rate. Usually, some amount of residual errors has to be tolerated for low-delay applications due to the nonergodic behavior of the channel and the imperfectness of the applied signal processing. Therefore, in addition to high compression efficiency and reasonable complexity, a video coding standard to be applicable for conversational services in wireless environments has to be error resilient.

In addition, it is worth noting at this point that new directions in the design of wireless systems do not necessarily attempt to minimize the error rates in the system, but to maximize the throughput. This is especially appealing for services with relaxed delay constraints such as PSS and MMS. The nonergodic behavior of the channel is exploited such that in case of good channel states significantly higher data rate is supported than in bad channel states. In addition, reliable link layer protocols with persistent Automatic Repeat reQuest (ARQ) are usually used to guarantee error-free delivery. For example, in the high-speed downlink packet access (HSDPA) concept [15] ARQ, adaptive modulation schemes, and multiuser scheduling taking into account the channel states are combined to significantly enhance the throughput in wireless systems.

3G wireless transmission stacks usually consist of two different bearer types, dedicated and shared channels. Whereas in dedicated channels one user gets assigned a fixed data rate for the entire transmission interval, shared channels allow a dynamic bit-rate allocation similar to ATM or GSM GPRS. HSDPA will be an extension of the shared channel concept on the air interface. Except for MMS all streaming and conversational applications are assumed to use dedicated channels in the initial phase of 3G wireless systems due to their almost constant bit-rate behavior. In modern system designs, an application can request one of many different quality-of-service (QoS) classes. QoS classes contain parameters like a maximum error rate,

TABLE I
QoS SERVICE CLASSES IN PACKET RADIO SYSTEMS [15]

| Traffic Class | Fundamental Characteristics | Typical Examples |
|---|---|---|
| Conversational | Preserve time relation between information entities of the stream Conversational pattern (stringent and low delay) | Voice and video telephony, Video conferencing |
| Streaming | Preserve time relation (variation) between information entities of the stream | Streaming multimedia (video, audio, etc.) |
| Interactive | Request response pattern, Preserve data integrity | Web browsing, network games |
| Background | Destination is not expecting the data within a certain time Preserve data integrity | Background download of e-mails, files, etc. |

maximum delay, and a guaranteed maximum data rate. Furthermore, according to [16], applications are usually divided into different service classes: conversational, streaming, interactive, and background traffic. Characteristics and typical examples are shown in Table I.

### B. *Transport of H.264/AVC Video in Wireless Systems*

According to Fig. 2, H.264/AVC distinguishes between two different conceptual layers, the video coding layer (VCL) and the network abstraction layer (NAL). Both the VCL and the NAL are part of the H.264/AVC standard. The VCL specifies an efficient representation for the coded video signal. The NAL of H.264/AVC defines the interface between the video codec itself and the outside world. It operates on NAL units which give support for the packet-based approach of most existing networks. At the NAL decoder interface, it is assumed that the NAL units are delivered in decoding order and that packets are either received correctly, are lost, or an error flag in the NAL unit header can be raised if the payload contains bit errors. The latter feature is not part of the standard as the flag can be used for different purposes. However, it provides a way to signal an error indication through the entire network. Additionally, interface specifications are required for different transport protocols that will be specified by the responsible standardization bodies. The exact transport and encapsulation of NAL units for different transport systems, such as H.320 [17], MPEG-2 Systems [18], and RTP/IP [19], are also outside the scope of the H.264/AVC standardization. The NAL decoder interface is normatively defined in the standard, whereas the interface between the VCL and the NAL is conceptual and helps in describing and separating the tasks of the VCL and the NAL.

For real-time video services over 3G mobile networks, two protocol stacks are of major interest. 3GPP has specified a multimedia telephony service for circuit-switched channels [8] based on ITU-T Recommendation H.324M. For IP-based packet-switched communication, 3GPP has chosen to use SIP and SDP for call control [20] and RTP for media transport [9]. In other words, the IP-based protocol stack as presented in [21] will be used in packet-switched 3G mobile services. While the H.324 and the RTP/UDP/IP stacks have different roots and a completely different switching philosophy, the loss and delay effects on the media data when transmitting over wireless dedicated channels are very similar.

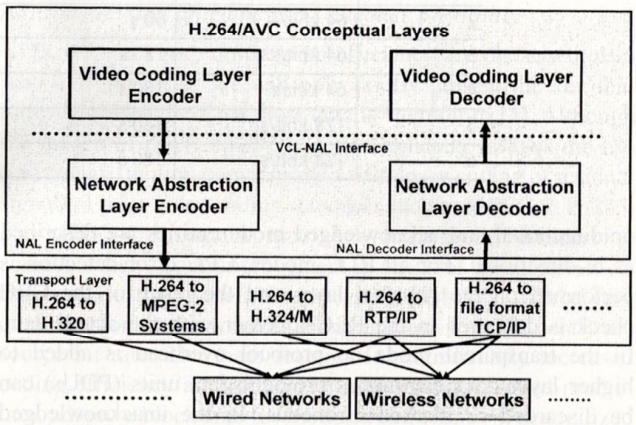



Fig. 2. H.264/AVC standard in transport environment.

H.324 [22] was primarily established by the ITU-T for low-bit-rate circuit-switched modem links and error prone extensions for mobile circuit switched low-bit-rate conversational services have been added. H.324M officially known as ITU-T Rec. H.324 Annex C, allows transmission over low, moderate, and high bit-error rate circuit switched links. 3GPP adopted H.324M including an error robust extension of the multiplexing protocol H.223 known as H.223 Annex B as the protocol used for circuit-switched video communication. This multiplexing protocol includes two layers: an error-resilient packet-based multiplex layer and an adaptation layer featuring common error detection capabilities, such as sequence numbering and cyclic redundancy checks (CRSs). Therefore, it is very similar to the RTP/UDP/IP stack (see [21]).

For packet-switched services, 3GPP/3GPP2 agreed on an IP-based stack. Fig. 3 shows a typical packetization of a NAL unit encapsulated in RTP/UDP/IP [19] through the 3GPP2 user plane protocol stack. After robust header compression (RoHC) [23] this IP/UDP/RTP packet is encapsulated into one packet data convergence protocol/point-to-point protocol (PDCP/PPP) packet that becomes a radio link control (RLC)-service data unit (SDU). The RLC protocol can operate in three modes: 1) transparent; 2) unacknowledged; and 3) acknowledged mode [26]. The RLC protocol provides segmentation and retransmission services for both users and control data. The transparent and unacknowledged mode RLC entities are defined to be

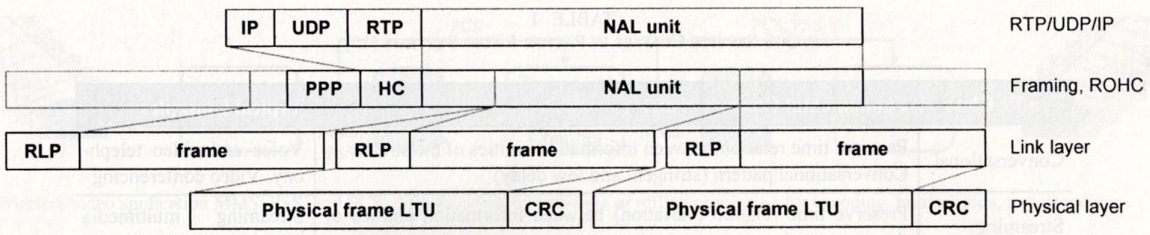



Fig. 3. Packetization through 3GPP2 user plane protocol stack.

TABLE II
BIT-ERROR PATTERNS

| No. | Bit rate | Length | BER | Mobile Speed | Application |
|---|---|---|---|---|---|
| 1 | 64 kbit/s | 60 s | 9.3e-3 | 3 km/h | Streaming |
| 2 | 64 kbit/s | 60 s | 2.9e-3 | 3 km/h | Streaming |
| 3 | 64 kbit/s | 180 s | 5.1e-4 | 3 km/h | Conversational |
| 4 | 64 kbit/s | 180 s | 1.7e-4 | 50km/h | Conversational |
| 5 | 128 kbit/s | 180 s | 5.0e-4 | 3 km/h | Conversational |
| 6 | 128 kbit/s | 180 s | 2.0e-4 | 50km/h | Conversational |

unidirectional and acknowledged mode entities are described as bi-directional. For all RLC modes, CRC error detection is performed on the physical layer and the result of the CRC check is delivered to the RLC together with the actual data. In the transparent mode no protocol overhead is added to higher layer data. Erroneous protocol data units (PDUs) can be discarded or marked erroneous. In the unacknowledged mode, no retransmission protocol is in use and data delivery is not guaranteed. Received erroneous data is either marked or discarded depending on the configuration. In the acknowledged mode, an automatic repeat request mechanism is used for backward error correction.

As video packets are of varying length by nature, the length of RLC-SDU's varies as well. If an RLC-SDU is larger than an RLC-PDU, the SDU is segmented into several PDUs. In the used, unacknowledged, and acknowledged modes, the flow of variable-size RLC-SDUs is continuous to avoid padding of bits as necessary for the transparent mode. In unacknowledged mode, if any of the RLC-PDUs containing data from a certain RLC-SDU have not been received correctly, the RLC-SDU is typically discarded. In acknowledged mode, the RLC/radio link protocol (RLP) layer can perform re-transmissions.

Additionally, both protocol stacks H.324 and RTP/IP/UDP use reliable setup and control protocols, H.245 and SIP, respectively. Hence, it can be assumed that a small amount of control information can be transported out-of-band in a reliable way. The resulting properties of real-time low-delay video transmission are therefore very similar in both cases. Packets are transmitted over underlying transports protocols and channels, which provide framing, encapsulation, error detection, and reliable setup. We focus hereafter on the RTP/IP-based transmission over wireless channels [21].

### C. Common Test Conditions for Wireless Video

In the H.264/AVC standardization process, the importance of mobile video transmission has been recognized by adopting appropriate common test conditions for 3G mobile transmission for circuit switched conversational services based on H.324M [24] and for packet-switched conversational and streaming services [25]. These test conditions permit the selection of appropriate coding features, testing and evaluating error-resilience features, as well as meaningful anchor results. In this paper, we focus on the IP-based test conditions. The common test conditions define six test-case combinations for packet-switched conversational services as well as packet-switched streaming services over 3G mobile networks. Additionally, the test conditions include simplified offline 3GPP/3GPP2 simulation software, programming interfaces and evaluation criteria. Radio channel conditions are simulated with bit-error patterns, which were generated from simulated mobile radio channel conditions. The bit-error patterns are captured above the physical layer and below the RLC/RLP layer and, therefore, they are used as the physical layer simulation in practice. The properties bit rate, length, bit-error rate, and the mobile speed of the bit-error patterns are presented in Table II.

The bit errors in the files are statistically dependent, as channel coding and decoding included in 3G systems produces burst errors. This has been taken into account by evaluating the bit-error pattern files in the following. Patterns 1 and 2 are mostly suited to be used in video streaming applications, where RLP/RLC layer re-transmissions can correct many of the frame losses. The applied channel-coding scheme is a Turbo code scheme and power control targeting throughput maximization rather than error minimization. Patterns 1 and 2 are unrealistic for conversational service, as an acceptable quality cannot be achieved with such high error rates without retransmissions.

Patterns 3–6 are meant to simulate a more reliable, lower error-rate bearer that is required in conversational applications. Assuming a random byte starting position within the file the packet error probability $p_e(r)$ depending of the length of the packet $r$ in bytes can be determined. These error probabilities $p_e(r)$ for all bit-error patterns are shown in Fig. 4. It is obvious

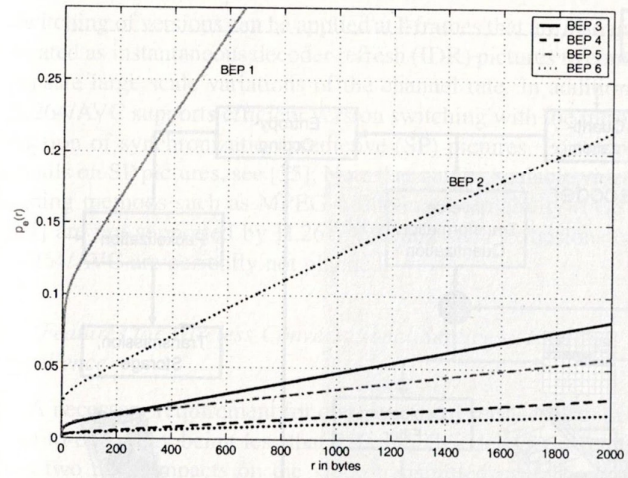



Fig. 4. Packet-loss probability $p_e(r)$ over packet length $r$ in bytes for bit-error patterns according to [25].

that the error rate increases significantly with increasing packet length. For bit-error patterns (BEP1 and 2), the loss rates are not acceptable as already short packets, e.g., 500 bytes, have loss probabilities up to 25%. The high error rates require retransmission on the link layer.

The patterns useful for conversational services provide very decent error characteristics for reasonable packet lengths. The loss probability for reasonable packet sizes up to 1000 bytes is below 5%. This means that typical fixed Internet packet loss probabilities (compare [21]) are not exceeded. Note that with higher speed (50 km/h), the channel tends to be "more ergodic" than in case of the walking user (3 km/h). Therefore, the error rates are usually higher for slowly moving users than for fast-moving users.

During the standardization process, it was agreed in the very beginning that the standard should include error-resilience features for IP-based wired and wireless transmission. Usually two kinds of errors are present in today's transmission systems: bit inversion errors or packet losses. However, all relevant multiplexing protocols like H.223 and UDP/IP and almost all underlying mobile systems include packet loss and bit-error detection capabilities applying sequence numbering and block check sequences, respectively. Therefore, it can be assumed that a vast majority of erroneous transmission packets can be detected. Moreover, even if packets were detected to contain bit errors, decoding could be attempted. Some research has been conducted in this area and, in a few scenarios, gains have been reported for still image transmission [27]. However, in the standardization of H.264/AVC for the development of the reference software, bit-erroneous packets have been considered as being discarded by the receiver for the following reasons.

1) Processing of bit-erroneous packets is likely to be possible in a receiving mobile device only, as gateways and receivers having fixed network connection typically drop erroneous packets.
2) Joint source-channel decoding, such as trellis-based decoding of variable-length codes or iterative source and channel decoding might be applied. However, these techniques have not yet shown significant improvements for video decoding.
3) The decoding based on lost packets serves as a lower bound on the performance in bit-error-prone environments and, therefore, provides a valid benchmark of the performance of H.264/AVC in error-prone environment.
4) Finally, handling of bit errors generally complicates the implementation of decoder software significantly. As the test model software for H.264 was developed for multiple purposes, only simple but meaningful network interfaces have been integrated.

## III. H.264/AVC—An Efficient and Flexible Video Coding Toolbox

### A. Compression Efficiency and Encoder Flexibility

The features for compression efficiency are discussed elsewhere in this issue, we will only briefly present the key features of the standard, for more details we refer to [2]. Although the design of the H.264/AVC codec basically follows the hybrid design (motion compensation with lossy coding of residual signal) of prior video coding standards such as MPEG-2, H.263, and MPEG-4, it contains many new features that enable it to achieve a significant improvement in terms of compression efficiency. This is the main reason why H.264/AVC will be very attractive for use in wireless environments with the costly resource bit rate. The main features for significantly increased coding efficiency are multiframe motion-compensated prediction, adaptive block size for motion compensation, generalized B-pictures concepts, quartel-pel motion accuracy, intra coding utilizing prediction in the spatial domain, in-loop deblocking filters, and efficient entropy-coding methods.

The normative part of a video coding standard in general only consists of the appropriate definition of the order and semantics of the syntax elements and the decoding of error-free bit streams. This allows a significant flexibility at the encoder, which can, on the one hand, be exploited for pure compression efficiency, and on the other hand, several included features in the standard can be selected by the encoder for other purposes such as error resilience, random access, etc. A typical encoder with the main encoding options is shown in Fig. 5.

The encoding options relevant for wireless transmission are highlighted. The recorded video data is preprocessed by appropriate spatial and temporal preprocessing such that the data rates and displays in a wireless environment are well-matched. For the quantization of transform coefficients, H.264 coding uses scalar quantization. The quantizers are arranged in a way that there is an increase of approximately 12.5% from one quantization parameter (QP) to the next. The quantized transform coefficients are converted into coding symbols and all syntax elements of a macroblock (MB) including the coding symbols are conveyed by entropy coding methods. A MB can always be coded in one of several intra modes with and without prediction, as well as various efficient inter modes. Each motion-compensated mode corresponds to a specific partition of the MB into fixed-size blocks used for motion description, and up to 16 motion vectors may be transmitted for a MB. In addition, for each MB, a different reference frame can be selected. Finally, a NAL

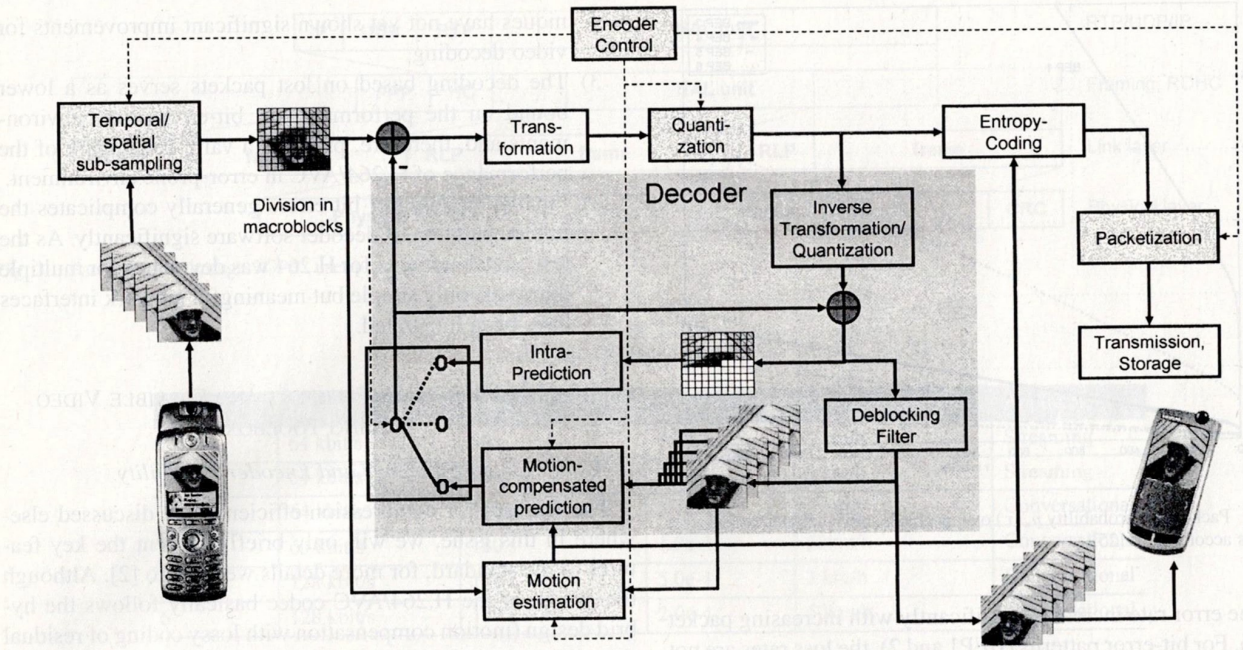



Fig. 5. H.264/AVC encoder realization with coding options.

unit in single-slice mode consists of the coded MBs of an entire frame or a subset of frame. More details on the packetization as well as on the appropriate selection of the presented different modes are discussed in Section IV.

### *B. Features for Multimedia Messaging and Wireless Packet-Based Streaming*

In addition to pure compression efficiency features, additional tools for different purposes have been included in H.264/AVC. We will highlight those features with application to wireless video transmission. Because of the strict separation of encoding, transmission, and decoding, the main issue for MMS is compression efficiency. Other helpful features include the insertion of regular intra frames with instantaneous decoder refresh (IDR) for random access and fast forward. The rate control is typically applied such that video quality is almost constant over the sequence, regardless of the scene complexity except for constraints from the hypothetical reference decoder (HRD) [28]. If time, memory capabilities, and battery power permit, even several encoding passes for optimized rate-distortion (R-D) performance would be possible. On the link layer, reliable transmission strategies as known for data transmission such as file download are used.

Due to on-line transmission and decoding, streaming applications involve more technical challenges than MMS. Usually, pre-encoded data is requested by the user, which inherently does not allow an adaptation to the transmission conditions such as bit rate or error rate in the encoding process. However, the receiver usually buffers the received data and starts play-back after a few seconds. Once starting playback, a continuous presentation of the sequence should be guaranteed. As wireless channels usually show ergodic behavior within a window of a few seconds, reliable transmission schemes can be applied on the link layer, especially when the channel is known at the transmitter or retransmissions for erroneous link layer packets can be used as for example in the acknowledged mode. Slow variance due to distance, shadowing, or varying multiuser topology in the supported cell with renewed resource allocation transform the wireless channel in a slowly varying variable-bit-rate channel. With an appropriate setting of the initial delay and receiver buffer a certain quality of service can be guaranteed [28].

Furthermore, general channel-adaptive streaming technologies, which allow reacting to variable bit-rate channels, have gained significant interest recently. According to [30], these techniques can be grouped into three different categories. Firstly, *adaptive media playout* [31] is a new technique that allows a streaming media client, without the involvement of the server, to control the rate at which data is consumed by the playout process. Therefore, the probability of decoder buffer underflows and overflows can be reduced, but still noticeable artifacts in the displayed video occur. A second technology for a streaming media system is proposed, which makes decisions that govern how to allocate transmission resources among packets. Recent work [32] provides a flexible framework to allow *R-D optimized packet scheduling*. Finally, it is shown that this R-D-optimized transmission can be supported, if media streams are pre-encoded with appropriate packet dependencies, possibly adapted to the channel (*channel-adaptive packet dependency control*) [33].

The latter techniques are supported by H.264/AVC by various means. As the streaming server is in general aware of the current channel bit rate, the transmitter can decide to send one of several pre-encoded versions of the same content taking into account the expected channel behavior. If the channel rate fluctuates only in a small range, frame dropping of nonreference frames might be sufficient resulting in well-known temporal scalability.

Switching of versions can be applied at I-frames that are also indicated as instantaneous decoder refresh (IDR) pictures to compensate large scale variations of the channel rate. In addition, H.264/AVC supports efficient version switching with the introduction of synchronization-predictive (SP) pictures. For more details on SP pictures, see [35]. Note that quality scalable video coding methods such as MPEG-4 fine-grain scalability (FGS) [34] are not supported by H.264/AVC and such extensions of H.264/AVC are currently not planned.

### C. Features for Wireless Conversational Services—Error Resilience

A necessary requirement for conversational services is a low end-to-end delay being less that 250 ms. This delay constraint has two main impacts on the video transmitted over wireless bearer services with constant bit rate. Firstly, features have to be provided which allow adapting the bit-rate such that over a short window a constant bit-rate can be maintained. In addition, usually only temporally backward references in motion compensation are used in conversational applications, since prediction from future frames would introduce additional delay. Secondly, within the round-trip time of the communication, the channel usually shows nonergodic behavior and transmission errors cannot be avoided. Whereas the first issue can be solved by adapting the QP appropriately, the second issue requires error-resiliency tools in the video codec itself. More exactly, an error-resilient video coding standard suitable for conversational wireless services has to provide features to combat two problems: on the one hand, it is necessary to minimize the visual effect of errors within one frame. On the other hand, as errors cannot be avoided, the well-known problem of spatio-temporal error propagation in hybrid video coding has to be limited. We will present all error-resilience features included in the H.264/AVC standard and provide further details on the exact application in Section IV.

Packet loss probability and the visual degradation from packet losses can be reduced by introducing slice-structured coding. A slice is a sequence of MBs and provides spatially distinct resynchronization points within the video data for a single frame. No intra-frame prediction takes place across slice boundaries. With that, packet loss probability can be reduced if slices and, therefore, transmission packets are relatively small, since the probability of a bit-error hitting a short packet is generally lower than for large packets (see, e.g., Fig. 4). Moreover, short packets reduce the amount of lost information and, hence, the error is limited and error concealment methods can be applied successfully. However, the loss of intra-frame prediction and the increased overhead associated with decreasing slice sizes adversely affect coding performance and requires additional overhead per slice. Especially for mobile transmission, where the packet size clearly affects loss probability, a careful selection of the packet size is necessary. H.264/AVC specifies several enhanced concepts to reduce the artifacts caused by packet losses within one frame. Slices can be grouped by the use of aggregation packets into one packet and, therefore, concepts such as group-of-block (GOB) and *slice interleaving* [37], [38] are possible. This does not reduce the coding overhead in the VCL, but the costly RTP overhead of up to 40 bytes per packet can be avoided.

A more advanced and generalized concept is provided by a feature that has been called by the proponents *flexible MB ordering* (FMO) [39]. FMO permits the specification of different patterns for the mapping of MBs to slices including checkerboard-like patterns, sub-pictures within a picture (e.g., splitting a CIF picture into four QCIF pictures), or a dispersed mapping of MBs to slices. FMO is especially powerful in conjunction with appropriate error concealment when the samples of a missing slice are surrounded by many samples of correctly decoded slices. For more details on FMO, see [21].

Another error-resilience feature in H.264/AVC is *data partitioning*, which can also reduce visual artifacts resulting from packet losses, especially if prioritization or unequal error protection is provided by the network. For more details on the data-partitioning mode, we refer to [21]. In general, any kind of *forward error protection* (FEC) in combination with interleaving for packet lossy channels can be applied. A simple solution is provided by RFC2733 [40], more advanced schemes have been evaluated in many papers, e.g., [41], [42]. However, in the following, we do not consider FEC schemes in the transport layer as this requires a reasonable number of packets per codeword.

Despite all these techniques, packet losses and resulting reference frame mismatches between encoder and decoder are usually not avoidable. Then, the effects of spatio-temporal error propagation are, in general, severe. The impairment caused by transmission errors decays over time to some extent. However, the leakage in standardized video decoders, such as H.264/AVC, 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 previously coded frame. Completely intra-coded frames are usually not inserted in real-time and conversational video applications as the instantaneous bit rate and the resulting delay is increased significantly. Instead, H.264/AVC allows encoding of single MBs for regions that cannot be predicted efficiently as it is also known from other standards. In H.264/AVC, the efficient intra prediction can be constrained to intra MBs only to avoid error propagation from inter-coded MBs to refreshing intra-coded MBs. Another feature in H.264/AVC is the possibility to select the reference frame from the multiframe buffer. Both features have mainly been introduced for improved coding efficiency, but they can efficiently be used to limit the error propagation. Conservative approaches transmit a number of intra-coded MBs anticipating transmission errors. In this situation, the selection of intra-coded MBs can be done either randomly or preferably in a certain update pattern. For details and early work on this subject, see [43]–[45]. Multiple reference frames can also be used to limit the error propagation, for example in *video redundancy coding* schemes (see, e.g., [46]). In addition, a method known from H.263 under the acronym *redundant slices* will be supported in JVT coding. This will allow sending the same slice predicted from different reference frames which provides the decoder the possibility to predict this slice from error-free reference areas. Finally, multiple reference frames can be successfully combined with a feedback channel, which will be discussed in detail among others in Section IV.

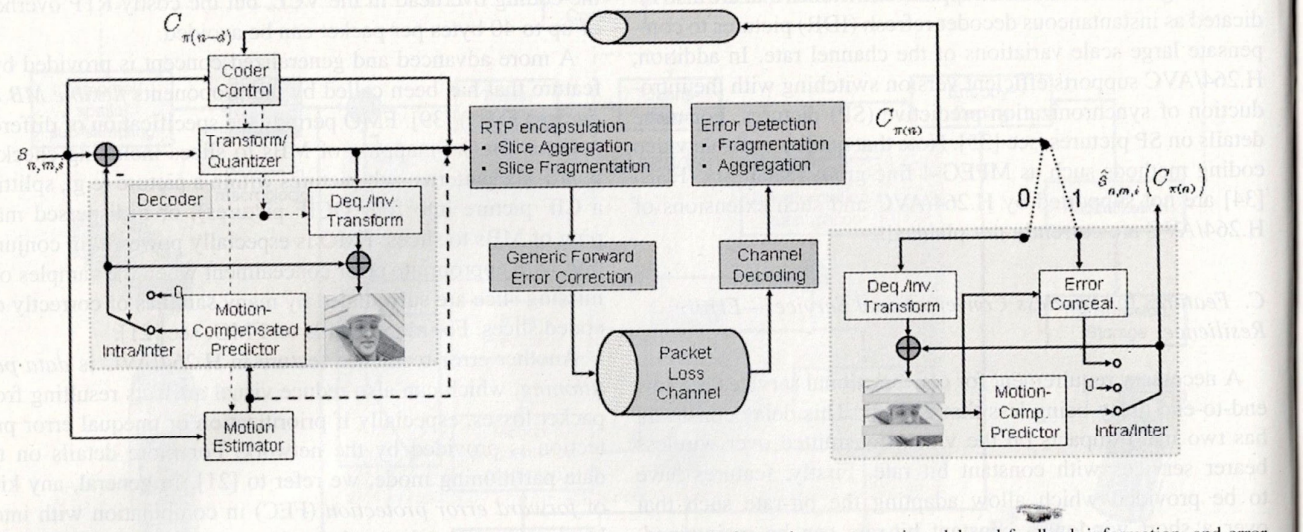



Fig. 6. H.264/AVC in IP-based packet-lossy environment with RTP encapsulation, generic forward error correction, delayed feedback information, and error concealment.

## IV. Using H.264/AVC in Wireless Conversational Services

### A. Problem Formulation and System Setup

The various error-resilience tools as described in the previous section provide a significant amount of freedom and flexibility for the implementation of H.264/AVC in wireless conversational services. A main criterion for the performance of the system is the appropriate selection of one or several of the mentioned tools together with the exact parameters, e.g., the position and number of intra MBs.

We will discuss in the following channel behavior and error concealment by formalizing the notation for sample representation. The investigated video transmission system is shown in Fig. 6. H.264/AVC video encoding is based on a sequential encoding of frames denoted with the index $n = 1, \ldots, N$ with $N$ being the total number of frames to be encoded. In most existing video coding standards including H.264/AVC, within each frame video encoding is typically based on sequential encoding of MBs (except for FMO) denoted by index $m = 1, \ldots, M$, where $M$ specifies the total number of MBs in one frame and depends on the spatial resolution of the video sequence. The encoding process creates slices by grouping a certain number of MBs. Picture number $n$ and start MB address $m_j$ are binary coded in the slice header. The coded representation of a slice is the payload of a NAL unit. The RTP payload specification specifies simple encapsulation of NAL units. In addition, several NAL units can be combined into one aggregation packet or one NAL unit can be fragmented into several transport packets [19], [21].

For notational convenience, let us define the number of transmission packets to transmit all frames up to $n$ as $\pi(n)$. With that, we can define the packet loss or channel behavior $c$ as a binary sequence $\{0,1\}^{\pi(n)}$ indicating whether a slice is lost (indicated by 1) or correctly received (indicated by 0). Obviously, if a NAL unit with the encapsulated slice is lost, all MBs contained by this slice are lost. It can be assumed that the decoder is aware of any lost packet as discussed previously. The channel-loss sequence is random and, therefore, we denote it as $C_{\pi(n)}$, where the statistics are in general unknown to the encoder. According to Fig. 6, in addition to the forward link it possible that a low-bit-rate reliable back-channel from the decoder to the encoder is available which allows reporting a $d$-frame delayed version $C_{\pi(n-d)}$ of the observed channel behavior at the decoder to the encoder. In RTP/IP environments, this is usually based on RTCP messages, and in wireless environments, internal protocols might be used.

The decoder processes the received sequence of packets. Whereas correctly received packets are decoded as usual for the lost packet, an error-concealment algorithm has to be invoked. The reconstructed sample $s_{n,m,i}$ at position $i$ in MB $m$ and frame $n$ depends on the channel behavior and on the decoder error concealment. In inter-coding mode, i.e., when motion-compensated prediction (MCP) is utilized, the loss of information in one frame has a considerable impact on the quality of the following frames, if the concealed image content is referenced for MCP. Because errors remain visible for a longer period of time, the resulting artifacts are particularly annoying. Therefore, due to the motion-compensation process and the resulting error propagation, the reconstructed image depends not only on the lost packets for the current frame, but in general on the entire channel-loss sequence $C_{\pi(n)}$. We denote this dependency by $\hat{s}_{n,m,i}(o, C_{\pi(n)})$

In the following, we will discuss appropriate extensions of the encoder, the decoder, and the transmission environment, which are either necessary or at least beneficial to enhance the quality of the transmitted video.

### B. Decoder Extensions—Loss Detection and Error Concealment

The H.264 standard defines how a decoder reacts to an error-free bit stream. In addition, a decoder implementation has also to deal with transmission errors. As we discussed earlier, it is assumed that bit errors are detected by the lower layer

entities and any remaining transmission error results in a packet loss. Therefore, we address the reaction of the decoder to slice losses. Two major issues are important for an error-resilient decoder: a robust video decoder has to detect transmission errors, and appropriate concealment on detected errors has to be applied. In this section, we present how the H.264 test model decoder meets the goal of error resiliency.

The error detection capabilities of the test model decoder are based on two assumptions about the error detection operation of the underlying system. First, bit-erroneous slices are discarded prior to passing the slices to the test model decoder. Second, received data is buffered in a way that the correct decoding order is recovered. In other words, the test model decoder expects noncorrupted slices in a correct decoding order. Temporal and spatial localization of lost packets is left to the decoder. In particular, the decoder has to detect if an entire picture or one or more slices of a picture were lost. Losses of entire pictures are detected using frame numbers associated with each frame and carried in slice headers. A frame number $n$ is incremented by one for each coded and transmitted frame that is further used for motion compensation. These frames are herein referred to as reference frames. For disposable nonreference pictures, such as conventional B-pictures, the frame number is incremented relative to the value in the most recent reference frame that precedes the disposable picture in the bit-stream order.

The decoder generates a slice structure for each received single slice packet and forwards it to the VCL decoder which maintains a state machine that keeps track of the expected frame number $n_e$. Moreover, the decoder maintains a loss indication for each MB within the current picture. At the beginning of a new picture, the binary map is reset to indicate that all MBs were lost. If the frame number $n$ of the next slice to be decoded equals $n_e$, the decoder decodes the slice and updates the binary map. If $n$ is greater than $n_e$, it is deduced that all the received slices of the previous picture have been decoded. Then, the binary map is investigated, and if the picture is not fully covered by correctly decoded MBs, a slice loss is inferred and losses are concealed as presented in the following. Moreover, if $n$ is greater than $n_e + 1$, the decoder infers a loss of pictures and inserts concealed pictures to the reference picture buffer as if the lost pictures were decoded. The concealment is accomplished by copying the previous decoded picture. Finally, the decoder resets the binary map and decodes the next slice of the next picture.

Error concealment is a nonnormative feature in the H.264 test model. The target for the selected error concealment is to provide a basic level of error resiliency for the decoder. Any error-robust coding scheme proposed for H.264 should be compared against the H.264 test model equipped with the selected error concealment algorithms. Two well-known concealment algorithms, weighted pixel value averaging for intra pictures [47] and boundary-matching-based motion vector recovery for inter pictures [48], were tailored for H.264 as summarized below and described in details in [49] and [50].

Weighted pixel value averaging operates as follows. If a MB has not been received, it is concealed from the pixel values of spatially adjacent MBs. If a lost MB has at least two correctly decoded neighboring MBs, only these neighbors are used in the concealment process. Otherwise, previously concealed neighboring MBs take part in the process, too. Each pixel value in a MB to be concealed is formed as a weighted sum of the closest boundary pixels of the selected adjacent MBs. The weight associated with each boundary pixel is relative to the inverse distance between the pixel to be concealed and the boundary pixel. The performance of the intra concealment method is shown in Fig. 7



Fig. 7. Intra frame error concealment.

In the motion vector recovery algorithm, the motion activity of the correctly received slices of the current picture is investigated first. If the average length of a motion vector component is smaller than a pre-defined threshold (currently 1/4 pixels), all the lost slices are copied from co-located positions in the reference frame. Otherwise, motion-compensated error concealment is used, and the motion vectors of the lost MBs are predicted as described in the following paragraphs. The image is scanned MB-column-wise from left and right edges to the center of the image. Consecutive lost MBs in a column are concealed starting from top and bottom of the lost area toward to center of the area. This processing order is used to ensure that lost MBs at the center of an image are concealed using as many neighboring concealed MBs as possible.

Each $8 \times 8$ luminance block of a MB to be concealed is handled separately. If a block has spatially adjacent blocks whose motion vectors are correctly received, these motion vectors and their reference pictures are used to obtain a candidate prediction block. If all the motion vectors in the adjacent blocks are lost, neighboring concealed motion vectors are used similarly. In addition, the spatially co-located block from the previous frame is always one of the candidates. The candidate prediction block whose boundary matching error is the smallest is chosen. The boundary matching error is defined as the sum of the pixel-wise absolute differences of the adjacent luminance pixels in the concealed block and its decoded or concealed neighbor blocks. The lost prediction error block is not concealed. The performance of the inter-frame concealment is shown in Fig. 8.

### C. Encoder Extensions

Let us now consider algorithms and rules for the appropriate selection of different encoding parameters according to the presentation in Fig. 5 for wireless conversational services. First, the rate control has to guarantee that the delay jitter is as small as possible. A good choice is to adapt for each frame to obtain almost constant encoding, but keep the QP within one frame constant. In the case that the highest QP cannot achieve the required bit rate, frame dropping is introduced; otherwise, the frame rate

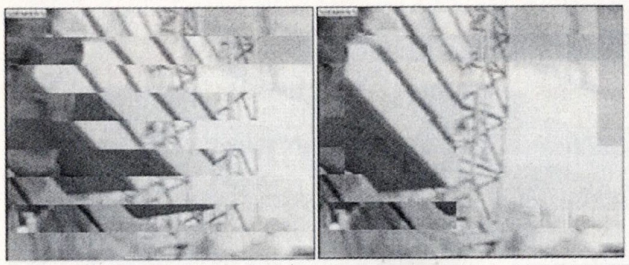


Fig. 8. Inter-frame error concealment.

is kept constant. Obviously, this results in varying quality depending of the complexity of the scene. Low-motion scenes with little complexity have higher quality than for example frames at the beginning of a sequence or at a scene cut.

As discussed in Section III, H.264/AVC provides a large flexibility on the selection of appropriate coding modes. The concept of selecting appropriate coding options in optimized encoder designs for many video coding standards is based on R-D optimization algorithms [51],[52]. The two cost terms "rate" and "distortion" are linearly combined and the mode is selected such that the total cost is minimized. This can be formalized by defining the set of selectable coding options for one MB as $\mathcal{O}$. Assuming the introduced distortion when encoding with a certain mode $o$ as $D(o)$ and the corresponding rate as $R(o)$, the rate-constrained mode decision selects the coding option $o^*$ such that the Lagrangian cost functional is minimized, i.e.,

$$o^* = \arg\min_{o \in \mathcal{O}} (D(o) + \lambda R(o)) \tag{1}$$

with $\lambda$ being the Lagrange parameter for appropriate weighting of rate and distortion. In the H.264/AVC test model, the Lagrangian mode selection is used for motion vector search as well as MB mode and reference frame selection. Then, for the distortion $D(o)$, the sum of squared sample differences (SSD) or the sum of absolute sample difference (SAD) are used, and for the rate $R(o)$, the number of bits for the encoding is used. The selection of the Lagrange parameter depends on the selected QPs (for details, see [53]). In addition, the mode selection might be extended by selecting the appropriate QP for each MB, as the QP can be changed at least in small ranges for each MB. This is not considered in the following.

As discussed in Section III-C, the tools for increased error resilience, in particular those to limit error propagation, do not significantly differ from those used for compression efficiency. Features like multiframe prediction or intra MB coding are not exclusively error-resilience tools. They are also used to increase coding efficiency in error-free environments providing a trade-off that is left to the encoder. This also means that bad decisions at the encoder can lead to poor results in coding efficiency or error resiliency or both. Therefore, the selection of the coding mode according to (1) is modified taking into account the influence of the random lossy channel. In the case of error-prone transmission, the distortion in (1) is replaced with the expected decoder distortion when encoding with mode o. Assuming that the encoder can access information about the channel statistics $C_{\pi(n)}$, the encoder can get an estimate of the expected decoder distortion $D_{n,m}(o, C_{\pi(n)})$ when encoding MB $m$ in frame $n$ at the decoder by the expected distortion as

$$D_{n,m}\left(o, C_{\pi(n)}\right) = \sum_{i=1}^{I} E_{C_{\pi(n)}} \left| s_{n,m,i} - \hat{s}_{n,m,i}\left(o, C_{\pi(n)}\right)\right|^2 \tag{2}$$

where the expectation is over the random process characterizing the channel $C_{\pi(n)}$. With the expected distortion measure, the mode selection for lossy channels is identical to that in (1) except for modified distortion term according to (2). However, the same parameter $\lambda$ [54] and the same set of possible coding options $\mathcal{O}$ are used.

This leaves the problem of computing the expected decoder distortion at the encoder which depends on the coding mode $o$, the channel statistics $C_{\pi(n)}$, and the applied error concealment in the decoder. The estimate of the expected sample distortion in packet loss environment has been addressed in several publications. For example, in [58], [55], or [56], methods to estimate the distortion introduced due the transmission errors and the resulting drift are presented. A similar approach has recently been proposed within the H.264/AVC standardization process which attempts to measure the drift noise between encoder and decoder [57]. In suboptimal approaches [55]–[57], the quantization noise and the distortion introduced by the transmission errors are linearly combined. The encoder keeps track of an estimated sample distortion and, therefore, requires additional complexity in encoder, which is dependent on the actual method chosen.

The most recognized out of these methods, called recursive optimal per-sample estimate (ROPE) algorithm [58], provides an estimation by keeping track of the first- and second-order moment of $\hat{s}_{n,m,i}, E\left\{\hat{s}_{n,m,i}(o, C_{\pi(n)})\right\}$, and $E\left\{\hat{s}^2_{n,m,i}(o, C_{\pi(n)})\right\}$, respectively. For H.263-like encoding ROPE can provide an exact estimate of the expected decoder distortion. As two moments for each sample have to be tracked in the encoder, the added complexity of ROPE is approximately twice the complexity of the decoder. However, the extension of the ROPE algorithm to H.264/AVC is not straightforward. The in-loop deblocking filter, the fractional sample motion accuracy, the complex intra prediction and the advanced error concealment require taking into account the expectation of products of samples at different positions to obtain an accurate estimation which makes the ROPE either infeasible or inaccurate in this case.

Therefore, a powerful yet complex method has been introduced into the H.264/AVC test model to estimate the expected decoder distortion [54]. Let us assume that we have $K$ copies of the random variable channel behavior at the encoder, denoted as $C_{\pi(n)}(k)$. Additionally, assume that the set of random variables $C_{\pi(n)}(k), k = 1, \cdots, K$ are *identically* and *independently* distributed (*i.i.d.*). Then, as $K \to \infty$, it follows by the strong law of large numbers that

$$\frac{1}{K}\sum_{k=1}^{K} \left| s_{n,m,i} - \hat{s}_{n,m,i}\left(C_{\pi(n)}(k)\right)\right|^2 = E_{C_{\pi(n)}} \left| s_{n,m,i} - \hat{s}_{n,m,i}\left(C_{\pi(n)}\right)\right|^2 \tag{3}$$

holds with probability 1. An interpretation of the left-hand side leads to a simple solution of the previously stated problem to estimate the expected sample distortion. In the encoder, $K$ copies of the random variable channel behavior and the decoder are operated. The reconstruction of the sample value depends on the channel behavior $C_{\pi(n)}(k)$ and the decoder including error concealment. The $K$ copies of channel and decoder pairs in the encoder operate independently. Therefore, the expected distortion at the decoder can be estimated accurately in the encoder if $K$ is chosen large enough. However, the added complexity in the encoder is obviously at least $K$ times the decoder complexity. For details on the implementation as well as comparison of suboptimal modes based on ROPE, we refer to [54].

It was later demonstrated in [59] and [60] that a technique known as isolated regions, which combines periodical intra MB coding and limitation of inter prediction, can provide competitive simulation results compared to the error-robust MB mode selection of H.264 test model, and the best simulation results were obtained when the two MB mode-selection algorithms were combined.

The packet size is usually upper bounded by a MTU size of the underlying network. However, if bit errors cause entire packet losses, it might be an advantage to reduce the packet size. Conservative approaches limit the packet or (in our simulation environment) slice size to a fixed number of MBs, e.g., a line of MBs is transmitted in one packet. However, this makes especially important image parts very susceptible to packet losses as for example suddenly appearing objects need intra coding or high motion areas require an extensive motion vector rate. A more suitable, but still conservative, approach is the introduction of slice boundaries such that slices have almost the same length in number of bytes. This makes all packets similarly susceptible to bit errors. This method has been proven to work quite well already for MPEG-4 and can also be applied with the flexible encoding framework of H.264/AVC.

### D. Exploiting Feedback in H.264/AVC

Multiple reference frames used in error-prone environment can most successfully be combined with a feedback channel. Due to the bidirectional nature of conversational applications, it is common that the encoder has knowledge of the experienced channel at the decoder, usually with a small delay. In our framework this can be expressed by the knowledge of a $d$-frame delayed version of the random channel $C_{\pi(n-d)}$ at the encoder. This characteristic can be conveyed from the decoder to the encoder by acknowledging correctly received slices (ACK), sending a not-acknowledge message (NAK) for missing slices or both types of messages. In general, it can be assumed that the reverse channel is error-free and the overhead is negligible. In common data transmission applications, the lost packets would obviously be re-transmitted. However, in a low-delay environment, this is not feasible, but the observed channel characteristic are still useful at the encoder even if the erroneous frame has already been decoded and concealed. The support of these techniques is out of the focus of the standardization of H.264, and a full support of the presented concepts might not be possible with current transport and control protocols. However, it seems worth to mention and discuss these concepts in here in detail as it shows the flexibility of H.264 as well as it provides motivation for inclusion of feedback in system designs.

In previous standards and transport environments similar approaches have already been discussed, usually limited by the reduced syntax capabilities of the video standard. A simple yet powerful approach suitable for video codecs using just one reference frame such as MPEG-2, H.261, or H.263 version 1 has been introduced in [61] and [62] under the name *error tracking*. When receiving a NAK on parts of frame $n-d$ or the entire frame $n-d$, the encoder attempts to track the error to obtain an estimate of the quality of frame $n-1$, which serves as reference for frame $n$. Appropriate actions after having tracked the error are for example presented in [56], and [61]–[64]. Note that with this concept, error propagation in frame $n$ is only removed if frames $n-d+1,\ldots,n-1$ have been received at the decoder without any error.

A technique addressing the problem of continuing error propagation has been introduced, among others, in [65]–[67] under the acronym *NEWPRED*. Based on these early nonstandard compliant solutions in H.263 Annex N [68], a reference picture selection (RPS) for each GOB is specified such that the NEWPRED technique can be applied. RPS can be operated in two different modes. In the negative acknowledgment mode (NAM), the encoder only alters its operation in the case of reception of a NAK. Then, the encoder attempts to use an intact reference frame for the erroneous GOBs. To completely eliminate error propagation, this mode has to be combined with independent segment decoding (ISD) according to Annex R of H.263 [68]. In the positive acknowledgment mode (PAM), the encoder is only allowed to reference confirmed GOBs. If no GOBs are available to be referenced, intra coding has to be applied. NEWPRED allows to completely eliminate error propagation in frame $n$, even if additional errors have occurred in frames $n-d+1,\ldots,n-1$.

The flexibility provided in H.263 Annex U [68] and especially H.264/AVC to select the MB mode and reference frames on MB or subMB basis allows incorporating NEWPRED in a straightforward manner [56]. Therefore, let us define three different states of transmitted packets at the encoder: ACK, NAK, and outstanding acknowledgment (OAK). Then, based on the exploitation of these messages in the encoder, we discuss three modes which can elegantly be integrated into the R-D optimized mode selection according to (1).

*1) Feedback Mode 1: Restricting Reference Areas to ACK:* In this case, the reference area is restricted to frames or slices which have been acknowledged. This can be formalized in the context of (1) by applying the encoding distortion $D(o)$ in (1), but altering the accessible coding options $\mathcal{O}$ such that only acknowledged areas can be used for reference. In addition, if no reference area is available, or if no satisfying match is found in the allowed area, intra coding is applied. Although in the presentation of a single frame an error might be visible, error propagation and reference frame mismatch between encoder can be completely avoided independent of the error concealment applied at the decoder if correctly decoded samples are not altered. However, in the used test model JM1.7, the deblocking filter operation in the motion-compensation

loop is applied over slice boundaries, which restricts the area to be referenced significantly, or a complete removal of encoder and decoder mismatch is not possible. Although the influence of this mismatch is, in general, negligible, in the final design of H.264/AVC the deblocking filter can adaptively be switched on and off at slice boundaries to allow mismatch-free operation.

*2) Feedback Mode 2: Referencing ACK and Error Concealed NAK:* In this case, the reference area is again restricted; however, in addition to acknowledged areas, also the areas which the decoder signaled as lost can be referenced. Therefore, the reference frames in the encoders multi-frame frame buffer are updated with reception of each ACK and NAK by applying the identical error concealment as the decoder applies. This is obviously very critical, as the error concealment is—for good reasons—nonnormative in common video coding standards including H.264/AVC. Only if the encoder is aware of the decoder's error concealment by any external means, this mode can provide benefits compared to feedback mode 1. In the context of the mode decision in (1), we have again a restricted set of coding modes $\mathcal{O}$ and, in addition, the encoding distortion is replaced by the deterministic decoder distortion.

*3) Feedback Mode 3: Unrestricted Reference Areas With Expected Distortion Updating:* In [56] and [58], techniques have been proposed which allow combining the error-resilient MB mode selection with feedback information. In this case, the expected decoder distortion computation is altered such that, for all packets $1, \ldots, \pi(n-d)$, the channel is deterministic at the encoder, and the expected distortion is computed for the packets with status OAK. In our case, packets containing MBs in frames $\pi(n-d)+1, \ldots, \pi(n)$ are random. The set of selectable coding options $\mathcal{O}$ is not altered compared to pure coding efficiency mode selection. This method is especially beneficial compared to mode 1 and 2, if the feedback is significantly delayed. In the case of the multiple decoder approach, this can be integrated by applying feedback mode 2 not only to the reference frames, but also to all decoders in the encoder. In combination with ROPE, however, the complexity of this method increases since the moments of the previous $d$ frames have to be re-tracked [58].

## V. Selected Simulation Results for Different System Concepts

### A. Simulation Conditions and Evaluation Criteria

In the following, we will present simulation results based on the test conditions that show the influence of the selection of different error-resilience features for the quality of the decoded images. For all following tests the H.264/AVC test model software version JM1.7 is used. Note that in the final standard [1], the zig-zag scanning and run-length coding is replaced by context-adaptive variable length codes (CVLC). All UVLCs are replaced by CVLC, which are adapted to the statistics of different syntax elements. In addition, quantizer values have been shifted. However, all changes from JM1.7 to the final standard are of little relevance to the presented results and conclusions in this paper.

The reported PSNR is the arithmetic mean over the decoded luminance PSNR over all frames of the encoded sequence and over 100 transmission and decoding runs. The 100 starting positions for the error patterns have been selected such that they are almost equally distributed over the error pattern. For all comparable results, the same starting positions and, therefore, the same channel statistics have been applied. In addition, we present results for the cumulative distribution of the decoded luminance PSNR for each frame, i.e., the likelihood that the PSNR of the frames of the sequence is smaller than the value on the x axis. This shows the variance in the decoded quality. It is assumed that the high-level syntax parameters have been transmitted in advance and out-of-band applying a reliable setup protocol. The NAL overhead, the RTP/UDP/IP overhead after RoHC, and the link layer overhead is taken into account in the bit-rate constraints according to [25].

For the following simulations, we concentrate on test case 5 from [25], which includes the QCIF test sequence "Foreman" (30 Hz, 300 frames) at a constant frame rate of 7.5 Hz at bit-error pattern 3 according to Table II, i.e., a mobile user at 3 km/h and maximum bit-rate 64 kbit/s. This is the most critical case in terms of error probability, additional test results will be made available online[2]. For the following tests, entropy coding based on the UVLC and only one reference frame has been applied, if not stated otherwise. The encoded sequences are I-P-P-P... sequences; B-pictures are excluded due to the unacceptable delay involved in the encoding process. Note that due to the repeated decoding of an encoded file, every 75th frame is an I-frame, i.e., an I-frame occurs every 10 s. Constrained intra has been used to avoid error propagation from inter MBs to intra MBs. For all encoding runs, R-D optimized mode selection and motion vector selection according to (1) have been used. The distortion $D(o)$ and the set of coding modes $\mathcal{O}$ is appropriately selected according to the applied features. In the case of using the expected decoder distortion, the number of decoders operated in the encoder has been fixed to $K = 100$. Unless stated otherwise, the error concealment in the decoder is based on the advanced error concealment as presented in Section IV-B, whereas the multiple decoders in the encoder always apply previous frame concealment. This reduces encoding complexity significantly and results only in negligible performance losses for the investigated cases.

### B. Low-Delay Rate Control

Version JM1.7 of the H.264/AVC test model encoder which is used in the experiments does not include a rate control to achieve a constant bit rate for wireless conversational services. Moreover, the rate control introduced in later versions of H.264/AVC test model encoder is not suitable for constant rate encoding. For this reason we have added a rate control which provides an almost constant bit-rate encoding for each frame by adapting the QP for each frame appropriately. Therefore, before we investigate the error-resilience features in H.264/AVC, we will first focus on the effect of bit-rate control that is necessary for constant bit-rate conversational applications. Fig. 9 shows the cumulative distribution of the encoding PSNR for the applied rate control and a fixed QP such that both encoded files result in the

[2]For additional simulation results, we refer to http://www.lnt.ei.tum.de/~stockhammer.

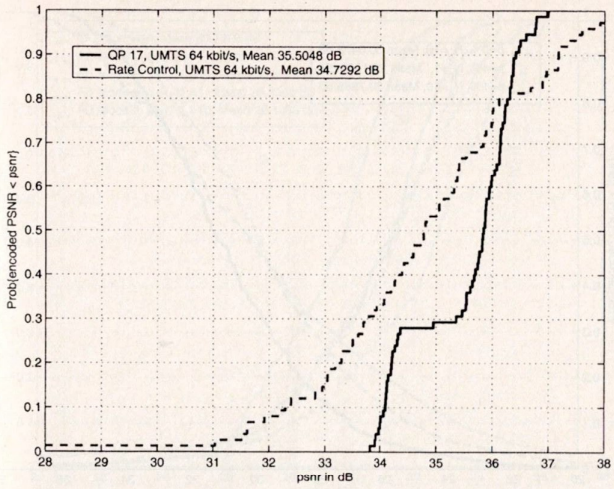



Fig. 9. Cumulative distribution of encoding luminance PSNR for each frame for constant QP 17 and constant bit rate such that rate constraint of 64 kbit/s for UMTS is fulfilled.

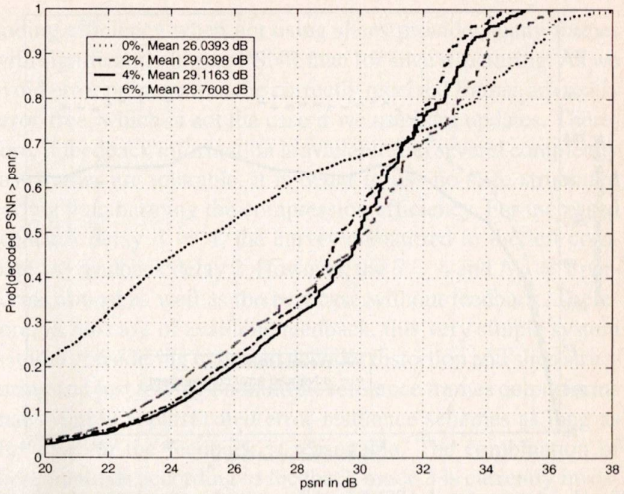



Fig. 10. Cumulative distribution of decoded PSNR for different NAL unit erasure rates for the estimation of the expected distortion in the encoder.

same total bit-rate, and, additionally the file can be transmitted over the 64 kbit/s link including NAL unit, packet, and link layer overhead. However, the fixed QP results in an extremely varying data rate and, therefore, the introduced delay jitter is not acceptable. In the encoding process, no error-resilience tools have been used, i.e., one frame is encapsulated into one NAL unit, for $D(o)$ the encoding distortion is applied, and the set of coding modes $\mathcal{O}$ is unrestricted.

The average PSNR for the rate control is about 0.8 dB below the PSNR for the fixed QP. In addition, the probability for low encoded PSNR is significantly higher as for complex scenes the QP has to be adapted appropriately. Therefore, it can be seen that the rate control necessary for conversational applications involves an inherent decrease in quality if the average PSNR is the measure of interest.

### C. $R - E\{D\}$-Optimized MB Mode Selection

In this section we investigate the performance of the system in case that the channel statistics are taken into account into the selection of the coding options in the encoder. For this purpose we replace the encoding distortion $D(o)$ in (1) by the expected decoder distortion assuming a channel producing independent packet losses with probability $p$. As we use the mapping of one frame to one transport packet and apply the strict rate control producing almost a constant number of bytes for each encoded frame, the size of each frame results in roughly 1000 bytes. The corresponding loss rate for packets of this size for bit-error pattern 3 is approximately 4%–5% according to Fig. 4. Fig. 10 shows the cumulative distribution of decoded PSNR for different NAL unit erasure rates $p = \{0, 0.02, 0.04, 0.06\}$ for the estimation of the expected distortion in the encoder. Looking at the results for the average PSNR, it can be seen that the introduction of loss-aware R-D optimization ($p > 0$) increases the decoded quality significantly compared to the results with pure encoding distortion ($p = 0$). The average PSNR increases by at least 3 dB when compared to the to pure R-D optimization. The advantage of $R-E\{D\}$ optimization is even more evident when looking at the cumulative distribution of the different strategies. Whereas in the case of no error resilience, the probability of bad frames (frames below 22 dB) is at an unacceptable ratio of about 30%, for loss-aware coding, this is reduced significantly to less than 8%. It is also obvious that if the expected error rate matches the experienced error rate on the channel, the performance is optimal (see $p = 4\%$). However, it can also be seen that a mismatch in the expected error rate in the encoder does not have significant influence. The performance of $p = 2\%$ and $p = 6\%$ is only slightly inferior to the matching expected error rate. Therefore, a rough estimation of the expected decoder distortion at the encoder seems to be good enough for a good mode selection. Note the significant loss in average PSNR for wireless transmission compared to the error-free transmission according to Fig. 9 of more than 5 dB.

### D. Slices and Error Concealment

The introduction of slices in the encoding has two beneficial aspects when transmitting over wireless channels, but adversely affects the coding efficiency due to increased packet overhead and reduced prediction within one frame, as e.g., motion vector prediction and spatial intra prediction is not allowed over slice boundaries. The two positive effects with the introduction of slices are the reduced error probability of shorter packets (see Fig. 4) and, the re-synchronization possibility within one frame. The latter technique allows restarting the decoding process at each slice and, in addition, it allows applying advanced error concealment as for example presented in Section IV-B. However, for packet-lossy transmission over the Internet, the introduction of slices does not, in general, provide gains in the decoded quality [69] as long as the slice size is below the MTU size, as the loss of a packet is then independent of its length. This is different for wireless transmission: Fig. 11 shows the average decoded PSNR for different number of packets per frame $N_p$ and MB mode decision with encoding distortion. The experimental conditions are similar to the above section otherwise. For a given number of packets per frame, the size of each packet is selected such that the packets roughly have the same number of bytes. This makes their susceptibility to bit errors almost identical.

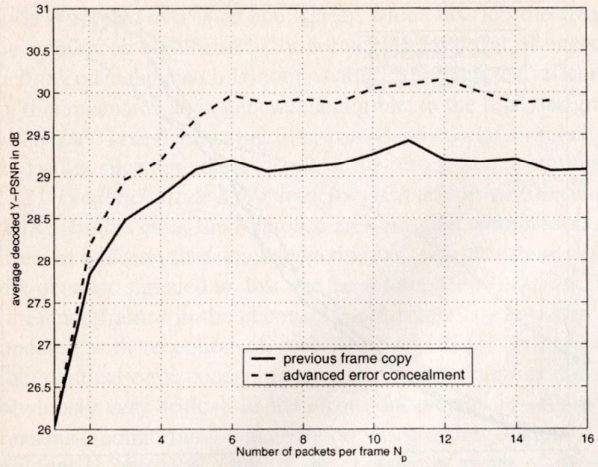



Fig. 11. Average decoded PSNR for different number of packets per frame and different error-concealment schemes; mode selection with encoding distortion.

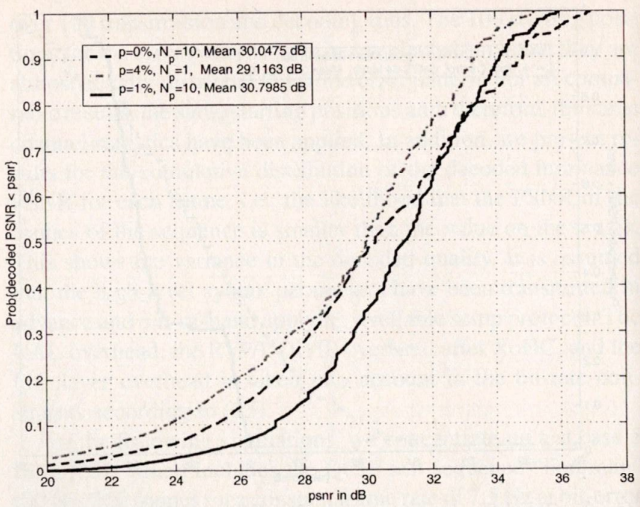



Fig. 12. Cumulative distribution of decoded PSNR for different error-resilience strategies: $R - E\{D\}$-optimized intra updates with and without slice structuring and delay $d = 2$ for different assumed loss probabilities $p$.

The average decoded PSNR is shown for previous frame copy error concealment and advanced error concealment according to Section IV-B. The benefit of introducing slice structuring in the encoding process is obvious from the results. Compared to the "one frame—one packet" mode indicated by $N_p = 1$ the introduction of shorter packets increases the decoded quality significantly for both error concealment methods. For about $N_p = 6$, the curve flattens out and decreases again for higher $N_p > 12$ due to increasing packet overhead and the reduced compression efficiency. Although a clear maximum cannot be determined, for the wireless environment according to our test conditions, a reasonable number of packets per frame is about 10. The resulting packet size in this case is in the range of 100 bytes. However, note that for the simulations, RoHC was applied, which reduces the typical IP/UDP/RTP overhead from 40 bytes to about 3 bytes and, therefore, the packetization overhead is less significant. The benefits of the advanced error concealment are also obvious from Fig. 11. As can be seen, the gains for advanced error concealment increase with increasing number of packets, as better concealment is possible due to increased number of neighboring MBs in case of losing a single slice.

For the experiments in Fig. 11, no explicit means to reduce the error propagation have been used. However, we can obviously combine the MB mode selection based on the expected decoder distortion with the slice structured coding. As indicated, the loss rate for decreased packet size decreases compared to the single packet for one frame. The loss rate can be estimated by dividing the approximated number of bytes for each frame, roughly 1000 bytes, by the number of packets. The loss probability can then again be estimated with Fig. 4 using the resulting average packet length. The combination of slice-structured coding and adaptive intra MB updates has been investigated and a comparison with the best cases of the previous two system designs is provided in Fig. 12 based on the cumulative distribution of the decoded PSNR. For the slice-structured coding with encoding distortion ($p = 0\%$) the number of packets is selected as $N_p = 10$. For the expected decoder distortion without slice-structuring ($N_p = 1$), the adapted loss-rate $p = 4\%$ is chosen. Finally, for the combination of slice-structured coding and channel-adaptive intra updates, the number of packets per frame is selected as $N_p = 10$ and, therefore, the appropriate loss probability to compute the expected decoder distortion according to Fig. 4 is about $p = 1\%$.

The average decoded PSNR indicates that an optimized combination of both error-resilience schemes outperforms each of the presented error-resilience schemes significantly. The result that slice-structured coding is superior to intra updates cannot be generalized for all sequences. The repeated I-frame insertion and the camera pan in the test sequence "Foreman" results in a significant amount of intra information, even if only the encoding distortion is chosen in the mode selection. This might change for different or longer sequences with less intra information. From the cumulative distribution, it can be observed that the probability for bad frames below 22 dB in PSNR is almost vanishing for the combined mode. The presented results indicate that slices in combination with advanced error concealment significantly outperform the single packet for one frame approach. However, the loss compared to error-free transmission is still about 4 dB in average PSNR.

From the results, it can be conjectured that for wireless transmission as investigated in this case, other approaches, which reduce the artifacts within one frame, might provide additional benefits. This includes concepts such as FMO, slice interleaving, or even generic forward-error correction in combination with a NAL unit fragmentation scheme as recently introduced in the draft RTP payload specification for H.264/AVC [19]. Also, data partitioning with appropriate unequal error protection might enhance the quality of the decoded video. In addition, it is conjectured from the results that a better adaptation of the link layer error protection scheme with appropriate interleaving could increase the overall system performance. These features are currently investigated and are subject of future work.

### E. Exploiting Feedback in Video Encoding

Finally, we investigate a system which exploits multiple reference frames and network feedback. We restrict our simula-

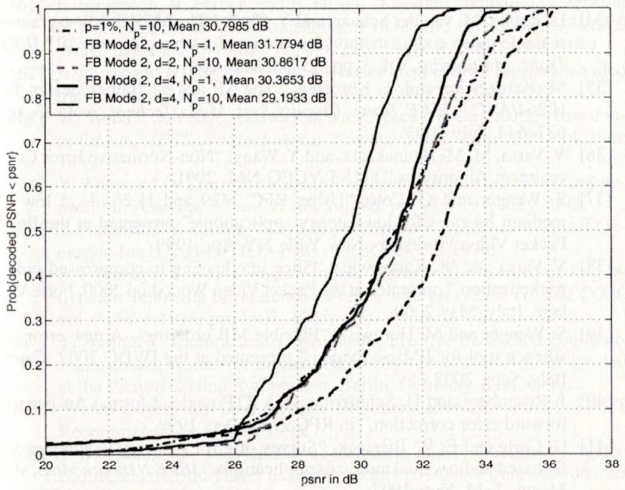



Fig. 13. Cumulative distribution of decoded PSNR for different error-resilience strategies: R-D-optimized intra updates with slice structuring, and feedback mode 2 with and without slice structuring for delay $d = 2$ and $d = 4$.

tion results to feedback mode 2 according to the presentation in Section IV-D with advanced error concealment. There are two reasons for this. On the one hand, the deblocking and mismatch problem present for JM1.7 can be avoided. On the other hand, it has been shown previously that feedback mode 1 and feedback mode 2 result in almost identical performance [70]. Therefore, we chose feedback mode 2 due to the simpler implementation, at least in our simulation environment. Feedback mode 3 is excluded as, on the one hand, the complexity of this mode is rather high and, on the other hand, as we operate with low feedback delays, the expected benefits are only marginal. In contrast to the previous simulations, we use five reference frames for the feedback mode. Fig. 13 shows the cumulative distribution of decoded PSNR for different error-resilience strategies: R-D-optimized intra updates with slice structuring, as well as feedback mode 2 with and without slice structuring for delay $d = 2$ and $d = 4$ frames, which corresponds to a round-trip time of about 250 and 500 ms, respectively.

Let us focus on the delay $d = 2$ case first. The results indicate that the optimized intra mode with slice structuring and MB mode selection with expected decoder distortion and feedback mode 2 perform very similar based on the cumulative distribution and the average decoded PSNR. The feedback mode might still be beneficial in this case as the complex estimation of the decoder distortion is not necessary for the feedback case. However, much more interesting is the case with feedback and no slice structuring. In contrast to the case without feedback (see Fig. 12), the renouncement of slice-structured coding provides a significantly higher average decoded PSNR. Initially, this is obviously surprising, as packet loss rate is still much lower when several slices are used and also the visual effects for a decoded frame when losing a single slice should be lower than in case of losing an entire frame. The first effect can indeed be observed from the cumulative distribution. The probability of bad frames (PSNR below 22 dB) is higher for $N_p = 1$ than for $N_p = 10$. However, in the case of no errors, the increased coding efficiency when not using slices provides many frames with significantly higher PSNR than for slice structuring. As we avoid error propagation, the correctly received frames are really error-free, which is not the case if we use intra updates. Therefore, if feedback information is available and several completely lost frames are tolerable, it is better to use no slice structured coding than harming the compression efficiency. For increased feedback delay $d = 4$, the curves are shifted to the left compared to feedback delay 2. However, the $d = 4$ and $N_p = 1$ performs almost as well as the best case without feedback. Therefore, in the case of available feedback, this very simple system without considering expected decoder distortion and slice structuring and just relying on multiple reference frames outperforms many highly sophisticated error-resilience schemes as long as the delay of the feedback is reasonable. The combination of these methods according to feedback mode 3 is currently investigated and should allow adaptively selecting the best methods, however, with significantly increased encoding complexity.

## VI. Conclusions

H.264/AVC promises some significant advances of the state-of-the-art of standardized video coding in mobile applications. In addition to excellent coding efficiency, the design of H.264/AVC also takes into account network adaptation providing large flexibility for its use in wireless applications. The tools provided in H.264/AVC for error resilience do not necessarily differ from the compression efficiency features such as intra MBs or multiple reference frames. However, in the case of error-prone transmission, the selection methods have to be changed by using the expected decoder distortion or by restricting the set of accessible coding options. In experimental results based on common test conditions, it has been shown that in case without any feedback, several slices in combination with channel-adaptive R-D optimized mode selection is a promising approach. In this case, further investigation with advanced error-resilience tools such as flexible MB ordering, data partitioning, and generic forward error correction, might provide benefits. However, in the case of available feedback, the application of multiple reference frames to exclude error propagation without slice structuring provides excellent results.

## Acknowledgment

The authors would like to thank T. Oelbaum, D. Kontopodis, Y.-K. Wang, V. Varsa, and A. Hourunranta for implementing and testing parts of the algorithms, V. Varsa and G. Liebl for providing the test conditions and software simulator, S. Wenger, N. Färber, K. Stuhlmüller, E. Steinbach, and B. Girod for useful discussions, and JVT for the collaborative work and the technically outstanding discussions and contributions.

## References

[1] "Final committee draft: Editor's proposed revisions," in *Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG*, T. Wiegand, Ed., Feb. 2003, JVT-F100.

[2] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 560–576, July 2003.

[3] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 688–703, July 2003.
[4] "Video Coding for Low Bitrate Communication, Version 1," ITU-T, ITU-T Recommendation H.263, 1995.
[5] "Coding of Audio-Visual Objects—Part 2: Visual," ISO/IEC JTC1, ISO/IEC 14 496-2 (MPEG-4 visual version 1), 1999.
[6] "Multimedia Messaging Service (MMS); Media Formats and Codecs," 3GPP Technical Specification 3GPP TR 26.140.
[7] S. Wenger, M. Hannuksela, and T. Stockhammer, "Identified H.26L Applications," ITU-T SG 16, Doc. VCEG-L34, Eibsee, Germany, 2001.
[8] "Codec for Circuit Switched Multimedia Telephony Service; General Description," 3GPP Technical Specification 3GPP TR 26.110.
[9] "Packet Switched Conversational Multimedia Applications; Default Codecs," 3GPP Technical Specification 3GPP TR 26.235.
[10] "Transparent End-to-End Packet Switched Streaming Service (PSS); RTP Usage Model," 3GPP Technical Specification 3GPP TR 26.937.
[11] "Multimedia Messaging Service (MMS); Media Formats and Codecs," 3GPP Technical Specification 3GPP TR 26.140.
[12] "Multimedia Broadcast/Multicast Services," 3GPP Technical Specification 3GPP TR 29.846.
[13] M. Horowitz, A. Joch, F. Kossentini, and A. Hallapuro, "H.264/AVC decoder complexity analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 704–716, July 2003.
[14] V. Lappalainen, A. Hallapuro, and T. D. Hämäläinen, "Complexity of optimized H.264/AVC video decoder implementation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 717–725, July 2003.
[15] "High Speed Downlink Packet Access (HSDPA); Overall UTRAN Description," 3GPP Technical Specification 3GPP TR 25.855.
[16] H. Holma and A. Toskala, Eds., *WCDMA for UMTS: Radio Access For Third Generation Mobile Communications*. New York: Wiley, 2000.
[17] *Narrow-Band Visual Telephone Systems and Terminal Equipment, Rev. 4*, ITU-T Recommendation H.320, 1999.
[18] "Generic Coding of Moving Pictures and Associated Audio Information," ISO/IEC International Standard 13 818, 1994.
[19] S. Wenger, T. Stockhammer, and M. M. Hannuksela, "RTP payload format for H.264 video," in *Internet Draft, Work in Progress*, Mar. 2003, Draft-wenger-avt-rtp-h264-01.txt.
[20] "3rd GPP; Technical Specification Group Core Network; IP Multimedia Call Control Protocol Based on SIP and SDP," 3GPP Technical Specification 3GPP TS 24.229.
[21] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 645–656, July 2003.
[22] D. Lindberg, "The H.324 multimedia communication standard," *IEEE Commun. Mag.*, vol. 34, pp. 46–51, Dec. 1996.
[23] H. Hannu, L.-E. Jonsson, R. Hakenberg, T. Koren, K. Le, Z. Liu, A. Martensson, A. Miyazaki, K. Svanbro, T. Wiebke, T. Yoshimura, and H. Zheng, "RObust header compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed," in RFC 3095, July 2001.
[24] "Common conditions for video performance evaluation in H.324/M error-prone systems, VCEG (SG16/Q15)," in Ninth Meeting, Redbank, NJ, Oct. 1999, ITU-T Q15-I-60.
[25] G. Roth, R. Sjöberg, G. Liebl, T. Stockhammer, V. Varsa, and M. Karczewicz, "Common Test Conditions for RTP/IP Over 3GPP/3GPP2,", Austin, TX, ITU-T SG16 Doc. VCEG-M77, 2001.
[26] "Radio Link Control (RLC) Protocol Specification," 3GPP Technical Specification 3GPP TS 25.322.
[27] S. S. Hemami, "Robust image transmission using resynchronizing variable-length codes and error concealment," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 927–939, June 2000.
[28] J. Ribas-Corbera, P. A. Chou, and S. Regunathan, "A generalized hypothetical reference decoder for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 674–687, July 2003.
[29] H. Jenkac, T. Stockhammer, and G. Kuhn, "Streaming media in variable bit-rate environments," presented at the Packet Video Workshop, Nantes, France, Apr..
[30] B. Girod, M. Kalman, Y. J. Liang, and R. Zhang, "Advances in video channel-adaptive streaming," in ICIP 2002, Rochester, NY, Sept. 2002.
[31] E. G. Steinbach, N. Färber, and B. Girod, "Adaptive play-out for law latency video streaming," presented at the ICIP 2001, Thessaloniki, Greece, Oct. 2001.
[32] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, submitted for publication.
[33] Y. J. Liang and B. Girod, "Rate-distortion optimized low—Latency video streaming using channel-adaptive bitstream assembly," presented at the ICME2002, Lausanne, Switzerland, Aug. 2002.
[34] H. Radha, M. van der Schaar, and Y. Chen, "The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP," *IEEE Trans. Multimedia*, vol. 3, pp. 53–68, Mar. 2001.
[35] M. Karczewicz and R. Kurçeren, "The SP and SI frames design for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 637–644, July 2003.
[36] V. Varsa, M. M. Hannuksela, and Y. Wang, "Non-Normative Error Concealment Algorithms," ITU-T VCEG-N62, 2001.
[37] S. Wenger and G. Cotê, "Using RFC 2429 and H.263+ at low to medium bit-rates for low-latency applications," presented at the Proc. Packet Video Workshop, New York, NY, Apr. 1999.
[38] V. Varsa and M. Karczewicz, "Slice interleaving in compressed video packetization," presented at the Packet Video Workshop 2000, Forte Village, Italy, May 2000.
[39] S. Wenger and M. Horowitz, "Flexible MB ordering—A new error resilience tool for IP-based video," presented at the IWDC 2002, Capri, Italy, Sept. 2002.
[40] J. Rosenberg and H. Schulzrine, "An RTP payload format for generic forward error correction," in RFC 2733, Dec. 1999.
[41] G. Carle and E. W. Biersack, "Survey of error recovery techniques for IP-based audio-visual multicast applications," *IEEE Network Mag.*, vol. 11, pp. 2–14, Nov. 1997.
[42] U. Horn, K. Stuhlmüller, M. Link, and B. Girod, "Robust internet video transmission based on scalable coding and unequal error protection," *IEEE Trans. Image Processing*, vol. 15, pp. 77–94, Sept. 1999.
[43] Q. F. Zhu and L. Kerofsky, "Joint source coding, transport processing, and error concealment for H.323-based packet video," *Proc. SPIE VCIP*, vol. 3653, pp. 52–62, Jan. 1999.
[44] P. Haskell and D. Messerschmitt, "Resynchronization of motion-compensated video affected by ATM cell loss," *Proc. IEEE ICASSP*, vol. 3, pp. 545–548, 1992.
[45] J. Liao and J. Villasenor, "Adaptive intra update for video coding over noisy channels," *Proc. ICIP*, vol. 3, pp. 763–766, Oct. 1996.
[46] S. Wenger, G. Knorr, J. Ott, and F. Kossentini, "Error resilience support in H.263+," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 867–877, Nov. 1998.
[47] P. Salama, N. B. Shroff, and E. J. Delp, "Error concealment in encoded video," in *Image Recovery Techniques for Image Compression Applications*. Norwell, MA: Kluwer, 1998.
[48] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," in *Proc. ICASSP*, vol. 5, Mar. 1993, pp. 417–420.
[49] V. Varsa, M. M. Hannuksela, and Y.-K. Wang, "Non-Normative Error Concealment Algorithms," ITU-T VCEG-N62, 2001.
[50] Y.-K. Wang, M. M. Hannuksela, V. Varsa, A. Hourunranta, and M. Gabbouj, "The error concealment feature in the H.26L test model," in *Proc. ICIP*, vol. 2, Sept. 2002, pp. 729–732.
[51] 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 Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 182–190, Apr. 1996.
[52] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Mag.*, vol. 15, pp. 74–90, Nov. 1998.
[53] T. Wiegand and B. Girod, "Lagrangian multiplier selection in hybrid video coder control," presented at the Proc. ICIP 2001, Oct. 2001.
[54] T. Stockhammer, D. Kontopodis, and T. Wiegand, "Rate-distortion optimization for H.26L video coding in packet loss environment," presented at the Packet Video Workshop 2002, Pittsburgh, PA, Apr. 2002.
[55] G. Cote, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 952–965, Dec. 2000.
[56] T. Wiegand, N. Färber, K. Stuhlmüller, and B. Girod, "Error-resilient video transmission using long-term memory motion-compensated prediction," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1050-–1050-, Dec. 2000.
[57] C. W. Kim, D. W. Kang, and I. S. Kwang, "High-complexity mode decision for error prone environment," in JVT-C101, May 2002.
[58] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 966–976, 2000.
[59] Y.-K. Wang and M. M. Hannuksela, "Error-robust video coding using isolated regions," in JVT-C073, May 2002.
[60] Y.-K. Wang, M. M. Hannuksela, and M. Gabbouj, "Error-robust inter/intra mode selection using isolated regions," in Proc. Int. Packet Video Workshop 2003, Nantes, France, Apr. 2003.

[61] E. Steinbach, N. Färber, and B. Girod, "Standard compatible extension of H.263 for robust video transmission in mobile environments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 872–881, Dec. 1997.
[62] B. Girod and N. Färber, "Feedback-based error control for mobile video transmission," *Proc. IEEE*, vol. 97, pp. 1707–1723, Oct. 1999.
[63] W. Wada, "Selective recovery of video packet losses using error concealment," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 807–814, June 1989.
[64] Y. Wang and Q. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, pp. 974–997, May 1998.
[65] T. Nakai and Y. Tomita, "Core Experiments on Feedback Channel Operation for H.263+," ITU-T SG15 LBC96-308, 1996.
[66] S. Fukunaga, T. Nakai, and H. Inoue, "Error resilient video coding by dynamic replacing of reference pictures," in *Proc. IEEE GLOBECOM*, vol. 3, Nov. 1996, pp. 1503–1508.
[67] Y. Tomita, T. Kimura, and T. Ichikawa, "Error resilient modified inter-frame coding system for limited reference picture memories," presented at the Picture Coding Symposium, Berlin, Germany, Sept. 1997.
[68] "Video Coding for Low Bit-Rate Communication, Version 1," ITU-T Recommendation H.263, 1995.
[69] T. Stockhammer, T. Wiegand, T. Oelbaum, and F. Obermeier, "Video coding and transport layer techniques for H.264-based transmission over packet-lossy networks," presented at the ICIP, Barcelona, Spain, Sept. 2003.
[70] T. Stockhammer and S. Wenger, "Standard-compliant enhancement of JVT coded video for transmission over fixed and wireless IP," presented at the IWDC 2002, Capri, Italy, Sept. 2002.



**Thomas Stockhammer** received the Diplom.-Ing. degree in electrical engineering in 1996 from the Munich University of Technology (TUM), Munich, Germany, where he is currently working toward the Dr.-Ing. degree in the area of source and video transmission over mobile and packet-lossy channels.

In 1996, he visited Rensselear Polytechnic Institute (RPI), Troy, NY to perform his diploma thesis in the area of combined source channel coding for video and coding theory. There he began research in video transmission and combined source and channel coding. In 2000, he was Visiting Researcher in the Information Coding Laboratory, University of San Diego at California (UCSD). Since then, he has published several conference and journal papers and holds several patents. He regularly participates and contributes to different standardization activities, e.g., ITU-T H.324, H.264, ISO/IEC MPEG, JVT, and IETF. He acts as a member of several technical program committees, as Reviewer for different journals, and as an Evaluator for the European Commission. His research interests include joint source and channel coding, video transmission, system design, rate-distortion optimization, information theory, and mobile communications.



**Miska M. Hannuksela** received the M.S. degree in engineering from Tampere University of Technology, Tampere, Finland, in 1997.

He is currently a Research Manager in the Visual Communications Laboratory, Nokia Research Center, Tampere, Finland. From 1996 to 1999, he was a Research Engineer with Nokia Research Center in the area of mobile video communications. From 2000 to 2003, he was a Project Team Leader and a specialist in various mobile multimedia research and product projects in Nokia Mobile Phones. He has been an active participant in the ITU-T Video Coding Experts Group since 1999 and in the Joint Video Team of ITU-T and ISO/IEC since its foundation in 2001. He has co-authored more than 80 technical contributions to these standardization groups. His research interests include video error resilience, scalable video coding, and video communication systems.



**Thomas Wiegand** received the Dr.-Ing. degree from the University of Erlangen-Nuremberg, Erlangen-Nuremberg, Germany, in 2000 and the Dipl.-Ing. degree in electrical engineering from the Technical University of Hamburg-Harburg, Hamburg-Harburg, Germany, in 1995.

He is the Head of the Image Communication Group in the Image Processing Department, Fraunhofer–Institute for Telecommunications – Heinrich Hertz Institute (HHI), Berlin, Germany. During 1997 to 1998, he was a Visiting Researcher at Stanford University, Stanford, CA, and served as a Consultant to 8x8, Inc., Santa Clara, CA. From 1993 to 1994, he was a Visiting Researcher at Kobe University, Kobe, Japan. In 1995, he was a Visiting Scholar at the University of California at Santa Barbara, where he began his 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 Video Coding Experts Group (ITU-T SG16 Q.6—VCEG)/ISO/IEC Moving Pictures Experts Group (ISO/IEC JTC1/SC29/WG11—MPEG)/Joint Video Team (JVT) standardization efforts and holds various international patents in this field. He has been appointed as the Associated Rapporteur of the ITU-T VCEG (October 2000), the Associated Rapporteur/Co-Chair of the JVT that has been created by ITU-T VCEG and ISO/IEC MPEG for finalization of the H.264/AVC video coding standard (December 2001), and the Editor of the H.264/AVC video coding standard (February 2002).

Wert
Bookbinding
Grantville, PA
JAN-JUNE 2004
"We're Quality Bound"

# APPENDIX WADA01

screenshot-catalog.libraries.psu.edu-2023.12.18-12_41_23
https://catalog.libraries.psu.edu/catalog/1181068
18.12.2023



# MARC View

| | | | | |
|---|---|---|---|---|
| LEADER 07877nas a22009611a 4500 | | | | |
| 001 1181068 | | | | |
| 003 SIRSI | | | | |
| 005 20151215050216.0 | | | | |
| 008 830531c19839999nyumr 0 0eng d | | | | |
| 010 | | | | a\| 83643067 z\| sn82005410 |
| 019 | | | | a\| MARS |
| 022 | 0 | | | a\| 0733-8716 |
| 035 | | | | a\| LIAS1351817 |
| 035 | | | | a\| (OCoLC)9450566 |
| 040 | | | | d\| UtOrBLW |
| 245 | 0 | 0 | | a\| IEEE journal on selected areas in communications : b\| a publication of the IEEE Communications Society. |
| 246 | 3 | 9 | | a\| IEEE journal on selected areas in communications |
| 246 | 3 | 0 | | a\| Selected areas in communications |
| 264 | | 1 | | a\| New York, N.Y. : b\| Institute of Electrical and Electronics Engineers, c\| [1983-] |
| 264 | | 4 | | c\| ©1983- |
| 300 | | | | a\| volumes : b\| illustrations ; c\| 28 cm |
| 310 | | | | a\| Monthly, b\| 2005- |
| 321 | | | | a\| Frequency varies, b\| 1983-2004 |
| 336 | | | | a\| text b\| txt 2\| rdacontent |
| 337 | | | | a\| unmediated b\| n 2\| rdamedia |
| 338 | | | | a\| volume b\| nc 2\| rdacarrier |
| 362 | 0 | | | a\| Vol. SAC-1, no. 1 (Jan. 1983)- v.SAC-5, no. 9 (1987) ; Vol. 6, no. 1 (Jan. 1988)- |
| 710 | 2 | | | a\| Institute of Electrical and Electronics Engineers. |
| 710 | 2 | | | a\| IEEE Communications Society. |
| 533 | | | | a\| Microform. b\| [New York, N.Y.? c\| IEEE], d\| 19__? e\| microfiches ; 11 x 15 cm. |
| 949 | | | | a\| Microfiche w\| ASIS c\| 1 i\| 1181068-1001 d\| 3/19/2001 l\| MICROS-HB m\| HARRISBURG r\| Y s\| Y t\| MICROFORM u\| 4/13/2001 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-1 1983 w\| LCPER c\| 1 i\| 000018892062 d\| 3/19/2001 e\| 3/19/2001 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 4/13/2001 z\| 862 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-3 no.4-6 1985 w\| LCPER c\| 1 i\| 000045496486 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 2/12/2002 z\| 862 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-2 1984 w\| LCPER c\| 1 i\| 000060038159 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 862 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-3 no.1-3 1985 w\| LCPER c\| 1 i\| 000060038166 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 862 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-4 no.1-5 1986 w\| LCPER c\| 1 i\| 000060038067 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-4 no.6-9 1986 w\| LCPER c\| 1 i\| 000060038074 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | | | | a\| TK5101.A1I16 v.SAC-5 no.1-4 1987 w\| LCPER c\| 1 i\| 000060037954 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |

| | |
|---|---|
| 949 | a\| TK5101.A1I16 v.SAC-5 no.5-9 1987 w\| LCPER c\| 1 i\| 000060037961 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | a\| TK5101.A1I16 v.SAC-6 no.1-5 1988 w\| LCPER c\| 1 i\| 000060037978 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | a\| TK5101.A1I16 v.6 no.6-9 1988 w\| LCPER c\| 1 i\| 000060037985 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | a\| TK5101.A1I16 v.7 no.1-5 1989 w\| LCPER c\| 1 i\| 000060037893 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | a\| TK5101.A1I16 v.7 no.6-9 1989 w\| LCPER c\| 1 i\| 000060037800 l\| ACAD-BLDG m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 10/3/2007 z\| 865 |
| 949 | a\| TK5101.A1I16 v.8 Jan-Apr 1990 w\| LCPER c\| 1 i\| 000018894073 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.8 May-Aug 1990 w\| LCPER c\| 1 i\| 000018894080 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.8 Sept-Dec 1990 w\| LCPER c\| 1 i\| 000018894097 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.9 Jan-Apr 1991 w\| LCPER c\| 1 i\| 000018894103 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.9 May-Aug 1991 w\| LCPER c\| 1 i\| 000018894110 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.9 Sept-Dec 1991 w\| LCPER c\| 1 i\| 000018894127 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.10 Jan-June 1992 w\| LCPER c\| 1 i\| 000018894134 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.10 Aug-Dec 1992 w\| LCPER c\| 1 i\| 000018894141 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.11 Jan-June 1993 w\| LCPER c\| 1 i\| 000018894158 d\| 10/11/2006 l\| CATO-2 m\| UP-ANNEX q\| 1 r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.11 Aug-Dec 1993 w\| LCPER c\| 1 i\| 000018894165 d\| 10/11/2006 l\| CATO-2 m\| UP-ANNEX q\| 1 r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.12 Jan-May 1994 w\| LCPER c\| 1 i\| 000018894172 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.12 June-Dec 1994 w\| LCPER c\| 1 i\| 000018894189 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.13 Jan-June 1995 w\| LCPER c\| 1 i\| 000018894196 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.13 Aug-Dec 1995 w\| LCPER c\| 1 i\| 000018894202 d\| 9/18/2008 l\| CATO-2 m\| UP-ANNEX q\| 1 r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.14 Jan-June 1996 w\| LCPER c\| 1 i\| 000018894219 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.14 July-Dec 1996 w\| LCPER c\| 1 i\| 000018894226 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.15 Jan-June 1997 w\| LCPER c\| 1 i\| 000018894233 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.15 July-Dec 1997 w\| LCPER c\| 1 i\| 000018894240 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.16 Feb-June 1998 w\| LCPER c\| 1 i\| 000018894257 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.16 Aug-Dec 1998 w\| LCPER c\| 1 i\| 000018894264 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.17 Jan-June 1999 w\| LCPER c\| 1 i\| 000018894271 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.17 July-Dec 1999 w\| LCPER c\| 1 i\| 000018894288 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |
| 949 | a\| TK5101.A1I16 v.18 Jan-June 2000 w\| LCPER c\| 1 i\| 000018894295 d\| 7/11/2003 l\| CATO-2 m\| UP-ANNEX q\| 1 r\| Y s\| Y t\| PERIODICAL u\| 3/5/2003 z\| 7769 |

| | | | |
|---|---|---|---|
| 949 | | | a\| TK5101.A1I16 v.18 July-Sept.2000 w\| LCPER c\| 1 i\| 000047609952 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 11/1/2001 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.18 Oct.-Dec.2000 w\| LCPER c\| 1 i\| 000047609808 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 11/1/2001 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.19 Jan-Mar 2001 w\| LCPER c\| 1 i\| 000048255431 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 11/28/2001 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.19 Apr-June 2001 w\| LCPER c\| 1 i\| 000048255424 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 11/28/2001 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.19 July-Sept 2001 w\| LCPER c\| 1 i\| 000048424660 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 11/4/2002 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.20 Jan-June 2002 w\| LCPER c\| 1 i\| 000051870089 d\| 7/7/2004 l\| CATO-2 m\| UP-ANNEX q\| 1 r\| Y s\| Y t\| PERIODICAL u\| 5/2/2003 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.20 Aug-Dec 2002 w\| LCPER c\| 1 i\| 000051870157 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 5/2/2003 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.21 Jan-June 2003 w\| LCPER c\| 1 i\| 000052812422 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/17/2004 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.21 Aug-Dec 2003 w\| LCPER c\| 1 i\| 000052812415 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 3/17/2004 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.22 Jan-June 2004 w\| LCPER c\| 1 i\| 000054847187 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 4/8/2005 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.22 Aug-Dec 2004 w\| LCPER c\| 1 i\| 000054847194 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 4/7/2005 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.23 Jan-Apr 2005 w\| LCPER c\| 1 i\| 000054976375 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 4/18/2006 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.23 May-Aug 2005 w\| LCPER c\| 1 i\| 000054976368 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 4/18/2006 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.23 Sept-Dec 2005 w\| LCPER c\| 1 i\| 000054976351 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 4/18/2006 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.24 Jan-Apr 2006 w\| LCPER c\| 1 i\| 000055217415 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 6/12/2007 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.24 May-Aug 2006 w\| LCPER c\| 1 i\| 000055217293 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 6/12/2007 z\| 7769 |
| 949 | | | a\| TK5101.A1I16 v.24 Sept-Dec 2006 w\| LCPER c\| 1 i\| 000055217422 l\| CATO-2 m\| UP-ANNEX r\| Y s\| Y t\| PERIODICAL u\| 6/12/2007 z\| 7769 |
| 852 | 0 | 1 | a\| PSt b\| UP-ANNEX c\| ANNEX h\| TK5101.A1 i\| I16 |
| 866 | | 0 | 8\| 0 a\| v.SAC-1(1983)-v.SAC-6:no.5(1988), v.6:no.6(1988)-v.15(1997), v.16:Feb.(1998)-v.24(2006) |
| 852 | 8 | | a\| PSt b\| HARRISBURG c\| MICROS-HB h\| Microfiche |
| 866 | | 0 | 8\| 0 a\| v.1 1983-v.17 1999 |

PennState

Copyright @2019 The Pennsylvania State University. All rights reserved. Except where otherwise noted, this work is subject to a Creative Commons Attribution 4.0 license. Details and exceptions.

Legal Statements | PSU Hotlines

PENN STATE UNIVERSITY LIBRARIES

Libraries Home (814) 865-6368
Libraries Intranet (Staff Only)
Accessibility Help
Website Feedback
Policies and Guidelines
Staff Directory

screenshot-catalog.libraries.psu.edu-2023.12.18-12_41_23
https://catalog.libraries.psu.edu/catalog/1181068
18.12.2023



# IEEE journal on selected areas in communications : a publication of the IEEE Communications Society

| | |
|---|---|
| Additional Titles: | IEEE journal on selected areas in communications and Selected areas in communications |
| Published: | New York, N.Y. : Institute of Electrical and Electronics Engineers, [1983-] |
| Copyright Date: | ©1983- |
| Physical Description: | volumes : illustrations ; 28 cm |
| Additional Creators: | Institute of Electrical and Electronics Engineers and IEEE Communications Society |

## Availability

Browse Nearby on Shelf I Want It

### Penn State Harrisburg (1 item)

| Call number | Material | Location |
|---|---|---|
| Harrisburg - Microforms - 2nd Floor - In-Library Use: Holdings Summary<br>v.1 1983-v.17 1999 | | |
| Microfiche | Microfilm, Microfiche, etc. | Harrisburg - Microforms - 2nd Floor - In-Library Use |

### Library Storage (52 items)

| Call number | Material | Location |
|---|---|---|
| Annexed Material: Holdings Summary<br>v.SAC-1(1983)-v.SAC-6:no.5(1988), v.6:no.6(1988)-v.15(1997), v.16:Feb.(1998)-v.24(2006) | | |
| TK5101.A1I16 v.6 no.6-9 1988 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.7 no.1-5 1989 | Bound Journal | Checked out |
| TK5101.A1I16 v.7 no.6-9 1989 | Bound Journal | Checked out |
| TK5101.A1I16 v.8 Jan-Apr 1990 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.8 May-Aug 1990 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.8 Sept-Dec 1990 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.9 Jan-Apr 1991 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.9 May-Aug 1991 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.9 Sept-Dec 1991 | Bound Journal | Annexed Material |

| | | |
|---|---|---|
| TK5101.A1I16 v.10 Aug-Dec 1992 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.10 Jan-June 1992 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.11 Aug-Dec 1993 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.11 Jan-June 1993 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.12 Jan-May 1994 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.12 June-Dec 1994 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.13 Aug-Dec 1995 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.13 Jan-June 1995 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.14 Jan-June 1996 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.14 July-Dec 1996 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.15 Jan-June 1997 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.15 July-Dec 1997 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.16 Aug-Dec 1998 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.16 Feb-June 1998 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.17 Jan-June 1999 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.17 July-Dec 1999 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.18 Jan-June 2000 | Bound Journal | Being transferred between libraries |
| TK5101.A1I16 v.18 July-Sept.2000 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.18 Oct.-Dec.2000 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.19 Apr-June 2001 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.19 Jan-Mar 2001 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.19 July-Sept 2001 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.20 Aug-Dec 2002 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.20 Jan-June 2002 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.21 Aug-Dec 2003 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.21 Jan-June 2003 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.22 Aug-Dec 2004 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.22 Jan-June 2004 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.23 Jan-Apr 2005 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.23 May-Aug 2005 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.23 Sept-Dec 2005 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.24 Jan-Apr 2006 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.24 May-Aug 2006 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.24 Sept-Dec 2006 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-1 1983 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-2 1984 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-3 no.1-3 1985 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-3 no.4-6 1985 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-4 no.1-5 1986 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-4 no.6-9 1986 | Bound Journal | Annexed Material |

| | | |
|---|---|---|
| TK5101.A1I16 v.SAC-5 no.1-4 1987 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-5 no.5-9 1987 | Bound Journal | Annexed Material |
| TK5101.A1I16 v.SAC-6 no.1-5 1988 | Bound Journal | Annexed Material |

**Dates of Publication and/or Sequential Designation:**
Vol. SAC-1, no. 1 (Jan. 1983)- v.SAC-5, no. 9 (1987) ; Vol. 6, no. 1 (Jan. 1988)-

**ISSN:**
0733-8716

**Publication Frequency:**
Monthly, 2005-
Frequency varies, 1983-2004

**Reproduction Note:**
Microform. [New York, N.Y.? IEEE], 19__? microfiches ; 11 x 15 cm.

View MARC record | catkey: 1181068

PennState

Copyright @2019 The Pennsylvania State University. All rights reserved. Except where otherwise noted, this work is subject to a Creative Commons Attribution 4.0 license. Details and exceptions.

Legal Statements | PSU Hotlines

PENN STATE UNIVERSITY LIBRARIES

Libraries Home (814) 865-6368
Libraries Intranet (Staff Only)
Accessibility Help
Website Feedback
Policies and Guidelines
Staff Directory

# APPENDIX WADA02

