In Proc. of Applications and Services in Wireless Networks (AWSN), Bern, Switzerland, July 2003

Semantic-aware Link Layer Scheduling of MPEG-4 Video Streams in Wireless Systems

Jirka Klaue, James Gross, Holger Karl, Adam Wolisz

Telecommunication Networks Group
Technical University of Berlin
Einsteinufer 25, 10587 Berlin, Germany

{jklaue|gross|karl|wolisz}@ee.tu-berlin.de

Abstract

Delivering video streams to terminals via a wireless last hop is a challenging task due to the varying nature of the wireless link. While a common approach suggests to exploit the variations of the wireless channel, an alternative is to exploit characteristics of the video stream to improve the transmission. In this paper we show how semantic stream variations of MPEG-4 coded video can be used to increase the number of terminals which can be supported in a wireless cell at a given minimum video quality. As example system we consider the performance of an OFDM system with different video sources. Simulations show that the number of supportable terminals can be increased on average by fifty percent. [1]

1 Introduction

For modern mobile communication systems, the promise to support multi-media applications exceeding the well-known telephony service has been made. Video streams shall be available to mobile users over a wireless transmission medium in a both cost-efficient and high-quality way. But the higher average bit rate and higher variability of video as compared to voice make this a challenging problem.

Trying to solve this problem at the wireless link layer, e.g., by improving efficiency or reducing the wireless channel's variability, can result in some improvements, but such link layer improvements are oblivious to the nature of the transported data. Also, adapting the video codec to the wireless transmission case is conceivable but is cumbersome and burdened with long delays: a possibly far away video source cannot react quickly enough to the variable conditions of a wireless link; it could perhaps adapt the video's average rate to a low expected value of wireless capacity, but this is unlikely to be optimal.

A different approach would be to exploit some information about the nature of video data closer to the wireless channel. Consider a wireless cell with several terminals, each receiving a video stream from a remote source via an access point. When one terminal experiences a bad channel, video frames for this terminal will start to queue up in the access point. A video-agnostic link layer could attempt to compensate for this bad channel by assigning more radio resources to this terminal, probably resulting in a degraded service level for the other terminals. But such a brute-force solution does not do full justice to the nature of the video stream: instead of forcing through all packets, a link layer aware of the video packet semantics could concentrate on the important ones, neglecting less important video packets in a bad channel state.

This is the idea of semantic-aware scheduling for video streams that we propose in this paper: use the number and kind of video frames queued up at an access point to decide which packets to hand down to the wireless link layer and which ones to delay or drop. The advantage of this scheme is its simplicity — MPEG-4 frame semantics

[1] This work has been supported by the German research funding agency 'Deutsche Forschungsgemeinschaft' under the program "Adaptability in Heterogeneous Communication Networks with Wireless Access" (AKOM)

are easy to detect — and its deployment right at the source of the problem, at the wireless link. We will investigate and compare two scheduling schemes: agnostic of queue content versus a semantic aware queue management, treating packets of a queue according to their application layer significance.

As a figure of merit for these schemes, two main options exist: either look at the quality improvements for a given set of terminals/video streams, or use a minimum target quality for a video transmission and try to maximize the number of terminals/ video streams that can be supported at such a minimum quality. In this paper, we opt for the second performance metric. We will show by simulations that semantic-aware link layer scheduling can considerably increase the number of supportable terminals.

In the following Section 3, the system model is described in more detail. Section 4 then introduces the semantic-aware link layer scheduling algorithms. Section 5 shows the scenarios and performance metric used for evaluation, and after presenting the results, we conclude the paper in Section 6.

2 Related Work

A lot of research regarding real-time video quality improvement has been done in the last decade. A challenging problem with streaming video is its high demand on the QoS features of the transport medium. But bandwidth, delay, and loss rate vary over time and the current Internet lacks QoS support. To overcome these problems, various mechanisms have been developed. They can roughly be divided in feedback control, source-rate adaptation, packetization, and error control [21]. Feedback control is done by estimating available network bandwidth based on packet loss information at the receiver. Source-rate adaptation is achieved with, e.g., frame skipping or the more sophisticated multi-layer coding aka fine-grained scalability (FGS) or rate-distortion theory based approaches. [8, 10, 21]. Packetization schemes try to minimize overhead while maximizing robustness against losses. Effectively inter-packet dependencies are minimized [21]. Error control approaches include forward error correction (FEC), retransmission schemes and error-resilient encoding [18, 19]. All these mechanisms address the problem of loss and delay during video transmissions and are, therefore, in principle applicable to wired and wireless connections.

Many of these approaches require end systems that are aware of the actual link or end-to-end feedback or both. Unlike these approaches, a base station connecting the internet to a wireless environment can use its knowledge about the wireless channel and the application requirements (packet importance). The classification of packets can be done with the schemes used in the differentiated services [16] approaches. The main advantage of prioritized transmission and priority drop [5] performed directly in the link layer of the base station is the independence of end-to-end feedback and the actual end systems.

3 System Model

We consider the downlink of a single cell. Data transmissions within this cell are managed by an access point, all J wireless terminals within this cell are associated to this access point. The access point is connected to a backbone, where data leaving or entering the cell is passed for and back. For data transmission a radio frequency band of bandwidth B is allocated. The transmission scheme used is OFDM, thus the bandwidth is divided into S subcarriers. We consider an OFDM-FDM system in the downlink, different subcarriers can be used for transmitting data to different terminals.

The terminals move constantly within the cell with a certain maximum speed $v_{\max}$. Therefore their channel gains vary constantly due to path loss, shadowing and fading. For each time instant do the gains differ for different subcarriers of the same terminal. Also the gains of the same subcarrier differ for different terminals, since they are not assumed to be at the same location of the cell.

Transmission of data within the cell is based on a timing structure provided. The basic element of this structure is one frame, which has the length of $2\ \mathrm{ms}$. Half of a frame is reserved for uplink transmissions (not considered further), the other half is reserved for downlink transmissions. Prior to a downlink transmission the access point assigns sets of subcarriers to wireless terminals. It may base its assignments on provided channel knowledge of each subcarrier regarding each terminal in the cell. If this is the case, then a dynamic assignment scheme is used. Alternatively, fixed sets of subcarriers are assigned to the terminals, which do not change from frame to frame. In the dynamic case, we assume an errorfree signaling system causing no additional cost in terms of system throughput to inform the terminals of their assignments before the data transmission starts. As dynamic algorithm we choose a heuristic from [3]. In order to transmit data on each subcarrier a set of modulation types (BPSK, QPSK, 16-QAM, 64-QAM and 256-QAM) is used adaptively, however during one frame length the assignments as well as the modulation types are fixed. Out of the available modulation types the system always chooses the one with the highest throughput while it still has a symbol error rate lower than P_s. Transmission power is equally distributed over all subcarriers, we do not assume a power control system to vary this. For forward error control we employ block codes, due to their easily handling in simulations.

Each terminal receives a data stream from a source outside of the cell. Streams consist of packets with a certain size in bits. These packets arrive constantly at the acccess point and are queued seperately for each terminal until they are transmitted to the respective terminal. The streams vary in the amount of bits which the access point receives for a given amount of time, their mean bit rate is denoted by r_{av}. Each transmitted packet has to be received by a terminal within a certain overall time span (for example $400\ \mathrm{ms}$). Since transmission of data within the backbone does cost some time, a certain time fraction of this span is alreday consumed, when the packet arrives at the access point. We assume that the remaining time span is known to the access point. Also the packets arriving do have semantical differences, as they belong to video streams encoded following MPEG-4 standard [8]. We assume that the access point also has knowledge of the content of each packet (i.e. if it belongs to an I-, B- or P-frame). This information is gathered by payload analyzing or, if encryption is applied, by packet marking in the UDP/RTP layer. Some streaming applications, like Apple's Darwin Streaming Server [2], provide these informations in the RTP-headers. Each terminal in the cell receives only such a stream, no heterogenous stream constellations are studied. Also the single streams are quite equal in terms of their average bit rate and their remaining transmit time. However, they might be time shifted to each other.

If packets are transmitted completely and after decoding from the applied block codes still bit errors remain within this packet, the receiver indicates this during the (not considered) uplink phase to the access point. Then, the packet might be retransmitted.

System performance is measured in terms of the number of terminals that can be served with video streams, whose quality still satisfies the user. This metric measuring the user satisfaction is explained in section 5.3.

4 Semantic Aware Scheduling

4.1 Video coding

Modern video coding methods exploit both the spatial and the temporal redundancy in the source data. Spatial redundancy is reduced using block-wise discrete cosine transformation (DCT) and quantization followed by entropy coding of the remaining coefficients. This is known as intraframe coding. Temporal redundancy is lowered by coding only the differences between any two successive images. Several methods exist to perform temporal coding, such as, frame differencing, motion estimation and motion-compensated prediction. These methods are known as interframe coding [12].

MPEG-4, as a popular example of state-of-the-art video coding techniques, takes advantage of intra- and interframe coding methods. It distinguishes between three frame types, namely I, P, and B-frames. I-frames are solely intra-coded frames, P-frames are predicted frames depending on the previous I-frame, and B-frames are bidirectional "predicted" frames (depending on the previous and following I-, or P-frame). Frames are arranged in so-called "groups of pictures" (GOP). Such a GOP consists of exactly one I-frame and some related P-frames and optionally some B-frames between these I- and P-frames [9]. In MPEG-4, I-frames contain by far the most information. Furthermore, loosing an I-frame would cause distortion of all following frames in a GOP. A P-frame loss would only influence the flanking B-frames and the loss of a B-frame would not influence any other frame.

4.2 Frame-type based scheduling

The basic idea of the scheduling algorithm is to exploit the significant differences between the MPEG-4 frame types regarding their information content and their influence on the error propagation. The hypothesis is that prioritized treatment of semantically more important frames results in a much better video quality. The scheduling algorithm manages:

- the order of transmission depending on the type of the frame
- the time at which a packet will be dropped from the queue.

If there are any packets in the queue containing parts of an MPEG-4 I-frame they are transmitted first, followed by packets related to P-frames and B-frames. If there are no video frames in the queue or there is still bandwidth left within the current downlink phase, other data possibly in the queue is transmitted.

The other scheduling parameter is the drop time. We define an overall maximum acceptable delay: Video packets which cannot be transmitted within the time limited by the maximum delay are dropped. A sensible approach is to derive this maximum delay from the play-out buffer size at the terminal and to drop a packet once the allowable delay is exceeded. While this would be straightforward for constant bit rate traffic, the unpredictability of the amount of data per time in a variable bit rate scheme requires an additional consideration: it is reasonable not to use the maximum allowable delay for all packets, since this could lead to unwanted loss of important data in the future. Therefore, we drop semantically less important packets earlier. Table 1 shows the exact values of the dropping parameter of the scheduler.

Frame type	Drop time
I	full deadline
P	3/4 of deadline
B	1/2 of deadline

Table 1: Drop times depending on the MPEG-4 frame type

These parameters were not optimized regarding their influence on the resulting video quality; this is an issue of further studies. But even these heuristically chosen values lead to a great improvement compared to semantically "blind" queue management.

5 Performance Analysis

5.1 Scenario

As transmission scenario we chose a system quite similar to IEEE 802.11a [6]. $S = 48$ subcarriers can be used for data transmission, every $4\ \mu\mathrm{s}$ a symbol is transmitted. For the transmission of data a maximum power of $0\ \mathrm{dBm}$ is allowed in the frequency band around $5.2\ \mathrm{GHz}$, each subcarrier is assigned a power of $-7\ \mathrm{dBm}$. The cell size is chosen to be $100\ \mathrm{m}$ in radius. Terminals moved with a maximum speed of $v_{\max} = 1\ \mathrm{m/s}$. Due to this movement, channel gains varied constantly as the impact of path loss, shadowing and fading changes from position to position of the terminal. For the path loss we assumed an exponent of $\alpha = 2.4$ and a path loss reference loss of $10log(K) = 46.7dB$. Following the ETSI C model for large open space environments

[11], the delay spread was set to equal $\Delta\sigma = 0.15\ \mu\mathrm{s}$ with an expontial power delay profile. For the Doppler power spectrum we assume a Jakes-like shape. As a result, channel gain variations were correlated in time and frequency for each terminal. An example environment for such a setting would be a large airport or exposition hall, with people moving at pedestrian speed within this area.

We used two different video sources in our simulations. One is a source with low motion video and the other a sequence of an movie with very high motion. These video sources can be coded by MPEG-4 with different efficiency and they behave differently in case of packet losses (see section 4.1).

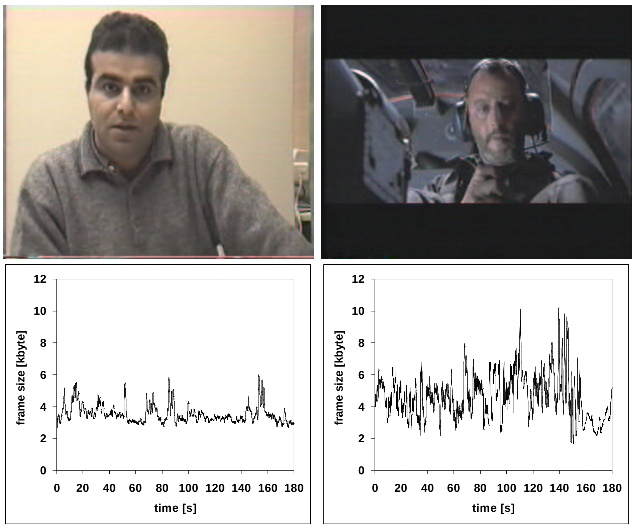


Figure 1: Sample image and bit-rate of low-motion (left) and high-motion (right) video source used for the simulations

Both videos are 25 frames per second, CIF (352x288 pixel) and 4500 frames (3 min) long. They are encoded according to MPEG-4 with variable bit-rate and a fixed 12-GOP with two consecutive B-frames (IBBPBBPBBPBB). MPEG-4, GOP and frame types are explained in section 4.1. Figure 1 shows exemplarily a sample picture from each video source and the resulting bit-rate of the MPEG-4 coded videos.

The weights and deadlines used by the semantic scheduling were chosen based on experience. We do not claim that they are optimal, especially the base deadline can only be "guessed". If the source of the video streams is located somewhere in the internet, it is difficult to get out the packet travelling time. Nevertheless, since acceptable end-to-end delays for video transmission systems are up to 400 ms, we assumed remaining packet deadlines of 100 to 250 ms in our simulations.

5.2 Frame-by-frame Video Quality Metric

Digital video quality measurements must be based on the perceived quality of the actual video being received by the users of the digital video system. Such a perception-based evaluation is appropriate as the subjective impression of the user is what only counts in the end. This intuitive impression of a human user watching a video is grasped by subjective quality metrics. These subjective metrics provide most information, but their determination is extremely costly: highly time consuming, high manpower requirements, and special equipment needed. Such subjective methods are described in detail by ITU [1, 14], ANSI [17] and MPEG [7] . Describing the human quality impression by a subjective quality metric is usually done with a "mean opinion scale" (MOS), on a scale from 5 (best) to 1 (worst) as in table 2.

The expensive and complex subjective tests are often not affordable. Instead, many tasks in industry and research require automated methods to evaluate video quality. Therefore, objective metrics have been developed to emulate the quality impression

Scale	Quality	Impairment
5	Excellent	Imperceptible
4	Good	Perceptible, but not annoying
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying

Table 2: ITU-R quality and impairment scale

of the human visual system (HVS). In [20], there is an exhaustive discussion of various objective metrics and their performance compared to subjective tests. The most widespread method is the calculation of peak signal to noise ratio (PSNR) image by image. It is a derivative of the well-known signal to noise ratio (SNR). The PSNR compares the maximum possible signal energy to the noise energy, which results in a higher correlation with the subjective quality perception than the conventional SNR [4]. Equation 1 gives the definition of the PSNR of source image s and destination image d [15].

$$\begin{aligned} PSNR(S,D) &= 20\log\frac{V_{peak}}{MSE(s,d)} \quad [dB] \qquad (1)\\ \text{where} & \\ V_{peak} &= 2^k - 1, k \text{ bit color depth} \\ MSE(s,d) &= \text{mean square error of } s \text{ and } d. \end{aligned}$$

While the PSNR does not directly correspond to the MOS, their exist heuristic mappings of PSNR to MOS (subjective quality) as shown in table 3. To evaluate the impact of the network (delay, loss) on the video quality, we need to compare the received (possibly distorted) video with the actually sent video. In fact, even the sent video is already distorted by the encoding process, e.g., MPEG-4, but this distortion cannot be avoided when striving for acceptable bit rates in video streams.

PSNR [dB]	MOS
> 37	5 (Excellent)
31 - 37	4 (Good)
25 - 31	3 (Fair)
20 - 25	2 (Poor)
< 20	1 (Bad)

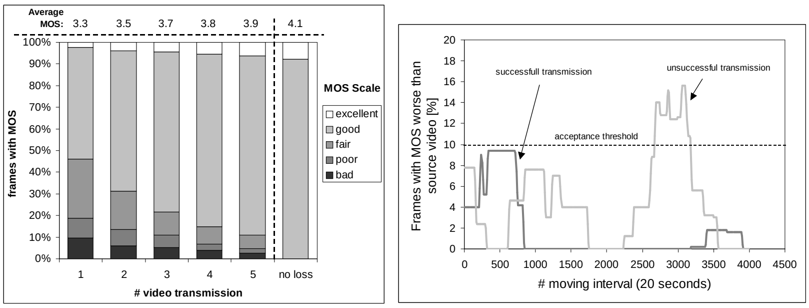
Table 3: Possible PSNR to MOS conversion, [13]

Hence, we have a computational approximation of the subjective human impression of every single frame at our disposal. Based on this frame by frame MOS calculation, we define a metric which reflects the user impression of the entire received video.

5.3 Video Quality Metric

Previously we described how the MOS is calculated frame by frame for both the original input video and for the received video. Extending such a quality metric to an entire video is difficult. The calculation of average PSNR (or MOS) values for the entire video does not map very well to the subjective impression, especially in the case of longer video clips. If, for instance, the first 10 seconds of the video stream are highly distorted the user is not satisfied with the video quality while the average MOS would not reflect this. Figure 2(a) shows the quality of five different video transmissions and the reference video quality. The average MOS is printed on top of each bar. Though this average MOS seems acceptable in all cases, it is likely that the frames with bad

MOS grades happen to appear continuously. In this case the subjective video quality will not match the average MOS.



(a) MOS

(b) MIV (**M**OS **I**nter**V**al)

Figure 2: Exemplification of average MOS (a) and video quality metric MIV (b); successful transmission means the percentage of frames worse than the original must be less than 10% within any interval.

To resolve this potential problem we follow a different approach here: We assume that the input video's quality is always acceptable for a human watching this anyway, i.e., that the coding distortions are acceptable. When is the received video also acceptable, i.e., when are the transmission-related distortions compared to the sent video still acceptable?

Intuitively, some small amount of distorted frames is likely to be acceptable, as long as this number does not become too big. This "too big" is formalized by requiring that, for any arbitrary part of the video, the number of received frames with a MOS smaller than that of the sent frame must not exceed 10% of the frames contained in that period — such a video transmission is called "successful". Applying this condition to any arbitrary period is necessary as videos which, e.g., completely loose their first 10% of all frames but perfectly transmit the others would still not be acceptable.

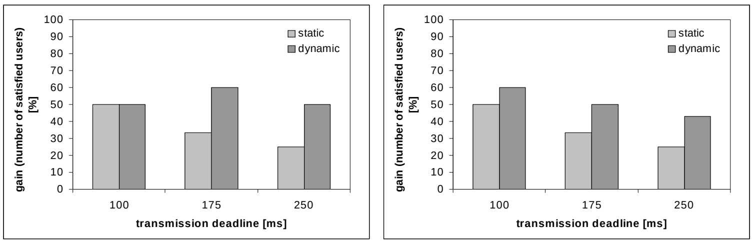
Figure 2(b) illustrates this metric, it shows if a video transmission was successful or not according to the metric defined above.

5.4 Results

In our simulations we varied the transmission deadline (100, 175, 250 ms), the subcarrier assignment algorithm (static, dynamic) and turned on and off the semantic scheduling mechanism explained in Section 4. All simulations were performed with both video sources (high-motion and low-motion). In all simulations the semantic scheduling outperformes the semantically "blind" scheduling. Up to 60% more terminals could be served with a qualitative acceptable video stream. In Figure 3 the gain is quantified for all simulations and Figure 4 shows the absolut numbers of successfully served terminals.

In all cases the improvement is achieved only by exploiting the semantical differences of the MPEG-4 frames. In fact the semantic scheduling policy results in a protection of I- and P-frames at the price of dropping B-frames more often. But since these B-frames have the least influence on the distortion of the resulting video, this strategy pays off well regarding the number of terminals served with good video quality.

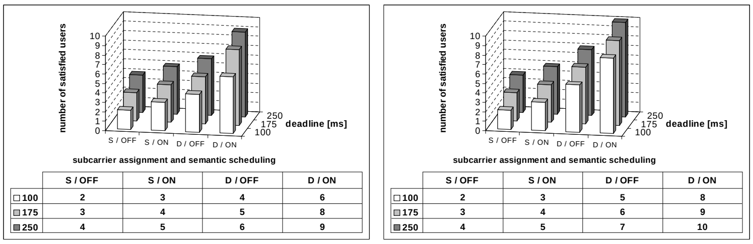
The potential of semantic scheduling has been shown for different video sources and different transmission deadlines. Especially under difficult conditions, e.g, very short deadlines, the gain of this method is very high.



(a) High motion video

(b) Low motion video

Figure 3: Gain of the semantic scheduling mechanism for different deadlines and subcarrier assignment algorithms



	S / OFF	S / ON	D / OFF	D / ON
□ 100	2	3	4	6
▨ 175	3	4	5	8
■ 250	4	5	6	9

	S / OFF	S / ON	D / OFF	D / ON
□ 100	2	3	5	8
▨ 175	3	4	6	9
■ 250	4	5	7	10

(a) High motion video

(b) Low motion video

Figure 4: Number of satisfied users with semantic scheduling turned on/off (each for different deadlines and static/dynamic subcarrier assignment)

6 Conclusions and Future Work

We have shown that applying some simple form of semantic-aware scheduling can significantly improve the characteristics of video transmissions over wireless channels. Specifically, the number of terminals supported in a cell can be improved, at a maximum of 60%.

The positive effect of the semantic scheduling arises in all investigated scenarios (different physical layers, different video sources, different transmission deadlines), showing that the method is quite stable. Nevertheless this assertion is subject to further investigation. Especially the stability of the approach when transmitting different video streams to each terminal and using other video coding strategies (e.g., varying the number of B-frames in the GOP structure) must be investigated. Furthermore, the choice of the scheduling parameters, like dropping deadlines, can surely be optimized. As a third open issue this approach might be connected to the common idea of adapting to the wireless link. In the case of an OFDM transmission system one might think of a semantically influenced subcarrier allocation strategy, which combines the adaption to the video stream as well as the adaption to the wireless link.

References

[1] ITU-R Recommendation BT.500-10. Methodology for the subjective assessment of the quality of television pictures, March 2000.

[2] Apple Computer. *Darwin Streaming Server*
. http://developer.apple.com/darwin/projects/streaming/.

[3] J. Gross and F. Fitzek. Channel state dependent scheduling policies for an ofdm physi-

cal layer using a m-ary state model. Technical Report TKN-01-010, Telecommunication Networks Group, Technische Universität Berlin, June 2001.

[4] Lajos Hanzo, Peter J. Cherriman, and Juergen Streit. *Wireless Video Communications*. Digital & Mobile Communications. IEEE Press, 445 Hoes Lane, Piscataway, 2001.

[5] Jie Huang, Charles Krasic, and Jonathan Walpole. Adaptive live video streaming by priority drop. *to appear in Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance*, July 2003.

[6] IEEE. *Supplement to Standard for Telecommunications and Information Exchange Between Systems — LAN/MAN Specific Requirements — Part 11: Wireless MAC and PHY Specifications: High Speed Physical Layer in the 5-GHz Band*, p802.11a/d7.0 edition, July 1999.

[7] ISO-IEC/JTC1/SC29/WG11. Evaluation methods and procedures for july mpeg-4 tests, 1996.

[8] ISO-IEC/JTC1/SC29/WG11. *ISO/IEC 14496: Information technology — Coding of audio-visual objects*, 2001.

[9] ISO/IEC JTC1/SC29/WG11. Overview of the mpeg-4 standard, July 2000.

[10] Weiping Li. Overview of fine granularity scalability in mpeg-4 video standard. *IEEE transaction on circuits and systems for video technology*, March 2001.

[11] J. Medbo and P. Schramm. *Channel Models for HIPERLAN/2*. ETSI EP BRAN document 3ERI085B, March 1998.

[12] King N. Ngan, Chi W. Yap, and Keng T. Tan. *Video Coding for Wireless Communication Systems*. Signal Processing and Communications. Marcel Dekker, Inc., 270 Madison Avenue, NY, 2001.

[13] Jens-Rainer Ohm. Bildsignalverarbeitung fuer multimedia-systeme. Skript, 1999.

[14] ITU-T Recommendation P.910. Subjective video quality assessment methods for multimedia applications, August 1996.

[15] Martyn J. Riley and Iain E. G. Richardson. *Digital Video Communications*. Artech House, 685 Canton Street, Norwood, 1997.

[16] Jitae Shin and JongWon Kim. Performance evaluation of differentiated services to mpeg-4 fgs video streaming. *J. of the Korean Insti. of Commun. Sciences*, 27:711–723, 2002.

[17] ANSI T1.801.01-1996. Digital transport of video teleconferencing / video telephony signals — video test scenes for subjective and objective performance assessment. ANSI, 1996.

[18] S. Wenger, G. Knorr, J. Ott, and F. Kossentini. Error resilience support in h.263+. *IEEE Transactions on Circuits and Systems for Video Technology*, 8:867–877, November 1998.

[19] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra. Rate-distortion optimized mode selection for very low bit-rate video coding and the emerging h.263 standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 6:182–190, April 1996.

[20] Stephen Wolf and Margaret Pinson. Video quality measurement techniques. Technical Report 02-392, U.S. Department of Commerce, NTIA, June 2002.

[21] D. Wu, Y. T. Hou, W. Zhu, H.-J. Lee, T. Chiang, Y.-Q. Zhang, and H. J. Chao. On end-to-end architecture for transporting mpeg-4 video over the internet. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(6):923–941, September 2000.