Rate-Constrained Coder Control and Comparison of Video Coding Standards

Thomas Wiegand, Heiko Schwarz, Anthony Joch, Faouzi Kossentini, and Gary J. Sullivan

***Abstract*—A unified approach to the coder control of video coding standards such as MPEG-2, H.263, MPEG-4, and the draft video coding standard H.264/AVC is presented. The performance of the various standards is compared by means of PSNR and subjective testing results. The results indicate that H.264/AVC compliant encoders typically achieve essentially the same reproduction quality as encoders that are compliant with the previous standards while typically requiring 60% or less of the bit-rate.**

***Index Terms*—Video, Standards, Coder Control, Rate-Constrained, Lagrangian, MPEG-2, H.263, MPEG-4, H.264/AVC.**

I. Introduction

THE specifications of most video coding standards including MPEG-2 Visual [1], H.263 [2], MPEG-4 Visual [3] and H.264/AVC [4] provide only the bit-stream syntax and the decoding process in order to enable interoperability. The encoding process is left out of the scope to permit flexible implementations. However, the operational control of the source encoder is a key problem in video compression. For the encoding of a video source, many coding parameters such as macroblock modes, motion vectors, and transform coefficient levels have to be determined. The chosen values determine the rate-distortion efficiency of the produced bit-stream of a given encoder.

In this paper, the operational control of MPEG-2, H.263, MPEG-4, and H.264/AVC encoders is optimized with respect to their rate-distortion efficiency using Lagrangian optimization techniques. The optimization is based on [5] and [6], where the encoder control for the ITU-T Recommendation H.263 [2] is addressed. The Lagrangian coder control as described in this paper was also integrated into the test models TMN-10 [7] and JM-2 [8] for the ITU-T Recommendation H.263 and H.264/AVC, respectively. The same Lagrangian coder control method was also applied to the MPEG-4 verification model VM-18 [9] and the MPEG-2 test model TM-5 [10]. In addition to achieving performance gains, the use of similar rate-distortion optimization methods in all encoders allows a useful comparison between the encoders in terms of coding efficiency.

This paper is organized as follows. Section II gives an overview of the syntax features of MPEG-2 Video, H.263, MPEG-4 Visual, and H.264/AVC. The rate-distortion-optimized coder control is described in Section III, and experimental results are presented in Section IV.

II. Standard Syntax and Decoders

All ITU-T and ISO/IEC JTC1 standards since H.261 [11] have in common that they are based on the so-called block-based hybrid video coding approach. The basic source coding algorithm is a hybrid of inter-picture prediction to utilize temporal redundancy and transform coding of the prediction error signal to reduce spatial redundancy. Each picture of a video signal is partitioned into fixed-size macroblocks of 16×16 samples, which can be transmitted in one of several coding modes depending on the picture or slice coding type. Common to all standards is the definition of INTRA coded pictures or I-pictures. In I-pictures, all macroblocks are coded without referring to other pictures in the video sequence. Also common is the definition of predictive-coded pictures, so-called P-pictures and B-pictures, with the latter being extended conceptually in H.264/AVC coding. In predictive-coded pictures, typically one of a variety of INTER coding modes can be chosen to encode each macroblock.

In order to manage the large number of coding tools included in standards and the broad range of formats and bit-rates supported, the concept of *profiles* and *levels* is typically employed to define a set of conformance points, each targeting a specific class of applications. These conformance points are designed to facilitate interoperability between various applications of the standard that have similar functional requirements. A profile defines a set of coding tools or algorithms that can be used in generating a compliant bit-stream, whereas a level places constraints on certain key parameters of the bit-stream, such as the picture resolution and bit-rate.

Although MPEG-2, H.263, MPEG-4, and H.264/AVC define similar coding algorithms, they contain features and enhancements that make them differ. These differences involve mainly the formation of the prediction signal, the block sizes used for transform coding, and the entropy coding methods. In the following, the description of the various standards is limited to those features relevant to the comparisons described in this paper.

A. ISO/IEC Standard 13818-2 / ITU-T Recommendation H.262: MPEG-2

MPEG-2 forms the heart of broadcast-quality digital television for both standard-definition and high-definition television (SDTV and HDTV) [1][12][13]. MPEG-2 video (IS 13818-2 / ITU-T Recommendation H.262) was designed to encompass MPEG-1 [14] and to also provide high quality with interlaced video sources at bit-rates in the range of 4-30 Mbit/s. Although usually thought of as an ISO standard, MPEG-2 video was developed as an official joint project of both the ISO/IEC JTC1 and ITU-T organizations, and was completed in late 1994.

MPEG-2 incorporates various features from H.261 and MPEG-1. It uses the basic coding structure that is still predominant today. For each macroblock, which consists of one 16×16 luminance block and two 8×8 chrominance blocks for 4:2:0 formatted video sequences, a syntax element indicating the macroblock coding mode (and signalling a quantizer change) is transmitted. While all macroblocks of I-pictures are coded in INTRA mode, macroblocks of P-pictures can be coded in INTRA, INTER-16×16, or SKIP mode. For the SKIP mode, runs of consecutive skipped macroblocks are transmitted and the representation of the picture in the skipped region is represented using INTER prediction without adding any residual difference representation. In B-pictures, the prediction signal for the motion-compensated INTER-16×16 mode can be formed by forward, backward, or bi-directionally interpolated prediction. The motion compensation is generally based on 16×16 blocks and utilizes half-pixel accurate motion vectors, with bilinear interpolation of half-pixel positions. The motion vectors are predicted from a single previously encoded motion vector in the same slice.

Texture coding is conducted using a DCT on blocks of 8×8 samples, and uniform scalar quantization (with the exception of the central dead-zone) is applied that can be adjusted using quantization values from 2 to 62. Additionally, a perceptually weighted matrix based on the frequency of each transform coefficient (except the Intra DC coefficient) can be used. The entropy coding is performed using zigzag scanning and two-dimensional run-level variable length coding (VLC). There are two available VLC tables for transmitting the transform coefficient levels, of which one must be used for predictive-coded macroblocks and either can be used for INTRA macroblocks, as selected by the encoder on the picture level.

For the coding of interlaced video sources, MPEG-2 provides the concept of field pictures and field-coded macroblocks in frame pictures. The top and bottom field of an interlaced frame can be coded together as frame picture or as two separate field pictures. In addition to the macroblock coding modes described above, field-picture macroblocks can also be coded in INTER-16×8 prediction mode, in which two different prediction signals are used, one for the upper and one for the lower half of a macroblock. For macroblocks in frame pictures, a similar coding mode is provided that uses different prediction signals for the top and bottom field lines of a macroblock. Macroblocks of both field and frame pictures can also be transmitted in dual prime mode. In this coding mode, the final prediction for each field is formed by averaging two prediction signals, of which one is obtained by referencing the field with the same parity and the other is obtained by referencing the field with the opposite parity as the current field. For coding of the residual data, MPEG-2 provides the possibility to use an alternative scanning pattern, which can be selected on picture level, and to choose between a frame-based and a field-based DCT coding of the prediction error signal.

The most widely implemented conformance point in the MPEG-2 standard is the Main Profile at the Main Level (MP@ML). MPEG-2 MP@ML compliant encoders find application in DVD-video, digital cable television, terrestrial broadcast of standard definition television, and direct-broadcast satellite (DBS) systems. This conformance point supports coding of CCIR 601 content at bit-rates up to 15 Mbit/s and permits use of B-pictures and interlaced prediction modes. In this work, an MPEG-2 encoder is included in the comparisons of video encoders for streaming and entertainment applications. The MPEG-2 bit-streams generated for our comparisons are compliant with the popular MP@ML conformance point with exception of the HDTV bit-streams, which are compliant with the MP@HL conformance point.

B. ITU-T Recommendation H.263

The first version of ITU-T Recommendation H.263 [2] defines a basic source coding algorithm similar to that of MPEG-2, utilizing the INTER-16×16, INTRA, and SKIP coding modes. But H.263 baseline contains significant changes that make it more efficient at lower bit-rates including median motion vector prediction and three-dimensional run-level-last variable length coding with tables optimized for lower bit-rates.

Moreover, version 1 of H.263 contains 8 Annexes (Annexes A-G) including four Annexes permitting source coding options (Annexes D, E, F, and G) for improved compression performance. Annexes D and F are in frequent use today. Annex D specifies the option for motion vectors to point outside the reference picture and to have longer motion vectors than H.263 baseline. Annex F specifies the use of overlapped block motion compensation and four motion vectors per macroblock with each motion vector assigned to an 8×8 sub-block, i.e., the use of variable block sizes. Hence, an INTER-8×8 coding mode is added to the set of possible macroblock modes.

H.263+ is the second version of H.263 [2][15], where several optional features are added to H.263 as Annexes I through T. Annex J of H.263+ specifies a de-blocking filter that is applied inside the motion prediction loop and is used together with the variable block-size feature of Annex F. H.263+ also adds some improvements in compression efficiency for the INTRA macroblock mode through prediction of intra DCT transform coefficients from neighboring blocks and specialized quantization and VLC coding methods for intra coefficients. This advanced syntax is described in Annex I of the ITU-T Recommendation H.263+. Annex I provides significant rate-distortion improvements between 1 and 2 dB compared to the H.263 baseline INTRA macroblock coding mode when utilizing the same amount of bits for both codecs [15]. Annex T of H.263+ removes some limitations of the baseline syntax in terms of quantization and also improves chrominance fidelity by specifying a smaller step size for chrominance coefficients than for luminance. The remaining Annexes contain additional functionalities including specifications for custom and flexible video formats, scalability, and backward-compatible supplemental enhancement information.

A second set of extensions that adds three more optional modes to H.263 [2] was completed and approved late in the year 2000. This version is often referred to as H.263++. The data partitioned slice mode (Annex V) can provide enhanced resilience to bit-stream corruption, which typically occurs during transmission over wireless channels, by separating header and motion vector information from transform coefficients. Annex W specifies additional backwards-compatible supplemental enhancement information including interlaced field indications, repeated picture headers, and the indication of the use of a specific fixed-point inverse DCT. Compression efficiency and robustness to packet loss can be improved by using the enhanced reference picture selection mode (Annex U), which enables long-term memory motion compensation [22][23]. In this mode, the spatial displacement vectors that indicate motion compensated prediction blocks are extended by variable time delay, permitting the predictions to originate from reference pictures other than the most recently decoded reference picture. Motion compensation performance is improved because of the larger number of possible predictions that are available by including more reference frames in the motion search. In Annex U, two modes are available for the buffering of reference pictures. The sliding-window mode – in which only the most recent reference pictures are stored – is the simplest and most commonly implemented mode. In the more flexible adaptive buffering mode, buffer management commands can be inserted into the bit-stream as side information, permitting an encoder to specify how long each reference picture remains available for prediction, with a constraint on the total size of the picture buffer. The maximum number of reference pictures is typically 5 or 10 when conforming to one of H.263's normative profiles, which are discussed next.

The ITU-T has recently approved Annex X of H.263, which provides a normative definition of profiles, or preferred combinations of optional modes, and levels, which specify maximum values for several key parameters of an H.263 bit-stream. Similar to their use in MPEG-2, each profile is designed to target a specific key application, or group of applications that require similar functionality. In this work, the rate-distortion capabilities of the Baseline Profile and the Conversational High Compression (CHC) Profile are compared to other standards for use in video conferencing applications. The Baseline Profile supports only baseline H.263 syntax (i.e. no optional modes) and exists to provide a profile designation to the minimal capability that all compliant decoders must support. The CHC Profile includes most of the optional modes that provide enhanced coding efficiency without the added delay that is introduced by B-pictures and without any optional error resilience features. Hence, it is the best profile to demonstrate the optimal rate-distortion capabilities of the H.263 standard for use in interactive video applications. Additionally, the High-Latency Profile (HL Profile) of H.263, which adds support for B-pictures to the coding efficiency tools of the CHC Profile, is included in the comparison of encoders for streaming applications, in which the added delay introduced by B-pictures is acceptable.

C. ISO/IEC Standard 14496-2: MPEG-4

MPEG-4 Visual [3] standardizes efficient coding methods for many types of audiovisual data, including natural video content. For this purpose, MPEG-4 Visual uses the baseline H.263 algorithm as a starting point so that all compliant MPEG-4 decoders must be able to decode any valid baseline H.263 bit-stream. However, MPEG-4 includes several additional features that can improve coding efficiency.

While spatial coding in MPEG-4 uses the 8×8 DCT and scalar quantization, MPEG-4 supports two different scalar quantization methods that are referred to as MPEG-style and H.263-style. In the MPEG-style quantization, perceptually weighted matrices, similar to those used in MPEG-2 assign a specific quantizer to each coefficient in a block, whereas in the H.263 method, the same quantizer is used for all AC coefficients. Quantization of DC coefficients uses a special non-linear scale that is a function of the quantization parameter. Quantized coefficients are scanned in a zigzag pattern and assigned run-length codes, as in H.263. MPEG-4 also includes alternate scan patterns for horizontally and vertically predicted INTRA blocks and the use of a separate VLC table for INTRA coefficients. These techniques are similar to those defined in Annex I of H.263.

Motion compensation in MPEG-4 is based on 16×16 blocks and supports variable block sizes, as in Annex F of H.263, so that one motion vector can be specified for each of the 8×8 sub-blocks of a macroblock, permitting the use of the INTER-8×8 mode. Version 1 of MPEG-4 supports only motion compensation at half-pixel accuracy, with bilinear interpolation used to generate values at half-pixel positions. Version 2 of MPEG-4 additionally supports the use of quarter-pixel accurate motion compensation, with a windowed 8-tap sinc function used to generate half-pixel positions and bilinear interpolation for quarter-pixel positions.

Motion vectors are permitted to point outside the reference picture and are encoded differentially after median prediction, according to H.263. MPEG-4 does not include a normative de-blocking filter inside the motion compensation loop, as in Annex J of H.263, but post-filters may be applied to the reconstructed output at the decoder to improve visual quality.

The MPEG-4 Simple Profile includes all features mentioned above, with the exception of the MPEG-style quantization method and quarter-pixel motion compensation. The Advanced Simple Profile adds these two features, plus B-pictures, global motion compensation (GMC) and special tools for efficient coding of interlaced video. A video coder compliant with the Simple Profile and the Advanced Simple Profile will be used in our experiments.

D. ITU-T Recommendation H.264 / ISO/IEC Standard 14496-10 AVC: H.264/AVC

H.264/AVC [4] is the latest joint project of the ITU-T VCEG and ISO/IEC MPEG. The H.264/AVC design covers a Video Coding Layer (VCL) and a Network Adaptation Layer (NAL). Although the VCL design basically follows the design of prior video coding standards such as MPEG-2, H.263, and MPEG-4, it contains new features that enable it to achieve a significant improvement in compression efficiency in relation to prior coding standards. For details please refer to [16]. Here, we will give a very brief description of the necessary parts of H.264/AVC in order to make the paper more self-contained.

In H.264/AVC, blocks of 4×4 samples are used for transform coding, and thus a macroblock consists of 16 luminance and 8 chrominance blocks. Similar to the I-, P-, and B-pictures defined for MPEG-2, H.263, and MPEG-4, the H.264/AVC syntax supports I-, P-, and B-slices. A macroblock can always be coded in one of several INTRA coding modes. There are two classes of INTRA coding modes, which are denoted as INTRA-16×16 and INTRA-4×4 in the following. In contrast to previous standards where only some of the DCT-coefficients can be predicted from neighboring INTRA-blocks, in H.264/AVC, prediction is always utilized in the spatial domain by referring to neighboring samples of already coded blocks. When using the INTRA-4×4 mode, each 4×4 block of the luminance component utilizes one of nine prediction modes. The chosen modes are transmitted as side information. With the INTRA-16×16 mode, a uniform prediction is performed for the whole luminance component of a macroblock. Four prediction modes are supported in the INTRA-16×16 mode. For both classes of INTRA coding modes, the chrominance components are predicted using one of four possible prediction modes.

In addition to the INTRA modes, H.264/AVC provides various other motion-compensated coding modes for macroblocks in P-slices. Each motion-compensated mode corresponds to a specific partition of the macroblock into fixed size blocks used for motion description. Macroblock partitions with block sizes of 16×16, 16×8, 8×16, and 8×8 luminance samples are supported by the syntax corresponding to the INTER-16×16, INTER-16×8, INTER-8×16, and INTER-8×8 macroblock modes, respectively. In case the INTER-8×8 macroblock mode is chosen, each of the 8×8 sub-macroblocks can be further partitioned into blocks of 8×8, 8×4, 4×8, or 4×4 luminance samples. H.264/AVC generally supports multi-frame motion-compensated prediction. That is, similar to Annex U of H.263, more than one prior coded picture can be used as reference for the motion compensation. In H.264/AVC, motion compensation is performed with quarter-pixel accurate motion vectors. Prediction values at half-pixel locations are obtained by applying a one-dimensional 6-tap FIR filter in each direction requiring a half-sample offset (horizontal or vertical or both, depending on the value of the motion vector), and prediction values at quarter-pixel locations are generated by averaging samples at the integer- and half-pixel positions. The motion vector components are differentially coded using either median or directional prediction from neighboring blocks.

In comparison to MPEG-2, H.263, and MPEG-4, the concept of B-slices is generalized in H.264/AVC. For details please refer to [17]. B-slices utilize two distinct reference picture lists, and four different types of INTER prediction are supported: list 0, list 1, bi-predictive and direct prediction. While list 0 prediction indicates that the prediction signal is formed by motion compensation from a picture of the first reference picture list, a picture of the second reference picture list is used for building the prediction signal if list 1 prediction is used. In the bi-predictive mode, the prediction signal is formed by a weighted average of a motion-compensated list 0 and list 1 prediction signal. The direct prediction mode differs from the one used in H.263 and MPEG-4 in that no delta motion vector is transmitted. Furthermore, there are two methods for obtaining the prediction signal referred to as temporal and spatial direct prediction, which can be selected by an encoder on the slice level. B-slices utilize a similar macroblock partitioning to P-slices. Besides the INTER-16×16, INTER-16×8, INTER-8×16, INTER-8×8 and the INTRA modes, a macroblock mode that utilizes direct prediction, the DIRECT mode, is provided. Additionally, for each 16×16, 16×8, 8×16, and 8×8 partition, the prediction method (list 0, list 1, bi-predictive) can be chosen separately. An 8×8 partition of a B-slice macroblock can also be coded in DIRECT-8×8 mode. If no prediction error signal is transmitted for a DIRECT macroblock mode, it is also referred to as B-slice SKIP mode.

H.264/AVC is basically similar to prior coding standards in that it utilizes transform coding of the prediction error signal. However, in H.264/AVC the transformation is applied to 4×4 blocks and, instead of the DCT, H.264/AVC uses a separable integer transform with basically the same properties as a 4×4 DCT. Since the inverse transform is defined by exact integer operations, inverse-transform mismatches are avoided. An additional 2×2 transform is applied to the four DC-coefficients of each

chrominance component. If the INTRA16×16-mode is in use, a similar operation extending the length of the transform basis functions is performed on the 4×4 DC-coefficients of the luminance signal.

For the quantization of transform coefficients, H.264/AVC uses scalar quantization, but without an extra-wide dead-zone around zero as found in H.263 and MPEG-4. One of 52 quantizers is selected for each macroblock by the quantization parameter Q. The quantizers are arranged in a way that there is an increase of approximately 12.5% in quantization step size when incrementing Q by one. The transform coefficient levels are scanned in a zigzag fashion if the block is part of a macroblock coded in frame mode; for field-mode macroblocks, an alternative scanning pattern is used. The 2×2 DC coefficients of the chrominance components are scanned in raster-scan order. All syntax elements of a macroblock including the vectors of scanned transform coefficient levels are transmitted by entropy coding methods.

Two methods of entropy coding are supported by H.264/AVC. The default entropy coding method uses a single infinite-extend codeword set for all syntax elements except the residual data. The vectors of scanned transform coefficient levels are transmitted using a more sophisticated method called Context-Adaptive Variable Length Coding (CAVLC). This scheme basically uses the concept of run-length coding as it is found in MPEG-2, H.263, and MPEG-4; however, VLC tables for various syntax elements are switched depending on the values of previously transmitted syntax elements. Since the VLC tables are well designed to match the corresponding conditional statistics, the entropy coding performance is improved in comparison to schemes using a single VLC table. The efficiency of entropy coding can be improved further if the Context-Adaptive Binary Arithmetic Coding (CABAC) is used. On the one hand, the usage of arithmetic coding allows the assignment of a non-integer number of bits to each symbol of an alphabet, which is extremely beneficial for symbol probabilities much greater than 0.5. On the other hand, the usage of adaptive codes permits adaptation to non-stationary symbol statistics. Another important property of CABAC is its context modeling. The statistics of already coded syntax elements are used to estimate conditional probabilities of coding symbols. Inter-symbol redundancies are exploited by switching several estimated probability models according to already coded symbols in the neighborhood of the symbol to encode. For details about CABAC please refer to [18].

For removing block-edge artifacts, the H.264/AVC design includes a de-blocking filter, which is applied inside the motion prediction loop. The strength of filtering is adaptively controlled by the values of several syntax elements.

Similar to MPEG-2, a frame of interlaced video can be coded as a single frame picture or two separate field pictures. Additionally, H.264/AVC supports a macroblock-adaptive switching between frame and field coding. Therefore, a pair of vertically adjacent macroblocks is considered as a coding unit, which can be either transmitted as two frame macroblocks or a top and a bottom field macroblock.

In H.264/AVC, three profiles are defined. The Baseline Profile includes all described features except B-slices, CABAC, and the interlaced coding tools. Since the main target application area of the Baseline Profile is the interactive transmission of video, it is used in the comparison of video encoders for video conferencing applications. In the comparison for video streaming and entertainment applications, which allow a larger delay, the Main Profile of H.264/AVC is used. The Main Profile adds support for B-slices, the highly efficient CABAC entropy coding method, as well as the interlaced coding tools.

III. Video Coder Control

One key problem in video compression is the operational control of the source encoder. This problem is compounded because typical video sequences contain widely varying content and motion, necessitating the selection between different coding options with varying rate-distortion efficiency for different parts of the image. The task of coder control is to determine a set of coding parameters, and thereby the bit-stream, such that a certain rate-distortion trade-off is achieved for a given decoder. This article focuses on coder control algorithms for the case of error-free transmission of the bit-stream. For a discussion of the application of coder control algorithms in the case of error-prone transmission, please refer to [19]. A particular emphasis is on Lagrangian bit-allocation techniques, which have emerged to form the most widely accepted approach in recent standard development. The popularity of this approach is due to its effectiveness and simplicity. For completeness, we briefly review the Lagrangian optimization techniques and their application to video coding.

A. Optimization Using Lagrangian Techniques

Consider K source samples that are collected in the K-tuple $\mathbf{S}= (\boldsymbol{S}_1,\ldots,\boldsymbol{S}_K)$. A source sample $\boldsymbol{S}_k$ can be a scalar or vector. Each source sample $\boldsymbol{S}_k$ can be quantized using several possible coding options that are indicated by an index out of the set $\boldsymbol{O}_k =(O_{k1},\ldots,O_{kN_k})$. Let $I_k \in \boldsymbol{O}_k$ be the selected index to code $\boldsymbol{S}_k$. Then the coding options assigned to the elements in $\mathbf{S}$ are given by the components in the K-tuple $\mathbf{I} =(I_1,\ldots,I_K)$. The problem of finding the combination of coding options that minimizes the distortion for the given sequence of source samples subject to a given rate constraint R_c can be formulated as

$$\min_{\mathbf{I}} D(\mathbf{S},\mathbf{I}) \quad \text{subject to } R(\mathbf{S},\mathbf{I}) \le R_c \tag{1}$$

Here, $D(\mathbf{S},\mathbf{I})$ and $R(\mathbf{S},\mathbf{I})$ represent the total distortion and rate, respectively, resulting from the quantization of $\mathbf{S}$ with a particular combination of coding options $\mathbf{I}$. In practice, rather than solving the constrained problem in (1), an unconstrained formulation is employed, that is

$$\mathbf{I}^* = \arg\min_{\mathbf{I}} J(\mathbf{S},\mathbf{I} \mid \lambda) \quad \text{with } J(\mathbf{S},\mathbf{I} \mid \lambda) = D(\mathbf{S},\mathbf{I}) + \lambda \cdot R(\mathbf{S},\mathbf{I}) \tag{2}$$

and $\lambda \ge 0$ being the Lagrange parameter. This unconstrained solution to a discrete optimization problem was introduced by Everett [20]. The solution $\mathbf{I}^*$ to (2) is optimal in the sense that if a rate constraint R_c corresponds to λ, then the total distortion $D(\mathbf{S},\mathbf{I}^*)$ is minimum for all combinations of coding options with bit-rate less or equal to R_c.

We can assume additive distortion and rate measures, and let these two quantities be only dependent on the choice of the parameter corresponding to each sample. Then, a simplified Lagrangian cost function can be computed using

$$J(\boldsymbol{S}_k,\mathbf{I} \mid \lambda) = J(\boldsymbol{S}_k, I_k \mid \lambda). \tag{4}$$

In this case, the optimization problem in (3) reduces to

$$\min_{\mathbf{I}} \sum_{k=1}^{K} J(\boldsymbol{S}_k,\mathbf{I} \mid \lambda) = \sum_{k=1}^{K} \min_{I_k} J(\boldsymbol{S}_k, I_k \mid \lambda) \tag{5}$$

and can be easily solved by independently selecting the coding option for each $\boldsymbol{S}_k \in \mathbf{S}$. For this particular scenario, the problem formulation is equivalent to the bit-allocation problem for an arbitrary set of quantizers, proposed by Shoham and Gersho [21].

B. Lagrangian Optimization in Hybrid Video Coding

The application of Lagrangian techniques to control a hybrid video coder is not straightforward because of temporal and spatial dependencies of the rate-distortion costs. Consider a block-based hybrid video codec such as H.261, H.263, H.264/AVC or MPEG-1/2/4. Let the image sequence s be partitioned into K distinct blocks $\boldsymbol{A}_k$ and the associated pixels be given as $\boldsymbol{S}_k$. The options $\mathbf{O}_k$ to encode each block $\boldsymbol{S}_k$ are categorized into INTRA and INTER, i.e. predictive coding modes with associated parameters. The parameters are transform coefficients and quantizer value Q for both modes plus one or more motion vectors for the INTER mode. The parameters for both modes are often predicted using transmitted parameters of preceding modes inside the image. Moreover, the INTER mode introduces a temporal dependency because reference is made to prior decoded pictures via motion compensated prediction. Hence, the optimization of a hybrid video encoder would require the minimization of the Lagrangian cost function in (2) for all blocks in the entire sequence. This minimization would have to proceed over the product space of the coding mode parameters. This product space is by far too large to be evaluated. Therefore, various publications elaborate on reductions of the product space and thus reducing complexity. For an overview, please refer to [24].

A simple and widely accepted method of INTER coding mode selection is to search for a motion vector that minimizes a Lagrangian cost criterion prior to residual coding. The bits and distortion of the following residual coding stage are either ignored or approximated. Then, given the motion vector(s), the parameters for the residual coding stage are encoded. The minimization of a Lagrangian cost function for motion estimation as given in (3) was first proposed by Sullivan and Baker [25].

Therefore, we split the problem of optimum bit allocation for INTER modes in a motion estimation and successive macroblock mode decision process between INTER or INTRA coding modes. The utilized macroblock mode decision is similar to [26] but without consideration of the dependencies of distortion and rate values on coding mode decisions made for past or future macroblocks. Hence, for each macroblock, the coding mode with associated parameters is optimized given the decisions made for prior coded blocks only. Consequently, the coding mode for each block is determined using the Lagrangian cost function in (3). Let the Lagrange parameter λ_{MODE} and the quantizer value Q be given. The Lagrangian mode decision for a macroblock $\boldsymbol{S}_k$ proceeds by minimizing

$$J_{MODE}(\boldsymbol{S}_k, I_k \mid Q, \lambda_{MODE}) = D_{REC}(\boldsymbol{S}_k, I_k \mid Q) + \lambda_{MODE} R_{REC}(\boldsymbol{S}_k, I_k \mid Q), \tag{7}$$

where the macroblock mode I_k is varied over the sets of possible macroblock modes for the various standards. As an example, the following sets of macroblock modes can be used for P-pictures (or P-slices) when coding progressive-scanned video:

- **MPEG-2:** INTRA, SKIP, INTER-16×16
- **H.263/MPEG-4:** INTRA, SKIP, INTER-16×16, INTER-8×8
- **H.264/AVC:** INTRA-4×4, INTRA-16×16, SKIP, INTER-16×16, INTER-16×8, INTER-8×16, INTER-8×8

Please note that although sometimes named identically here, the various modes are different between the above various standards.

H.264/AVC additionally provides the following set of sub-macroblock types for each 8×8 sub-macroblock of a P-slice macroblock that is coded in INTER-8×8 mode: INTER-8×8, INTER-8×4, INTER-4×8, and INTER-4×4.

The distortion $D_{REC}(\boldsymbol{S}_k,I_k|Q)$ and rate $R_{REC}(\boldsymbol{S}_k,I_k|Q)$ for the various modes are computed as follows: For the INTRA modes, the corresponding 8×8 (MPEG-2, H.263/MPEG-4) or 4×4 (H.264/AVC) blocks of the macroblock $\boldsymbol{S}_k$ are processed by transformation and subsequent quantization. The distortion $D_{REC}(\boldsymbol{S}_k,\text{INTRA}|Q)$ is measured as the sum of the squared differences (SSD) between the reconstructed (s') and the original (s) macroblock pixels

$$SSD = \sum_{(x,y)\in \boldsymbol{A}} \left| s[x,y,t] - s'[x,y,t] \right|^2 , \quad (8)$$

where $\boldsymbol{A}$ is the subject macroblock. The rate $R_{REC}(\boldsymbol{S}_k,\text{INTRA}|Q)$ is the rate that results after entropy coding.

For the SKIP mode, the distortion $D_{REC}(\boldsymbol{S}_k,\text{SKIP}|Q)$ and rate $R_{REC}(\boldsymbol{S}_k,\text{SKIP}|Q)$ do not depend on the current quantizer value. The distortion is determined by the SSD between the current picture and the value of the inferred INTER prediction, and the rate is given as one bit per macroblock for H.263 and MPEG-4, and approximately one bit per macroblock for MPEG-2 and H.264/AVC.

The computation of the Lagrangian costs for the INTER modes is much more demanding than for the INTRA and SKIP modes. This is because of the block motion estimation step. The size of the blocks $\boldsymbol{S}_i$ within a macroblock is $A\times B$ pixels for the INTER-A×B mode. Given the Lagrange parameter λ_{MOTION} and the decoded reference picture s', rate-constrained motion estimation for a block $\boldsymbol{S}_i$ is performed by minimizing the Lagrangian cost function

$$\boldsymbol{m}_i = \arg\min_{\boldsymbol{m}\in \boldsymbol{M}} \left\{ D_{DFD}(\boldsymbol{S}_i,\boldsymbol{m}) + \lambda_{MOTION} R_{MOTION}(\boldsymbol{S}_i,\boldsymbol{m}) \right\} \quad (9)$$

where M is the set of possible coding modes and with the distortion term being given by

$$D_{DFD}(\boldsymbol{S}_i,\boldsymbol{m}) = \sum_{(x,y)\in \boldsymbol{A}_i} \left| s[x,y,t] - s'[x-m_x, y-m_y, t-m_t] \right|^p \quad (10)$$

with p=1 for the SAD and p=2 for the SSD. $R_{MOTION}(\boldsymbol{S}_i,\boldsymbol{m})$ is the number of bits to transmit all components of the motion vector (m_x, m_y), and, in case multiple reference frames are used, m_t. The search range $\boldsymbol{M}$ is ±32 integer pixel positions horizontally and vertically and either 1 or more prior decoded pictures are referenced. Depending on the use of SSD or SAD, the Lagrange parameter λ_{MOTION} has to be adjusted.

The motion search that minimizes (9) proceeds first over integer-pixel locations. Then, the best of those integer-pixel motion vectors is tested whether one of the surrounding half-pixel positions provides a cost reduction in (9). This procedure of determination of a sub-pixel position is called half-pixel refinement. In the case quarter-pixel motion accuracy is used, the previously determined half-pixel location is used as the center for the corresponding sub-pixel refinement step, respectively. The sub-pixel refinement yields the resulting motion vector $\boldsymbol{m}_i$. The resulting prediction error signal $u[x,y,t,\boldsymbol{m}_i]$ is processed by transformation and subsequent quantization, as in the INTRA mode case. The distortion D_{REC} is also measured as the SSD between the reconstructed and the original macroblock pixels. The rate R_{REC} is given as the sum of the bits for the mode information, the motion vectors as well as the transform coefficients.

A final remark should be made regarding the choice of the Lagrange parameters λ_{MODE} and λ_{MOTION}. In [27][24] it was shown via experimental results that the following relationship is efficient for H.263/MPEG-4

$$\lambda_{MODE} = 0.85 \cdot Q_{H.263}^2 \quad (11)$$

and for SAD in (9),

$$\lambda_{MOTION} = \sqrt{\lambda_{MODE}} \, . \quad (12)$$

Correspondingly for SSD in (9), we would use

$$\lambda_{MOTION} = \lambda_{MODE} \, . \quad (13)$$

The experiment that lead to the relationship in (10) has also been conducted for H.264/AVC providing the following equation

$$\lambda_{MODE} = 0.85 \cdot 2^{(Q_{H.264}-12)/3} \quad (14)$$

for λ_{MODE}, and (12) and (13) for λ_{MOTION} .

Thus, rate control in those codecs is conducted via controlling for instance the quantization parameter and adjusting the Lagrange parameters accordingly using Eqs. (11)-(14).

IV. COMPARISON

We performed three separate experiments, each targeting a particular application area. The first experiment evaluates performance for video streaming while the second experiment targets video conferencing. The coding features used in these two applications differ primarily in that the low delay constraints that are imposed in the video conferencing experiment are relaxed in the video streaming case. Additionally, appropriate content is selected to represent each application space. The third experiment addresses entertainment-quality applications. In this experiment, the coding features are similar to those used in the video streaming case, but high-resolution video sources are used.

A. Video Streaming Applications

Table 6 of Appendix D shows results for the set of test sequences and test conditions specified in MPEG's recent Call for Proposals for New Tools to Further Improve Video Coding Efficiency (CfP) [28]. In that call, only MPEG-4 ASP and an H.264/AVC codec compliant with the outdated TML-8 [29] were tested. We have extended this comparison by the results of MPEG-2 ML@MP and H.263 HLP; and the results for H.264/AVC have been updated using the latest reference software version JM-61e. Details about the input sequences used in the tests are listed in Appendix B. All coders used only one I-picture at the beginning of a sequence, and 2 B-pictures have been inserted between each two successive P-pictures. Full search motion estimation with a range of ±32 integer pixels was used by all encoders along with the Lagrangian Coder Control described in the previous section.

The MPEG-2 Visual encoder generated bit-streams that are compliant with the popular ML@MP conformance point and the H.263 encoder used the HLP features. For MPEG-4 Visual, the ASP was used with quarter-sample accurate motion compensation and global motion compensation enabled. Additionally, the recommended de-blocking/de-ringing filter was applied as a post-processing operation. For the H.264/AVC JM-61e coder, the Main Profile was used with CABAC as entropy coding method. We have generally used five reference frames for both H.263 and H.264/AVC. The usage of B-pictures in the H.264/AVC encoder was restricted in a way that B-pictures are not used as reference pictures, and that all preceding reference pictures (in decoding order) are inserted in reference picture list 0, while only the future reference picture is placed in reference picture list 1. That is, this restricted B-picture concept for H.264/AVC used in the comparison is very similar to that of MPEG-2, H.263, and MPEG-4. To comply with the MPEG CfP conditions, the bit-rates were adjusted by using fixed quantization parameters.

The target bit-rates were always hit with a difference smaller than 2%. The quantization parameter for B-pictures was set in such a way that the corresponding quantization step size was approximately 20% larger than that for P-pictures for all codecs. Table 6 shows that with the H.264/AVC compliant encoder, performance gains of 1-3 dB are achieved in comparison with the MPEG-4 coder, 1-5 dB are achieved in comparison with H.263, and 3-6 dB are achieved in comparison with MPEG-2.

In the left column of Figure 1, rate-distortion curves for the four codecs are plotted for selected sequences. The test points corresponding to the MPEG CfP (Table 6) are marked inside the plots by white circles. For all sequences, H.264/AVC significantly outperforms the other codecs. In the right column of Figure 1 the bit-rate saving relative to the worst tested video coding standard, MPEG-2, is plotted against the PSNR of the luminance component for H.263 HLP, MPEG-4 ASP, and H.264/AVC MP.

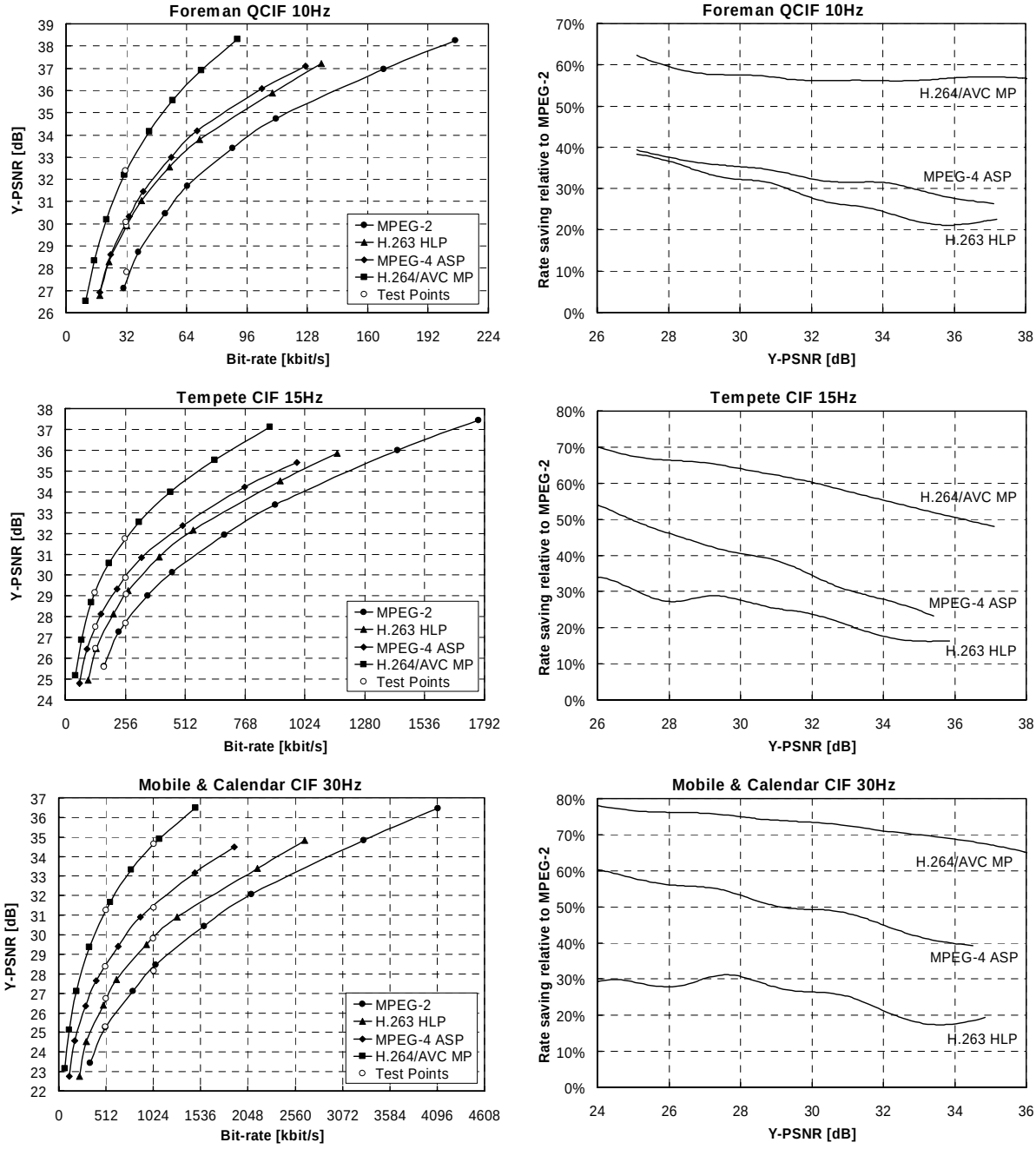


Figure 1: Selected rate-distortion curves and bit-rate saving plots for video streaming applications.

Table 1 presents the average bit-rate savings provided by each encoder relative to all other tested encoders over the entire set of sequences and bit-rates. It can be seen that H.264/AVC Coding significantly outperforms all other standards. On the most complex sequence of the test set, Mobile & Calendar (CIF, 30Hz), average bit-savings of more than 70% relative to MPEG-2 are realized. Bit-rate savings are as low as 50% on the Flower Garden sequence in CIF resolution (15Hz), with an average of 63% over the entire test set. H.264/AVC Main Profile provides more than 35% bit-rate savings relative to its two nearest competitors, MPEG-4 Advanced Simple and H.263 CHC. Note that H.264/AVC includes all of the main technical features used in these other encoder configurations, plus several additional features. The highly flexible motion model, the loop filtering and the very efficient context-based arithmetic coding scheme are the three primary factors that enable the superior rate-distortion performance of H.264/AVC Main Profile.

Table 1: Average bit-rate savings for video streaming applications

	Average bit-rate savings relative to:		
Coder	MPEG-4 ASP	H.263 HLP	MPEG-2
H.264/AVC MP	37.44%	47.58%	63.57%
MPEG-4 ASP	-	16.65%	42.95%
H.263 HLP	-	-	30.61%

B. Video Conferencing Applications

This experiment evaluates coding performance for interactive video applications, such as videoconferencing, in which a small delay and real-time encoding capability are the key requirements. Such applications generally support low to medium bit-rates and picture resolutions, with QCIF resolution at 10-128 kbit/s and CIF resolution at 128-512 kbit/s being the most common. The set of input sequences for this comparison consists of four QCIF (10Hz and 15Hz) and four CIF (15Hz and 30Hz) sequences. Refer to Appendix B for details about these sequences. Encoders included in this comparison are compliant with the following standards/profiles: the H.263 Baseline and Conversational High Compression (CHC) Profiles, the MPEG-4 Simple Profile, and the H.264/AVC Baseline Profile.

In all bit-streams, only the first picture was intra coded, with all of the subsequent pictures being temporally predicted (P-pictures). Both the H.263 CHC and H.264/AVC Baseline encoders used five reference pictures for long-term prediction. (This is the maximum number allowed for CIF sequences in Level 40 of H.263's normative profile and level definitions). A motion search range of ±32 integer pixels was employed by all encoders with the exception of H.263 Baseline, which is constrained by its syntax to a maximum range of ±16 integer pixels.

Since profiles are used to indicate decoder support for a set of optional modes, an encoder that is compliant with a particular profile is permitted – but not required – to use any of the optional modes supported in a that profile. With this in mind, encoders were configured by only including the optional modes from each profile that would produce the best possible rate-distortion performance, while satisfying the low delay and complexity requirements of interactive video applications.

As in the first experiment, we present both rate-distortion curves for luminance component, as well as plots of bit-rate savings relative to the poorest performing encoder. As should be expected, it is the H.263 Baseline encoder that provides the worst rate-distortion performance, and therefore it serves as the common basis for comparison. Figure 2 shows the rate-distortion plots as well as the bit-rate saving plots for three selected test sequences. The average bit-rate savings results over the entire test set are given in Table 2. In addition to the selected rate-distortion and bit-rate saving plots of Figure 2, results for fixed target bit-rates between 24 kbit/s for 10 Hz QCIF sequences and 256 kbit/s for 30 Hz CIF sequences are shown in Table 7 of Appendix D.

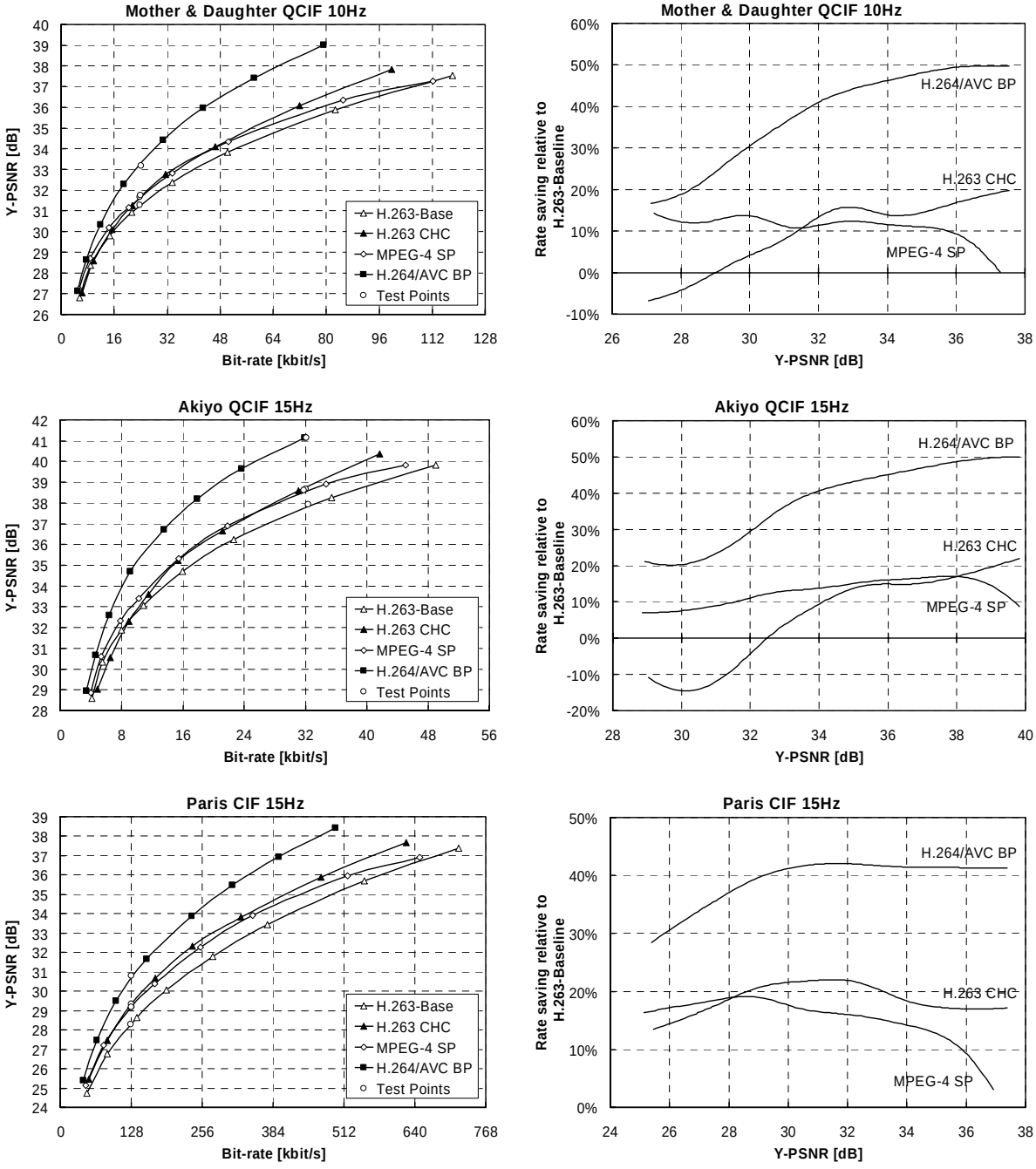


Figure 2: Selected rate-distortion curves and bit-rate saving plot for video conferencing applications.

Table 2: Average bit-rate savings for video conferencing applications

	Average bit-rate savings relative to:		
Coder	H.263 CHC	MPEG-4 SP	H.263 Base
H.264/AVC BP	27.69%	29.37%	40.59%
H.263 CHC	-	2.04%	17.63%
MPEG-4 SP	-	-	15.69%

It is immediately clear from these results that the next-generation H.264/AVC standard outperforms all of the other standards by a substantial margin. Bit-rate savings of more than 40% relative to H.263 Baseline are realized. Relative to H.263 CHC, H.264/AVC provides more than 25% bit-rate savings. These reported bit-rate savings are lower than the bit-rate savings measured in the first experiment for video streaming applications. This is mainly related to the fact that we have chosen typical videophone/videoconferencing sequences for the second experiment. These sequences are generally characterized by low or medium motion as well as low spatial detail. However, for H.264/AVC, the largest improvements of coding efficiency are obtained for complex sequences such as Mobile & Calendar. Furthermore, the H.264/AVC MP results for video streaming applications benefit from the usage of the highly efficient context-based arithmetic coding scheme that is not included in the Baseline Profile of H.264/AVC.

By examining the relative rate-distortion performance of various standards and profiles included in this experiment, further insight into the gains in coding efficiency provided by some of their key features can be obtained. For example, the MPEG-4 Simple Profile provides approximately 15% bit-rate savings over H.263 Baseline. The technical features that should contribute to this improvement include allowing motion compensation on 8×8 blocks, extrapolation of motion vectors over picture boundaries, and improved intra coding efficiency[1]. Additional bit-rate savings of –7 to 14 % are provided by H.263 CHC. The main technical difference between H.263 CHC and MPEG-4 SP is that H.263 CHC supports multi-frame motion compensated prediction and uses a modified chrominance quantization step size, which noticeably improves the chrominance fidelity[2].

C. Entertainment-Quality Applications

Our third experiment seeks to address coding efficiency for entertainment-quality applications, such as DVD-Video systems and HDTV. In such applications, sequences are generally encoded at resolutions of 720×480 pixels and higher at average bit rates of 3 Mbit/s and up. Since the MPEG-2 standard is the only standard commonly used in this application space, only its performance was compared to that of the H.264/AVC standard.

For this comparison we used a set of four interlaced-scan standard definition sequences at resolutions of 720×576 pixels (25 Hz) and four progressive-scan high definition sequences at resolutions of 1280×720 pixels (60 Hz); details about these sequences are specified in Appendix B.

Aside from the higher resolution source content, the experimental setup is very similar to that used in the video streaming applications test. The same encoding software was used for both standards, as well as similar coding options, including 2 B-pictures between each pair of anchor pictures, Lagrangian Coder Control, and full search motion estimation with a range of ±32 pixels. The MPEG-2 Visual encoder generated bit-streams that are compliant with the ML@MP and HL@MP conformance point for the standard definition and high definition sequences, respectively. For H.264/AVC, the Main Profile was used with 5 reference frames and CABAC as entropy coding. One key difference is that an I-picture was inserted every 480 ms for encoding the 25 Hz standard definition sequences and every 500 ms for encoding the 60 Hz high definition sequences. Frequent periodic INTRA coded pictures are typical in entertainment-quality applications in order to enable fast random access. As in the streaming test, the quantization parameter for B-pictures was set in a way that the resulting quantization step size is approximately 20% larger than that for P-pictures for both codecs.

[1] The maximum bit rate supported in any level of the Simple Profile is only 384 Kbps – a value that is exceeded by nearly every data point generated by the rate-distortion optimized encoder used in this test. Thus, only the simplest 30 Hz CIF content can really be encoded with acceptable visual quality while conforming to the bit rate restrictions of this profile. We have chosen to ignore this constraint in our analysis in order to measure the performance of the underlying technology rather than the confining the analysis only to cases within all limits of the MPEG-4 Visual specification.

[2] The rate-distortion as well as the bit-rate saving plots only consider the reconstruction quality of the luminance component.

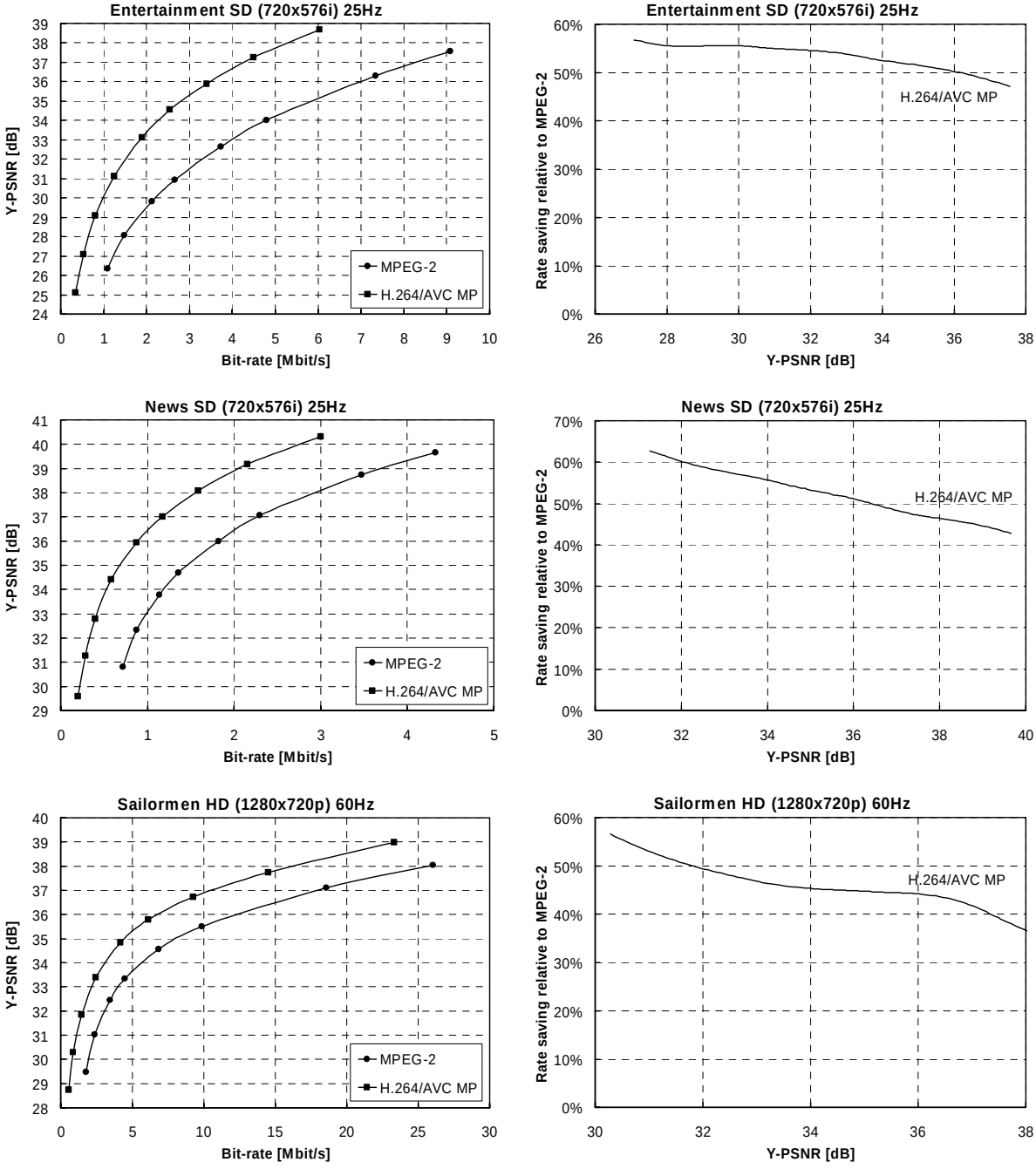


Figure 3: Selected rate-distortion curve and bit-rate saving plot for entertainment-quality applications.

The rate-distortion curves generated by the two encoders, along with the computed bit-rate savings realized by H.264/AVC over MPEG-2 based on points of equal PSNR, are shown in Figure 3 for three selected sequences. As in the previous tests, the H.264/AVC codec offers a significant rate savings advantage. At lower bit rates, savings lie between 45 and 65%, while at the higher bit rates, which are more common in entertainment-quality applications, rate savings of 25-45% are realized. The average rate saving measured over the entire set of sequences and bit-rates range is about 45%.

D. Subjective Comparisons

While PSNR is a convenient measure of distortion in video applications, it does not take into account all of the intricacies of the human visual system, which is the ultimate judge of video quality. With this in mind, we have carried out a set of informal subjective visual tests in order to validate and complement the results that have been derived using PSNR-based comparison. The

results of these tests indicate that H.264/AVC codec provides greater improvements in subjective quality over earlier standards than PSNR-based comparisons indicate.

One informal subjective test that was carried out at the HHI showed that for all cases in the streaming test set, sequences coded at 512 kbit/s with H.264/AVC are subjectively equivalent to the same sequences coded at 1024 kbit/s with MPEG-4 Visual. This corresponds to a bit-rate saving of H.264/AVC against MPEG-4 Visual of about 50% for these bit-rates, which is in general larger than the savings indicated on the rate-distortion plots. At lower bit-rates, the tests seem to indicate bit-rate savings between 30-40%.

In a second set of informal subjective tests performed at UB Video, several H.264/AVC, MPEG-4 ASP and H.263 CHC sequences with equivalent luminance PSNR were compared by a large number of viewers. Constant quantization parameters and no B-pictures were used by all encoders. The recommended de-blocking and de-ringing filters were applied as a post-process to the MPEG-4 ASP decoded sequences. The Test Model de-ringing filter was also applied to the H.263 CHC decoded sequences. Comparisons were made on each sequence between H.264/AVC and each of the other encoders, at both low and high bit-rates. While each pair of sequences had nearly identical PSNR values, the test subjects indicated a significant preference for the H.264/AVC sequences relative to the MPEG-4 ASP sequences. The preference towards H.264/AVC was strongest on the low bit rate coded sequences. Again, these results indicate that the bit-rate savings that can be achieved using H.264/AVC to achieve essentially equivalent visual quality as other standards are even larger than what the PSNR-based results indicate. Finally, we note that in the H.264/AVC to H.263 CHC comparison, only a minor preference towards H.264/AVC was expressed, on average. The results of these tests suggest that the use of a de-blocking filter inside the motion compensation loop, as found in H.263 CHC and H.264/AVC but not MPEG-4 ASP, may have an impact on subjective visual quality beyond what is reflected in PSNR-based results.

Similar subjective comparisons were made between the MPEG-2 and H.264/AVC sequences encoded for the entertainment-quality test. Again, the results illustrated that the bit rate savings that are provided by H.264/AVC are larger when subjective visual quality is used rather than PSNR measurements to determine points of equivalent quality. Approximately 10-15% greater rate savings were observed for the H.264/AVC codec over a range of bit-rates through subjective evaluation. At low bit-rates, H.264/AVC was perceived to provide equivalent quality at a bit rate reduced by 70% from that of MPEG-2. At higher bit rates, rate savings of approximately 40% were determined based on the evaluation by the test subjects.

V. Conclusions

The performance of the H.264/AVC compliant encoder in all experiments clearly demonstrates the potential importance of this standard in future applications of video streaming as well as interactive video coding. Although H.264/AVC coding shares the common hybrid video coding structure with previous standards, there are significant differences that provide substantial coding gains. The main difference between H.264/AVC and most previous standards is the largely increased flexibility, which provides increased coding efficiency for potentially increased computational complexity at the encoder. This would require intelligent implementation and coder control strategies, especially in streaming and broadcast applications.

Acknowledgement

The authors would like to thank the JVT for the collaborative work and the technically outstanding discussions and contributions that enabled this analysis.

Appendix A: Video Codecs

The software implementations used in the comparisons are as follows:

- **MPEG-2**: MPEG Software Simulation Group version 1.2. Public software, modified to include Lagrangian rate-distortion optimization. See http://www.mpeg.org/MSSG.
- **H.263**: University of British Columbia Signal Processing and Multimedia Group (UBC-SPMG), H.263 code library version 0.3. Available to ITU-T members and academic research organizations. See http://www.ece.ubc.ca/spmg/h263plus/h263plus.html.
- **MPEG-4**: The HHI MoMuSys-based rate-distortion optimized coder and the UB Video's *UB-Stream* version 2.0. Those two codecs were used to generate the anchors in MPEG's recent coding efficiency tests. See http://bs.hhi.de/~wiegand/ICG-Project-RDO.html and http://www.ubvideo.com.
- **H.264/AVC**: JVT JM-61e implementation developed by JVT members and with rate-distortion optimization by the HHI. Available at http://bs.hhi.de/~suehring/tml/download/jm61e.zip.

The various standard decoders together with bit-streams of all test cases presented in this paper can be down-loaded at ftp://ftp.hhi.de/ieee-tcsvt/.

APPENDIX B: TEST SEQUENCES

Details about the input video sequences used in the comparisons for video streaming, video conferencing, and entertainment applications are listed in Table 3, Table 4, and Table 5, respectively. All sequences use the YUV 4:2:0 color format, in which the two chrominance components are down-sampled by a factor of two in each spatial direction. The sequences used in the first two comparisons are popular QCIF and CIF resolution test sequences used in the video standards community.

Table 3: Input sequences used in the comparison for video streaming applications

Name	Res.	Duration	Characteristics
Foreman	QCIF	10 sec.	Fast camera and content motion with pan at the end
Container Ship	QCIF	10 sec.	Still camera on slow moving scene
News	QCIF	10 sec.	Still camera on human subjects with synthetic background
Tempete	QCIF	8.67 sec.	Camera zoom; spatial detail; fast random motion
Bus	CIF	5 sec.	Fast translational motion and camera panning; moderate spatial detail
Flower Garden	CIF	8.33 sec.	Slow and steady camera panning over landscape; spatial and color detail
Mobile & Calendar	CIF	8.33 sec.	Slow panning and zooming; complex motion; high spatial and color detail
Tempete	CIF	8.67 sec.	Camera zoom; spatial detail; fast random motion

Table 4: Input sequences used in the comparison for video conferencing applications

Name	Res.	Duration	Characteristics
Akiyo	QCIF	10 sec.	Still camera on human subject with synthetic background
Foreman	QCIF	10 sec.	Fast camera and content motion with pan at the end
Silent	QCIF	10 sec.	Still camera but fast moving subject
Mother & Daughter	QCIF	10 sec.	Still camera on human subjects
Carphone	CIF	10 sec.	Fast camera and content motion with landscape passing
Foreman	CIF	10 sec.	Fast camera and content motion with pan at the end
Paris	CIF	10 sec.	Still camera on human subjects; typical videoconferencing content
Sean	CIF	10 sec.	Still camera on human subject with synthetic background

Table 5: Input sequences used in the comparison for entertainment applications

Name	Res.	Duration	Characteristics
Harp & Piano	720×576i	8.8 sec.	Fast camera zoom; local motion
Basketball	720×576i	9.92 sec.	Fast camera and content motion; high spatial detail
Entertainment	720×576i	10 sec.	Camera and content motion; spatial detail
News	720×576i	10 sec.	Scene cut between slow and fast moving scene
Shuttle Start	1280×720p	10 sec.	Jiggling camera, low contrast, lighting change
Sailormen	1280×720p	10 sec.	Translational and random motion; high spatial detail
Night	1280×720p	7.67 sec.	Static camera, fast complex motion
Preakness	1280×720p	10 sec.	Camera zoom, highly complex motion, high spatial detail

APPENDIX C: PERFORMANCE MEASURES

Since it is the most widely accepted objective measure of visual distortion, PSNR of the luminance component is our primary means of measuring visual distortion. The PSNR between the reconstructed (s') and the original (s) video signal for the set of pixels in $\boldsymbol{A}$ is determined via

$$\mathrm{PSNR} = 10 \log_{10} \frac{255^2}{\mathrm{MSE}} \mathrm{dB} \quad (15)$$

with

$$\mathrm{MSE} = \frac{1}{|\boldsymbol{A}|} \mathrm{SSD}\,, \quad (16)$$

where the SSD is given via (8) and $|\boldsymbol{A}|$ specifies the number of pixels in $\boldsymbol{A}$.

For each test case and sequence, results are presented in a set of rate-distortion curves, with one curve for each encoder being evaluated. A curve is generated by encoding each sequence several times with different quantization step sizes, which are held constant throughout each of the coding passes. The average PSNR for each of the three components over all of the frames in the sequence is recorded and plotted versus the average bit-rate. These results indicate differences in achievable rate-distortion performance between different standards.

A more practical and tangible quantity to measure is the percentage bit-rate savings that one standard can provide relative to another, while achieving equivalent visual quality. These calculations can be made by interpolating between points on two rate-distortion curves, aligning points of equal distortion, and then computing the difference in bit-rate between these points. In order to make such comparisons between several rate-distortion curves, the curve of the encoder with the poorest performance is used as a common base for comparison against all of the other encoders. This can be expressed as:

$$S(\mathrm{PSNR}) = 100 \cdot \frac{A(\mathrm{PSNR}) - B(\mathrm{PSNR})}{A(\mathrm{PSNR})} \% \quad (17)$$

where B and A represent the bit-rates necessary to achieve a given PSNR value, using the encoder in question (B) and the common anchor encoder (A), respectively.

While these objective measures are convenient and widely accepted, we recognize that the ultimate judge of quality is the human viewer. To this end, small-scale informal subjective tests were conducted in order to validate the results found using PSNR measures. Sequences used in the tests achieved a target bit-rate, within a tolerance of ±2% by selecting the necessary fixed quantizer to achieve the rate. One change in the quantizer value was permitted at some point in the sequence, to facilitate meeting the target rate within the small tolerance. This procedure is similar to that used for subjective testing in MPEG's recent Call for Proposals for new techniques to improve the efficiency of video coding [28].

APPENDIX D: TABLES

The fixed bit-rate results for video conferencing and video streaming applications are summarized in Table 6 and Table 7, respectively, for various test cases. The achieved bit-rates are given in kbit/s, and the PSNR values for the luminance and the two chrominance components are given in dB.

Table 6: Fixed bit-rate results for video streaming applications

	MPEG-2				H.263 HLP				MPEG-4 ASP				H.264/AVC MP			
Sequence	Rate	PSNR-Y	PSNR-U	PSNR-V	Rate	PSNR-Y	PSNR-U	PSNR-V	Rate	PSNR-Y	PSNR-U	PSNR-V	Rate	PSNR-Y	PSNR-U	PSNR-V
A: QCIF, 10 Hz, 32 kbit/s																
Foreman	32.12	27.81	35.14	34.96	32.18	29.90	37.73	37.70	31.92	30.09	37.33	37.33	31.49	32.40	38.68	38.98
Container	32.22	32.71	39.75	39.04	31.97	35.96	41.38	41.12	31.82	36.42	42.46	42.23	31.89	38.57	43.00	42.98
News	32.44	29.97	35.07	36.75	32.50	34.06	38.68	39.31	32.24	33.30	37.50	38.58	31.96	35.75	39.45	40.05
Tempete	36.91	24.83	29.38	32.04	32.24	26.62	32.50	34.74	31.68	27.87	31.61	34.21	31.83	29.62	33.58	36.02
B: QCIF, 15 Hz, 64 kbit/s																
Foreman	63.45	30.36	37.07	37.28	65.14	32.38	38.65	38.93	64.38	32.81	38.73	39.15	63.42	35.21	40.00	40.67
Container	63.95	34.34	40.95	40.40	63.97	38.26	43.32	43.22	63.87	38.47	44.21	43.87	63.67	40.67	44.80	44.92
News	63.45	32.61	37.33	38.55	63.80	36.25	39.79	40.43	64.00	35.78	39.37	40.67	63.98	38.80	41.71	42.27
Tempete	65.21	26.36	30.65	33.14	64.39	28.39	33.34	35.57	64.13	29.39	32.59	35.14	63.43	31.78	34.65	36.89
C: CIF, 15Hz, 128 kbit/s																
Foreman	130.37	28.94	35.78	36.30	128.40	30.91	38.35	39.26	127.83	31.30	38.16	38.99	128.70	33.66	39.49	40.87
Container	127.90	32.63	39.94	39.95	129.02	34.99	42.00	41.84	128.62	35.28	42.16	41.91	128.67	36.74	42.40	42.40
News	129.84	32.73	37.92	38.98	129.02	36.68	40.82	41.47	126.97	35.71	39.20	40.58	128.25	38.21	41.21	42.09
Tempete	165.75	25.60	30.67	33.33	129.07	26.47	33.42	35.66	129.11	27.51	32.03	34.78	126.34	29.16	34.41	36.71
D: CIF, 15 Hz, 256 kbit/s																
Bus	260.78	25.96	35.78	36.25	258.76	26.97	37.60	38.87	256.15	28.31	37.57	39.15	256.14	29.86	38.44	39.96

Mobile	256.01	24.59	29.96	30.17	259.20	25.66	31.97	32.40	258.88	27.07	32.24	32.63	254.87	29.73	34.26	34.69
Flower	261.67	23.93	28.82	32.37	257.85	24.89	31.58	33.56	255.97	26.07	30.89	33.90	257.89	28.08	33.02	35.08
Tempete	257.65	27.68	32.45	34.82	259.28	29.06	34.54	36.75	256.58	29.86	34.09	36.60	254.37	31.74	35.83	37.98
E: CIF, 30 Hz, 512 kbit/s																
Bus	506.29	27.35	36.43	37.62	511.98	28.77	38.16	39.41	511.88	29.75	38.28	39.89	511.85	31.89	39.29	40.85
Mobile	506.26	25.31	30.26	30.47	513.05	26.74	32.40	32.85	505.03	28.36	33.12	33.54	512.58	31.27	35.18	35.65
Flower	518.64	25.71	30.25	33.08	517.90	26.35	31.99	34.14	511.76	27.96	32.16	34.79	514.59	30.16	33.95	35.67
Tempete	521.40	28.43	32.91	35.14	513.73	29.45	34.94	37.11	510.55	30.84	34.74	37.18	515.49	32.79	36.36	38.38
F: CIF, 30 Hz, 1024 kbit/s																
Bus	1022.54	30.72	38.70	40.12	1025.80	31.91	39.55	41.21	1022.54	32.82	39.94	41.60	1025.51	35.24	40.77	42.59
Mobile	1029.58	28.16	33.00	33.27	1024.27	29.82	34.43	34.83	1029.18	31.37	35.29	35.74	1026.00	34.64	37.27	37.74
Flower	1034.33	28.66	32.92	35.10	1033.05	29.77	33.77	35.27	1024.30	31.20	34.58	36.61	1020.08	33.67	36.23	37.32
Tempete	1029.56	31.30	35.17	37.13	1022.81	32.55	36.53	38.50	1025.77	33.34	36.51	38.69	1020.06	35.54	37.90	39.68

Table 7: Fixed bit-rate results for video conferencing applications

	H.263 Baseline				H.263 CHC				MPEG-4 SP				H.264/AVC Baseline			
Sequence	Rate	PSNR-Y	PSNR-U	PSNR-V	Rate	PSNR-Y	PSNR-U	PSNR-V	Rate	PSNR-Y	PSNR-U	PSNR-V	Rate	PSNR-Y	PSNR-U	PSNR-V
A: QCIF, 10 Hz, 24 kbit/s																
Akiyo	24.14	37.34	39.73	41.31	24.06	38.54	41.89	42.93	24.19	38.01	40.24	41.95	24.00	40.68	42.90	43.58
Foreman	24.21	27.73	35.39	34.95	24.25	28.52	37.39	37.37	24.09	29.10	36.27	35.95	23.87	30.08	37.45	37.58
Mother & Daughter	23.78	31.27	36.49	36.32	23.82	31.68	37.80	37.65	23.97	31.75	36.62	36.37	24.08	33.19	37.96	37.71
Silent	24.08	31.12	35.44	36.93	23.90	32.31	37.28	38.83	24.14	31.68	35.51	37.02	24.09	32.42	36.34	38.07
B: QCIF, 15 Hz, 32 kbit/s																
Akiyo	32.31	37.93	40.53	41.87	32.05	38.68	41.97	42.98	31.76	38.62	41.12	42.60	32.07	41.15	43.22	43.95
Foreman	31.78	28.17	35.38	35.01	32.10	28.66	37.39	37.34	32.13	29.35	36.19	36.13	32.37	30.51	37.58	37.60
Mother & Daughter	31.77	31.56	36.62	36.49	31.74	31.87	37.81	37.61	32.27	31.96	36.73	36.70	32.14	33.66	37.99	37.81
Silent	31.79	31.21	35.46	36.90	31.88	32.58	37.58	38.89	31.97	31.95	35.74	37.39	32.18	32.47	36.45	38.04
C: CIF, 15 Hz, 128 kbit/s																
Carphone	129.71	31.53	35.94	37.03	127.64	32.32	38.02	39.24	127.82	32.50	36.62	37.73	125.64	33.50	37.75	39.23
Foreman	128.32	29.92	36.40	37.00	127.97	30.76	38.50	39.39	128.65	31.52	37.71	38.45	127.24	32.96	38.77	40.06
Paris	127.38	28.30	33.30	33.84	128.29	29.34	35.56	36.32	127.95	29.18	33.59	34.25	128.52	30.81	35.80	36.18
Sean	129.74	36.64	40.56	41.07	128.47	37.91	41.71	42.29	127.37	36.75	40.50	41.31	129.89	39.46	42.22	43.05
D: CIF, 30 Hz, 256 kbit/s																
Carphone	258.89	32.47	36.35	37.54	256.20	33.31	38.20	39.62	256.71	33.34	36.99	38.20	257.42	34.39	37.79	39.21
Foreman	254.66	31.60	37.23	37.86	256.49	32.06	38.96	40.05	258.48	32.39	38.08	39.03	253.62	34.27	39.59	40.85
Paris	257.05	29.55	34.08	34.70	258.19	30.56	36.19	36.65	254.91	30.34	34.44	34.95	256.43	32.24	36.67	36.93
Sean	254.91	37.94	41.42	42.06	258.52	39.53	43.03	43.65	258.09	37.89	41.59	42.45	257.54	40.72	43.26	44.17

VI. REFERENCES

[1] ITU-T and ISO/IEC JTC1, "Generic coding of moving pictures and associated audio information – Part 2: Video," ITU-T Recommendation H.262 – ISO/IEC 13818-2 (MPEG-2), Nov. 1994.

[2] ITU-T, "Video coding for low bitrate communication," ITU-T Recommendation H.263; version 1, Nov. 1995; version 2, Jan. 1998; version 3, Nov. 2000.

[3] ISO/IEC JTC1, "Coding of audio-visual objects – Part 2: Visual," ISO/IEC 14496-2 (MPEG-4 visual version 1), Apr. 1999; Amendment 1 (version 2), Feb. 2000; Amendment 4 (streaming profile), Jan. 2001.

[4] T. Wiegand and G. J. Sullivan, "Draft ITU-T Recommendation H.264 and Final Draft International Standard of Joint Video Specification (ITU-T Recommendation H.264 | ISO/IEC 14496-10 AVC)", Joint Video Team of ISO/IEC JTC1/SC29/WG11 and ITU-T SG16/Q.6 Doc. JVT-G050, Pattaya, Thailand, Mar. 2003.

[5] T. Wiegand and B. D. Andrews, "An Improved H.263 Coder Using Rate-Distortion Optimization," ITU-T/SG16/Q15-D-13, Tampere, Finland, Apr. 1998.

[6] M. Gallant, G. Cote, and F. Kossentini, "Description of and Results for Rate-Distortion Based Coder," ITU-T/SG16/Q15-D-47, Tampere, Finland, Apr. 1998.

[7] ITU-T/SG 16/VCEG (formerly Q.15, now Q.6), "Video Codec Test Model Near-Term Number 10 (TMN-10)," Tampere, Finland, Apr. 1998.

[8] T. Wiegand (ed.), "Working Draft Number 2, Revision 8 (WD-2 rev 8)," Joint Video Team of ISO/IEC MPEG and ITU-T VCEG, JVT-B118r8, Apr. 2002.

[9] ISO/IEC JTC1/SC29/WG11, "MPEG-4 Video Verification Model 18.0 (VM-18)," Doc. MPEG-N3908, Jan. 2001.

[10] ISO/IEC JTC1/SC29/WG11, "Test Model 5," Doc. MPEG-N0400, Apr. 1993.

[11] ITU-T, "Video Codec for Audiovisual Services at px64 kbit/s," ITU-T Recommendation H.261, Version 1, Nov. 1990; Version 2, Mar. 1993.

[12] J.L. Mitchell, W.B. Pennebaker, C. Fogg, and D.J. LeGall, "MPEG Video Compression Standard," Chapman and Hall, New York, USA, 1997.

[13] B.G. Haskell, A. Puri, A.N. Netravalli, "Digital Video: An Introduction to MPEG-2," Chapman and Hall, NewYork, USA, 1997.

[14] ISO/IEC JTC1, "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s – Part 2: Video," ISO/IEC 11172-2 (MPEG-1), Mar. 1993.

[15] G. Cote, B. Erol, M. Gallant, and F. Kossentini. "H.263+: Video Coding at Low Bit Rates", IEEE Transactions on Circuits and Systems for Video Technology, vol. 8, pp. 849–866, Nov. 1998.

[16] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra: "Overview of the H.264/AVC Video Coding Standard", in IEEE Transactions on Circuits and Systems for Video Technology, this issue.

[17] M. Flierl and B. Girod. "Generalized B Pictures and the Draft H.264/AVC Video Compression Standard", in IEEE Transactions on Circuits and Systems for Video Technology, this issue.

[18] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Adaptive Binary Arithmetic Coding for H.264/AVC," in IEEE Transactions on Circuits and Systems for Video Technology, this issue.

[19] T. Stockhammer, M. M. Hannuksela, and T. Wiegand. "H.264/AVC in Wireless Environments", in IEEE Transactions on Circuits and Systems for Video Technology, this issue.

[20] H. Everett, "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources," Operations Research, vol. 11, pp. 399-417, 1963.

[21] Y. Shoham and A. Gersho, "Efficient Bit Allocation for an Arbitrary Set of Quantizers," IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 36, pp. 1445-1453, Sep. 1988.

[22] T. Wiegand, X. Zhang, and B. Girod, "Long-Term Memory Motion-Compensated Prediction," IEEE Transactions on Circuits and Systems for Video Technology, vol. 9, pp. 70-84, Feb. 1999.

[23] T. Wiegand and B. Girod, "Multi-frame Motion-Compensated Prediction for Video Transmission," Kluwer Academic Publishers, Sep. 2001.

[24] G. J. Sullivan and T. Wiegand "Rate-Distortion Optimization for Video Compression," IEEE Signal Processing Magazine, vol. 15, pp. 74-90, Nov. 1998.

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

[26] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra. "Rate-Distortion Optimized Mode Selection for Very Low Bit Rate Video Coding and the Emerging H.263 Standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 6, pp. 182–190, Apr. 1996.

[27] T. Wiegand and B. Girod, "Lagrangian Multiplier Selection in Hybrid Video Coder Control," in Proc. ICIP 2001, Thessaloniki, Greece, Oct. 2001.

[28] ISO/IEC JTC1/SC29/WG11, "Call for Proposals On New Tools For Video Compression Technology," Doc. MPEG-N4065, Mar. 2001.

[29] ITU-T/SG 16/VCEG (formerly Q.15 now Q.6), "H.26L Test Model Long Term Number 8 (TML-8) draft 0," Doc. VCEG-N10, Jul. 2001.

Thomas Wiegand is the head of the Image Communication Group in the Image Processing Department of the Heinrich Hertz Institute Berlin, Germany. He received the Dr.-Ing. degree from the University of Erlangen-Nuremberg, Germany, in 2000 and the Dipl.-Ing. degree in Electrical Engineering from the Technical University of Hamburg-Harburg, Germany, in 1995.

From 1993 to 1994, he was a Visiting Researcher at Kobe University, Japan. In 1995, he was a Visiting Scholar at the University of California at Santa Barbara, USA, where he started 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 -

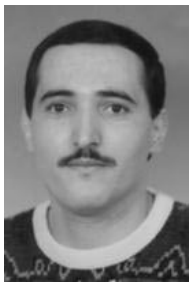
VCEG) / ISO/IEC Moving Pictures Experts Group (ISO/IEC JTC1/SC29/WG11 - MPEG) / Joint Video Team (JVT) standardization efforts and holds various international patents in this field. From 1997 to 1998, he has been a Visiting Researcher at Stanford University, USA, and served as a consultant to 8x8, Inc., Santa Clara, CA, USA.

In October 2000, he has been appointed as the Associated Rapporteur of the ITU-T VCEG. In December 2001, he has been appointed as 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. In February 2002, he has been appointed as the Editor of the H.264/AVC video coding standard.

Heiko Schwarz received the Dipl.-Ing. degree in electrical engineering from the University of Rostock, Germany, in 1996 and the Dr.-Ing. degree from the University of Rostock in 2000. In 1999, he joined the Fraunhofer-Institute for Communications HHI, Berlin, Germany. His research interests include image and video compression, video communication as well as signal processing.

Anthony Joch received the B. Eng. degree in computer engineering from McMaster University, Hamilton, ON, Canada, in 1999, and the M.A.Sc. degree in electrical engineering from the University of British Columbia, Vancouver, BC, Canada, in 2002.

In 2000, he joined UB Video Inc., Vancouver, BC, where he is currently a Senior Engineer involved in the development of software codecs for the H.264/MPEG-4 AVC standard. His research interests include reduced-complexity algorithms for video encoding, video pre- and post-processing, and multimedia systems. He has been an active contributor to the H.264/MPEG-4 AVC standardization effort, particularly in the area of deblocking filtering and as a co-chair of the ad-hoc group for bitstream exchange.



Faouzi Kossentini received the B.S., M.S., and Ph.D. degrees from the Georgia Institute of Technology, Atlanta, in 1989, 1990, and 1994, respectively. He is presently the President and Chief Executive Officer of UB Video. Dr. Kossentini is also an associate professor at the department of Electrical and Computer Engineering at the University of British Columbia, doing research in the areas of signal processing, communications and multimedia. Dr. Kossentini has co-authored more than one hundred and fifty journal papers, conference papers and book chapters. He has also participated in numberous international ISO and ITU-T activities involving the standardization of coded representation of audiovisual information. Dr. Kossentini is a senior member of the IEEE. He was also a Vice General Chair for ICIP-2000, and he was an associate editor for the IEEE transactions on Image Processing and the IEEE Transactions on Multimedia.

Gary J. Sullivan is the chairman of the Joint Video Team (JVT) for the development of the next-generation H.264/MPEG4-AVC video coding standard, which is in the final stages of approval as a joint project between the ITU-T video coding experts group (VCEG) and the ISO/IEC moving picture experts group (MPEG).

He is also the Rapporteur of Advanced Video Coding in the ITU-T, where he has led VCEG (ITU-T Q.6/SG16) for about six years. He is also the ITU-T video liaison representative to MPEG and served as MPEG's (ISO/IEC JTC1/SC29/WG11) video chairman from March of 2001 to May of 2002.

He is currently a program manager of video standards and technologies in the eHome A/V platforms group of Microsoft Corporation. At Microsoft he designed and remains lead engineer for the DirectX(r) Video Acceleration API/DDI feature of the Microsoft Windows(r) operating system platform.

Prior to joining Microsoft in 1999, he was the Manager of Communications Core Research at PictureTel Corporation, the quondam world leader in videoconferencing communication. He was previously a Howard Hughes Fellow and Member of the Technical Staff in the Advanced Systems Division of Hughes Aircraft Corporation and was a terrain-following radar system software engineer for Texas Instruments. He received his Ph.D. and Engineer degrees in Electrical Engineering from the University of California, Los Angeles, in 1991.

His research interests and areas of publication include image and video compression, rate-distortion optimization, motion representation, scalar and vector quantization, and error and packet loss resilient video coding.