

# CONCEPTS AND PERFORMANCE OF NEXT-GENERATION VIDEO COMPRESSION STANDARDIZATION

*Till Halbach*

Department of Telecommunications  
Norwegian University of Science and Technology  
7491 Trondheim, Norway  
Email: halbach@tele.ntnu.no  
Phone: +47 73 59 44 88  
Fax: +47 73 59 26 40

*Mathias Wien*

Institute of Communications Engineering  
Aachen University  
52056 Aachen, Germany  
Email: wien@ient.rwth-aachen.de  
Phone: +49 241 80 2 76 81  
Fax: +49 241 80 2 21 96

## ABSTRACT

The emerging video coding standard ITU-T H.264 | ISO/IEC MPEG-4 AVC is introduced and described. The concept shows a significantly improved performance as compared to former video coding standards like H.263 or MPEG-4 Visual. Enhanced inter and intra prediction techniques are the key to high compression efficiency.

In this paper, the main features of the standard are described, and a performance analysis is given. The description outlines the algorithms used in the Baseline and in the Main profile of the standard. The focus of the performance analysis is on intra coding, giving a detailed comparison to the state-of-the-art still-image compression standard JPEG2000.

The simulation results reveal an excellent objective and subjective compression performance for the new standard, especially at very low bit rates.

## 1. INTRODUCTION

The Joint Video Team (JVT) of the ITU-T Video Coding Experts Group VCEG and the ISO/IEC Moving Picture Experts Group MPEG have recently drafted a joint video coding standard that will be called ITU-T Recommendation H.264 and ISO/IEC MPEG-4 Advanced Video Coding [1]. H.264, also known under its project name H.26L, represents the latest generation of standardized video compression systems, and is expected to replace older standards like MPEG-4 Part 2 (Visual) [2] and H.263++ [3]. At the moment, H.264 has the status Final Committee Draft (FCD), and is planned to be finished by December 2002.

This paper is organized as follows. First, we give an overview of the standard and describe employed techniques. The focus of the description is on the so-called Baseline and Main profiles of the standard. It

includes both motion and intra prediction as well as the concept of adaptive block-size transforms that are introduced in the Main profile.

We briefly review some prior performance comparisons before we compare H.264's intra-frame coding efficiency to the known still image compression standards JPEG, JPEG2000, and Motion JPEG2000.

## 2. KEY FEATURES OF H.264

The concepts of H.264 are very similar to established standards like H.261, H.263, and MPEG-1,2,4. The standard is based on the hybrid coding scheme, i.e. motion between frames of the sequence is predicted using motion vectors, and the prediction error is then transformed, quantized, and transmitted. Motion prediction in inter-coded frames can be performed using one or two reference frames that may be temporally located before or after the current frame. Prediction is also the method of choice in intra coding, however based only on content of the current frame. All components of the codec are designed to allow for a 16-bit implementation to enable realization on 16-bit architectures, e.g. DSPs.

While the former video coding standards specified the Discrete Cosine Transform (DCT) for transform coding of the prediction error, H.264 employs integer transforms. In contrast to the DCT, these transforms allow for a bit-exact specification.

In order to remove blocking artifacts generated by the block-wise prediction and reconstruction, a deblocking filter is placed in the feed-back loop after inverse quantization and inverse transform for both encoder and decoder. The strength of the filter is driven by the prediction type, motion vectors, the prediction error energy, and by the employed quantization level.

For encoding, frames are divided into slices, that represent independently decodable parts of a frame,

hereby increasing the codec’s error robustness. Slices consist of macroblocks (MBs) representing a region of  $16 \times 16$  luminance pixels and their corresponding chrominance pixels. Currently, YUV with 4:2:0 sub-sampling is the only color space used in the specification. This means, a macroblock consist of a  $16 \times 16$  luminance block and two (subsampled)  $8 \times 8$  chrominance blocks.

A further element to enhance robustness is the split of the system into two hierarchically grouped layers. The Video Coding Layer is responsible for all coding matters as in conventional systems. It is the task of the lower-lying Network Adaptation Layer to account for different applications like conversational communication types, video transmission by means of H.32x series packets, or considering RTP/UDP/IP-like communications.

The standard comprises three profiles, called Baseline, Main, and X, that represent different tool sets specified for different applications. The Baseline profile consists of low-complexity low-latency technical features and aims at interactive applications. Potentially error-prone environments are accounted for as well. The tools of the Baseline profile are common to all profiles. The Main profile aims at high coding efficiency, e.g. for broadcasting applications. The tools of the Baseline and the Main profile are described in more detail in the following. Profile X is designed mainly for streaming applications. Here, error robustness and error resilience as well as techniques for flexible access and switching between bit streams are included.

The profiles are organized in so-called levels that define certain parameter values like maximum picture size and processing rate. Each level implies inclusion of the requirements of all lower-numbered levels.

### 2.1. Baseline profile

A Baseline decoder has to operate on intra (I) slices, consisting of intra-frame MBs only, and inter (P) slices which contain mainly temporally predicted MBs but may also include intra-frame MBs. P slices are predicted by one or more temporally preceding P or I slices. In order to adapt the prediction more precisely than previous standards to movement within one video picture, MBs can be further divided into subblocks with one or both dimensions cut into halves as depicted in Fig. 1.

$8 \times 8$  subblocks may in turn be divided into further subblocks according to the same scheme. This leads to a tree-structured motion segmentation with possible block sizes ( $X \times Y$ )  $16 \times 16$ ,  $16 \times 8$ ,  $8 \times 16$ ,  $8 \times 8$ ,  $8 \times 4$ ,  $4 \times 8$ , and  $4 \times 4$ . All subblocks within one MB are of the same type (I/P).

For intra prediction, there are 8 different direc-

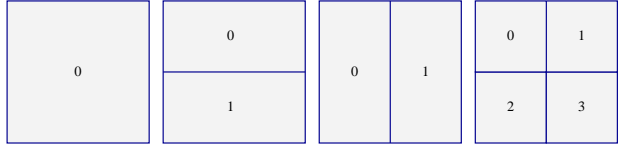


Figure 1: Possible block splits for motion compensation on  $16 \times 16$  and  $8 \times 8$  blocks.

tional prediction modes, and an additional DC (mean) prediction mode for  $4 \times 4$  blocks. The pixels along the block boundaries from the decoded neighboring upper and left blocks are used for the prediction. The corresponding directions are depicted in Fig. 2. In addition, there are 4 intra-frame prediction modes on a  $16 \times 16$ -block basis, including DC, horizontal, vertical and a so-called plane prediction.

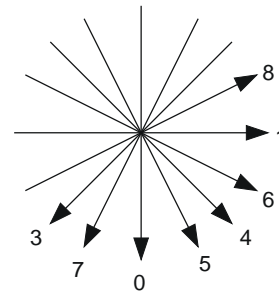


Figure 2: Prediction directions for intra  $4 \times 4$  prediction modes.

Motion estimation is done with  $1/4$ -pel accuracy by means of interpolation with a 6-tap filter. The determined motion vectors are median-predicted from neighboring blocks, i.e. a vector offset is computed, and the vector difference is coded. Up to 15 different reference frames can be employed for prediction, depending on the chosen level.

The prediction error is – independently of the block mode – transformed by means of a low-complexity  $4 \times 4$  integer transform which, together with an appropriate scaling in the quantization stage, approximates the  $4 \times 4$  DCT. The transform is applied in both horizontal and vertical direction. The transform matrix  $T_4$  is given by

$$T_4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix}. \quad (1)$$

The transformed coefficients in the encoder are then quantized by means of one quantizer out of the set of 52 uniform scalar quantizers, scanned in a zig-zag manner and finally entropy-encoded by context-adaptive variable-length coding (CAVLC). All other

data, e.g. header information, is encoded with regard to a single VLC table, an exponential Golomb code with parameter zero, i.e. without suffix. The Baseline profile supports not only progressively scanned pictures but, level-dependent, can also handle interlaced frames.

As mentioned before, MBs in H.264 are grouped into slices which in turn make up a single picture in a tile-like manner. Slices may arrive at the decoder in an arbitrary order; this is abbreviated to ASO. Further, the data of one slice may be repeated in the bit stream, thus generating redundant slice (RS) descriptions that can be utilized in case a slice with identical data has been error-affected previously. There are no spatial-prediction dependencies between slices to stop error propagation in the picture plane under erroneous transmission, hereby increasing the bit stream's error resilience. The affiliation of MBs to a slice is either determined by simple raster scan order or by an allocation map as used by a technique called flexible macroblock ordering (FMO).

## 2.2. Main profile

The Main profile compounds of all Baseline profile algorithms plus additional features, but except the functionalities FMO, ASO, and RS. The decoder has also to expect an arbitrary number of B slices placed between I and P slices, which use temporally succeeding or preceding pictures of all types for prediction, even including other B slices. Two predictions may be weighted for prediction.

In the Main profile, the concept of adaptive block-size transforms (ABT) is employed. The basic idea of ABT for inter coding is to connect the block size used for transform coding of the prediction error to the block size used for motion compensation [4]. The maximum feasible signal length is exploited for transform coding. For complexity reasons, the maximum transform block size is restricted to  $8 \times 8$ .

In general, the separable 2-D transform of a 2-D signal of size  $n \times m$  pixel can be written as

$$C_{n,m} = T_v \cdot B_{n,m} \cdot T_h^T, \quad (2)$$

where  $B_{n,m}$  denotes the signal block of  $n \times m$  pixels,  $C_{n,m}$  are the transform coefficients, and  $T_v$  and  $T_h$  are the  $m \times m$  and  $n \times n$  transform matrices in vertical and horizontal direction, respectively. Hence, only an additional  $8 \times 8$  transform matrix has to be defined for the application of ABT [5].

Since the macroblock mode is already transmitted for motion compensation, no additional side information needs to be coded for inter ABT. For intra coding, blocks of size  $4 \times 4$ ,  $4 \times 8$ ,  $8 \times 4$ , and  $8 \times 8$  are used for prediction and transform. Prediction is done with re-

spect to the directions given for the fundamental  $4 \times 4$  modes.

For higher coding efficiency, VLC is replaced by context-based adaptive binary arithmetic coding (CABAC). Here, non-binary values are binarized, and the resulting bins are coded with a binary arithmetic encoder. The general building blocks of CABAC are depicted in Fig. 3.

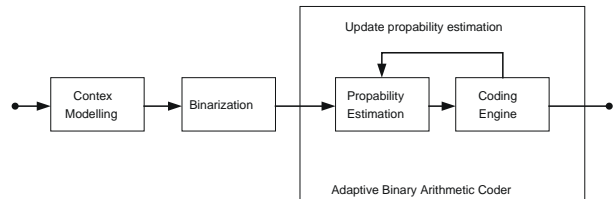


Figure 3: CABAC building blocks.

To demonstrate the performance gain introduced by ABT, rate-distortion curves for the CCIR test sequence FOOTBALL are shown in Fig. 4. The sequence was encoded with the frame structure [IBBPBBP...]. The rate-distortion-optimized Joint Model JM-3.9a software with CABAC was used to produce the results for both ABT and non-ABT.

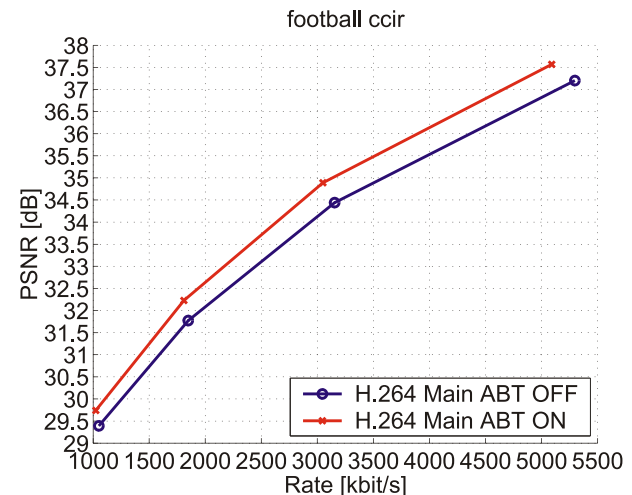


Figure 4: Comparison of ABT and non-ABT coding in Main profile on the test sequence FOOTBALL, CCIR 601 format.

All these mentioned features contribute to H.264's high adaptivity which in turn is quantified in Sec. 3 by performance comparisons with respect to other well known standards.

## 2.3. Encoding issues

The encoder will preferably select the prediction mode according to a Lagrange rate-distortion optimization

criterion, which in turn is based on the encoded rate and distortion for each block and each prediction mode. This mode selection achieves optimum rate-distortion performance but is highly complex. A simplified method is therefore to choose the one prediction mode for which the overall transformed prediction error, chosen for instance to be the summed absolute transformed difference (SATD), is minimized. This can e.g. be done by Lagrange functional minimization

$$J = SATD + \lambda \cdot R, \quad (3)$$

with a rate constraint  $R$  and the optimization parameter  $\lambda$ . SATD for a  $4 \times 4$  block is defined as

$$SATD = \frac{1}{2} \sum_{i,j=0}^3 |T_H\{D(i,j)\}|. \quad (4)$$

$T_H\{\cdot\}$  is the 2-D Hadamard transform, and the definition of the prediction error is

$$D(i,j) = X_{\text{org}}(i,j) - X_{\text{pred}}(i,j), \quad (5)$$

with the original and predicted samples  $X$ , at position pixel  $i$  in line  $j$ . The minimization is done for all intra- and inter-frame MB coding modes (the latter one implying all reference frames).

In general, the techniques for finding the prediction mode are encoder-specific and thus non-normative.

### 3. PERFORMANCE EVALUATION

The performance of the upcoming standard is exemplified by results for one of its former software implementations, TML-8, from September 2001 [6]. Several video sequences of different sizes are tested at various rates. All employed codecs are based on Lagrange rate-distortion optimization. The following are  $PSNR$  luminance values. Compared to MPEG-2 (Verification Model VM-5, SW-1.2) at equal bandwidth, H.264's TML-8 yields an average gain of 5.8 dB and a maximum gain of 7.2 dB for CIF picture size at a transmission rate of 1024 Kbps and a frame rate of 30 fps. In comparison to H.263++ High-Latency profile, the gain spans from 2.5 dB (max. 3.0 dB) for QCIF at 32 Kbps and 10 fps, to 3.8 dB (max. 5.2 dB) for CIF at 1024 Kbps and 30 fps. Related to the optimized MPEG-4 Advanced Simple profile, there is a gain range of 1.9 dB (max. 2.8 dB) for QCIF at 32 Kbps and 10 fps to 2.5 dB (max. 3.6 dB) for CIF at 1024 Kbps and 30 fps. Considering subjective evaluations, TML-8.0 operates on approximately half the bandwidth of MPEG-4 for the same visual fidelity.

A detailed comparison of the intra coding scheme to still image codecs is given in the following.

#### 3.1. Comparison H.264 intra coding to JPEG-2000 and JPEG

In order to evaluate the performance of H.264's intra-frame coding scheme, some tests are carried out under the following conditions. We let H.264's Joint Model JM-3.9a Main profile without ABT compete with the default mode of VM-9.0 of JPEG2000 [7]. Since H.264 lacks a rate distortion control, the quantization parameters (QPs) of JM, which control the choice of employed quantizer, have been chosen beforehand, and then VM-9.0 is adjusted to the achieved rate of JM-3.9a, picture per picture, in several iterations with a tolerance of 0.001 bpp. Tab. 1 shows the results in terms of luminance  $PSNR$  [dB] and bit rate [bpp]. The achieved bit rate corresponds to the QPs 20, 28, 36, and 44. All values are averaged over 20 pictures of equispaced frame indices from the respective sequence. The last column contains the difference, i.e. JM minus VM  $PSNR$  values. Negative values here means that JPEG2000 outperforms H.264.

H.264 performs very well in intra mode. For almost all sequences at all bit rates, it always outperforms JPEG2000. This may be explained by the various excellent intra-frame prediction modes of H.264. JPEG2000, on the other hand, consists of a wavelet decomposition without prediction, followed by scalar quantization and arithmetic encoding. It is observed that the gain shrinks somewhat for larger image sizes. Moreover, the highest gains are achieved especially at lower rates (higher QPs). The average gain over all sequences and all bit rates is 1.12 dB, i.e. H.264 outperforms JPEG2000 significantly on the video test material.

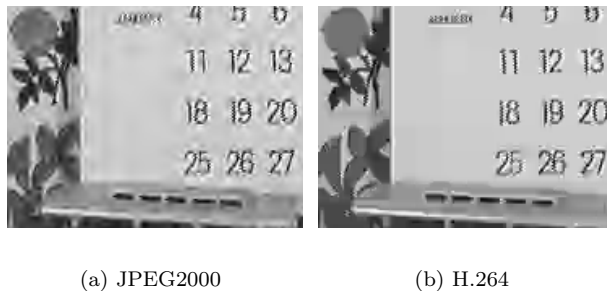


Figure 5: Detail from sequence Mobile at luminance rate of 0.253 bpp, picture 121.

The visual evaluation reveals that, for all rates, JPEG2000 is slightly better in preserving details than its competitor. H.264 further shows some for a block-based compression scheme typical blocking artifacts, even though not very distinct. At higher rates (above approx. 1.5 bpp), the subjective comparison shows no perceivable differences. However, considering a

Sequence (Size)	Av.Q	QP	Bit rate	PSNR			Difference JM to	
	(JPEG)	(H.264)	(av.)	JPEG	JPEG2000	H.264	JPEG	JPEG2000
Container (QCIF)	82	20	1.715	37.26	43.28	43.28	6.02	0.87
	43	28	0.892	31.64	35.63	37.24	5.60	1.61
	7	36	0.405	25.26	29.28	31.55	6.29	2.27
	0	44	0.153	18.42	23.68	26.05	7.63	2.37
News (QCIF)	80	20	1.667	37.22	42.69	43.77	6.55	1.08
	40	28	0.942	31.78	35.55	37.63	5.85	2.08
	9	36	0.462	26.07	29.23	31.27	5.20	2.04
	0	44	0.192	16.78	24.00	25.90	9.12	1.90
Tempete (CIF)	86	20	2.334	37.90	43.27	42.91	5.01	-0.36
	61	28	1.316	32.58	36.01	36.05	3.47	0.04
	16	36	0.580	27.38	29.43	29.55	2.17	0.12
	3	44	0.189	21.56	24.44	24.14	2.58	-0.30
Mobile (CCIR 601)	86	20	2.632	36.16	42.05	42.68	6.52	0.63
	63	28	1.530	30.21	34.94	36.02	5.81	1.08
	21	36	0.724	25.66	28.19	29.53	3.87	1.34
	4	44	0.253	21.01	23.03	23.96	2.95	0.93
Total							5.29	1.12

Table 1: H.264’s I picture coding performance. PSNR and difference values in dB.

very low bit rate (below 0.5 bpp), the JPEG2000 images look generally blurred; they contain further much ringing artifacts. This is due to the wavelet transform employed. With H.264, the subjective image representation is much better at these rates, as shown in Fig. 5.

In addition to JPEG2000, we also record the coding efficiency of the JPEG standard in default mode, as Tab. 1 shows. We use the software implementation version 6b of the Independent JPEG Group. The *PSNR* values given must be interpreted with caution since it is not possible to put a rate constraint on the software directly. Instead, the quality parameter of the JPEG algorithm is adjusted picture-wise such that the resulting rate is closest possible to the target bit rate. JPEG consists of an  $8 \times 8$  DCT with subsequent coefficient scan, quantization, and entropy encoding, but lacks an intra prediction mechanism as H.264 [8]. Mainly because of this difference, we can see that H.264 outperforms JPEG significantly for all image size at all rates. The average gain over all sequences and all bit rates is 5.3 dB. Given these large gains, it is clear that also the subjective evaluation is in favor of H.264.

A useful side result is the *PSNR* gain of JPEG-2000 over the old JPEG standard; it is on the average 4.2 dB, spanning a range from approximately 2 dB to over 7 dB.

It should be mentioned that JPEG2000 has been developed for real still images and a broad range of applications, demanding so various features like e.g.

SNR/spatial scalability and region-of-interest coding, abilities that are not standardized in H.264.

### 3.2. Comparison for chrominance

As chrominance coding is not implemented for the JPEG2000 software, we also record the performance of the verification model of the currently evolving Motion JPEG2000 (MJ2) project, and compare its results in *PSNR* to those of H.264 Main profile without ABT in intra-coding mode. An MJ2 bit stream is basically a sequence of coded still images as defined in JPEG2000 [9]. In addition, the same YUV color space and chroma subsampling is supported by MJ2 as in H.264. MJ2’s software version VM-2.0 is used. The software model of H.264 is identical with the one in Sec. 3.1, i.e. JM-3.9a. The tests are carried out with the sequences, QPs, frame skipping, etc. as specified before. The resulting chrominance *PSNR* values in dB and their differences are listed in Tab. 2. The values have been averaged beforehand over the respective sequence and both U and V components. It is stressed that the bit rate, which is put picture-wise as an constraint on MJ2, is the rate over all color components and hence somewhat higher than the rate shown in Tab. 1.

We see that Motion JPEG2000 achieves better chroma *PSNR* values than H.264 with nearly all test conditions. The average gain is 1.42 dB. The explanation of this fact is that H.264 processes chroma samples with less accuracy than luminance values, e.g. by means of a simple  $2 \times 2$  integer transform. The up-

Seq	QP	PSNR		Difference
	(H.264)	MJ2	H.264	H.264 – MJ2
Cont.	20	48.24	46.15	-2.09
	28	41.53	40.52	-1.01
	36	37.32	36.63	-0.69
	44	33.87	34.81	0.94
News	20	47.04	45.58	-1.46
	28	41.32	39.76	-1.56
	36	36.85	36.13	-0.72
	44	32.99	33.86	0.87
Temp.	20	47.12	43.53	-3.59
	28	41.10	38.41	-2.69
	36	37.23	35.25	-1.98
	44	33.73	33.10	-0.63
Mob.	20	46.34	43.27	-3.07
	28	39.94	37.51	-2.43
	36	35.47	33.73	-1.74
	44	32.11	31.17	-0.94
Total				-1.42

Table 2: H.264’s chrominance coding performance. PSNR and difference values in dB.

coming still-image compression standard on the other hand uses in its default mode visual weighting of color components and processes the chroma samples with relatively high accuracy, both of which lead obviously to superior results. However, the difference in chrominance coding seems to be large but is visually imperceptible.

#### 4. CONCLUSIONS AND OUTLOOK

Many have postulated that conventional, i.e. block-based, video compression ‘is dead’. H.264’s technical concepts, however, prove once more to be successful, complemented by several new efficient compression methods as explained in this paper, which together constitute the final codec. Hence, an upper limit in video compression seems not to be reached yet. The standard shows further that great adaptivity to the source, the video sequence, is the key to high-performance coding. This adaptivity is accomplished by enlarged complexity as compared to older standards, and in addition by larger memory requirements as for the picture reference buffer.

H.264 targets not only video conferences but the whole spectrum from television broadcasting, Internet streaming, and digital storage media to databases, telemedicine systems, and digital cinema, etc. Real-time solutions are already in position for the market.

#### 5. REFERENCES

- [1] Thomas Wiegand, “Joint Final Committee Draft (JFCD) of joint video specification ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC,” Tech. Rep. D157, ITU-T VCEG | ISO/IEC MPEG (JVT), Aug. 2002.
- [2] ISO/IEC IS 14496-2: 2001, *Information Technology – Coding of Audio-Visual Objects – Part 2: Visual*, (MPEG-4).
- [3] ITU-T Recommendation H.263, *Video Coding for Low Bitrate Communication*, 1995.
- [4] Mathias Wien and Achim Daxhof, “ABT coding for higher resolution video,” in *JVT, Meeting B, Document JVT-B053*, Geneva, Switzerland, Jan. 2002, ITU-T VCEG and ISO/IEC MPEG.
- [5] Mathias Wien and Jens-Rainer Ohm, “Simplified adaptive block transforms,” in *JVT, Meeting A, Document JVT-A003*, Pattaya, Thailand, Dec. 2001, ITU-T VCEG and ISO/IEC MPEG.
- [6] Pankaj Topiwala, Gary Sullivan, Anthony Joch, and Faouzi Kossentini, “Performance evaluation of H.26L, TML 8 vs. H.263++ and MPEG-4,” Tech. Rep. N18, ITU-T Q.6/SG 16 (VCEG), Sept. 2001.
- [7] ISO/IEC IS 15444-1: 2000, *JPEG2000 Image Coding System, Part I*, (JPEG2000).
- [8] ISO/IEC IS 10918-1: 1994, *Digital Compression and Coding of Continuous-tone Still Images, Part I: Requirements and Guidelines*, (JPEG, also CCITT Recommendation T.81).
- [9] Takahiro Fukuhara and David Singer, “Motion JPEG2000 Verification Model Ver. 2.0 (Technical Description),” Tech. Rep. N1785, ISO/IEC JTC 1/SC 29/WG 1, July 2000.