Performance Comparison of JPEG2000 and H.264/AVC High Profile Intra–Frame Coding on HD Video Sequences

Pankaj Topiwala, Trac Tran, Wei Dai {pankaj, trac, daisy} @ fastvdo.com
FastVDO, LLC, Columbia, MD 21046

ABSTRACT

This paper reconsiders the rate-distortion performance comparison of JPEG2000 with H.264/AVC High Profile I–frame coding for high definition (HD) video sequences. This work is a follow-on to our paper at SPIE 05 [14], wherein we further optimize both codecs. This also extends a similar earlier study involving H.264/AVC Main Profile [2]. Coding simulations are performed on a set of 720p and 1080p HD video sequences, which have been commonly used for H.264/AVC standardization work. As expected, our experimental results show that H.264/AVC I–frame coding offers consistent R-D performance gains (around 0.2 to 1 dB in peak signal-to-noise ratio) over JPEG2000 color image coding. As in [1, 2], we do not consider scalability, complexity in this study (JPEG2000 is used in non-scalable, but optimal mode).

Keywords: H.264, AVC, High Profile, JPEG2000, image coding, video coding.

1. INTRODUCTION

H.264, or MPEG-4 Part 10, is an international video coding standard developed by the ITU-T Video Coding Experts Group (VCEG) together with the ISO/IEC Moving Picture Experts Group (MPEG) as the product of a collective partnership effort known as the Joint Video Team (JVT) [3, 4]. H.264 technology, also known as Advanced Video Coding (AVC), is designed to provide good video quality at substantially lower bit rates comparing to previous standards. It is also intended to have a reasonable level of computational complexity and to be flexible enough for a wide range of applications, from broadcasting, DVD storage, teleconferencing, to wireless multimedia communications. On the other hand, JPEG2000 is the wavelet-based still image compression standard [5, 6] whose aim is not only to improve coding performance over the original DCT-based JPEG standard [7] but also to add or improve features such as scalability, editability, and lossless coding capability. Although H.264 and JPEG2000 are developed for different signals, there are several application areas where they overlap: video applications requiring fast, frequent, and convenient frame access for editing purposes, for instance, Digital Cinema; high-quality high-resolution medical and satellite imaging; video applications requiring real-time simple encoding, etc.

There have been several performance comparisons evaluating JPEG2000 and H.264/AVC I–frame coding [1, 2]. The rate-distortion performance of Motion-JPEG2000 and H.264/AVC Main Profile (MP) intra coding was first reported by Marpe et al. in [2]. Using a set of test video sequences with different resolutions, [2] showed that H.264/AVC intra coding has around 0.2 ~ 2 dB PSNR gains over JPEG2000 for the low and middle resolution sequences, e.g., CIF and ITU-R 601 720x576i (25Hz) sequences. However, in their opinion, JPEG2000 has an advantage over H.264/AVC for sequences with very high-resolution content, e.g., 1080p sequences, while for 720p HD sequences, it is reported that both perform virtually at the same level [1, 2]. By contrast, our paper [14] last year at SPIE showed that for 720p, AVC High Profile (HP) holds a clear advantage over JPEG2000, while for 1080p, the results are more even, or a slight advantage for AVC. Recently, it has also been shown that H.264/AVC Fidelity Range Extensions (FRExt) amendment [8] HP provides a major breakthrough in compression efficiency over MP generally. Consistent R-D gains in favor of H.264/AVC FRExt intra coding over JPEG2000 for a set of monochrome ISO/IEC images are also reported in [1].

In this paper, we reinvestigate the intra-frame coding performance of H.264/AVC HP in comparison with JPEG2000 (Part 1) [5] for six 720p (1280x720-pixel resolution) HD sequences and four 1080p (1920x1080-pixel resolution) HD sequences. The chosen test sequences cover three different classes of still
frames based on their level of spatial contents, ranging from smooth to moderate to high. We confirm that H.264 HP intra coding consistently outperforms JPEG2000 in the rate-distortion sense on almost all of our test sequences, especially at high bit-rate high-quality setting. A key difference in this paper over the paper [14] a year ago is that we have further optimized both H.264/AVC HP and JPEG2000. In particular, we have removed the visual weighting in JPEG2000 (a default parameter setting), while using some more advanced settings in the AVC HP coding [15-18]. In fact, H.264/AVC HP has so many encoder parameter settings, even for Intra coding, that optimizing it parametrically is itself a fine art. The important test conditions and coding tools used are described in more detail below in the experimental results section.

The organization of the paper is as follows. To make this paper self-consistent, we provide a quick overview of the two codecs first in Section 2. Our evaluating methodology – including various H.264 and JPEG2000 settings – and our high-definition test video sequences are discussed in Section 3, followed by our experimental results and discussions of RD performance in Section 4. Finally, Section 5 concludes the paper with a few remarks.

2. DESCRIPTION OF EVALUATING COMPRESSION ALGORITHMS

We provide in this section a brief description of three image compression algorithms under common performance evaluation: JPEG2000, H.264/AVC Main Profile intra-frame coding, and H.264/AVC High Profile FRExt intra-frame coding. All three compression schemes are based on the classic three-stage transform-coding paradigm, consisting of a signal decomposition stage, followed by uniform scalar quantization, and context-adaptive entropy coding. Our tests dispense with the Main Profile (MP) in favor of HP, since HP is a superset of MP, and in fact superior. We discuss MP mainly to illustrate HP better.

2.1. JPEG2000

Unlike its predecessor JPEG [7], which is based on the 8x8 block DCT decomposition, JPEG2000 relies on the wavelet transform as its main de-correlation engine. This multi-resolution transform with length-varying basis functions decomposes an input image into wavelet coefficients grouped by sub-bands, representing different spatial-frequency components. The set of resulting wavelet coefficients are further split into small coding units called code-blocks, which are independently processed by a coding scheme called Embedded Bitplane Coding with Optimal Truncation (EBCOT) followed by adaptive context-based binary arithmetic coding.

JPEG2000 has a few distinctive features that we did not enable in this evaluation. Most notable is the scalability feature, allowing one to extract different regions, components, images of different fidelities and/or spatial resolutions out of one single compressed bit-stream. The drawback of scalability is its adverse effect on rate-distortion performance. To max its performance, our comparisons are conducted with the non-scalable single-layer mode. Another feature that we also elect to disable is the tiling mode, which partitions the input image into non-overlapped rectangular tiles to be encoded independently. The tiling feature, intended for lower-complexity and parallel processing, also most likely lowers R-D performance.

2.2. H.264/AVC Main Profile Intra-Frame Coding

Both based on the transform-coding paradigm, the main difference between H.264/AVC Main Profile intra-coding and JPEG2000 is at the transformation stage. Other differences in the quantization and entropy coding stage are dictated by the characteristics of the produced transform coefficients. While JPEG2000 employs the global wavelet transform (tiling is its only option for image partitioning), H.264 follows the block coding philosophy, which is more in line of the block-translational motion model employed in its inter-frame coding framework.

Unlike all of its video coding standard predecessors, H.264’s transform block size is reduced from 8x8 to 4x4. As a pre-processing step, H.264 relies on spatial prediction using neighboring pixels from previously encoded blocks to take advantage of inter-block spatial correlation. The residual prediction error is de-correlated by a 4x4 low-complexity multiplier-less integer transform that approximates the original 4x4 DCT well but can be implemented in 16-bit fixed-point architectures. The DC coefficients of neighboring blocks are collected into 4x4 blocks and then further processed using the same 4x4 integer transform (2x2 blocks and 2x2 Hadamard transform are used in the chrominance space). The combination of spatial prediction and the wavelet-like 2-level transform iteration has proven to be very effective in
smooth image regions – one reason why H.264 can stay competitive with JPEG2000 in high-resolution high-quality applications whereas the block-coding based JPEG is not. This H.264 R-D performance result is rather consistent with a few recent reports that the block DCT coding framework can be very competitive with the global wavelet coding framework if inter-block correlation is properly taken into account coupled with appropriately designed context-adaptive entropy coding [9, 10, 11].

After transformation, the H.264 transform coefficients are scalar quantized, zig-zag scanned, and entropy coded by Context-based Adaptive Binary Arithmetic Coding (CABAC). Another entropy coding choice that provides a faster simpler implementation but sacrifices some coding efficiency is called Context-Adaptive Variable-Length Coding (CAVLC), switching from different VLC tables designed from exponential-Golomb codes based on locally available contexts collected from neighboring blocks.

2.3. H.264/AVC FRExt High Profile Intra-Frame Coding
The JVT completed the development of some extensions to the original H.264 standard in July, 2003. The resulting codec is known as H.264 Fidelity Range Extensions (FRExt), also known as the High Profile [8]. Amongst the extensions as expected from the naming are support for higher-fidelity video pixel resolution (including 10-bit and 12-bit video samples) and support for higher-resolution color spaces such as YUV 4:2:2 and YUV 4:4:4. The main FRExt feature that improves coding efficiency (our top criterion in this paper) is the addition of the 8x8 integer transform – another DCT approximation – and all coding modes as well as prediction schemes associated with the adaptive selection between the 4×4 and 8×8 integer transforms. The addition of the larger block size of 8x8 is critical in high-resolution high bit-rate applications as shown in later sections.

3. EVALUATING METHODOLOGY

3.1. Video Test Sequences
In our performance evaluation, we select six progressive-scan high-definition video sequences (60Hz) at 720p resolution and four at 1080p resolution. All of the test sequences are in the YUV 4:2:0 color format where two chrominance components (U, V) are down-sampled by a factor of two in each spatial dimension. These six 720p sequences can be grouped into three different categories according to their different spatial contents:

- Smooth spatial details: Jets (first 60 frames) and ShuttleStart (first 60 frames)
- Moderate spatial details: BigShip (first 60 frames) and Crew (first 60 frames)
- High spatial details: City (first 60 frames) and Harbour (first 120 frames).

Since we are conducting I-frame coding, 60 frames is quite adequate to establish the trend. The only exception is Harbour where we consider the first 120 frames. Both 1080p video sequences Hollywood and Kungfu can be classified as having moderate spatial details: a typical landscape panning sequence of the famous Hollywood hill and a fast-action Kungfu sequence with relatively smooth, stable background. 1080p video sequences Duck and Crowd can be classified as having high spatial details. All sequences are available on our web site at www.fastvdo.com.

3.2. Codec Settings
In our coding experiments, we use publicly available software implementations of H.264/AVC and JPEG2000. The latest release of the reference software (JM 11.0) is used for H.264/AVC encoder, and each frame of the test sequences is coded in the I–frame mode. For JPEG2000 coding, D. Taubman’s "Kakadu" (version 2.2) software [6] is used to code each frame to reach the target bit rates. Note that we turned off the visual weighting.

The configuration of the H.264/AVC JM encoder [4] is chosen as follows:

- 8x8 transform mode: enabled, allowing adaptive choice between 4x4/8x8 transform and all associated prediction modes
- CABAC: enabled
- R-D optimization: enabled
- De-blocking filter: enabled
- AdaptiveRounding: 1, AdaptRndPeriod: 1, AdaptRndChroma: 1, AdaptRndWFactorIRref: 8, AdaptRndWFactorINref: 8, OffsetMatrixPresentFlag: 0
The "Kakadu" JPEG2000 encoder [6] is driven in default mode (except visual weighting is turned off):
- One tile per frame (no tiling); some limited results with 4 tiles for comparison.
- 9/7-tap biorthogonal Daubechies wavelet filters (default floating-point transform)
- 5 levels of wavelet decomposition
- Single-layer mode (no scalability option)
- Code-block size of 64x64 wavelet coefficients
- EBCOT encoding scheme
- R-D optimization for a given target bit rate.
- No_weights on (in particular, visual weighting is off)

3.3. Evaluating Criteria
To compare the objective performance, we illustrate the curves of average PSNR values of the luminance and chrominance components over all encoded frames versus the final bit rate. For each experiment, H.264/AVC FRExt codes each frame in I-frame mode with one fixed quantization step size. Also, the quantization values for luminance and chrominance components are the same (which is the default mode of HP profile). For JPEG2000, we code each frame with the target bit rate derived from the set total bit rates, frame rate, and sequence resolution. Since this experiment concerns high-bit-rate scenarios, we choose the test psnr points to cover the range between 30 ~ 55dB.

4. EXPERIMENTAL RESULTS
Fig. 1 to Fig. 4 depicts the rate-distortion curves as the outcome of our coding experiments for each test sequence. Overall, H.264/AVC FRExt has an average 0.5dB PSNR gain over JPEG2000 in the Luminance component and it also works well with the Chrominance components for all three types of HD sequences. The typical result comes from the ShuttleStart sequence, where H264 FRExt has a consistent 0.5+ dB gain over JPEG2000 as shown in Fig. 2. From Fig. 1 ~ Fig.2 and Fig. 4 ~ Fig.6, we can observe that H264/AVC FRExt has higher gains in all three components than JPEG2000 at high bit rates. The PSNR gains reduce while the bit rates decreases. At the high end, we generally get a clear differentiation of performance in favor of H.264/AVC HP, with different asymptotic behavior. That asymptotic difference we attribute to recent encoder optimizations, mainly to the adaptive quantization method of [15-18].

---

1 In the experiments, we find that enabling the scaling quantization matrix in FRExt (i.e., “ScalingMatrixPresentFlag=1” which is the default setting in FRExt configuration file) severely lowers the coding efficiency, especially at the high bit-rate setting.
Figure 1: R-D curves for three components of the smooth spatial content 720p Jets sequence (left) and the ShuttleStart sequence (right) comparing H.264/AVC FRExt intra coding and JPEG2000.
Figure 2: R-D curves for three components of the moderate spatial content 720p BigShip sequence and the City sequence comparing H.264/AVC High Profile FRExt intra coding and JPEG2000.
Figure 3: R-D curves for three components of the 1080p Hollywood sequence (left) and the Kungfu sequence (right) comparing H.264/AVC High Profile FRExt intra coding and JPEG2000.
Figure 4: R-D curves for three components of the 1080p Duck sequence (left) and the Crowd sequence (right) comparing H.264/AVC High Profile FRExt intra coding and JPEG2000.
Figure 5: R-D curves for three components of the 1080p Duck sequence (left) and the Crowd sequence (right) comparing H.264/AVC High Profile FRExt intra coding and JPEG2000.

5. CONCLUSION

This comparative study points out the objective RD-performance superiority of the latest H.264/AVC FRExt I-frame coding scheme of the High Profile over the international still image compression standard JPEG2000 in high-resolution high-bit-rate video coding applications where fast and convenient frame access is of highest priority. Along with benchmarks in [1, 2], our experiment again confirms that as far as R-D performance is concerned, H.264/AVC FRExt is currently the leader not only in video compression but also in still image compression as well. At the 720p resolution, H.264 consistently offers significant
improvement in peak signal-to-noise ratio, especially in the luminance component. In fact, we even notice an asymptotic difference in slopes between the two codecs (e.g., with the Jets sequence, figure 1). The asymptotic difference, we believe, is mainly due to the use of adaptive quantization offset [15], in which the dequantization is more accurately near the centroid of a quantization bin than in the default settings (wherein a constant rather than computed offset from the center of the bin is used), as motivated by RD optimization. This advance, included in the AVC standard, is not currently available in JPEG2000.

In fact, the concept of [15] may not be properly implemented in the chroma channels yet, and there may be room for further improvements. In any case, there is little bitrate allocated to chroma channels, and differences in chroma psnr values are hard to observe. And at the 1080p resolution, contrasting to popular conceptions and results from previous coding experiments using H.264 Main-Profile [1, 2], our study points out that H.264 FRExt High-Profile I-frame coding algorithm is at least very competitive with JPEG2000 in the rate distortion sense. Furthermore, we remark that modern codecs, especially when used in high quality, high bandwidth applications, are more memory constrained rather than processor constrained, especially memory bandwidth. In this context, since the wavelet transform is a global transform (whereas AVC uses block transforms), the memory bandwidth requirements of JPEG2000 far exceed those of AVC. When tiling is used in JPEG2000 to constrain memory bandwidth (e.g., for 128x128 tiles, 1080p would have 16x8=128 tiles), H.264 HP may in fact be superior, even at 1080p or higher. In this paper, we tested only up to 4 tiles, and noted no significant variation in performance. In future papers, we will conduct more realistic tiling tests.

In figures 6-9, we reprint our results from the previous SPIE 05 paper [14], for comparison. In fact, in that paper, the parametric settings for neither the AVC HP nor the JPEG2000 codec were fully optimized. AVC HP did not have adaptive quantization offset turned on, for example, which is responsible for the asymptotic difference at high bit rate between HP and JPEG2000. Also, we had one default setting for JPEG2000, which included visual-weighting. However, note that even then, visual weighting turns off automatically when the bitrate exceeds one bit per sample, the threshold for which is at about 55 Mb/s. Thus, in retrospect, we can now be assured that the results were valid, since there is little difference between the new results and the previous ones.

6. ACKNOWLEDGEMENTS

We thank Gary Sullivan for useful comments and suggestions regarding optimizing AVC encoding, and especially in the use of the adaptive quantization offset method of [15-18].
REFERENCES


Figure 6: R-D curves for three components of the smooth spatial content 720p Jets sequence (left) and the ShuttleStart sequence (right) comparing H.264/AVC FRext intra coding and JPEG2000. This figure is quoted from [14].
Figure 7: R-D curves for three components of the moderate spatial content 720p BigShip sequence and the Crew sequence comparing H.264/AVC High Profile FRExt intra coding and JPEG2000. This figure is quoted from [14].
Figure 8: R-D curves for three components of the high spatial content 720p City sequence and the Harbour sequence comparing H.264/AVC High Profile FRExt intra coding and JPEG2000. This figure is quoted from [14].
Figure 9: R-D curves for three components of the 1080p Hollywood sequence (left) and the Kungfu sequence (right) comparing H.264/AVC High Profile FRExt intra coding and JPEG2000. This figure is quoted from [14].