# Intra Coding Performance Comparison of HEVC, H.264/AVC, Motion-JPEG2000 and JPEGXR Encoders

Yi Lui, Wassim Hamidouche, Olivier Dforges

# Intra Coding Performance Comparison of HEVC, H.264/AVC, Motion-JPEG2000 and JPEGXR Encoders

Yi Lui, Wassim Hamidouche and Olivier Dforges
Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, France
E-mail: firstname.lastname@insa-rennes.fr

September 18, 2018

**Abstract**

In this document the intra coding performance comparison of HEVC, H.264/AVC, Motion-JPEG2000 and JPEG XR is investigated. The performance of these four codecs is evaluated for various videos coded at different bitrates in Intra coding configuration. The results prove the excellent coding efficiency of the HEVC. In addition, the performance of the JPEG2000 is improved at high resolution videos and is close to the ones of HEVC and H.264/AVC at low bit-rates.

## 1 Introduction

The Joint Collaborative Team on Video Coding (JCT-VC) developed the latest video coding standard, High Efficiency Video Coding (HEVC), and released the first version in January 2013 [1]. Similar to the previous video coding standard H.264/AVC [2], HEVC exploits both the temporal redundancy between frames and the spatial redundancy within a frame. While evaluations demonstrate that HEVC can achieve a bitrate reduction by approximately 50% on average compared with H.264/AVC for an equivalent subjective quality of the reproduced video [3]. The improved intra coding method, which employs more prediction modes for luminance and chrominance components than those used in H.264/AVC, contributes a lot to the superior coding efficiency of HEVC [4]. The intra coding focuses on the spatial redundancy within the frame and also can be applied to the still image coding which is dominated by the JPEG standard series. So it is meaningful to compare the spatial coding efficiency of different solutions from both image and video coding.

Marpe *et al.* compared the intra coding performance of H.264 with the one of Motion-JPEG2000 [5], which encodes each frame by the use of JPEG2000 still image coding standard [6]. The experiment calculates the average luminance (Y) PSNR over the average bitrate and concludes that H.264/AVC has a significantly advantage of the rate-distortion at low resolution source materials ($352 \times 288$p and $720 \times 576$p in YUV 4:2:0), while Motion-JPEG2000 clearly outperforms H.264/AVC at high resolutions ($1920 \times 1080$p) and high bit-rates. In [7], Simone *et al.* transformed ten high definition images from RGB to YUV 4:4:4 and used them to measure JPEG2000, H.264/AVC High 4:4:4 Intra Profile and HD Photo which is standardized in JPEGXR [8]. The experiment shows that JPEG2000 outperforms HD Photo for all the images and H.264/AVC performs well compared to HD Photo, while JPEG2000 is better than H.264/AVC for most high definition images. In contrast, another literatures offer different conclusions based on their experiments. Shi *et al.* test sequences with resolutions from $512 \times 256$ to $4096 \times 2048$. They concludes that H.264/AVC intra frame performs better than JPEG2000 at higher bitrates for different resolutions, while vice versa at lower bitrates [9]. In [10], Cai *et al.* compared HEVC with H.264/AVC and Motion-JPEG2000. According to their experiment results, Motion-JPEG2000 is more efficient than H.264/AVC and HEVC in low bitrate performance, while gradually surpassed by H.264/AVC and HEVC as the bit-rate increases. Nguyen *et al.* also presented the rate-distortion performances of HEVC Main Still Picture (MSP) profile, H.264/AVC and JPEG2000 [11]. They tested 36 still images and concluded the superior performance of HEVC for the average bit-rate reduction.

Although some comparison work has been presented as shown above, a divergence of the performances between the H.264/AVC and JPEG2000 is noticed. The experimental conditions, such as different image/video sequences, color spaces and PSNR calculation methods probably influence the comparison conclusion. In our experiment, the original video sequence is preferred rather than the individual images as done in [5] [10]. Each test sequence contains hundreds of frames (10 seconds duration) which provide sufficient examples. The format of sequence is YUV 4:2:0 and not converted to other color space. Each intra coding solution chooses the corresponding format mode during the test. The Evaluating criterion is the combination of weighted luma and chroma PSNR used in HM, which is the reference software of HEVC [12].

The following parts are organized as follows. a brief introduction of the related coding technologies is presented in Section 2. The test video sequences, reference softwares and related coding parameter configurations are provided in Section 3. The comparison results are shown and discussed in Section 4. The conclusion is provided in Section 5.

# 2 Overview of the Compression Schemes

This section gives an overview of the evaluated coding algorithms, HEVC, H.264/AVC, JPEG 2000 and JPEGXR. The two video coding standards (HEVC, H.264/AVC) constrained to intra-coding mode which only considers the spatial redundancy, while JPEG2000 and JPEGXR are still image coding standards. In the next sections, we give a brief introduction of these four standards.

## 2.1 H.264/AVC

H.264/AVC is based on the block coding. It divides each frame into non-overlapping square or rectangular blocks. Each block is put into intra or inter prediction. The prediction residuals are then transformed by the block-based approach and quantized.

In the intra-coding mode, 9 prediction modes are used for the $4 \times 4$ block, while 4 modes for the $16 \times 16$ block. For the prediction residual signal, an adaptive selection between the $4 \times 4$ and $8 \times 8$ transform block sizes for the integer transform operation is applied to improve the coding efficiency. For the entropy coding part, H.264/AVC has two kinds of methods, CABAC and Context-adaptive variable-length coding (CAVLC). CABAC compresses the quantized transform coefficient values more efficiently than CAVLC, but also requires more processing operations. In our experiments, CABAC is adopted for H.264/AVC to test the coding efficiency.

## 2.2 HEVC

In HEVC, each frame is partitioned into coding tree blocks (CTB). The blocks of CTB can be further partitioned into coding blocks (CB). The size of the CB varies from the same size of CTB to the minimum size $8\times8$. The size is specified by a syntax which should be coded. The CB and the associated syntax form a coding unit (CU).

To each CU, an intra or inter mode is applied. For the intra mode, it considers the associate pixels adjacent to the current position and chooses one of 35 intra prediction mode [4]. Specifically, there are 33 angular prediction modes, one DC mode, and one planar mode.

The angular prediction is designed to model different directional structures in the video or image and performs well in the presence of edges. For the smooth content in the image, DC prediction generates a prediction block by averaging values in the neighboring reconstructed blocks. It provides an alternative but a coarse approximation. In addition, the planar prediction mode is used to describe the continuities along block boundaries. In the planer mode, the border samples are linearly interpolated and the center samples are linearly interpolated from the border ones. After the prediction, kinds of filters are applied to remedy the discontinuities along block boundaries. The complete description of the filtering process is referred to [13]. The prediction residuals are transformed by the Discrete cosine transform (DCT) in each transform block (TB) which is partitioned individually from the CU. For the inter coding mode, CB corresponds to one to four prediction blocks (PB). PB and the associate syntax form a prediction unit (PU) which has one or two motion vectors. The inter prediction is done by the use of the advanced motion vector prediction (AMVP): A candidate list is built for each motion vector to record the motion vectors of neighboring blocks with the same index as the predicted motion vector.

Finally, all syntax data are entropy coded by CABAC, which is similar to the entropy coding scheme used in H.264/AVC [14]. A sample adaptive offset (SAO) operation is also adopted in HEVC [15]. SAO classifies reconstructed pixels into different categories and then reduce the distortion by adding an offset for each category of pixels. According to a report on HEVC compression performance verification testing in 2014, the average bit-rate reduction for HEVC are estimated at approximately 52% for 480p, 56% for 720p, 62% for 1080p and 64% for 4K UHD, compared with H.264/AVC [16].

## 2.3 JPEG2000

JPEG2000 is designed for the still image coding. It partitions the image into rectangular regions called tiles. The *tile* can be any size, but once the size is chosen, all the *tiles* have the same size during the coding. Each *tile* is compressed individually in four main steps.

The first step is to apply the one-dimensional wavelet transform to the *tile* in the horizontal and vertical directions to get subbands of wavelet coefficients. The achieved coefficients are located in four blocks and describe the horizontal and vertical characteristics of the original *tile* component: one is the low resolution representation (LL), one responds strongly to vertical edges and line segments (HL), one responds to horizontal edges and line segments (LH), and one responds primarity to diagonally oriented features (HH). The low resolution one LL can be transformed further to get the more low resolution image blocks. This process is then repeated several times until a certain low resolution is achieved. This procedure can be seen as a decomposition of the image block. There are two kinds of transforms, integer and floating point. The integer one is reversible and designed for lossless coding.

After transformation, the next step is to quantize the transformed coefficients. Each coefficient is divided by a quantization step size $Q$ and rounded down. This operation is lossy. The larger $Q$ is, the coarser the coefficients are quantized, and the lower bitrate can be achieved. For the lossless coding, $Q$ is essentially set to 1.0 and the coefficients are integers produced by the reversible transform.

The third step is using an arithmetic coding which is called MQ coder to encode the quantized wavelet coefficients. The coder encodes the bits of coefficients, starting with the most significant bits and progressing to less significant ones by a process called EBCOT (Embedded Block Coding with optimal Truncation) [17].

The last step is the construction of the bitstream. The bitstream is composed of packets and many "markers". The marker is a sign of certain code parts. By the use of markers, the deocoder can skip some parts of the bitstream to decode a certain code part, and display certain regions of the image before others. The bitstream is also organized by layers. Each layer contains a certain resolution information and the decoder can achieve the image progressively by releasing the layers one by one.

Motion-JPEG2000 is defined as Part 3 of the ISO Standard for JPEG2000. It does not involve inter-frame coding, but each frame is coded independently using JPEG2000 [18].

## 2.4 JPEGXR

JPEGXR is based on the technology developed by Microsoft under the name HD Photo [19]. It uses an integer transform adopting a lifting scheme [20]. This transform is close to a $4 \times 4$ DCT but is lossless. JPEGXR allows an optional overlap step. This step operates on $4 \times 4$ blocks which are offset by 2 samples in each direction from the $4 \times 4$ core transform blocks. Its purpose is to improve compression capability and reduce block-boundary artifacts at low bitrates. At high bitrates, when the block-boundary artifacts are not obvious, this operlap step is skiped for reducing the encoding and decoding time. At the final stage, JPEGXR employs the adaptive Huffman codes for entropy coding.

Compared to JPEG2000, JPEGXR offers a low computational complexity coding solution. It is popularized mainly by Microsoft and supported in Adobe Flash Player 11, Windows Imaging Component and Windows operating systems.

Table 1: Overview of the Test Sequences

| Sequence Name | Resolution | Frame rate (fps) | Frame Amount |
|---|---|---|---|
| BlowingBubbles | 416×240 | 50 | 500 |
| BasketballPass | | 50 | 500 |
| BQMall | 832×480 | 60 | 600 |
| PartyScene | | 50 | 500 |
| BasketballDrill | | 50 | 500 |
| ChinaSpeed | 1024×768 | 30 | 500 |
| FourPeople | 1280×720 | 60 | 600 |
| Johnny | | 60 | 600 |
| KristenAndSara | | 60 | 600 |
| SlideEditing | | 30 | 300 |
| SlideShow | | 20 | 500 |
| Kimono1 | 1920×1080 | 24 | 240 |
| ParkScene | | 24 | 240 |
| Cactus | | 50 | 500 |
| BQTerrace | | 60 | 600 |
| BasketballDrive | | 50 | 500 |
| Traffic | 2560×1600 | 30 | 150 |
| PeopleOnStreet | | 30 | 150 |
| Beauty | 3840×2160 | 30 | 120 |
| Bosphorus | | 30 | 120 |
| HoneyBee | | 30 | 120 |

# 3 Evaluation Methodology

The test sequences are mainly from those used in the standardization process of HEVC and shown in Table 1. The test sequences in 4K are available in [21]. All the sequences are in the format YUV 4:2:0 and have the 8-bit depth. These sequences cover a wide range of characteristics of the image contents and have hundreds of frames to describe the movements of the traffic, the people and the screen context. Besides, the sequences provide examples in different resolutions with sizes from $416 \times 240$ to $3840 \times 2160$ in order to test the spatial compression efficiencies of coding solutions for low and high resolution images. It is expected to provide convincing test results.

## 3.1 Encoder configuration

In general, the default configurations of the reference softwares of the test compression schemes are used in this comparison experiment.

For HEVC, the released version of HM 14 [12] is tested as the reference solution. The intra coding mode is controlled by the configuration file "encoder_intra_main.cfg". The details of options are shown in Table 2. The quantization parameter QP is changed during the test to get the qualities of coded sequences at different bit-rates.

H.264/AVC is tested by the use of the reference solution JM 18.6 which is available in [22]. JM is configured to closely emulate HEVC coding based on the HM-like configurations as used in [23]. The configuration file is "encoder_JM_Intra_HE.cfg" which is included in the JM folder. The main parameters are listed in Table 3.

For JPEG2000, the Motion-JPEG2000 is used to encode video sequences. The implementation is based on the OpenJPEG v2.1 [24]. The options also adopt default values and the reversible discrete wavelet transform (DWT) is used. However, the reversible DWT is the choice to implement the lossless coding by the JPEG2000. For the lossy coding, the irreversible DWT can achieve a better objective quality than the one the reversible mode as reported in [25] [26] [27]. In this test, the irreversible DWT is adopted to show the lossy coding performance of JPEG2000, while the results of the reversible DWT are also added for comparison. The main options of Motion-JPEG2000 are shown in Table 4.

The implementation of JPEGXR is available in [28]. The options are also mainly configured in default and some parameters are listed in Table 5.

Table 2: Options of HM

| Coding Options | Parameters |
|---|---|
| Profile | main |
| MaxCUWidth | 64 |
| MaxCUHeight | 64 |
| MaxPartitionDepth | 4 |
| QuadtreeTULog2MaxSize | 5 |
| QuadtreeTULog2MinSize | 2 |
| QuadtreeTUMaxDepthInter | 3 |
| QuadtreeTUMaxDepthIntra | 3 |
| IntraPeriod | 1 |
| FastSearch | 1 (TZ search) |
| SearchRange | 64 |
| HadamardME | 1 (Enabled) |
| QP | Quantization parameter (0∼51) |
| RDOQ | 1 |
| RDOQTS | 1 |
| LoopFilterDisable | 0 (0=Filter) |
| InternalBitDepth | 8 |
| SAO | 1 (Enabled) |
| AMP | 1 (Enabled) |
| TransformSkip | 1 (Enabled) |
| TransformSkipFast | 1 (Enabled) |

Table 3: Options of JM

| Coding Options | Parameters |
|---|---|
| ProfileIDC | 100 (FREXT Profiles) |
| IntraPeriod | 1 |
| IDRPeriod | 1 |
| QPISlice | Quant. param for I Slices (0∼51) |
| FastCrIntraDecision | 0 (off) |
| CbQPOffset | 0 |
| Transform8x8Mode | 1 |
| SymbolMode | 1 (CABAC) |
| RDOptimization | 1 (Enabled) |
| DFDisableRefISlice | 0 (0=Filter) |

## 3.2 Measurement Criteria

The Bjøntegaard-Delta bit-rate (BD-BR) measurement for the calculation of average bit-rate differences between rate-distortion curves for the same objective quality. The negative BD-BR values indicate actual bit-rate savings [29]. The objective quality used here is the $PSNR_{YUV}$ which is calculated from the $PSNR$ values of components Y, U and V as did in [10].

$$PSNR_{YUV} = (6 * PSNR_Y + PSNR_U + PSNR_V)/8 \tag{1}$$

The $PSNR$ of each component is the average of the PSNR sequence of the frames.

Table 4: Options of MotionJPEG2000

| Coding Options | Parameters |
|---|---|
| Size of code-block | 64×64 |
| Progression order | LRCP |
| DWT mode | Reversible DWT 5-3<br>Irreversible DWT 9-7 |
| Frame size and sub-sampling | Width,height,2,2 (YUV420) |
| Compression ratios | 1 ∼ 300 |

Table 5: Options of JPEGXR

| Coding Options | Parameters |
|---|---|
| -l (the overlapped block filtering) | HP only |
| -d (quanti. for U/V derived from Y) | enabled |
| -q (quanti. values) | 0 ∼ 255 |
| -r (encoding with RAW image) | enabled |
| -M (raw image format) | 19 (YCC420) |

# 4 Experiment Results and Discussion

Fig. 1 to Fig. 3 show the rate-distortion performances of each coding standards on different test video sequences. The point in the curve represents the objective quality of the decoded frame sequence at a specific bit-rate. MJPEG2K$_{re}$ represents the results of the Motion JPEG2000 with reversible DWT, while MJPEG2K$_{irre}$ with the irreversible DWT.

According to the figures, the intra mode performance of HEVC outperforms AVC, Motion-JPEG2000 and JPEGXR in most cases. The quality of AVC is closed to the one of HEVC, but does not exceed nether at low or high bi-rate. This result has been tested and concluded in many researches [3] [4].

As reported in [25] and [27], MJPEG2K$_{irre}$ has higher PSNR performances than MJPEG2K$_{re}$ ones to most sequences except the 'SlideEditing' and 'SlideShow' which have lots of artificial graphics and much content in the same color. The reversible DWT shows more contributions for this characteristic. With the irreversible DWT, Motion-JPEG2000 has a better rate-distortion quality compared with JPEGXR as concluded in [7] where the test still images are high definition 4:4:4 color ones. While for some sequences in 1080p, such as 'ParkScene', 'Cactus' and 'BQTerrace', JPEGXR provides comparable results to MJPEG2K$_{irre}$ at high bit-rates. Compare the rate-distortion curves for different resolution sequences, MJPEG2K$_{irre}$ shows a better performance for the high resolution frames than the low ones. For the sequences lower than 1080p, the points of MJPEG2K$_{irre}$ are obviously lower than the ones of AVC. With the increasing of the resolution, MJPEG2K$_{irre}$ presents close results to the ones of AVC for 1600p sequences and higher results for 2160p, especially at low bit-rates. The reason is probably related to the wavelet transform. The inter correlation between pixels are quite strong in the high resolution frame and the wavelet transform shows an excellent decorrelation property.

As a low computational solution, JPEGXR is not developed with a very high PSNR in lossy coding. Its performance are lower than AVC and HEVC at all bit-rate parts.

Table 6 provides the bit-rate saving of HEVC compared with AVC, Motion-JPEG2000 and JPEGXR. The $BD-BR$ values indicate the required overhead in bit-rate to achieve a closed $PSNR_{YUV}$ value. For each test sequence, the bit-rate data of the different encoders are selected out according to the same $PSNR$ range area. These data are used to calculate the bit-rate saving compared with HEVC. As shown in the average data, HEVC achieves the gain of 22.3859% compared with AVC, 45.3480% with Motion-JPEG2000 and 58.8925% with JPEGXR.

Table 6: HEVC Bit-Rate Saving (BD-BR in %) compared with AVC, Motion-JPEG2000 with irreversible DWT (M2K$_{ir}$) and JPEGXR(JXR) for a specific range of $PSNR_{YUV}$

| Sequence Name | PSNR Range (dB) | HEVC vs. AVC | HEVC vs. M2K$_{ir}$ | HEVC vs. JXR |
|---|---|---|---|---|
| BlowingBubbles | $30 \sim 48$ | 12.7740 | 33.6876 | 37.8617 |
| PartyScene | $30 \sim 48$ | 11.6621 | 32.4413 | 39.1666 |
| BasketballDrill | $33 \sim 45$ | 44.0732 | 62.8349 | 103.0553 |
| ChinaSpeed | $30 \sim 50$ | 34.6287 | 66.2816 | 99.3773 |
| FourPeople | $35 \sim 50$ | 27.1632 | 41.7835 | 64.6286 |
| Johnny | $35 \sim 50$ | 44.6954 | 60.0140 | 105.0201 |
| KristenandSara | $35 \sim 50$ | 36.2561 | 55.6464 | 87.4065 |
| SlideEditing | $25 \sim 50$ | 34.4160 | 83.2319 | 124.1203 |
| SlideShow | $40 \sim 60$ | 39.9924 | 78.4710 | 118.7211 |
| Kimono1 | $37 \sim 48$ | 31.5492 | 24.3941 | 72.6893 |
| ParkScene | $35 \sim 48$ | 16.0876 | 26.5534 | 37.0676 |
| Cactus | $35 \sim 48$ | 20.0561 | 33.8119 | 46.9229 |
| BQTerrace | $34 \sim 44$ | 23.5953 | 44.3458 | 42.3828 |
| BasketballDrive | $35 \sim 50$ | 22.9524 | 48.1836 | 85.9476 |
| Traffic | $35 \sim 50$ | 23.8857 | 32.4333 | 42.8683 |
| PeopleOnStreet | $35 \sim 50$ | 24.4652 | 36.6050 | 50.6165 |
| Beauty | $36 \sim 48$ | 20.0446 | 10.1971 | 95.5694 |
| Average | | 22.3859 | 45.3480 | 58.8925 |

# 5 Conclusion

An intra coding performance comparison of HEVC, H.264/AVC, Motion-JPEG2000 and JPEG XR is presented in this study. The results prove the excellent coding efficiency of HEVC. In addition, the performance of the JPEG2000 can be improved with the increasing of the resolution and close to the ones of HEVC and H.264/AVC at low bit-rates.

# Reference


[1] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1649–1668, Dec 2012.

[2] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 560–576, July 2003.

[3] J. Ohm, G. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards – including high efficiency video coding (HEVC)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1669–1684, Dec 2012.

[4] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra Coding of the HEVC Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1792–1801, Dec 2012.

[5] D. Marpe, V. George, H. L. Cycon, and K. U. Barthel, "Performance evaluation of Motion-JPEG2000 in comparison with H.264/AVC operated in pure intra coding mode," *Proc. SPIE*, vol. 5266, pp. 129–137, 2004.

[6] "ITU-T Recommendation T.800 and ISO/IEC 15444-1 JPEG2000 Image Coding System: Core Coding System (JPEG2000 Part 1)," 2000.

[7] F. De Simone, M. Ouaret, F. Dufaux, A. G. Tescher, and T. Ebrahimi, "A comparative study of JPEG2000, AVC/H.264, and HD photo," *Proc. SPIE*, vol. 6696, pp. 669602–669602–12, 2007.

[8] "ITU-T Recommendation T.832 : Information technology - JPEG XR image coding system - Image coding specification," 2012.

[9] B. Shi, L. Liu, and C. Xu, "Comparison between JPEG2000 and H.264 for digital cinema," in *IEEE International Conference on Multimedia and Expo*, pp. 725–728, June 2008.

[10] Q. Cai, L. Song, G. Li, and N. Ling, "Lossy and lossless intra coding performance evaluation: HEVC, H.264/AVC, JPEG 2000 and JPEG LS," in *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*, pp. 1–9, Dec 2012.

[11] T. Nguyen and D. Marpe, "Objective Performance Evaluation of the HEVC Main Still Picture Profile," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, pp. 790–797, May 2015.

[12] JCT-VC, "Subversion Repository for the HEVC Test Model Version," 2014.

[13] B. Bross, W.-J. Han, G. J. Sullivan, J.-R. Ohm, and T. Wiegand, "High Efficiency Video Coding (HEVC) text specification draft 7," Tech. Rep. document JCTVC-I1003, ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC), May 2012.

[14] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 620–636, July 2003.

[15] C.-M. Fu, E. Alshina, A. Alshin, Y.-W. Huang, C.-Y. Chen, C.-Y. Tsai, C.-W. Hsu, S.-M. Lei, J.-H. Park, and W.-J. Han, "Sample Adaptive Offset in the HEVC Standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, pp. 1755–1764, Dec 2012.

[16] T. TK, M. Marta, B. Vittorio, and R. Naeem, "Report on HEVC compression performance verification testing," *JCTVC-Q1011 Document, MPEG document: m33518 and N 14420*, May 2014.

[17] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Transactions on Image Processing*, vol. 9, pp. 1158–1170, Jul 2000.

[18] ITU-T, "T.802 : Information technology - JPEG 2000 image coding system: Motion JPEG 2000," Jan 2005.

[19] S. Srinivasan, C. Tu, Z. Zhou, D. Ray, S. Regunathan, and G. Sullivan, "An introduction to the HD photo technical design," Tech. Rep. JPEG document wg1n4183, Microsoft, Apr. 2007.

[20] C. Tu, S. Srinivasan, G. J. Sullivan, S. Regunathan, and H. S. Malvar, "Low-complexity hierarchical lapped transform for lossy-to-lossless image coding in JPEG XR / HD photo," 2008.

[21] Ultra video group, "4K video test sequences," 2014.

[22] F. H. H. Institute, " H.264/AVC Software Coordination ," 2015.

[23] A. Pierre, S. Philippe, and B. Philippe, "Comparison of Compression Performance of HEVC Draft 10 with AVC for UHD-1 material," *JCT-VC of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Document: JCTVC-M0166*, Apr. 2013.

[24] " OpenJPEG Library ," 2014.

[25] S. Peter, S. Athanassios, and E. Touradj, eds., *The JPEG2000 Suite*. John Wiley & Sons, Ltd, 2009.

[26] M. Iwahashi and H. Kiya, "Non separable 2D factorization of separable 2D DWT for lossless image coding," in *IEEE International Conference on Image Processing*, pp. 17–20, Nov 2009.

[27] L. Wang, J. Wu, L. Jiao, L. Zhang, and G. Shi, "Lossy to lossless image compression based on reversible integer DCT," in *IEEE International Conference on Image Processing*, pp. 1037–1040, Oct 2008.

[28] ITU, "JPEG XR image coding system reference software," 2012.

[29] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," *ITU-T Q.6/SG16 (VCEG) 13th Meeting, Document VCEG-M33*, Apr. 2001.

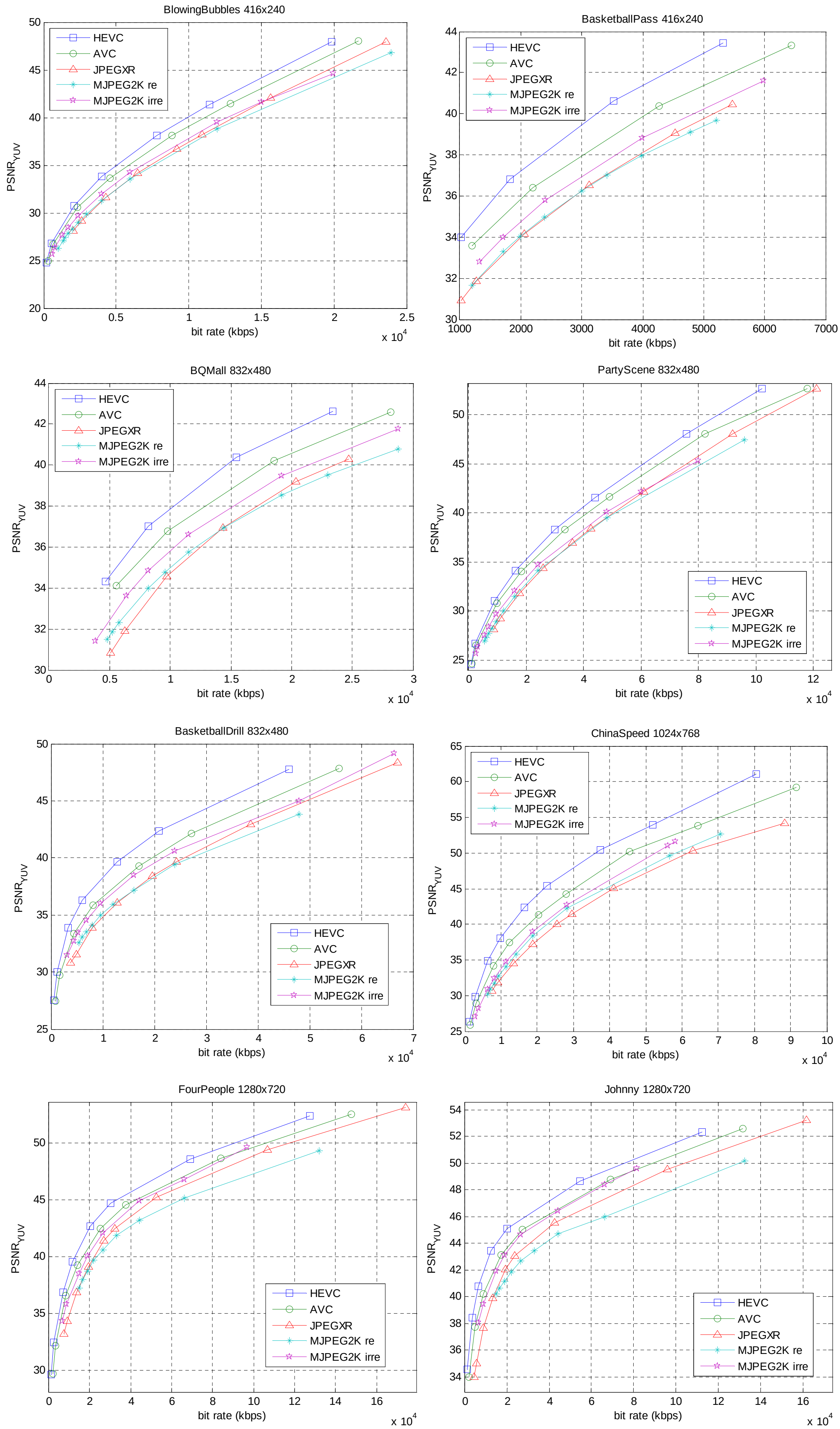


Figure 1: Performance curves

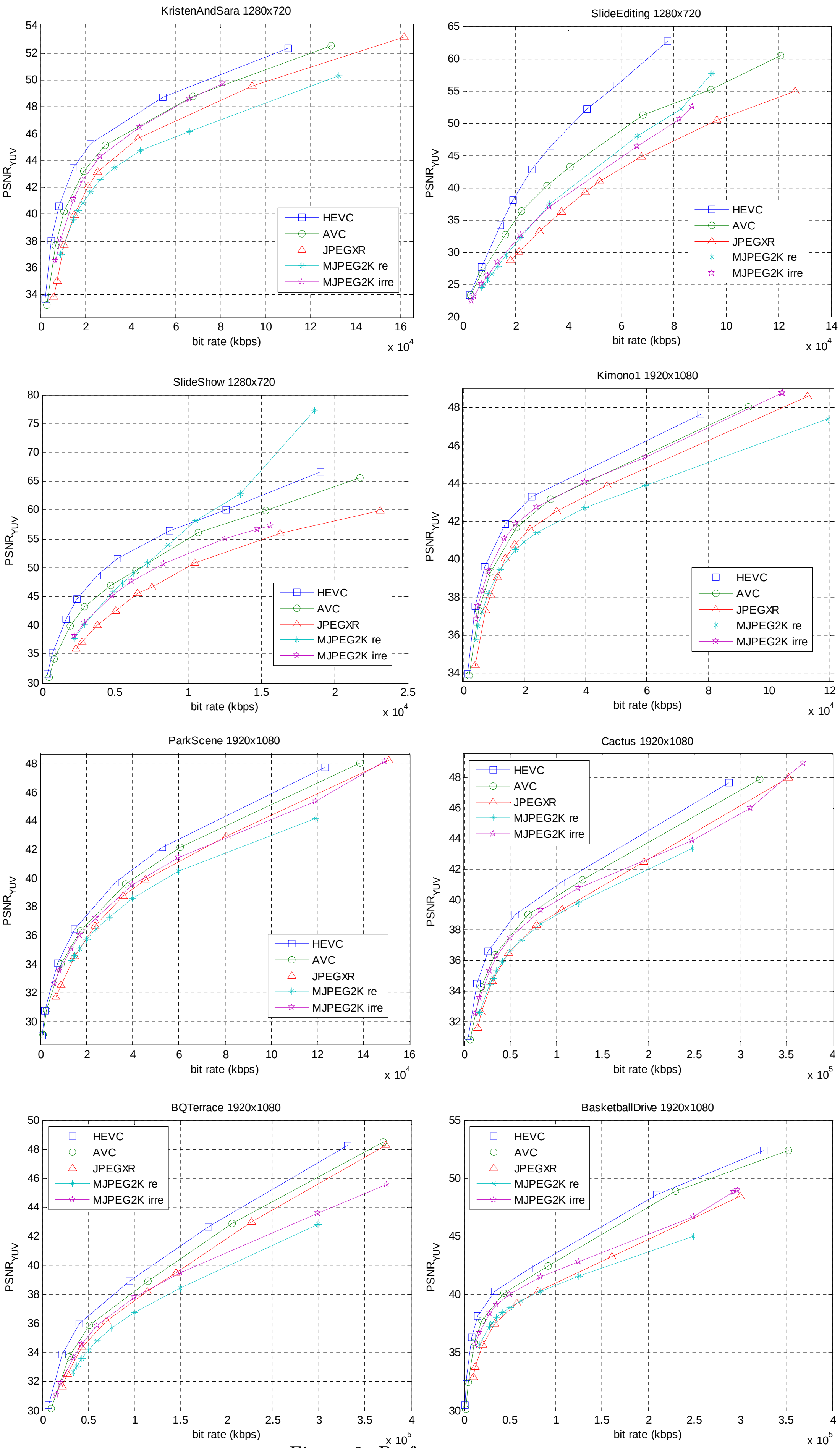


Figure 2: Performance curves

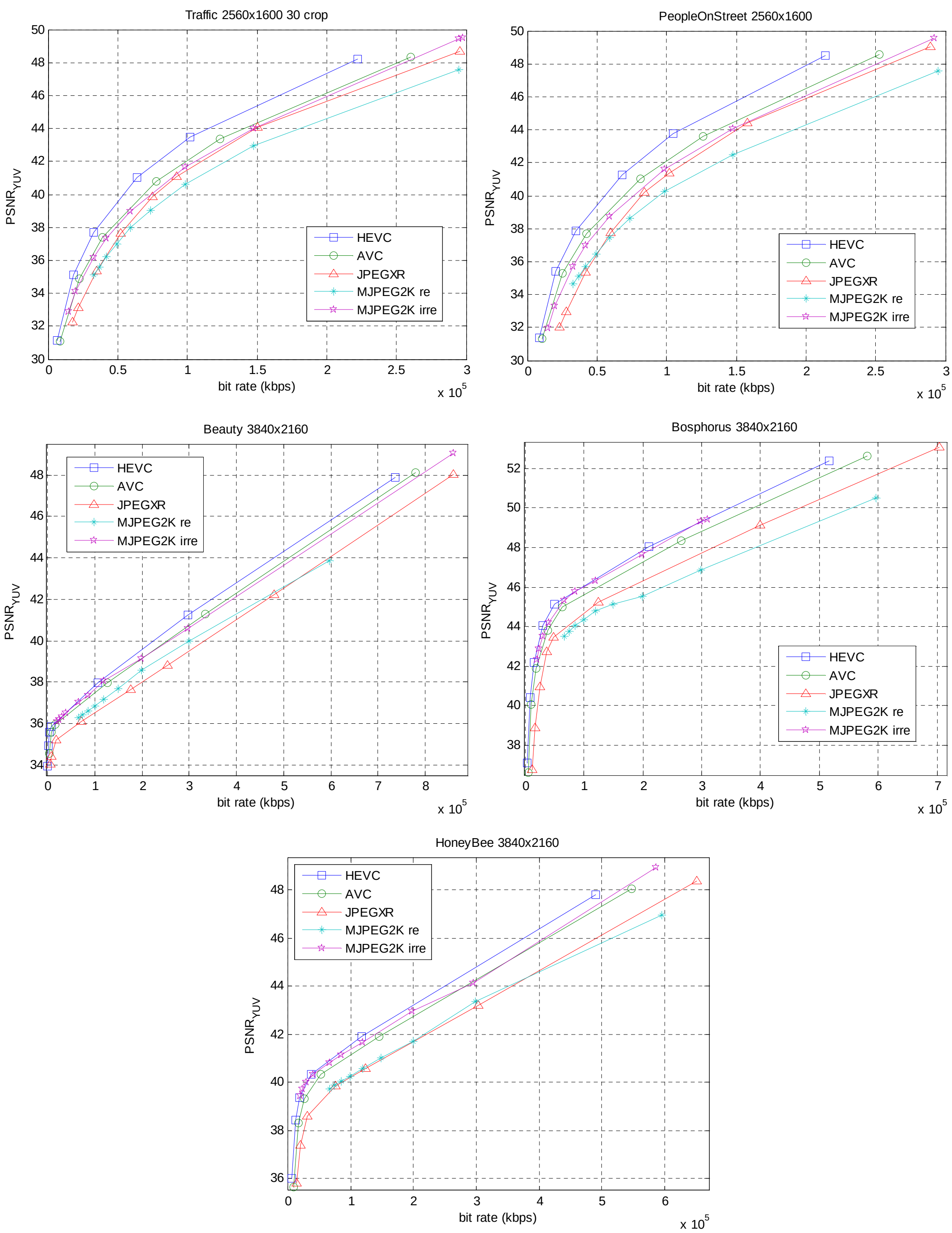


Figure 3: Performance curves