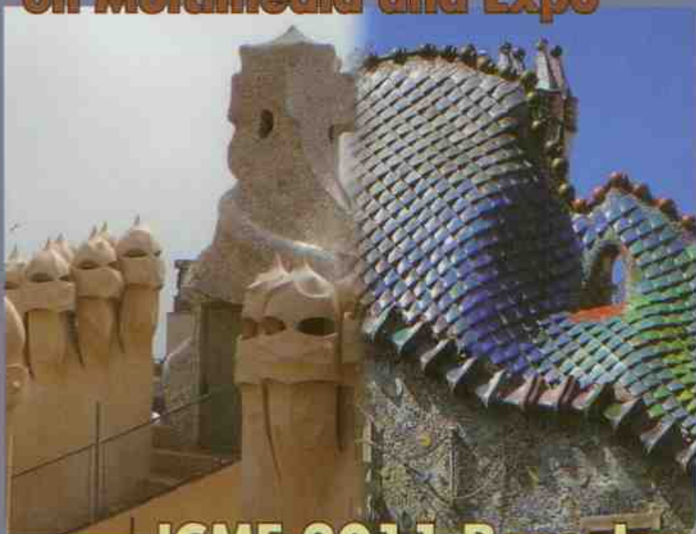
Program guide

IEEE International Conference on Multimedia and Expo



ICME 2011 Barcelona
July 11-15, 2011
Spain

IEEE

IEEE computer society

IEEE Communications Society

CAS

IEEE Signal Processing Society

PT2: POSTER 11
Room: JHM, Date: 7/12/2011, Time: 13:30-15:30
Session Chair(s): Oscar C. Au

DISTORTION OVER LATENCY: NOVEL METRIC FOR MEASURING INTERACTIVE PERFORMANCE IN REMOTE RENDERING SYSTEMS
Shu Shi, Klara Nahrstedt, Roy Campbell

IMAGE COMPRESSION ALGORITHM BASED ON HILBERT SCANNING OF EMBEDDED QUADTREES: AN INTRODUCTION OF THE HI-SET CODER.
Jesús Jaime Moreno Escobar, Xavier Otazu Porter

BILAYER VIDEO SEGMENTATION FOR VIDEOCONFERENCING APPLICATIONS
Alessandro Parolin, Guilherme Fickel, Claudio Jung, Tom Malzbender, Ramin Samadani

A NOVEL HOMOGRAPHY-BASED SEARCH ALGORITHM FOR BLOCK MOTION ESTIMATION IN VIDEO CODING
Zhaopeng Cui, Guang Jiang, Dujuan Wang, Chengke Wu

IMPROVED CABAC DESIGN IN H.264/AVC FOR LOSSLESS DEPTH MAP CODING
Jin Heo, Yo Sung Ho

TRANSMISSION DISTORTION-OPTIMIZED UNEQUAL LOSS PROTECTION FOR VIDEO TRANSMISSION OVER PACKET ERASURE CHANNELS
Yongfei Zhang, Shiyin Qin, Zhihai He

HYBRID LOW-DELAY COMPRESSION OF MOTION CAPTURE DATA
Choong Hoon Kwak, Ivan Bajic

H.264/AVC BASED NEAR LOSSLESS INTRA CODEC USING LINE-BASED PREDICTION AND MODIFIED CABAC
Jung Ah Choi, Jin Heo, Yo Sung Ho

JOINT SCALABLE VIDEO AND DIGITAL FOUNTAIN CODING FOR HETEROGENEOUS QOS VIDEO BROADCASTING
Wen Ji, Zhu Li, Yiqiang Chen

JOINT POWER ALLOCATION AND BIT LOADING FOR ENHANCED SVC VIDEO DOWNLINK TRANSMISSIONS OVER SDMA/OFDMA NETWORKS
Xiaodong Li, Zhenzhong Chen, Yap Peng Tan

H.264/AVC BASED NEAR LOSSLESS INTRA CODEC USING LINE-BASED PREDICTION AND MODIFIED CABAC

Jung-Ah Choi, Jin Heo, and Yo-Sung Ho

Gwangju Institute of Science and Technology
{jachoi, jinheo, hoyo}@gist.ac.kr

ABSTRACT

In this paper, we propose a new H.264/AVC based intra codec for near lossless coding. The proposed algorithm is composed of two parts: line-based intra prediction and modified context-based adaptive binary arithmetic coding (CABAC). Experimental results show that the proposed method provides about 8.95% bit savings, compared to the current H.264/AVC FRExt high profile.

Index Terms— H.264/AVC, intra coding, context-based adaptive binary arithmetic coding (CABAC), near lossless video coding, high quality video coding

1. INTRODUCTION

As demands for high quality video services continue to increase, the efficient method to encode high-definition (HD) contents becomes important issue of the next generation video coding. Compression of high quality video faces severe requirements to the quality of the reconstructed data. There are three kinds of video coding methods: lossless, lossy, and near lossless coding.

In order to encode HD sequences efficiently, it is very important to select an appropriate coding method. In terms of the decoded quality, lossless coding is suitable to encode HD contents, but it allows modest compression ratios, ranging normally from 1.5 to 3. Lossy coding yields much higher compression ratios by allowing some distortion in the reconstructed image; however, we incur a quality degradation of the decoded HD content.

Thus, near lossless coding seems to be the most suitable method for high quality video coding in terms of the introduced distortion and compression performance [1]. To date, since H.264/AVC [2] has been developed by mainly focusing on lossy coding, it does not provide good coding performance for near lossless video coding.

After finalizing the standardization of the first version of H.264/AVC, JVT developed extensions to the original standard, known as the fidelity range extensions (FRExt) [3]. When developing the FRExt amendment, JVT focused on improvement of functionality for lossy and lossless video coding. To enhance coding performance of H.264/AVC for near lossless video coding, more efficient coding techniques for core parts such as prediction, transform, quantization, or entropy coding is required.

In this paper, we have tried to improve both prediction and entropy coding. Generally, the magnitude of the error increases for prediction pixels farther away from the boundary pixels used for prediction. To maximize the prediction accuracy, we propose a line-based intra 16×16 prediction using line-of-pixels (LOP) as the prediction unit, not the entire macroblock.

For this task, we sequentially encode and reconstruct each LOP after prediction to utilize the reconstructed LOP as reference data of the next LOP. As the distance between pixels to be predicted and reference pixels is relatively short, we can take full advantage of correlation between neighboring pixels. In this way, higher prediction accuracy than the traditional block-based prediction method is obtained with no change of H.264/AVC syntax elements.

An improved context-based adaptive variable length coding (CAVLC) for lossless coding based on the statistical characteristics of residual data for lossless coding was proposed [4]. Although CAVLC is available for all profiles in H.264/AVC, main target of CAVLC is the baseline profile and the applications of this profile is limited.

We modified context-based adaptive binary arithmetic coding (CABAC) [5] in the H.264/AVC standard for near lossless coding. CABAC was originally developed for lossy coding; it was designed by taking into consideration the typically observed statistical properties of residual data. However, the statistical characteristics of residual data in near lossless coding are quite different from those of residual data in lossy coding. It means that use of H.264/AVC CABAC is inappropriate for near lossless intra coding. Thus, by considering the statistical properties of residual data for near lossless intra coding, we removed the coding stage for the unnecessary syntax element.

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2011-(C1090-1111-0003)).

978-1-61284-350-6/11/$26.00 ©2011 IEEE

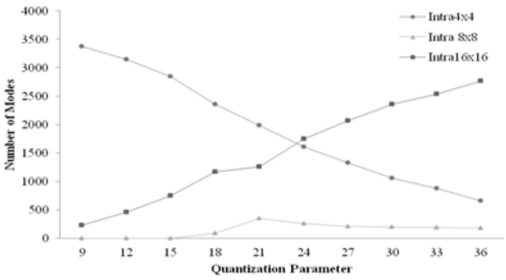
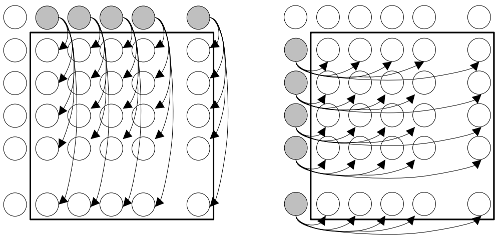


Fig. 1. Mode distribution according to QP (*Jets*, 720p HD).



(a) Vertical mode (b) Horizontal mode

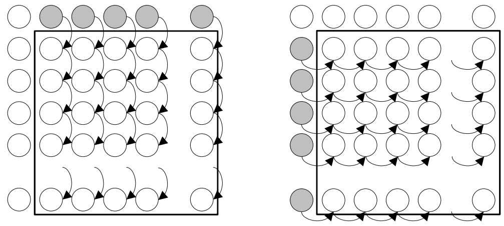
Fig. 2. H.264/AVC intra 16×16 prediction.

2. PROPOSED NEAR LOSSLESS INTRA CODING

2.1. Line-based intra prediction

We checked the selected intra mode distributions for various quantization parameters (QP), as shown in Fig. 1. In lossy coding (over QP = 21), the number of intra 16×16 prediction modes is larger than the number of intra 4×4 prediction modes; in near lossless coding (below QP = 21), the number of intra 16×16 prediction modes is smaller. This result indicates that the prediction accuracy of intra 16×16 prediction mode is not sufficient for selection as the best mode in near lossless coding. From this observation, we can know that the intra 16×16 prediction performance of H.264/AVC in near lossless coding should be improved.

In the conventional intra 16×16 prediction of H.264/AVC, the current macroblock is predicted by referencing neighboring pixels of previously coded upper and left macroblocks. In other words, the prediction accuracy is dependent on the location of pixels, i.e., pixels located further provide poor prediction performance. As shown in Fig. 2, in the intra 16×16 vertical and horizontal prediction modes, the prediction accuracy of the last LOP is the lowest. As a result, the number of coding bits increases. In particular, when the input sequence has a homogeneous texture pattern with variations such as gradation, the intra 16×16 mode in H.264/AVC cannot yield sufficient prediction accuracy. Therefore, in order to improve prediction accuracy of the intra 16×16 mode, we propose a more efficient line-based intra prediction method for near lossless intra coding in H.264/AVC by modifying the relevant coding procedure of intra 16×16 prediction.



(a) Vertical mode (b) Horizontal mode

Fig. 3. Line-by-line intra 16×16 prediction.

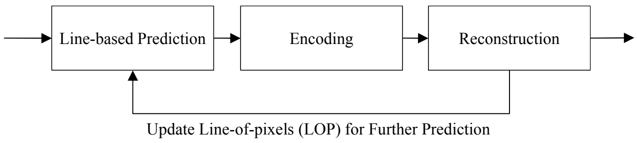


Fig. 4. Encoding structure of the proposed intra prediction method.

Table 1. Modified prediction equations

Mode	Prediction Equation
Vertical mode	**pred**(x,0) = **p**(x, -1) (1) **pred**(x,y) = **r**(x,y-1) (2)
Horizontal mode	**pred**(0,y) = **p**(-1,y) (3) **pred**(x,y) = **r**(x-1,y) (4)

In the vertical and horizontal mode of intra 16×16, 256 pixels in the current macroblock are predicted using 16 neighboring pixels. Generally, closer pixels give better prediction accuracy. The process of the line-by-line prediction is shown in Fig. 3.

The prediction equations of the proposed method are shown in Table 1. In the proposed method, we make the predicted pixel values of the first line within the current macroblock using Eq. (1) or Eq. (3) according to the mode. Then, the predicted pixel values are subtracted from the original pixel values and the residual data is formed. The transform, the quantization, the inverse quantization, the inverse transform are applied one after another to make the reconstructed pixel values, **r**(x,y). This reconstructed pixel values are used as the reference data of the next line. According to the prediction mode, we use Eq. (2) or Eq. (4) as the prediction formula. The above prediction process is continued until the last line of the macroblock.

In Fig. 4, we depict the encoding structure of the proposed method used to implement a line-based prediction. The coding procedure of the proposed intra prediction can be summarized in the following steps:

Step 1 Prediction of the first line of pixels (LOP).
Step 2 Transform and quantization of the residual LOP.
Step 3 Inverse quantization and inverse transform.
Step 4 Encoding next LOP using the reconstructed LOP.

The coded block pattern (*cbp*) signals as to whether there are non-zero coefficients in the transform block or not. In the conventional H.264/AVC, the *cbp* of the Intra_16×16 mode is just calculated one time. Thus, we should change the calculation procedure for *cbp* in the proposed algorithm. Since the prediction unit of the proposed Intra_16×16 coding is LOP, we can compute the cbp for each LOP, cbpline, where the subscript line represents the corresponding line position. The *cbp* for the macroblock is calculated from the cumulative value of cbp_{line}, as in Eq. (5).

$$cbp = cbp_0 + cbp_1 + \cdots + cbp_{15} = \sum_{line=0}^{15} cbp_{line} \qquad (5)$$

2.2. Proposed CABAC

In lossy coding, the degree of quantization is higher than that of near lossless coding. Hence, the probability distribution of a non-zero coefficient is likely to decrease as the scanning position increases. However, the characteristics of residual data in near lossless coding are different from those in lossy coding.

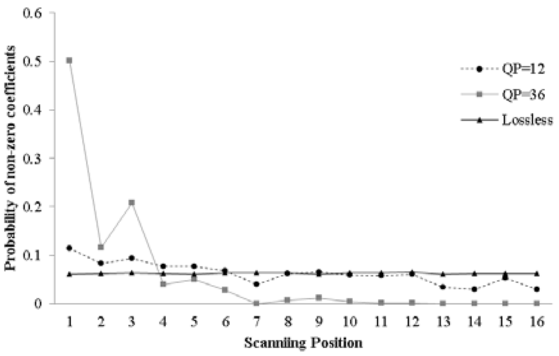


Fig. 5. Probability of non-zero coefficients.

In near lossless coding, the probability of existence of non-zero coefficients is independent of the scanning position. That is, in near lossless coding, non-zero coefficients frequently occur at the higher scanning position, compared to the case of lossy coding. Although near lossless coding is lossy coding, the statistical characteristics of residual data are much similar to lossless coding. Figure 5 shows the probability distribution of existence of non-zero coefficients according to the scanning position. As expected, a significant difference can be seen in the statistics between residual data of lossy and near lossless coding.

Therefore, in order to reflect the above observations more accurately, we propose more efficient CABAC for near lossless intra coding in H.264/AVC by modifying the relevant coding parts of CABAC.

In lossy coding, the occurrence probability of a non-zero coefficient is likely to decrease as the scanning position increases due to the higher QP value. Therefore, the coefficient tends to be zero at the higher scanning position. In this case, *last_significant_coeff_flag* plays an important role for the early termination of significance map coding.

Scanning position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Transform coefficient level	9	0	-5	3	0	-7	4	0	8	-11	-6	0	3	1	0	0
significant_coeff_flag	1	0	1	1	0	1	1	0	1	1	1	0	1	1		
last_significant_coeff_flag	0		0	0		0	0		0	0	0		0	1		

Fig. 6. Example for significance map coding in lossy coding.

Scanning position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Transform coefficient level	9	0	-5	3	0	-7	4	0	8	-11	-6	0	3	1	0	0
significant_coeff_flag	1	0	1	1	0	1	1	0	1	1	1	0	1	1	0	0

Fig. 7. Example for significance map coding in near lossless coding.

However, in near lossless coding, the occurrence probability of a non-zero coefficient is independent of the scanning position, as shown in Fig. 5. Thus, the last non-zero coefficient is terminated at the end of the scanning position. In this case, it is meaningless to encode *last_significant_coeff_flag* that indicates the position of the last significant coefficient. Therefore, we remove *last_significant_coeff_flag* coding process in significance map coding and directly encode all *significant_coeff_flag*s at the scanning position from 1 to 16.

An example for significance map coding of CABAC in lossy coding is represented in Fig. 6 when the scanning position of the last significant coefficient is 14; the gray shaded *significant_coeff_flag* and *last_significant _coeff_flag* are encoded in significant map coding. Note that both *significant_coeff_flag* and *last_significant _coeff_flag* for the last scanning position of a sub-block are never encoded.

However, since we removed *last_significant_coeff_flag* coding process, *significant_coeff_flag* is unconditionally encoded up to the last scanning position. Figure 7 indicates an example for significance map coding in near lossless coding. All gray shaded *significant_coeff_flag*s are encoded in the proposed significance map coding, as shown in Fig. 7.

3. EXPERIMENTAL RESULTS

In this paper, an efficient line-based intra prediction and an improved CABAC for near lossless coding have been presented. To verify efficiency of the proposed methods, experiments were performed on various test sequences of YUV420 and 8 bits per pixel (bpp) format with 720p HD and 1080p Full HD resolutions. We implemented our proposed method in the H.264/AVC reference software version JM 12.2 [6]. The encoding parameters for the reference software were as follows:

1) *ProfileIDC* = 100 (High)
2) *LevelIDC* = 51
3) *IntraPeriod* = 1 (only intra coding)
4) *QPISlice* = 8, 10, 12, 14
5) *Transform8x8Mode* = 1
6) *SymbolMode* = 1 (CABAC)

In order to evaluate coding performance of each proposed method, we experimented with two parts according to different settings as follows:

1) *Method I*: line-based intra coding
2) *Method II*: *Method I* + modify significance map coding

To verify efficiency of the proposed method, we have performed two kinds of experiments. In the first experiment, we compared the performance of the proposed method (*Method I* and *Method II*) with that of H.264/AVC, as shown in Table 2. The Bjøntegaard delta peak signal-to-noise ratio (dB) and the Bjøntegaard delta bit rate (%) are used to evaluate the performance of the proposed algorithm in Table 2 with respect to H.264/AVC.

From Table 2, we confirmed that the proposed method provided better coding performance by approximately 8.95% with 720p HD and 1080p Full HD resolution sequences, compared to the conventional H.264/AVC.

In the second experiment, we encoded test sequences using our proposed method (*Method II*), H.264/AVC, and a well-known near lossless coding technique, JPEG-LS [7] and then, compared coding performance of our proposed method to that of JPEG-LS. Comparisons were made in terms of compression ratio (CR) and PSNR in Table 3. The CR is calculated as follows:

$$\textit{Compression Ratio} = \frac{\textit{Original image size}}{\textit{Bitrate}_{\textit{proposed}}} \quad (6)$$

In Table 3, **M** is the max allowed loss per symbol in near lossless coding. Using **M** value, we can fix the decoded image quality and compare the coding bits. From the result, we validate that the proposed method guarantees the best coding performance in near lossless coding.

For an exact comparison, we plot the bits per pixel (BPP)-distortion curve for some test sequences in Fig. 8; the x-axis of the graph is BPP and the y-axis is the PSNR. In Fig. 8, the dashed curve represents the performance of *Method II*. Since the upper curve indicates better performance, we confirmed that the proposed method provides the best coding performance for all test sequences in near lossless coding.

4. CONCLUSIONS

In this paper, we have proposed an efficient H.264/AVC-based intra codec for near lossless coding. Considering the different characteristics between near lossless coding and lossy coding, we have modified the intra coding method using line-based prediction and modified context-based adaptive binary arithmetic coder (CABAC) methods. In the proposed method, we break the traditional block-based intra coding using new prediction unit and modified syntax element. Experimental results show that the proposed method provides approximately 8.95% bit saving, compared to the H.264/AVC FRExt high profile.

5. REFERENCES

[1] I. Avcibas, N. Memon, B. Sankus, and K. Sayood, "Lossless and Near-lossless image compression with successive refinement," *IEEE Transactions on Communications*, vol. 53, no. 3, March 2005.

[2] ITU-T Recommendation H.264 and ISO/IEC 14496-10, Advanced Video Coding for Generic Audiovisual Services, May 2003.

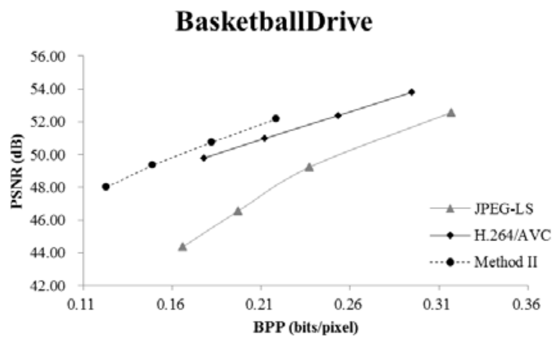
[3] Draft Text of H.264/AVC Fidelity Range Extensions Amendment to ITU-T Rec. H.264, document JVT-L047.doc, Joint Video Team of ISO/IEC 14496-10 AVC, ISO/IEC JTC1/SC29/WG11, and ITU-T Q.6/SG16, July 2004.

[4] J. Heo, S. H. Kim, and Y. S. Ho, "Improved CAVLC for H.264/AVC lossless intra-coding," *IEEE Trans. Circuits Syst. Video Technol.*,vol. 20, no. 2, pp. 213-222, Feb. 2010.

[5] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 620–636, July 2003.

[6] http://iphome/hhi.de/shehring/tml/download/old_jm/jm 12.2.zip, Joint Video Team, Reference Software Version 12.2.

[7] K. Sayood, "Lossless image compression," in Introduction to Data Compression, San Mateo, CA: Morgan Kaufmann, 2006.



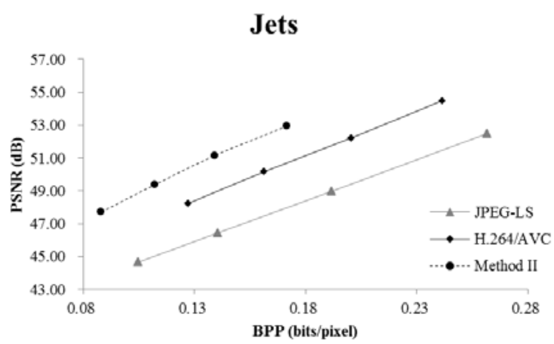


Fig. 8. Bits per pixel (BPP)-distortion curves.

Table 2. Performance of proposed method with respect to H.264/AVC near lossless intra coding

Image	Method	BDPSNR (dB)	BDRATE (%)
Rush_hour (1920 ×1080)	*Method I* *Method II*	0.47 0.51	-6.26 -6.63
Pedestrian (1920 ×1080)	*Method I* *Method II*	0.48 0.52	-6.12 -6.50
BasketballDrive (1920 ×1080)	*Method I* *Method II*	0.76 0.86	-9.94 -11.12
Toys and Calendar (1920 ×1080)	*Method I* *Method II*	0.56 0.89	-4.83 -7.55
Bigships (1280 ×720)	*Method I* *Method II*	0.22 0.37	-1.95 -3.17
Jets (1280 ×720)	*Method I* *Method II*	2.01 2.22	-21.42 -22.83
Vidyo1 (1280 ×720)	*Method I* *Method II*	0.58 0.64	-7.75 -8.48
Vidyo4 (1280 ×720)	*Method I* *Method II*	0.35 0.38	-4.98 -5.29
Average	*Method I* *Method II*	**0.68** **0.80**	**-7.91** **-8.95**

Table 3. Comparison of compression ratio for H.264/AVC, JPEG-LS, and proposed method

Image	Method	M=1		M=2		M=3		M=4	
		CR	PSNR (dB)	CR	PSNR (dB)	CR	PSNR (dB)	CR	PSNR (dB)
Rush_hour	H.264/AVC	4.20	52.52	5.43	49.17	6.62	46.58	6.63	46.58
	JPEG-LS	4.36	54.01	5.31	52.15	6.73	50.34	8.78	48.47
	Method II	5.00	53.29	6.19	51.44	7.90	49.70	10.43	47.90
Pedestrian	H.264/AVC	4.42	52.39	5.75	49.34	6.92	46.77	8.14	44.85
	JPEG-LS	4.52	54.52	5.43	52.63	6.79	50.83	8.63	48.99
	Method II	5.18	53.72	6.34	51.84	8.04	50.09	10.37	48.35
Basketball Drive	H.264/AVC	3.16	52.57	4.21	49.24	5.07	46.56	6.03	44.37
	JPEG-LS	3.39	53.80	3.95	52.38	4.72	50.98	5.62	49.79
	Method II	4.57	52.16	5.49	50.74	6.71	49.34	8.14	48.00
Toys and Calendar	H.264/AVC	3.03	52.61	4.00	49.30	4.76	46.66	5.47	44.65
	JPEG-LS	2.56	54.68	2.90	52.58	3.45	50.41	4.11	48.40
	Method II	2.96	53.19	3.45	51.24	4.18	49.28	5.04	47.48
Bigships	H.264/AVC	3.07	52.57	3.95	49.22	4.67	46.58	5.33	44.52
	JPEG-LS	2.93	54.60	3.34	52.65	3.92	50.73	4.59	48.90
	Method II	3.23	53.54	3.77	51.48	4.51	49.59	5.39	47.80
Jets	H.264/AVC	3.82	52.50	5.21	48.98	7.11	46.46	9.54	44.67
	JPEG-LS	4.14	54.48	4.98	52.23	6.20	50.16	7.87	48.23
	Method II	5.83	52.94	7.19	51.15	8.90	49.39	11.36	47.71
Vidyo1	H.264/AVC	4.17	52.55	5.27	49.30	6.32	46.74	7.39	44.77
	JPEG-LS	4.25	54.10	5.19	52.21	6.59	50.49	8.31	48.90
	Method II	5.25	52.92	6.45	51.28	8.00	49.71	9.73	48.30
Vidyo4	H.264/AVC	4.47	52.66	5.80	49.41	7.11	46.89	8.42	44.93
	JPEG-LS	4.75	54.29	5.89	52.39	7.50	50.73	9.47	49.21
	Method II	5.52	53.47	6.90	51.68	8.64	50.09	10.65	48.68
Average	H.264/AVC	**3.79**	**52.55**	**4.95**	**49.25**	**6.07**	**46.66**	**7.12**	**44.92**
	JPEG-LS	**3.86**	**54.31**	**4.62**	**52.40**	**5.74**	**50.58**	**7.17**	**48.86**
	Method II	**4.69**	**53.15**	**5.72**	**51.36**	**7.11**	**49.65**	**8.89**	**48.03**