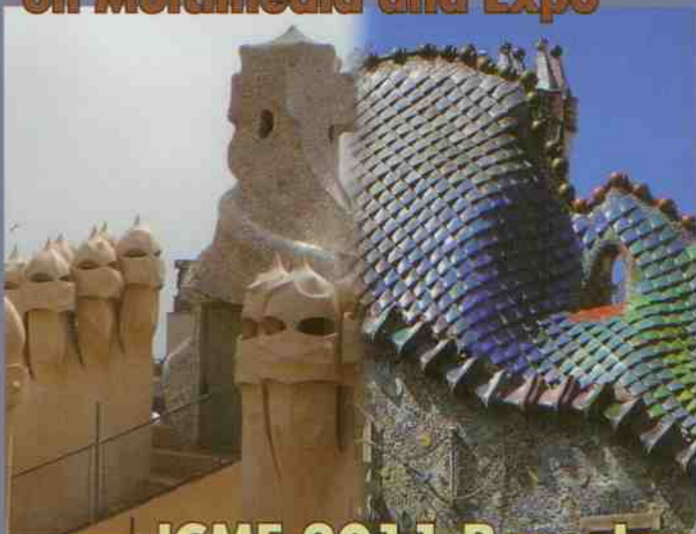
Program guide

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

IMPROVED CABAC DESIGN IN H.264/AVC FOR LOSSLESS DEPTH MAP CODING

Jin Heo, Yo-Sung Ho

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

ABSTRACT

The depth map represents three-dimensional (3D) data and is used for depth image-based rendering (DIBR) to synthesize virtual views. Since the quality of synthesized virtual views highly depends on the quality of the depth map, we encode the depth map by lossless coding mode. However, context-based adaptive binary arithmetic coding (CABAC) for the H.264/AVC standard does not guarantee the best coding performance for lossless depth map coding because CABAC was originally designed for lossy coding. In this paper, we propose an improved coding method of CABAC for lossless depth map coding considering the statistical properties of residual data from lossless depth map coding. Experimental results show that the proposed CABAC method provides approximately 4.3% bit saving, compared to the original CABAC in H.264/AVC.

Index Terms— H.264/AVC, CABAC, lossless coding, depth map

1. INTRODUCTION

Three-dimensional video (3DV) technologies have been studied to expand the user's sensation beyond what is offered by the traditional media. In the 3DV system [1], the main difficulty in the deployment of 3DV services appears to be the large bandwidth requirements associated with transport of multiple video streams. Therefore, the depth map is used as 3D information to synthesize virtual views corresponding to an arbitrary viewpoint because depth map coding consumes a small overhead, typically about 10-20%, on the video bitrate [2].

In order to synthesize a virtual view, we use a depth image-based rendering (DIBR) technique using video and depth map streams [3]. In DIBR, accuracy of the depth map directly affects the quality of synthesized virtual view. Therefore, several research works have focused on the acquisition of the accurate depth map [4], [5].

Before 3D video is rendered at the receiver side, we transmit a regular video stream as well as its depth map. In the 3DV system, the depth map is encoded by lossy coding. However, since the quality of the depth map is highly related to the rendering quality of virtual views, the depth map should be encoded without any loss.

In lossy depth map coding, residual data represent the quantized transform coefficients. On the contrary, in lossless depth map coding, residuals are not the quantized transform coefficients, but rather the differential pixel values between the original and predicted pixel values. Therefore, the statistical properties of residual data between lossy and lossless depth map coding are quite different.

Since the context-based adaptive binary arithmetic coder (CABAC) [6] for the H.264/AVC standard [7] was originally designed for lossy coding, CABAC was unable to provide the optimum coding performance for lossless depth map coding. Therefore, we have tried to improve coding performance of CABAC for lossless depth map coding.

2. OVERVIEW OF CABAC IN H.264/AVC

The encoding process of CABAC consists of four main coding steps: binarization, context modeling, binary arithmetic coding, and probability update. In the first step, a given non-binary valued syntax element is uniquely mapped to a binary sequence. When a binary valued syntax element is given, the first step is bypassed. In the regular coding mode, each binary value of the binary sequence enters the context modeling stage, where a probability model is selected based on the previously encoded syntax elements. Then, the arithmetic coding engine encodes each binary value with its associated probability model. Finally, the selected context model is updated according to the actual coded binary value.

Figure 1 illustrates encoding structure of CABAC for residual data coding in a 4×4 sub-block. First, the syntax element *coded_block_flag* is encoded for each sub-block with a 1-bit symbol. If *coded_block_flag* is zero, no further information is encoded for the sub-block. Otherwise, significance map and level information coding processes are sequentially performed.

This research was supported by MKE under ITRC support program supervised by NIPA (NIPA-2011-(C1090-1111-0003)).

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

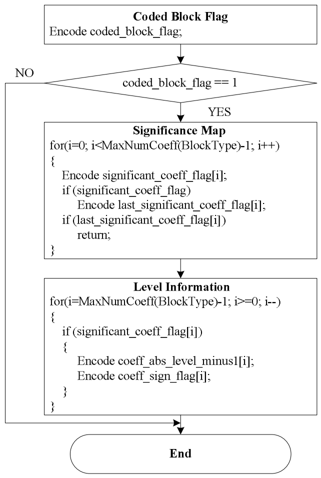


Fig. 1. Encoding structure of CABAC for residual data coding.

If *coded_block_flag* indicates that the sub-block has significant coefficients, a binary valued significance map is encoded. For each coefficient, a 1-bit symbol *significant_coeff_flag* is encoded in scanning order. If *significant_coeff_flag* is one, a further 1-bit symbol *last_significant_coeff_flag* is encoded. This syntax element indicates if the current significant coefficient is the last coefficient inside the sub-block or if further significant coefficients follow.

After the encoded significance map determines the locations of all significant quantized transform coefficients inside a sub-block, the values of the significant coefficients are encoded by using two syntax elements: *coeff_abs_level_minus1* and *coeff_sign_flag*. The syntax element *coeff_sign_flag* is encoded by a 1-bit symbol, whereas the *Unary/0th order Exp-Golomb* (UEG0) binarization scheme is used to encode the values of *coeff_abs_level_minus1* representing the absolute value of the level minus 1. The values of the significant coefficients are encoded in reverse scanning order.

3. PROPOSED METHOD

3.1. Significance Map Coding

In lossy depth map coding, the occurrence probability of a significant coefficient is likely to decrease as the scanning position increases. Therefore, the significant coefficient tends to be located at earlier scanning position. In this case, *last_significant_coeff_flag* plays an important role in the early termination of significance map coding.

However, in lossless depth map coding, residual data do not represent the quantized transform coefficients, but rather the differential pixel values. Therefore, the statistics of residual data from lossy and lossless depth map coding are very different. In lossless depth map coding, the occurrence probability of a significant differential pixel is independent of the scanning position, as shown in Fig. 2.

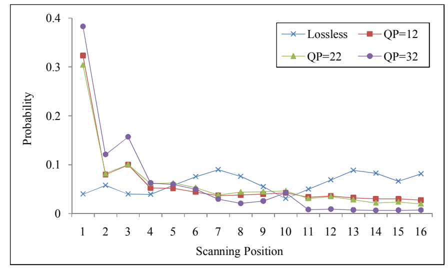


Fig. 2. Occurrence probability distribution of non-zero coefficients according to the scanning position.

From extensive experiments on lossless depth map coding, we observed that significance map coding is likely to be terminated at the end of the scanning position. In this case, it is meaningless to encode *last_significant_coeff_flag* indicating the position of the last significant differential pixel. Therefore, we can remove the *last_significant_coeff_flag* coding process and directly encode all *significant_coeff_flag*s at the scanning position from 1 to 16.

Figure 3 represents the examples of significance map coding in lossy and lossless coding. In lossy coding, the gray shaded *significant_coeff_flag* and *last_significant_coeff_flag* are encoded up to the scanning position number 14. However, since we removed *last_significant_coeff_flag* in lossless coding, *significant_coeff_flag* is unconditionally encoded up to the last scanning position. Therefore, all gray-shaded *significant_coeff_flag*s are encoded in the proposed 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		

(a) Lossy coding

Scanning position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Differential pixel value	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

(b) Lossless coding

Fig. 3. Example of significance map coding.

3.2. Level Coding

In level coding, the UEG0 binarization method is applied to the absolute values of the quantized transform coefficients (*abs_level*). UEG0 binarization for *abs_level* has a cutoff value S = 14 for the *truncated unary* (TU) prefix and the order k = 0 for the *Exp-Golomb* (EG0) suffix. The structure of UEG0 binarization is only accurate for small *abs_level*; for larger *abs_level*, adaptive modeling has limited functionality.

The statistics of the absolute value of the differential pixel (*abs_diff_pixel*) in lossless coding is different from that of *abs_level* in lossy coding. In lossy coding, the occurrence probability of *abs_level* is highly skewed on small *abs_level*. However, in lossless coding, the distribution of *abs_diff_pixel* is quite wide, as shown in Fig. 4. Therefore, UEG0 binarization used in lossy coding is not appropriate for lossless coding.

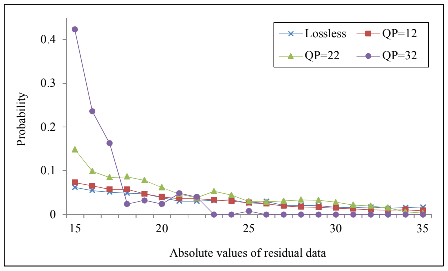


Fig. 4. Occurrence probability distribution of the absolute value.

In order to efficiently encode *abs_diff_pixel* in lossless coding, theoretically, the larger cutoff value for UEG0 binarization is required. UEG0 binarization with a very large cutoff value is convergent to TU binarization. Therefore, we use TU binarization with the unlimited upper bound instead of UEG0 binarization.

3.3. Context Modeling

Each model can be identified by the unique context index because the entirety of probability models used in CABAC can be arranged in a linear fashion. The context index for syntax element of residual data (γ_S) is specified by

$$\gamma = \Gamma_S + \Delta_S(ctx\_cat) + \chi_S \tag{1}$$

where Γ_S and χ_S represent the context index offset, which is defined as the lower value of the range and the context index increment of a given syntax element *S*, respectively. In addition, $\Delta_S(ctx\_cat)$ is the context category dependent offset Δ_S and is determined according to the block type.

In lossless coding, since neither transform nor quantization is performed, all pixel values within a sub-block do not separate DC and AC coefficients. As a result, block type corresponding to context category of 2 (luma block for intra4×4) is only used in lossless coding. Thus, we can fix the context category value for *significant_coeff_flag* and *coeff_abs_level_minus1*.

In lossy coding, the context index increment for *significant_coeff_flag* (χ_{SIG}) was designed based on the expectation that the occurrence probability of a significant coefficient is likely to decrease as the scanning position increases, as shown in Fig 2. Therefore, χ_{SIG} can be determined according to the scanning position. However, in lossless coding, the occurrence probability of a significant differential pixel is independent of the scanning position. Therefore, we do not consider χ_{SIG} in the context index decision for *significant_coeff_flag* (γ_{SIG}). As a result, γ_{SIG} is given by

$$\gamma_{SIG} = \Gamma_{SIG} + \Delta_{SIG}(ctx\_cat = 2) \tag{2}$$

In order to determine the context index increment for *coeff_abs_level_minus1* (χ_{Coeff}), we use two adequately designed sets of context models: one for the first bin and another for the remaining bins in the UEG0 prefix. In lossy coding, at the end of the scanning position, *abs_level* is likely to observe the occurrence of successive ±1. In addition, *abs_level* is going to be larger as the scanning position decreases. Based on these observation, χ_{Coeff} is determined according to the accumulated number of encoded trailing ones (*NumT1*(*i*)) and the accumulated number of encoded levels with an absolute value greater than one (*NumLgt1*(*i*)), where *i* is the scanning position.

However, the statistics of the absolute value of residuals from lossy and lossless coding are very different especially in trailing ones. The number of consecutive ±1 at the end of the scanning position in lossless coding is greater than that in lossy coding. Therefore, we modify the context index increment for the first bin and do not use the context index increment for the remaining bins. Finally, the context index for *coeff_abs_level_minus1* (γ_{Coeff}) is calculated by

$$\begin{aligned}&\gamma_{Coeff}(i, bin\_index = 0)\\&= \Gamma_{Coeff} + \Delta_{Coeff}(ctx\_cat = 2) + \chi_{Coeff}(i, bin\_index = 0)\end{aligned} \tag{3}$$

$$\begin{aligned}&\chi_{Coeff}(i, bin\_index = 0)\\&= \begin{cases} 6, & \text{if } NumLgt1(i) > 0 \\ \min(5, NumT1(i)), & \text{otherwise} \end{cases}\end{aligned} \tag{4}$$

$$\gamma_{Coeff}(i, bin\_index) = \Gamma_{Coeff} + \Delta_{Coeff}(ctx\_cat = 2) \tag{5}$$

4. EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify coding efficiency of the proposed method, we performed experiments on several depth map sequences. We implemented our proposed method in the H.264/AVC reference software version JM 13.2 [8]. The encoding parameters for the reference software were as follows.

1) *ProfileIDC* = 244 (High 4:4:4)
2) *IntraPeriod* = 1 (only intra coding)
3) *QPISlice* = 0 (lossless)
4) *SymbolMode* = 1 (CABAC)
5) *QPPrimeYZeroTransformBypassFlag* = 1 (lossless)

In order to evaluate coding performance of each proposed method, our experiment included two sections, based on the following settings:

1) *Method I*: proposed significance map coding
2) *Method II*: *Method I* + proposed level coding + proposed context modeling

To verify efficiency of the proposed method, we performed two kinds of experiments. In the first experiment, four depth map sequences were tested. In the second experiment, in order to compare coding performance of our proposed method we encoded only one frame (first frame) for each depth map sequence using our proposed method (*Method II*) and a well-known lossless coding technique; lossless joint photographic experts group (JPEG-LS) [9].

In our experiments, we compared bit-rate percentage differences (Table 1) and compression ratio differences (Table 2) with respect to the original CABAC in H.264/AVC and JPEG-LS, respectively. These changes were calculated as follows:

$$\Delta Saving\ \ Bits(\%) = \frac{Bitrate_{H.264/AVC} - Bitrate_{Method}}{Bitrate_{H.264/AVC}} \times 100 \quad (6)$$

$$Compression\ \ Ratio = \frac{Original\ image\ size}{Bitrate_{Method}} \quad (7)$$

Table 1. Comparison of saving bits for H.264/AVC CABAC and the proposed methods

Sequence (Original Size (bits))	Method	Total Bits (bits)	ΔSaving Bits (%)
Breakdancers (102×4768 (629145600))	Original	29858176	0
	Method I	29109048	2.50895
	Method II	28807464	3.51901
Ballet (1024×768 (629145600))	Original	28204088	0
	Method I	27602192	2.13407
	Method II	27174088	3.65195
Beer_garden (1920×1080 (1658880000))	Original	127849392	0
	Method I	122219448	4.40358
	Method II	121484624	4.97833
Mobile (720×540 (311040000))	Original	2918744	0
	Method I	2869736	1.67908
	Method II	2775848	4.89580

Table 2. Comparison of compression ratio for JPEG-LS and the proposed method

Sequence (Original Size)	Method	Compression Ratio
Breakdancers (102×4768)	Original	13.5979
	JPEG-LS	3.9388
	Method II	14.1016
Ballet (102×4768)	Original	14.9010
	JPEG-LS	3.9072
	Method II	15.4281
Beer_garden (1920×1080)	Original	8.3770
	JPEG-LS	2.6241
	Method II	8.8169
Mobile (720×540)	Original	60.7500
	JPEG-LS	4.8473
	Method II	62.7198

In Table 1, we confirmed that the proposed method provided a better coding performance of approximately 4.26% bit saving, compared to the conventional CABAC. Table 2 presents the experimental results comparing JPEG-LS in terms of lossless intra coding, which again shows that the proposed method showed better coding performance in lossless coding.

5. CONCLUSIONS

In this paper, we proposed an improved context-based adaptive binary arithmetic coding (CABAC) for lossless depth map coding. Considering the statistical differences in residual data between lossy and lossless depth map coding, we modified the CABAC encoding mechanism based on truncated unary binarization for differential sample value, modified significance map coding, and modified context modeling. Experimental results show that the proposed method provides approximately 4.3% bit saving, compared to the current H.264/AVC CABAC.

6. REFERENCES

[1] A. Redert, M.O. Beeck, C. Fehn, W. IJsselsteijn, M. Pollefeys, L.V. Gool, E. Ofek, I. Sexton, and P. Surman, "ATTEST: Advanced Three-dimensional Television System Technologies," in Proc. Of International Symposium on 3D Data Processing Visualization and Transmission, pp. 313–319, Nov. 2002.

[2] H.M. Ozaktas and L. Onural, Three-Dimensional Television Capture, Transmission, Display, Springer, 2008.

[3] C. Fehn, "Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3-D TV," in Proc. Of SPIE Conf. Stereoscopic Displays and Virtual Reality Systems, vol. 5291, pp. 93–104, Jan. 2004.

[4] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality Video View Interpolation Using a Layered Representation," in Proc. Of SIGGRAPH'04, pp. 600–608, Aug. 2004.

[5] S.Y. Kim, E.K. Lee, and Y.S. Ho, "Generation of ROI Enhanced Depth Maps Using Stereoscopic Cameras and a Depth Camera," IEEE Trans. Broadcast., vol. 54, no. 4, pp. 732–740, Dec. 2008.

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

[7] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 560–576, July 2003.

[8] http://iphome/hhi.de/shehring/tml/download/old_jm/jm13.2.zip

[9] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," IEEE Trans. Image Process., vol. 9, no. 8, pp. 1309–1324, Aug. 2000.