Improved Entropy Coder in H.264/AVC for Lossless Residual Coding in the Spatial Domain

Jin Heo and Yo-Sung Ho

School of Information and Communicatitions
Gwangju Institute of Science and Technology (GIST)
261 Cheomdan-gwagiro, Buk-gu, Gwangju 500-712, Republic of Korea
{jinheo,hoyo}@gist.ac.kr

Abstract. Since a block-based frequency transform is applied to residual data in lossy coding, it can reduce the spatial correlation efficiently. However, since residual data obtained from prediction is directly encoded without transform and quantization in lossless coding, there are some differences of the statistical properties in residuals between lossy and lossless coding. Based on the statistical characteristics of residuals in the spatial domain, we proposed an efficient context-based adaptive binary arithmetic coder (CABAC) for lossless residual coding. Experimental results show that the proposed CABAC provided approximately 19% bit saving, compared to the conventional CABAC.

Keywords: context-based adaptive binary arithmetic coding (CABAC), H.264/AVC, lossless coding, intra coding.

1 Introduction

Since H.264/AVC improves coding performance over previous video coding standards such as H.263 and MPEG-4 by using the state-of-the-art coding tools, it is currently the most powerful coding standard [1] [2]. Moreover, since H.264/AVC is known to provide high coding efficiency for lossy video coding, it has been used for a wide range of applications, including broadcast, storage, and video telephony.

Lossless compression has long been recognized as an important part for application areas that require high quality such as source distribution, digital cinema, and medical imaging. Recently, as the number of services for higher quality video representation is expanding, the importance for lossless coding is also increasing. However, since the majority of research pertaining to H.264/AVC has focused on lossy coding, it is not suitable for lossless coding.

In order to enhance coding performance for lossless coding, H.264/AVC included a transform-bypass lossless mode which uses prediction and entropy coding for encoding sample values, in the fidelity range extensions (FRExt) [3]. Although the new lossless mode of FRExt became a fairly efficient method for lossless video coder, it was not the best method for lossless coding. Therefore, more efficient coding techniques for prediction and entropy coding in lossless coding are still required.

Recently, new intra prediction methods, referred to as sample-wise *differential pulse-code modulation* (DPCM) [4] [5] were introduced for lossless intra coding.

However, since they still employed the conventional entropy coders, originally designed for discrete cosine transform (DCT) based lossy coding, new intra prediction methods have limited functionality.

The statistical characteristics of residual data are different between lossy and lossless coding; in lossy coding, residual data are the quantized transform coefficients in the frequency domain, whereas in lossless coding, residual data are just prediction residuals in the spatial domain without transform and quantization. Thus an improved context-based adaptive variable length coding (CAVLC) method was proposed for lossless coding [6]. In this paper, we proposed an improved context-based adaptive binary arithmetic coding (CABAC) method [7] for lossless intra coding based on the observed statistical properties of residual data in the spatial domain.

This paper is organized as follows. In Section 2, after we show an overview of H.264/AVC lossless coding, we present the CABAC encoder framework and the structure of CABAC for residual data coding. In Section 3, we explain our proposed CABAC method for lossless residual coding in the spatial domain. Experimental results are given in Section 4, and we draw our conclusion in Section 5.

2   Overview of H.264/AVC Lossless Coding and CABAC

2.1   H.264/AVC Lossless Coding

A typical encoding algorithm for lossy coding proceeds as follows. An input picture is split into macroblock and then each macroblock is encoded in intra or inter mode. The residual of the intra or inter prediction is transformed by a frequency transform. Finally the transform coefficients are quantized, entropy coded, and transmitted together with the side information. Fig. 1 shows the conventional H.264/AVC encoder structure.
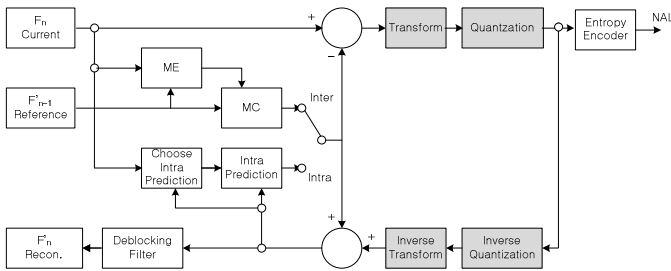

Fig. 1. Flowchart of CAVLC

However, an encoder structure for lossless coding is different from that for lossy coding. Since lossless data compression allows the exact original data to be reconstructed from the compressed bitstream, transform and quantization processes which cause a data loss are not included in the lossless encoder; the gray shaded stages in Fig. 1 are not included. Therefore, the residual obtained from prediction is directly encoded by the entropy coder. As a result, the residual data is handled only in the spatial domain during the entire lossless encoding process.

2.2 CABAC Framework and CABAC for Residual Data Coding

The encoding process of CABAC consists of four 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 the 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. Alternatively, in the bypass coding mode, each binary value is directly encoded via the bypass coding engine without using an explicitly assigned model.

Fig. 2 illustrates the CABAC encoding structure for a 4×4 sub-block of the quantized transform coefficients. First, the coded block flag is transmitted for each sub-block if the coded block pattern or the macroblock mode indicates that the specific sub-block has non-zero coefficients. If the coded block flag is zero, no further information is transmitted and the encoding process is terminated for the current sub-block; otherwise, the significance map and level information are sequentially encoded.
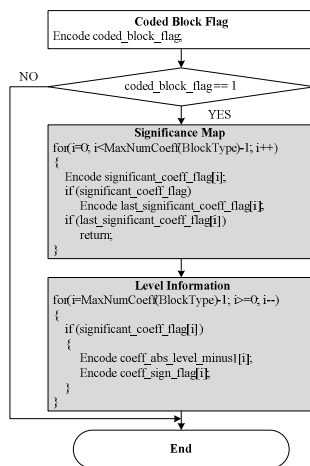

Fig. 2. Encoding structure of CABAC for residual data coding

Second, if *coded_block_flag* indicates that a sub-block has significant coefficients, a binary-valued significance map is encoded. For each coefficient, a 1-bit syntax element *significant_coeff_flag* is encoded in scanning order. If *significant_coeff_flag* is one, i.e., if a non-zero coefficient exists at this scanning position, a further 1-bit syntax element *last_significant_coeff_flag* is encoded. This syntax element states whether the current significant coefficient is the last coefficient or not.

After the encoded significance map determines the locations of all significant 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 Exponential Golomb* (UEG0) binarization method 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 CABAC Method for Lossless Coding

3.1   Analysis of the Statistical Properties of Residual Data in the Spatial Domain

In lossy coding, residual data represent the quantized transform coefficients in the frequency domain. The statistical characteristics of residual data in lossy coding are as follows. In a given sub-block, the probability of a non-zero coefficient existing is likely to decrease as the scanning position increases.

In lossless coding, however, residual data do not represent the quantized transform coefficients, but rather the differential pixel values between the original and predicted pixel values in the spatial domain. Therefore, the statistical characteristics of residual data in lossless coding are as follows. First, the probability of a non-zero differential pixel existing is independent of the scanning position, and the number of non-zero differential pixels is generally large, compared with the number of non-zero coefficients in the frequency domain.

Fig. 3 represents the probability distribution of non-zero residuals existing according to the scanning position. As expected, significant difference can be seen in the statistical characteristics between the residual data of the frequency (average value for the quantization parameter (QP) = 12, 24, and 36) and spatial domains.
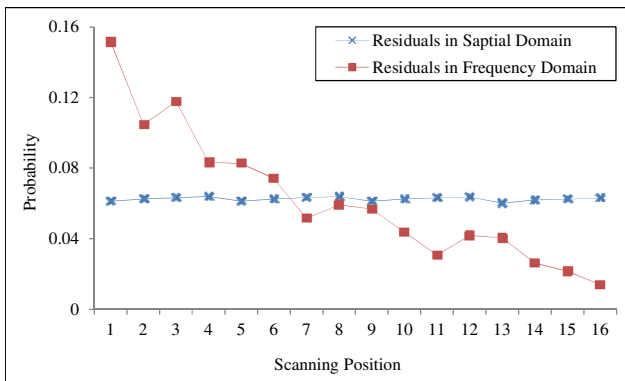

Fig. 3. Probability distribution of non-zero residuals according to the scanning position

Therefore, based on the above statistical characteristics of residual data in the spatial domain, we propose an efficient CABAC method for lossless coding in H.264/AVC by modifying the relevant coding parts of the conventional CABAC. In Fig. 2, the gray-shaded processes are modified in the proposed method for lossless

coding. The coding procedure of the proposed CABAC can be summarized in the following steps.

Step 1: Encode whether the current sub-block contains non-zero differential pixel values (*coded_block_flag*).

Step 2: Encode whether the differential pixel value at each scanning position is non-zero up to the last scanning position (*significant_diff_pixel_flag*).

Step 3: Encode the absolute value of a differential pixel minus 1 with the modified binarization and context modeling methods (*abs_diff_pixel_minus1*).

Step 4: Encode the sign of a differential pixel (*diff_pixel_sign_flag*).

3.2 Significance Map Coding

In lossy coding, the occurrence probability of a non-zero coefficient is likely to decrease as the scanning position increases because residual data are the quantized transform coefficients, as shown in Fig. 3. Therefore, the significant coefficient tends to be located at earlier scanning positions. In this case, *last_significant_coeff_flag* plays an important role in the early termination of significance map coding.

However, in lossless coding, since neither transform nor quantization is performed, the occurrence probability of a non-zero differential pixel is independent of the scanning position, as shown in Fig. 3. Thus, the last non-zero differential pixel is located at the end of the scanning position. From the extensive experiments on the location of the last non-zero residual in a sub-block, we observed that the average position of the last non-zero residual data in lossy and lossless coding are approximately 14.71 and 7.75 (average value for the quantization parameter (QP) = 12, 24, and 36), respectively. In this case, it is meaningless to encode *last_significant_coeff_flag* to indicate the position of the last significant differential pixel. Therefore, we remove the *last_significant_coeff_flag* coding process and directly encode *significant_diff_pixel_flag*s at all scanning positions from 1 to 16 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	8	-6	3	0	13	4	-9	1	0	11	-7	-2	5	-4	6	0
significant_coeff_flag	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	
last_significant_coeff_flag	0	0	0		0	0	0	0		0	0	0	0	0	1	

(a) Original method in lossy coding

Scanning position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Differential pixel value	8	-6	3	0	13	4	-9	1	0	11	-7	-2	5	-4	6	0
significant_diff_pixel_flag	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0

(b) Proposed method in lossless coding

Fig. 4. Example of significance map coding

3.3   Binarization for Differential Pixel Value

For the absolute value of the quantized transform coefficient (*abs_level*) in the frequency domain, the *Unary/k^{th} order Exponential Golomb* (UEGk) binarization method is applied. The design of the UEGk binarization method is motivated by the fact that the unary code is the simplest prefix-free code in terms of implementation cost and it permits the fast adaptation of individual symbol probabilities in the subsequent context modeling stage [7]. These observations are only accurate for small *abs_level*s; however, for larger *abs_level*s, adaptive modeling has limited functionality. Therefore, these observations have led to the idea of concatenating an adapted truncated unary (TU) code as a prefix and a static Exp-Golomb code [8] as a suffix.

The UEGk binarization of *abs_level* has a cutoff value S = 14 for the TU prefix and the order $k = 0$ for the *Exponential Golomb* (EG0) suffix. Previously, Golomb codes have been proven to be optimal prefix-free codes for geometrically distributed sources [9]. Moreover, EG0 is the optimal code for a *probability density function* (pdf) as follows:

$$p(x) = 1/2 \cdot (x+1)^{-2} \ \ with \ x \geq 0 \tag{1}$$

The statistical properties of the absolute value of the differential pixel (*abs_diff_pixel*) in the spatial domain are quite different from those of *abs_level* in the frequency domain, as shown in Fig. 5. The statistical distribution of *abs_level* in the frequency domain is highly skewed on small values; however, in the spatial domain, the statistical distribution of *abs_diff_pixel* is quite wide. Moreover, in the figure, we can also observe that the TU code is a fairly good model for the statistical distribution of *abs_level* in the frequency domain; whereas, it is not appropriate for the statistical distribution of *abs_diff_pixel* in the spatial domain.
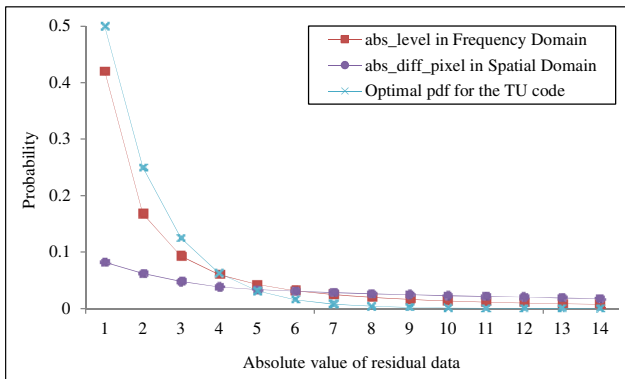
Fig. 5. Probability distribution of the absolute value and the optimal pdf of the TU code

In order to efficiently encode *abs_diff_pixel* in the spatial domain, we adjusted the cutoff value *S* of the TU prefix in UEG0 binarization. In Fig. 5, the optimal pdf curve for the TU code and the statistical distribution curve for *abs_diff_pixel* in the spatial domain intersect at an absolute value of 5. Moreover, as the absolute value increases,

the statistical difference between the TU code and *abs_diff_pixel* becomes larger. Based on this observation, we determined a new cutoff value $S = 5$ for the TU prefix in the proposed binarization method.

In order to provide a good prefix-free code for lossless coding, we also determined an appropriate parameter k for the EGk code. The prefix of the EGk codeword consists of a unary code corresponding to the value $l(x) = \lfloor \log_2(x/2^k + 1) \rfloor$. The suffix is then computed as the binary representation of $x + 2^k(1 - 2^{l(x)})$ using $k + l(x)$ significant bits. Consequently, for EGk binarization, the number of symbols having the same code length of $k + 2l(x) + 1$ grows geometrically. Then, by inverting Shannon's relationship between the ideal code length and the symbol probability, we can find each pdf corresponding to an EGk having an optimal code according to a parameter k.

$$p_k(x) = 1/2^{k+1} \cdot (x/2^k + 1)^{-2} \ \ with \ x \geq 0 \tag{2}$$

where $p_k(x)$ is the optimal pdf corresponding to the EGk code for a parameter k. This implies that for an appropriately chosen parameter k, the EGk code represents a fairly good prefix-free code for tails of typically observed pdfs.

Fig. 6 presents the probability distribution of $p_k(x)$ for $k = 0$, 1, 2, and 3 and the occurrence probability distribution of *abs_diff_pixel* from 6 to 20, where *abs_diff_pixel*s up to 5 are specified by the TU code. In the figure, the probability distribution of $p_k(x)$ for $k = 3$ is well matched to the occurrence probability distribution of *abs_diff_pixel* in the spatial domain. This result implies that the EG3 code represents a fairly good approximation of the ideal prefix-free code for encoding *abs_diff_pixel* in the spatial domain.
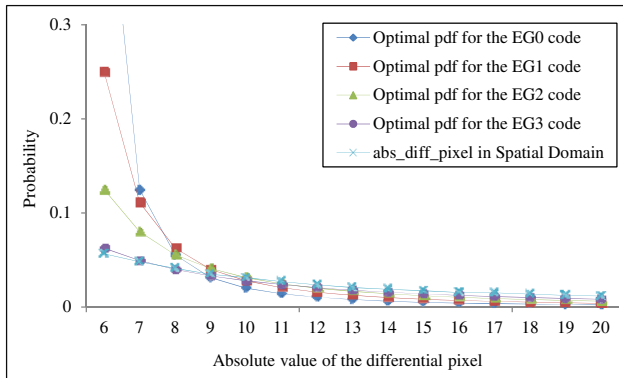
Fig. 6. Probability distribution of the optimal pdf of EGk code and *abs_diff_pixel*

Based on these observations, we designed an efficient binarization algorithm to encode *abs_diff_pixel* in the spatial domain. In the proposed algorithm, UEGk binarization of *abs_diff_pixel* is specified by new cutoff value $S = 5$ for the TU prefix and the order $k = 3$ for the EGk suffix.

4   Experimental Results and Analysis

In order to verify coding performance of the proposed method, we performed experiments on several test sequences for QCIF, CIF, and HD resolutions. We implemented our proposed method in the H.264/AVC reference software version JM 16.2 [10]. 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 is used)
    5) *ContextInitMethod* = 1 (adaptive)
    6) *LosslessCoding* = 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*: modify significance map coding
    2) *Method II*: *Method I* + modify binarization for differential pixel value coding

To verify efficiency of the proposed method, we performed two kinds of experiments. In the first experiment, six YUV420 format test sequences with QCIF, CIF, and HD resolutions were tested, as shown in Table 1. In the second experiment, we encoded only one frame (first frame) for each test sequence using our proposed method (*Method II*) and a well-known lossless coding technique, lossless joint photographic experts group (JPEG-LS) [11], used as a comparison for coding performance of our proposed method, as shown in Table 2.

Comparisons were made in terms of bit-rate percentage differences (Table 1) and compression ratio differences (Table 2) with respect to H.264/AVC CABAC and JPEG-LS, respectively. These changes were calculated as follows:

$$\Delta Saving\ Bits(\%) = \frac{Bitrate_{H.264/AVC} - Bitrate_{Method}}{Bitrate_{H.264/AVC}} \times 100 \tag{3}$$

$$Compression\ Ratio = \frac{Original\ image\ size}{Bitrate_{Method}} \tag{4}$$

In Table 1, we confirmed that the proposed method provided a better coding performance of approximately 18.9% bit saving via the QCIF, CIF, and HD resolution sequences, 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 displayed the better coding performance in lossless coding.

In general, since video sequences contain more redundancy in time than in space, the accuracy of inter prediction is better than that of intra prediction. Thus, there are significant statistical differences between lossless intra and lossless inter coding. In other words, it is not easy to determine the best CABAC method that can generally be used for lossless video coding (for both intra and inter coding). Therefore, in this paper, we focused on the improvement of an appropriate CABAC for lossless intra coding.

Table 1. Comparison of the Performance Measures

Sequence	Image size (bits)	Method	Encoding bits (bits)	Saving bits (%)
Foreman (QCIF) 100 frames	30412800	H.264/AVC CABAC	14344176	0
		Method I	12857928	10.361
		Method II	12572368	12.352
Container (QCIF) 100 frames	30412800	H.264/AVC CABAC	14482016	0
		Method I	13034368	9.996
		Method II	11913968	17.733
Mobile (CIF) 100 frames	121651200	H.264/AVC CABAC	91371512	0
		Method I	85034984	6.935
		Method II	68152408	25.412
Tempete (CIF) 100 frames	121651200	H.264/AVC CABAC	79063136	0
		Method I	72756080	7.977
		Method II	60830560	23.061
Crowdrun (HD) 100 frames	2488320000	H.264/AVC CABAC	1250235376	0
		Method I	1120777696	10.355
		Method II	1047171240	16.242
Parkjoy (HD) 100 frames	2488320000	H.264/AVC CABAC	1283550664	0
		Method I	1155350512	9.988
		Method II	1043186200	18.727
Average		H.264/AVC CABAC		0
		Method I		9.269
		Method II		18.921

Table 2. Comparison of the Performance Measures

Sequence	Method	Compression ratio
Foreman (QCIF)	JPEG-LS	1.8179
	Method II	2.4190
Container (QCIF)	JPEG-LS	1.9030
	Method II	2.5527
Mobile (CIF)	JPEG-LS	1.4865
	Method II	1.7850
Tempete (CIF)	JPEG-LS	1.6556
	Method II	1.9998
Crowdrun (HD)	JPEG-LS	1.6802
	Method II	2.3762
Parkjoy (HD)	JPEG-LS	1.8664
	Method II	2.3853
Average	JPEG-LS	1.7349
	Method II	2.2530

5   Conclusions

In this paper, we proposed an improved context-based adaptive binary arithmetic coding (CABAC) method for H.264/AVC lossless residual coding in the spatial domain. Considering the statistical differences in residual data between the frequency and spatial domains, we modified the CABAC method based on the observed statistical characteristics of residual data in the spatial domain. Experimental results show that the proposed method provided an approximately 19% bit saving, compared to the H.264/AVC CABAC.

References

1. Wiegand, T., Sullivan, G.J., Bjontegaard, G., Luthra, A.: Overview of the H.264/AVC video coding standard. IEEE Transactions on Circuits and Systems for Video Technology 13(7), 560–576 (2003)
2. Sullivan, G.J., Wiegand, T.: Video compression–from concepts to the H.264/AVC standard. Proc. IEEE 93(1), 18–31 (2005)
3. Sullivan, G.J., Topiwala, P., Luthra, A.: The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions. In: Proc. SPIE Conf., Special Session Adv. New Emerg. Standard: H.264/AVC, pp. 454–474 (2004)
4. Lee, Y.L., Han, K.H., Sullivan, G.J.: Improved lossless intra coding for H.264/MPEG-4 AVC. IEEE Transactions on Image Processing 15(9), 2610–2615 (2006)
5. Joint Video Team of ISO/IEC JTC1/SC29/WG11 and ITU-T Q.6/SG16, Lossless Intra Coding for Improved 4:4:4 Coding in H.264/MPEG-4 AVC, document JVT-P016.doc (2005)
6. Heo, J., Kim, S.H., Ho, Y.S.: Improved CAVLC for H.264/AVC lossless intra-coding. IEEE Transactions on Circuits and Systems for Video Technology 20(2), 213–222 (2010)
7. Marpe, D., Schwarz, H., Wiegand, T.: Context-based adaptive binary arithmetic coding in the H.264/AVC video compression. IEEE Transactions on Circuits and Systems for Video Technology 13(7), 620–636 (2003)
8. Teuhola, J.: A compression method for clustered bit-vectors. Information Processing Letters 7, 308–311 (1978)
9. Gallager, R.G., Van Voorhis, D.C.: Optimal source codes for geometrically distributed integer alphabets. IEEE Transactions on Information theory 21(2), 228–230 (1975)
10. Joint Video Team, Reference Software Version 16.2, `http://iphome/hhi.de/shehring/tml/download/old_jm/jm16.2.zip`
11. Weinberger, M.J., Seroussi, G., Sapiro, G.: The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS. IEEE Transactions on Image Processing 9(8), 1309–1324 (2000)