SIViP (2015) 9:1055–1066
DOI 10.1007/s11760-013-0545-z

ORIGINAL PAPER

Efficient residual data coding in CABAC for HEVC lossless video compression

Jung-Ah Choi · Yo-Sung Ho

Received: 21 November 2012 / Revised: 4 June 2013 / Accepted: 26 July 2013 / Published online: 20 September 2013

Abstract After the development of the next generation video coding standard, referred to as high efficiency video coding (HEVC), the joint collaborative team of the ITU-T video coding experts group and the ISO/IEC moving picture experts group has now also standardized a lossless extension of such a standard. HEVC was originally designed for lossy video compression, thus, not ideal for lossless video compression. In this paper, we propose an efficient residual data coding method for HEVC lossless video compression. Based on the fact that there are statistical differences of residual data between lossy and lossless coding, we improved the HEVC lossless coding using sample-based angular prediction (SAP), modified level binarization, and binarization table selection with the weighted sum of previously encoded level values. Experimental results show that the proposed method provides high compression ratio up to 11.32 and reduces decoding complexity.

Keywords High efficiency video coding (HEVC) · Binarization · Binarization table selection · Level coding · Lossless coding

1 Introduction

High efficiency video coding (HEVC) [1] is a new video coding standard developed by joint collaborative team on video coding (JCT-VC) of ITU-T video coding experts group (VCEG) and ISO/IEC moving picture experts group (MPEG). Currently, most of coding techniques are established and HEVC version 1 is released in January 2013 [2]. We expect that HEVC will be widely used in various applications for recording, compression, and distribution of high-resolution video contents [3].

Compression schemes can be divided into two major classes: lossy compression and lossless compression. Lossy compression is most commonly used to compress multimedia data that needs to be stored or transmitted. By contrast, lossless compression is useful when it is necessary to minimize the storage space or transmission bandwidth of data while still maintaining archival quality. In lossless coding, no distortion is allowed in reconstructed frames. Many applications such as medical imaging, preservation of artwork, image archiving, remote sensing, and image analysis require the use of lossless compression [4].

In the past two decades, capacity of mass storage devices has increased and display devices are of such quality that very little distortion can be tolerated. It results in a resurgence of interest in lossless compression. With growing demand of users, JCT-VC included the lossless compression part in the HEVC standard in consequence of the Ad Hoc group for lossless coding [5]. Figure 1 shows the HEVC lossless coding framework. To achieve lossless coding, transform, quantization, and their inverse operations are bypassed because they are not reversible in general [6]. Also, all in-loop filtering operations including deblocking filter and sample adaptive offset (SAO) are bypassed in the encoder and the decoder because there is no error occurred in lossless compression. As a result, remaining parts are prediction and entropy coding.

Before anything else, JCT-VC focused on improving prediction accuracy for lossless coding. In the 7th JCT-VC

J.-A. Choi (✉) · Y.-S. Ho
Gwangju Institute of Science and Technology (GIST),
123 Cheomdan-gwagiro, Buk-gu, Gwangju 500-712,
Republic of Korea
e-mail: jachoi@gist.ac.kr

Y.-S. Ho
e-mail: hoyo@gist.ac.kr

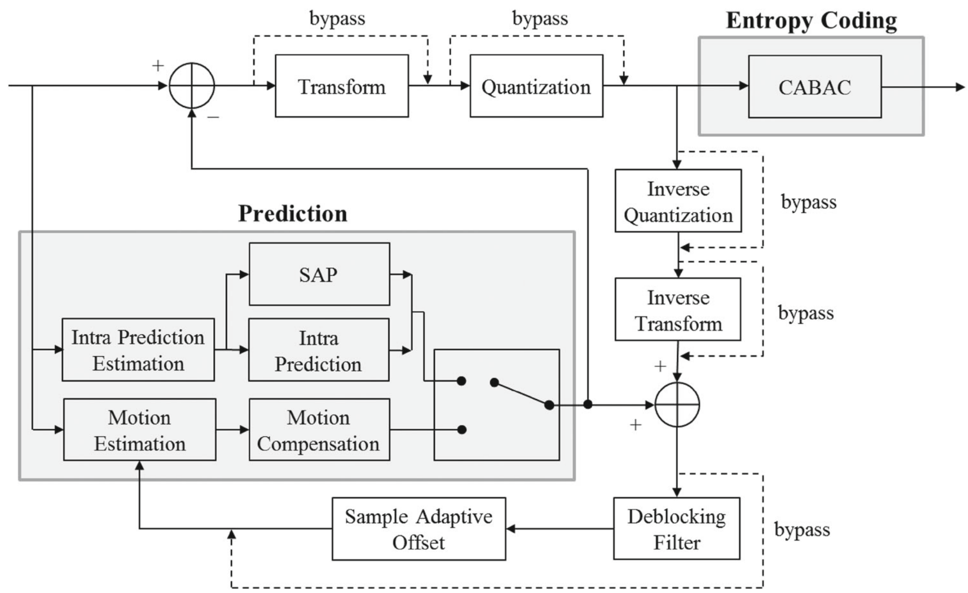

Fig. 1 The HEVC lossless coding framework

meeting, sample-based angular prediction (SAP) [7,8] is suggested to replace the existing block-based intra-prediction method. Since joint proposal that combines SAP and the lossless coding signaling method [5] in the 8th JCT-VC meeting, efforts to find the efficient entropy coding method are continued. Specifically, entropy coding method for coding prediction residuals attracts attention. A residual data coding method for lossless coding [9] and an improved binarization scheme for intra-luma prediction residual data [10] were introduced. Since the development of HEVC lossless coding mode is not yet finished, many experts are actively researching efficient algorithms for lossless coding.

In this paper, we have tried to improve the performance of entropy coding for HEVC lossless compression. Since HEVC entropy coder has been developed by mainly focusing on lossy coding to date, it cannot show outstanding coding performance for lossless coding. Thus, we analyze the statistical characteristics of residual data in lossless coding, and given the statistics, we design an improved entropy coding method. Note that the proposed method does not require significant modifications of HEVC encoding and decoding processes, so it can be easily applied to the conventional standard.

The paper is organized as follows. In Sect. 2, we briefly present an overview of core techniques in HEVC lossless coding. In Sect. 3, after we analyze characteristics of residual data in lossless coding, our proposed algorithm is explained in detail. In Sect. 4, the performance of the proposed algorithm is compared to that of HEVC lossless compression and other recently proposed algorithms. Finally, conclusions are presented in Sect. 5.

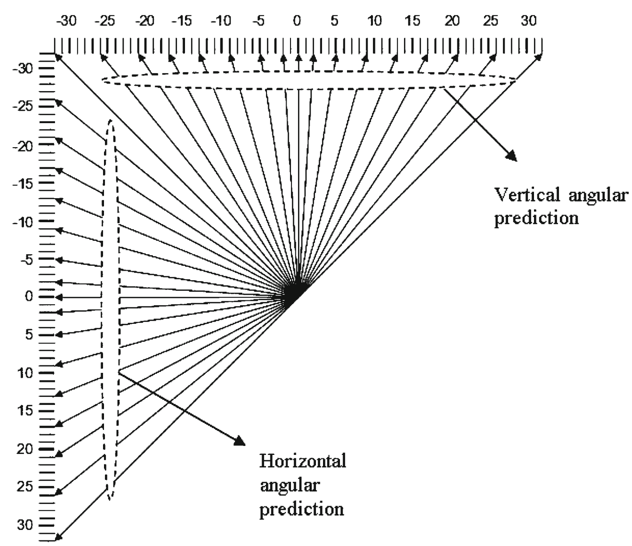

Fig. 2 Intra-prediction angles (*vertical* and *horizontal* angular prediction)

2 Overview of HEVC lossless coding

2.1 Sample-based angular prediction

In order to explore spatial sample redundancy of intra-coded frame in lossless coding, SAP is suggested instead of general HEVC intra-prediction. Unlike H.264/AVC intra-coding [11], HEVC employs 35 intra-prediction modes including DC and planar modes. As shown in Fig. 2, 33 angles are defined and these angles are categorized into two classes: vertical and horizontal angular prediction. Each prediction has both negative and positive angles.

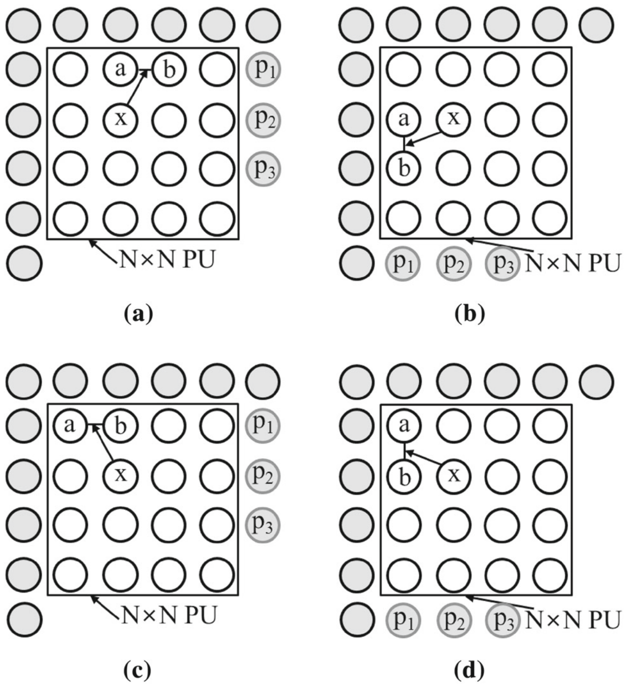

Fig. 3 Intra-prediction angles (*vertical* and *horizontal* angular prediction). **a** Vertical sample-based angular prediction with positive angle. **b** Horizontal sample-based angular prediction with positive angle. **c** Vertical sample-based angular prediction with negative angle. **d** Horizontal sample-based angular prediction with negative angle

In lossless coding, reference samples within the current prediction unit (PU) as well as neighboring samples of the current PU are available. Thus, prediction can be performed sample by sample to achieve better intra-prediction accuracy. All samples within a PU use a same prediction angle, and the signaling method of the prediction angle is exactly same as that in lossy intra-coding.

In SAP, samples in a PU are processed in predefined orders. The raster scanning and vertical scanning processing order is applied to vertical and horizontal angular prediction, respectively. In addition, reference samples around right and bottom PU boundaries of the current PU are padded from the closest boundary samples of the current PU.

Figure 3 presents the reference sample locations *a* and *b* relative to the current sample **x** to be predicted for horizontal and vertical angular prediction with negative and positive prediction angles. At most two reference samples are selected for each sample to be predicted in the current PU. Depending on the current sample location and the selected prediction angle, reference sample **a** and **b** can be neighboring PUs, padded samples, or samples inside the current PU. The interpolation for prediction sample generation is exactly same as that in lossy coding.

Table 1 CABAC syntax elements for residual data coding

last_significant_coeff_x_prefix
last_significant_coeff_y_prefix
last_significant_coeff_x_suffix
last_significant_coeff_y_suffix
significant_coeff_group_flag
significant_coeff_ flag
coeff_abs_level_greater1_flag
coeff_abs_level_greater2_flag
coeff_sign_flag
coeff_abs_level_remaining

2.2 CABAC entropy coder for residual data coding

As an entropy coder, HEVC employed context-based adaptive binary arithmetic coding (CABAC). The usage of arithmetic coding allows the assignment of a non-integer number of bits to each symbol, which is extremely beneficial for symbol probabilities that are greater than 0.5 [12]. CABAC typically provides better compression efficiency than variable length coding (VLC)-based entropy coders such as low complexity entropy coding (LCEC) [13] and context-based variable length coding (CAVLC) [14].

The specific syntax elements for residual data coding are shown in Table 1. The gray-shaded syntax elements are encoded in transform unit (TU) level and others are encoded in 4×4 subblock level.

First, the position of the last significant coefficient within a TU is coded by four syntax elements: *last_significant_coeff_x _prefix*, *last_significant_coeff_y_prefix*, *last_significant _coeff_x_suffix*, and *last_significant_coeff_y_suffix*. Since TU may take a size from 4×4 to 32×32, it is more efficient to encode the location of the last significant coefficient in a TU using the column and the row position.

Then, *significant_coeff_group_flag* is coded as a subblock unit. A subblock consists of 16 coefficients, i.e., a 4×4 subset. For example, a 16×16 TU is divided into 16 subblocks. When a subblock has one or more nonzero quantized transform coefficients, *significant_coeff_group_flag* is equal to 1. Otherwise, *significant_coeff_group_flag* is 0. When, *significant_coeff_group_flag* is equal to 1, following five syntax elements are signaled to represent the coefficients level information within the subblock.

(a) *significant_coeff_flag*: indicates whether the coefficient is nonzero or not.
(b) *coeff_abs_level_greater1_flag*: indicates whether the coefficient amplitude is larger than one for each nonzero coefficient (i.e., with significant_coeff_flag as 1).

(c) *coeff_abs_level_greater2_flag*: indicates whether the coefficient amplitude is larger than two for each coefficient with amplitude larger than one (i.e., with coeff_abs_level_greater1_flag as 1).
(d) *coeff_sign_flag*: indicates sign information of the nonzero coefficients.
(e) *coeff_abs_level_remaining*: indicates remaining absolute level value.

These syntax elements were designed to take advantage of several characteristics of residual data in lossy coding: after transform and quantization, high-frequency regions of subblocks typically contain small coefficients and the level of nonzero coefficients tends to be larger toward the low-frequency regions. Therefore, taking into consideration the above characteristics, above syntax elements are encoded in reverse order.

The absolute value of the transform coefficient levels are binarized using a concatenation of a truncated unary code, a truncated Golomb–Rice code, and a 0th order Exp-Golomb code as illustrated in Fig. 4. Two flags, *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* are coded using the truncated unary code.

In HEVC, *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* are only encoded for a few starting nonzero coefficients. The coefficient level information is coded in inverse scan order, which means the high-frequency components of transformed coefficients are scanned first. In such design, the amplitude of the first few coefficients in the reverse scan order tends to be small. Furthermore, the absolute level, s, is 1 or 2. For this case, it is beneficial to use flags *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* to reduce the length of binarized bins, and these flags can be efficiently encoded with well-adapted context model.

However, after encountering a certain number of greater than 1 coefficients, due to reverse scanning, it is much likely that the remaining significant coefficients also have high magnitudes. In such case, using *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* is not likely to improve the compression performance. Thus, in the current HEVC standard, *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* are encoded only for the first 8 and 1 nonzero coefficients in a subblock, respectively.

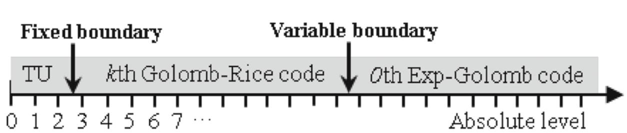

Fig. 4 The binarization method for the absolute value of the transform coefficient level in HEVC

For the remaining absolute value of a coefficient level *coeff_abs_level_remaining* is binarized by a concatenation of the truncated Golomb–Rice code and 0th order Exp-Golomb code [15]. First, the Golomb–Rice code is constructed as follows. Given a particular Rice parameter k, an absolute transform coefficient s consists of prefix part p and a remainder part r. The prefix is coded using a truncated unary code and the remainder is coded using a fixed length code. The length of the Golomb–Rice code is $k + 1 + p$.

$$p = \left\lfloor \frac{s}{2^k} \right\rfloor \tag{1}$$

$$r = s - p \cdot 2^k \tag{2}$$

In the current HEVC standard, the range of the Rice parameter k is from 0 to 4. Thus, five different binarization tables exist according to the Rice parameter k, as shown in Table 2.

When an absolute transform coefficient is greater than a certain cutoff value, the 0th order Exp-Golomb code is added. The cutoff value is defined according to the Rice parameter k, as shown in Table 3.

The initial value of the Rice parameter k is always 0. Selection of the Rice parameter depends on the current Rice parameter and the coded value of *coeff_abs_level_remaining*, s_i, as shown in Table 4.

Figure 5 shows an example of residual data coding in a subblock. Note that the value of *coeff_abs_level_remaining* is adjusted appropriately for the values of *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag*.

3 Proposed residual data coding for lossless compression

As shown in Fig. 1, HEVC and H.264/AVC have similar structures including hybrid video coding schemes with spatial/temporal prediction and entropy coding. However, HEVC achieves significant compression ratio improvement compared to H.264/AVC, since various coding tools are refined. In intra-prediction, HEVC employs more prediction modes up to 35 modes and flexible prediction units from 4×4 to 64×64.

Cai et al. [16] compared lossless intra-coding performances of several video coding methods such as HEVC lossless coding, H.264/AVC lossless coding [17], JPEG2000 [18], and JPEG-LS [19]. They encoded YUV 4:2:0 test sequences used in the standardization process of HEVC and compared compression performances. The lossless coding results are listed and compared in Table 5. We check the best performance in bold type. It is concluded that the lossless intra-coding performance of HEVC has matched that of H.264/AVC, although the results are in favor of JPEG-LS for most sequences.

Since the standardization process of the HEVC extension is not completed, the algorithm design and implementation of

Table 2 Five different binarization tables of the truncated Golomb–Rice code

s	Codeword				
	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 4$
0	0·	0 · 0	0·00	0·000	0·0000
1	10·	0·1	0·01	0·001	0·0001
2	110·	10·0	0·10	0·010	0·0010
3	1110·	10·1	0·11	0·011	0·0011
4	11110·	110·0	10·00	0·100	0·0100
5	111110·	110·1	10·01	0·101	0·0101
6	1111110·	1110·0	10·10	0·110	0·0110
7	11111110·	1110·1	10·11	0·111	0·0111
8	111111110·	11110·0	110·00	10·000	0·1000
9	1111111110·	11110·1	110·01	10·001	0·1001
10	11111111110·	111110·0	110·10	10·010	0·1010
11	111111111110·	111110·1	110·11	10·011	0·1011
12	1111111111110·	1111110·0	1110·00	10·100	0·1100
13	11111111111110·	1111110·1	1110·01	10·101	0·1101
14	111111111111110·	11111110·0	1110·10	10·110	0·1110
15	1111111111111110·	11111110·1	1110·11	10·111	0·1111

Table 3 Defined cutoff values

k	Cutoff value
0	7
1	14
2	26
3	46
4	78

Table 4 Threshold for determining whether to increase Rice parameter

k	Threshold $T(s_i)$ to increase k
0	3
1	6
2	12
3	23
4	>24

lossless coding are not finished. Among proposals for lossless coding, one of very efficient algorithms is SAP [7,8], while the acceptance of it is still under consideration in the standardization meeting. In this paper, we take the SAP technique into consideration together to improve HEVC lossless coding.

Figure 6a, b shows distributions of absolute levels in various coding environments according to the scanning position. In Fig. 6a, we compared distributions of absolute levels in HEVC lossless coding and HEVC lossy coding with various quantization parameters (QPs). Figure 6b shows distributions of absolute levels in H.264/AVC lossless coding, HEVC lossless coding, and SAP-based HEVC lossless coding.

From Fig. 6a, b, we can observe following features: (1) In Fig. 6a, the statistical characteristics of absolute levels in lossless coding become quite different from those in lossy coding. In lossy coding, absolute level is quantized transform coefficient. Hence, absolute level tends to decrease as the scanning position increases. In lossless coding, absolute level is not quantized transform coefficient, but the differential pixel value between the original and predicted pixel value. Since neither transform nor quantization is performed, absolute level in lossless coding is independent of the scanning position. Thus, it is necessary to modify the conventional residual data coding process in order to efficiently encode differential pixel values. (2) From the comparison, the absolute level distribution of HEVC lossless coding is quite different from that of H.264/AVC lossless coding. As shown in Fig. 6b, the magnitude of absolute level in HEVC lossless coding is lower than that in H.264/AVC lossless coding. Since HEVC employs more accurate prediction, the amount of the residual data becomes smaller. Therefore, we conclude that we need to design a new entropy coding method for residual data in HEVC lossless coding. (3) The magnitude of absolute level in SAP-based HEVC lossless coding is smaller than that in HEVC lossless coding. This indicates that SAP results in better prediction performance than the conventional intra-prediction of HEVC.

18	6	-6	-1
-12	4	-4	0
7	4	2	1
2	4	-1	0

4×4 sub-block

16	14	11	7
15	12	8	4
13	9	5	2
10	6	3	1

Entropy coding order

Coefficient	0	1	-1	0	2	4	-1	-4	4	2	-6	4	7	6	-12	18
significant_coeff_flag	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
coeff_abs_level_greater1_flag		0	0		1	1	0	1	1	1						
coeff_abs_level_greater2_flag					0											
coeff_sign_flag		0	1		0	0	1	1	0	0	1	0	0	0	1	0
coeff_abs_level_remaining						2		2	2	0	5	3	6	5	11	17

Fig. 5 An example of residual data coding in HEVC CABAC

Table 5 Compression ratio comparison of various lossless coding methods

Class	Sequence	HEVC	H.264/AVC	JPEG2000	JPEG-LS
A	Traffic	2.16	2.28	2.45	**2.56**
	PeopleOnStreet	2.21	2.29	2.50	**2.70**
B	Kimono	2.32	2.35	2.58	**2.59**
	ParkScene	1.96	1.93	2.17	**2.20**
	Cactus	1.94	1.85	2.03	**2.06**
	BQTerrace	1.91	1.77	1.98	**2.00**
	BasketballDrive	2.19	2.20	2.26	**2.29**
C	RaceHorses	2.00	1.82	2.20	**2.27**
	BQMall	2.08	2.05	2.16	**2.26**
	PartyScene	1.65	1.39	1.72	**1.79**
	BasketballDrill	2.10	1.91	2.06	**2.18**
D	RaceHorses	1.86	1.67	2.07	**2.14**
	BQSquare	1.69	1.42	1.72	**1.83**
	BlowingBubbles	1.59	1.33	1.67	**1.74**
	BasketballPass	2.15	2.18	2.33	**2.53**
E	FourPeople	2.55	2.64	2.82	**3.00**
	Johnny	2.78	2.85	2.99	**3.12**
	KristenAndSara	2.79	2.86	3.00	**3.19**
F	BasketballDrillText	2.14	1.95	2.07	**2.21**
	ChinaSpeed	3.00	3.23	2.85	**3.31**
	SlideEditing	3.73	3.80	2.95	**4.03**
	SlideShow	9.25	9.93	8.79	**10.45**

In order to more accurately reflect the above-mentioned statistical characteristics of residual data, we propose more efficient entropy coding schemes for lossless video coding by modifying the conventional CABAC scheme in HEVC. The proposed method consists of two parts: modified level binarization and modified binary table selection. The details of the proposed method are explained in following sub-sections.

3.1 Modified level binarization

As explained in Sect. 2, syntax elements *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* are used in level coding. Except bits for *coeff_abs_level_remaining* information, maximum two bits are always required to encode the level which is larger than two. In this

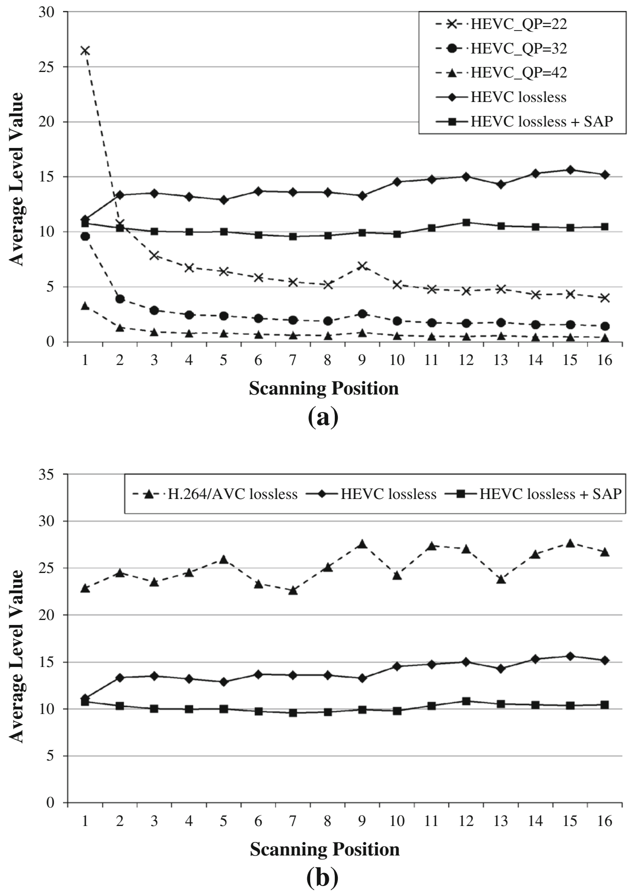

Fig. 6 Comparison of absolute level distribution (SlideEditing, 1280 × 720). **a** Magnitude of absolute level in HEVC lossy coding and HEVC lossless coding. **b** Magnitude of absolute level in H.264/AVC lossless coding, HEVC lossless coding, and SAP-based HEVC lossless coding

case, the bit length for the absolute level s with the Rice parameter k is as follows.

$$l(s,k) = \begin{cases} s, & s < 3 \\ 2 + \left\lfloor \frac{(s-3)}{2^k} \right\rfloor + 1 + k, & s \geq 3 \end{cases} \quad (3)$$

Here, '$\left\lfloor \frac{(s-3)}{2^k} \right\rfloor + 1 + k$' represents the length of the codeword for *coeff_abs_level_remaining* and '2' represents the length of the codeword for *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag*. If we do not use syntax elements *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* and directly encode *coeff_abs_level_remaining* using the Golomb–Rice code, the length for the coefficient s with the Rice parameter k becomes as follows.

$$l_p(s,k) = \left\lfloor \frac{(s-1)}{2^k} \right\rfloor + 1 + k, \quad \forall s \quad (4)$$

For low-magnitude coefficient s such as $s < 3$, $l(s, k)$ is same or shorter than $l_p(s, k)$. However, for $s \geq 3$, $l_p(s, k)$ is always shorter than $l(s, k)$. It means that two flags *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag* do not give codeword length compaction for high-magnitude coefficients ($s \geq 3$). In lossless coding, a lot of high-magnitude coefficients can occur, as well as low-magnitude coefficients.

In Fig. 7, we illustrated the histogram of the residual data of the *SlideEditing* test sequence in lossy and lossless coding, respectively [20]. In lossy coding, quantization parameter (QP) is equal to 27. The range of the absolute value of a nonzero coefficient in lossless coding is generally wide compared to those in lossy coding. From Fig. 7, we can observe that large residual data appears with relatively higher frequency in lossless coding, compared to lossy coding. Consequently, average magnitude of residual data in lossless coding is higher than that in lossy coding.

Therefore, it is more efficient to represent coefficients using the Golomb–Rice code directly. Hence, we introduce a new level coding method. Figure 8 describes the flowchart of the proposed level coding.

In the proposed method, we do not encode *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag*. That is, instead of using the truncated unary binarization, level is directly encoded by the truncated Golomb–Rice binarization, as illustrated Fig. 9. Compared to Fig. 4, the truncated unary binarization part is removed in Fig. 9.

3.2 Modified binarization table selection

In level coding, the absolute level of each nonzero coefficient is adaptively encoded in reverse scan order by the selected binarization table from Rice parameter k. Each binarization table is designed to encode efficiently in a specified range of *coeff_abs_level_remaining*, s_i, as described in Table 4. As previously mentioned, selection of the binarization table for CABAC level coding is based on the expectation that the absolute level within a subblock is likely to increase at low frequencies. Hence, the Rice parameter k for selection of the binarization table monotonically increases according to the previously encoded absolute level. It was designed to take advantage of the characteristics of residual data in lossy coding: after transform and quantization, subblocks typically contain many zeros, especially in high-frequency regions and the level of residual data tends to be larger toward the low-frequency regions.

However, the absolute level in lossless coding is independent of the scanning position within the subblock, as shown in Fig. 6b. That is because coefficient levels are not quantized transform coefficients, but the differential pixel values between the original and predicted pixel values in lossless coding. Thus, the next level does not necessarily increase at lower frequencies; we cannot assume that the next level is larger than the current level.

Therefore, we designed an adaptive method for binarization table selection that can decrease or increase according to

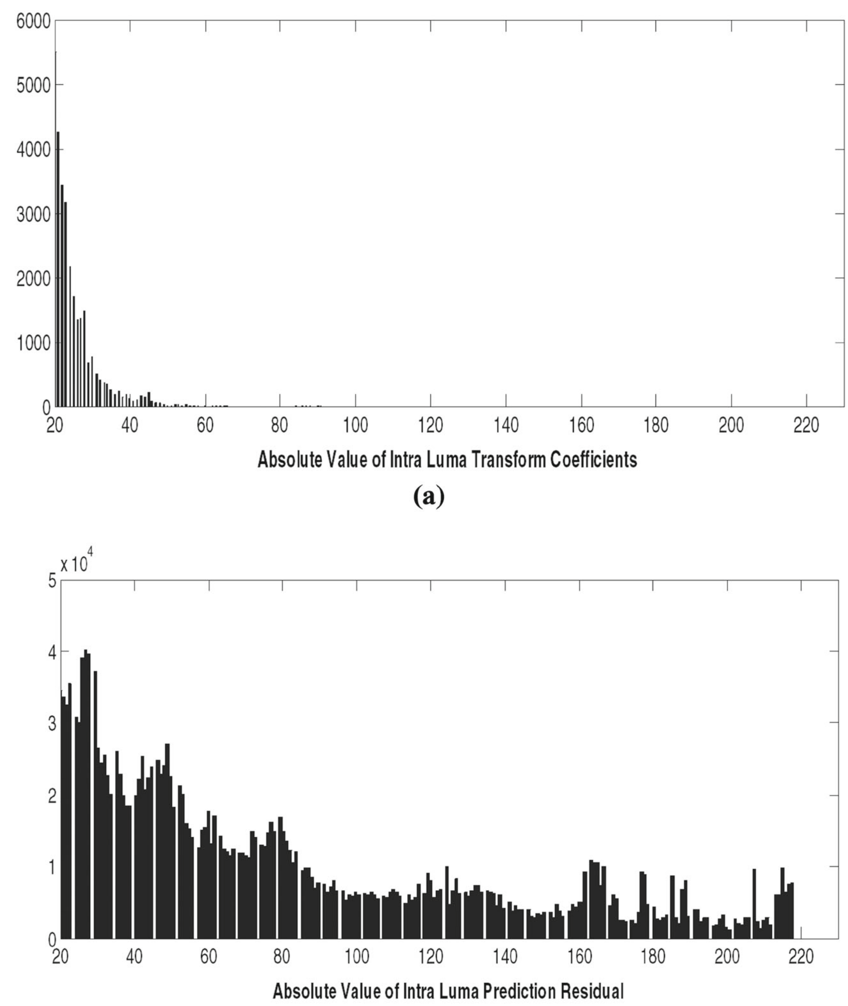

Fig. 7 The histogram of residual data in HEVC lossy (**a**) and lossless (**b**) coding

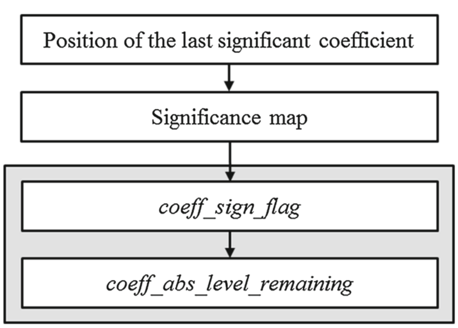

Fig. 8 The flowchart of the proposed method

the previously encoded absolute level. This method was used to predict the accurate variable length coding (VLC) table in context-based adaptive variable length coding (CAVLC) for H.264/AVC lossless coding [21]. Applying this scheme to CABAC binarization table selection, we can increase correctness of binarization table selection considering the structural characteristics in lossless coding.

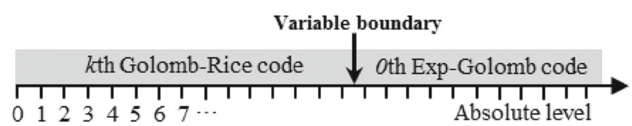

Fig. 9 The proposed binarization method for the absolute level for HEVC lossless coding

The basic idea for this concept is that the table for the next level can be determined using the weighted sum of the previously encoded level. The decision procedure for determining the level binarization table is described in Eqs. (5)–(7).

$$T(s_i) = \frac{1}{w_i + 1}\{w_i \cdot avg_i + s_i\} \tag{5}$$

$$\text{avg}_i = \frac{1}{(lastcoeff - i + 1)}\left\{\sum_{k=lastcoeff}^{i} s_k\right\} \tag{6}$$

$$w_i = \begin{cases} a, & i = lastcoeff \\ a+1, & i = lastcoeff - 1,\ lastcoeff - 2 \\ a+2, & \text{otherwise} \end{cases} \tag{7}$$

Table 6 Test sequences

Class	Sequence	Resolution	Total frame	Frame rate
A	Traffic	2560 × 1600	150	30
	PeopleOnStreet		150	30
B	Kimono	1920 × 1080	240	24
	ParkScene		240	24
	Cactus		500	50
	BQTerrace		600	60
	BasketballDrive		500	50
C	RaceHorses	832 × 480	300	30
	BQMall		600	60
	PartyScene		500	50
	BasketballDrill		500	50
D	RaceHorses	416 × 240	300	30
	BQSquare		600	60
	BlowingBubbles		500	50
	BasketballPass		500	50
E	FourPeople	1280 × 720	600	60
	Johnny		600	60
	KristenAndSara		600	60
F	BasketballDrillText	832 × 480	500	50
	ChinaSpeed	1024 × 768	500	30
	SlideEditing	1280 × 720	300	30
	SlideShow	1280 × 720	500	20

where w_i and s_i are the weighting coefficient and the absolute level value, respectively, where both values are related to the current scanning position i. $T(s_i)$ represents the threshold value for selecting the corresponding binarization table used to encode the next level (= $(i-1)$th absolute level, s_{i-1}). Note that absolute level s is encoded in reverse order. In Eqs. (6) and (7), *lastcoeff* represents the scanning position number of the last nonzero coefficient. In Eq. (7), a is a threshold value for the weighting coefficient, and it is determined empirically through simulations. The proposed binarization method can reflect the characteristics of absolute levels in lossless coding, since k can be increased or decreased.

Table 7 Encoding parameters

Parameter	Value	Description
CUWidth	64	LCU width
CUHeight	64	LCU height
QP	0	Lossless coding
InternalBitDepth	8	8 bit per pixel
LosslessCuEnabled	1	Lossless coding
LoopFilterDisable	1	No deblocking filter
SAO	0	No sample adaptive offset

4 Experimental results and analysis

In order to verify the performance of the proposed method, we implemented the proposed method in HEVC test model (HM) 8.0 [22]. In experiments, we used common test sequences and common test conditions [23] of the HEVC standard. The test sequences are grouped into six classes and details are presented in Table 6. Table 7 shows the encoding parameters for the reference software.

In our experiment, encoding results of HEVC lossless coding, H.264/AVC lossless coding, JPEG2000, JPEG-LS, SAP-based HEVC lossless coding, and improved HEVC lossless coding using SAP and proposed residual data coding were compared. The compression performance is measured in terms of compression ratio and it is calculated as follows.

$$\text{Compression ratio} = \frac{\text{Original file size}}{\text{Compressed file size}} \tag{8}$$

The lossless coding results are listed in Table 8. Also, we checked the best performance in bold type. It can be seen that improved HEVC lossless coding using SAP and proposed residual data coding ('SAP + Proposed Method' in Table 7) gives about 3.01 compression ratio on average and 11.32 compression ratio at maximum compared to the

Table 8 Compression ratio comparison of various lossless coding methods

Class	Sequence	HEVC	H.264 /AVC	JPEG 2000	JPEG-LS	SAP [7,8]	SAP + proposed method
A	Traffic	2.16	2.28	2.45	**2.56**	2.43	2.50
	PeopleOnStreet	2.21	2.29	2.50	**2.70**	2.45	2.53
B	Kimono	2.32	2.35	2.58	**2.59**	2.50	2.55
	ParkScene	1.96	1.93	2.17	**2.20**	2.08	2.10
	Cactus	1.94	1.85	2.03	**2.06**	2.02	2.02
	BQTerrace	1.91	1.77	1.98	2.00	2.00	**2.02**
	BasketballDrive	2.19	2.20	2.26	2.29	2.36	**2.38**
C	RaceHorses	2.00	1.82	2.20	**2.27**	2.16	2.21
	BQMall	2.08	2.05	2.16	**2.26**	2.21	2.24
	PartyScene	1.65	1.39	1.72	1.79	1.74	**2.90**
	BasketballDrill	2.10	1.91	2.06	2.18	2.27	**2.31**
D	RaceHorses	1.86	1.67	2.07	**2.14**	2.05	2.09
	BQSquare	1.69	1.42	1.72	**1.83**	1.78	1.81
	BlowingBubbles	1.59	1.33	1.67	1.74	1.74	**1.75**
	BasketballPass	2.15	2.18	2.33	**2.53**	2.42	2.48
E	FourPeople	2.55	2.64	2.82	**3.00**	2.87	2.93
	Johnny	2.78	2.85	2.99	3.12	3.09	**3.15**
	KristenAndSara	2.79	2.86	3.00	**3.19**	3.12	3.18
F	BasketballDrillText	2.14	1.95	2.07	2.21	2.33	**2.38**
	ChinaSpeed	3.00	3.23	2.85	3.31	3.59	**4.16**
	SlideEditing	3.73	3.80	2.95	4.03	4.14	**5.21**
	SlideShow	9.25	9.93	8.79	10.45	10.15	**11.32**
Average		2.55	2.53	2.61	2.84	2.80	**3.01**

HEVC lossless mode. From experimental results, we verified that the proposed method shows better compression performance than HEVC lossless coding, H.264/AVC lossless coding, JPEG2000, and SAP for all test sequences. Moreover, SAP and the proposed method highly improve the compression performance of conventional HEVC lossless coding, and it provides better compression performance than JPEG-LS for some test sequences.

Figure 10 shows compression ratio of various lossless coding methods for each class of test sequences. This can be obtained by averaging the compression ratio with the same resolution. Although core algorithms of various lossless coding methods are different, there still exist some common features. For class F, the proposed method outperforms other lossless coding methods. Class F consists of screen contents which represent images or videos rendered by electronic devices such as computers or mobile phones. Compared to natural images, signals of screen contents changed much. The examples of screen contents are a computer screen with characters, graphics, webpage scrolling, video playing, horizontal and vertical lines, as well as video game content. Since the proposed method can properly detect the changed characteristics by estimating suitable binarization table well.

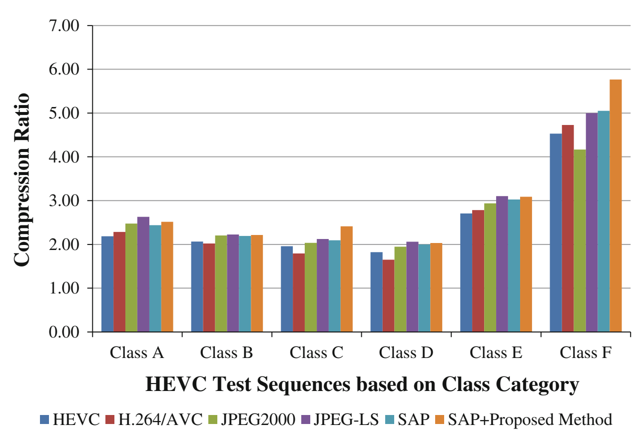

Fig. 10 Compression ratio of HEVC, H.264/AVC, JPEG2000, JPEG-LS, SAP, and improved HEVC lossless coding for intra-only configuration

Except class F, higher-resolution sequences tend to achieve better compression performance.

In Table 9, the decoding time-saving results for all test sequences are shown. Using SAP, the amount of the residual data is decreased. Accordingly, the decoding time is decreased. Although the proposed method has some

Table 9 The decoding time saving of the proposed method in terms of HEVC lossless coding

Class	Sequence	SAP [7,8] (%)	SAP + proposed method (%)
A	Traffic	−4.86	−4.60
	PeopleOnStreet	−4.00	−3.98
B	Kimono	−2.79	−2.97
	ParkScene	−0.36	−0.35
	Cactus	0.02	−0.30
	BQTerrace	−0.76	−1.24
	BasketballDrive	−1.40	−2.38
C	RaceHorses	−2.80	−3.65
	BQMall	−1.62	−2.01
	PartyScene	−0.30	−0.56
	BasketballDrill	−3.21	−3.60
D	RaceHorses	−2.76	−3.28
	BQSquare	0.00	−0.70
	BlowingBubbles	−2.41	−2.92
	BasketballPass	−3.06	−3.23
E	FourPeople	−2.92	−3.55
	Johnny	−1.49	−1.67
	KristenAndSara	−2.26	−2.70
F	BasketballDrillText	−3.17	−3.65
	ChinaSpeed	−6.53	−8.78
	SlideEditing	−2.11	−5.34
	SlideShow	−3.31	−4.97
Average		−2.37	−3.02

Table 10 The comparison of the maximum number of context-coded bins in level coding

Syntax element	HEVC	SAP + proposed method
significant_coeff_flag	16	16
coeff_abs_level_greater1_flag	8	0
coeff_abs_level_greater2_flag	1	0
Total	25	16

additional process for estimating the binarization table, we do not use two flags *coeff_abs_level_greater1_flag* and *coeff_abs_level_greater2_flag*. Thus, we can save complexity for parsing these two flags.

Due to sophisticated operations, CABAC is regarded as the bottleneck of the high throughput codec system. Especially, context-coded bin coding through regular mode in CABAC level coding is a main factor of the limitation of throughput. In order to check the throughput differences between HEVC and the proposed method, we checked the number of context-coded bins in level coding.

Table 10 shows the number of context-coded bins for a 4×4 subblock in worst-case scenario. Since the proposed method removes two flags that are coded as context-coded bins, the proposed scheme can reduce the maximum numbers of context-coded bins from 25 to 16. It means that 36 % context-coded bins in CABAC level coding can be reduced by the proposed method.

5 Conclusions

In this paper, we proposed an improved residual data coding method for HEVC lossless intra-coding. Considering statistical differences of residual data between lossy and lossless coding, we modified the context-based adaptive binary arithmetic coding (CABAC) mechanism for lossless coding. We first analyze the statistical characteristics of lossless coding and design an efficient level binarization method. Consequently, we present a binarization table selection method using the weighted sum of previously encoded results. Experimental results show that the proposed method provides approximately 3.01 compression ratio on average and reduces decoding complexity. From various experiments, we can verify that the proposed method efficiently improves the compression performance of HEVC lossless coding.

Acknowledgments This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2012-0009228).

References

1. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: High efficiency video coding (HEVC) text specification draft 8. In: 10th JCT-VC Meeting, JCTVC-J1003, Stockholm, SE, July (2012)
2. Ho, Y., Choi, J.: Advanced video coding techniques for smart phones. In: Proceeding of the International Conference on Embedded Systems and Intelligent Technology (ICESIT) 2012, 27–29, Nara, Japan, Jan (2012)
3. ISO/IEC JTC1/SC29/WG11: Vision, application, and requirements for high performance video coding (HVC). MPEG document, N11096, Kyoto, JP, Jan (2010)
4. Sayood, K. (ed.): Lossless Compression Handbook. Academic Press, San Diego (2003)
5. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: AHG19: A lossless coding solution for HEVC. In: 8th JCT-VC Meeting, JCTVC-H0530, San José, CA, Feb (2012)
6. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: AHG19: a QP-based enabling method for lossless coding in HEVC. In: 8th JCT-VC Meeting, JCTVC-H0528, San José, CA, Feb (2012)
7. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: AHG22: sample-based angular prediction (SAP) for HEVC lossless coding. In: 7th JCT-VC Meeting, JCTVC-G093, Geneva, CH, April (2012)
8. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: AHG19: method of frame-based lossless coding mode for HEVC. In: 8th JCT-VC Meeting, JCTVC-H0083, San José, CA, Feb (2012)

9. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: Improved CABAC for lossless compression. In: 9th JCT-VC Meeting, JCTVC-I0341, Geneva, CH, April (2012)
10. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: JCT-VC AHG report: lossless coding (AHG13). In: 9th JCT-VC Meeting, JCTVC-I0013, Geneva, CH, April (2012)
11. Choi, J., Ho, Y.: Implicit line-based intra 16 × 16 prediction for H.264/AVC high-quality video coding. Circuits Syst. Signal Process. **31**(5), 1829–1845 (2012)
12. Wiegand, T., Sullivan, G., Bjøntegaard, G., Luthra, A.: Overview of the H.264/AVC video coding standard. IEEE Trans. Circuits Syst. Video Technol. **13**(7), 560–576 (2003)
13. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: Description of video coding technology proposal by Tandberg, Nokia, Ericsson. In: 1st JCT-VC Meeting, A119, Dresden, DE, April (2010)
14. ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6: Context-adaptive VLC coding of coefficients. In: 3rd JVT Meeting, C028, Seattle, WA, May (2002)
15. Nguyen, T., Marpe, D., Schwarz, H., Wiegand, T.: Reduced-complexity entropy coding of transform coefficient levels using truncated Golomb–Rice codes in video compression. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), pp. 753–756, Sept. (2011)
16. Cai, Q., Song, L., Li, G., Ling, N.: Lossy and lossless intra coding performance vvaluation: HEVC, H.264/AVC, JPEG2000 and JPEG LS. In: Proceeding of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1–9, Dec. (2012)
17. Sullivan, G., Topiwala, P., Luthra, A.: The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions. In: Proceeding of the SPIE Conference on Applications of Digital Image Processing XXVII, pp. 454–474, Aug. (2004)
18. Skodras, A., Christopoulos, C., Ebrahimi, T.: The JPEG 2000 still image compression standard. IEEE Signal Process. Mag. **18**(5), 36–58 (2001)
19. Weinberger, M., Seroussi, G., Sapiro, G.: The LOCO-1 lossless image compression algorithm: principles and standardization into JPEG-LS. IEEE Trans. Image Process. **9**(8), 1309–1324 (2000)
20. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: AHG13: improved binarization scheme for intra luma prediction residuals in lossless coding. In: 9th JCT-VC Meeting, JCTVC-I0311, Geneva, CH, May (2012)
21. Heo, J., Kim, S., Ho, Y.: Improved CAVLC for H.264/AVC lossless intra-coding. IEEE Trans. Circuits Syst. Video Technol. **20**(2), 213–222 (2010)
22. Jct-Vc, HEVC Test Model (HM). http://hevc.kw.bbc.co.uk/trac/browser/tags/HM-8.0
23. ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11: Common HM test conditions and software reference configurations. In: 9th JCT-VC Meeting, JCTVC-I1100, Geneva, CH, April (2012)