J Real-Time Image Proc (2019) 16:883–899
DOI 10.1007/s11554-016-0571-5

ORIGINAL RESEARCH PAPER

A context-adaptive fast intra coding algorithm of high-efficiency video coding (HEVC)

Ismail Marzuki[1] · Jonghyun Ma[1] · Yong-Jo Ahn[1] · Donggyu Sim[1]

Received: 21 March 2015 / Accepted: 26 February 2016 / Published online: 15 March 2016

Abstract This paper proposes a context-adaptive fast intra coding algorithm of high efficiency video coding (HEVC). Intra coding of HEVC improves the coding efficiency of intra coded picture with an exhaustive computation and evaluation. In this paper, the proposed algorithm consists of an adaptive rough mode selection to decrease the prediction modes for rough mode decision (RMD), based on the upper coding unit (CU) and the neighboring prediction units (PU), by minimizing BD-rate increment and encoding complexity. For further complexity reduction, the early termination approaches for RDO and RQT stages are also introduced by the adaptive thresholding based on the average value of rate distortion (RD) costs. In terms of complexity evaluation, experimental results show that the proposed algorithms reduce encoding time at about 40 % with 1.3 % BD-rate loss on average, compared to the reference software of HEVC.

Keywords Fast intra prediction · High efficiency video coding (HEVC) · Mode selection · Video coding

1 Introduction

High efficiency video coding (HEVC) project was officially launched based on a project partnership by ISO/IEC MPEG and ITU-T VCEG in January 2010 [1–4]. The state-of-the-art video compression technology developed by collaboration work of the two international standardization organizations was released as a new international video compression standard in January 2013 [5–8]. By releasing the reference software called HEVC Test Model (HM) [9], including the draft specification [10], inspection on HEVC conformance has been cultivated abruptly by both industry and academic researchers [11–16]. Nowadays, HEVC has been well known as a promising video coding standard with superior capability in compression by improving the bit-rate reduction about 50 % in relative to its predecessors, advanced video coding (AVC) high profile, in subjective quality [7, 17, 18]. Furthermore, due to the increasing demand for higher quality video, the dominant AVC applications used worldwide are expected to be soon replaced by the HEVC.

HEVC is investigated to support greater color and bit-depth precision of a new video content format. It is developed to meet the goal of video applications targeting the higher-resolution videos format (e.g., 4 and 8 K resolutions), such as ultra-high-definition television (UHDTV) that will be launched in the near future [17–20]. Main and Main 10 [9, 10] profiles have been specified for various applications. The Main profile for typical consumer applications is being deployed today, and the Main 10 profile is also employed for supporting up to 10 bits precision [16]. Both Main profile and Main 10 profile have the same basic tool configuration [10].

Against its prior video coding standards, the remarkable compression performance of HEVC originates from the nested quad-tree coding unit structure called coding tree unit (CTU) [21–23]. The CTU supports larger and various block-shape structures against the fixed size of macroblock structure concept introduced in H.264/AVC [21], by splitting quad-tree-based CU partitioning. Each CU can have various prediction unit (PU) and quad-tree transform unit (TU) structures [21–25]. HEVC also supports many

✉ Donggyu Sim
dgsim@kw.ac.kr

[1] Department of Computer Engineering, Kwangwoon University, 20, Gwangun-ro, Nowon-gu, Seoul, Korea

prediction modes including intra and inter prediction algorithms. Intra prediction modes of HEVC consist of 33 directional, DC, and planar prediction modes [21–28]. Two in-loop filtering methods are employed as deblocking filter and sample adaptive offset [29, 30] for significant compression and visual quality performance of HEVC. However, these coding tools also significantly increase computational complexity that may lead to impracticality for real-time video applications [21, 27, 32, 33] by keeping the best visual quality.

Intra coding is performed to exploit spatial redundancy among the neighboring intra blocks by extrapolation values derived from the reconstructed pixels of neighboring blocks. In HEVC, intra coding is designed for each 2N × 2N and N × N PU sizes ranging from 64 × 64 to 4 × 4 by utilizing 35 intra prediction modes derived from the previous coded block. Intra prediction mode 0 refers to the planar mode and mode 1 refers to the DC mode. In addition, other 33 intra modes are for angular modes. This design evidently improves coding efficiency of HEVC for intra coded pictures over the prior video [27] and the still picture codecs [61, 62]. It is reported that the HEVC intra coding improves bit-rate reduction to be 22.3 % on average and up to 36 % over the H.264/AVC reference. In addition, the average objective bit-rate reductions provided by the HEVC intra coding against the still picture codes are 61.6 and 20 % with respect to JPEG and JPEG 2000 [61], and 32.2 % with respect to JPEG XR [62], respectively. Furthermore, subjective picture quality is also significantly improved when comparing the resulting pictures at fixed bit-rate compared both prior video and still picture codecs. However, due to various PU sizes and intra prediction modes, intra coding of HEVC requires heavy computational load [27]. The rate distortion optimization (RDO) employed to select the best performance for intra coding of HEVC requires exhaustive search computation [26] and leads impractical for most of the applications [16].

To find the optimal prediction mode for the current block with a lower complexity, HM intra coder consists of three stages such as the rough mode decision (RMD), the rate distortion optimization (RDO), and the residual quadtree process (RQT) [31]. However, it still requires heavy computational complexity. The detailed information for intra prediction of HM will be further discussed in Sect. 2. To improve intra prediction complexity while maintaining coding efficiency for HEVC, various proposed algorithms have been published. PU-based RMD algorithms were basically proposed to decrease the number of prediction modes, instead of utilizing the modes supported in HM [34–36]. However, decreasing intra prediction modes contributes only small encoding time saving. Therefore, other papers proposed the fast intra mode decision that exploits not only in RMD stage, but also in the other stages [24, 41, 42]. In addition, the early termination technique or other approaches for CU, PU, or TU decision in intra coding are also presented to accelerate the current version of intra coding [39, 44–46].

In this paper, a context-adaptive fast intra coding algorithm is presented to decrease the computational complexity and to maintain the Bjongtegaard Distortion rate (BD rate) [60] increment for all mentioned stages of intra coding in HM. The proposed algorithm is designed to improve our previous work [24] for better encoding time saving and low BD-rate loss. Two new approaches for RQT stage are introduced in this work to enhance encoding time reduction including the early skip for RQT stage and the early terminate of RQT partition. The proposed algorithm for RQT stage is designed only for intra coding case which is organized very differently from our previous work [37]. First, an adaptive rough mode selection is employed to reduce candidate modes for RMD stage. It is organized based on the upper CU layer and the neighboring PU blocks. Then, the rate distortion (RD) cost estimated from RMD stage and an average RD cost from the available neighboring CU blocks are estimated as thresholds to early skip RDO and RQT stages, respectively. The experimental results show that the proposed algorithm can achieve encoding time saving to about 40 % with negligible BD-rate loss of 1.3 %.

The rest of this paper is organized as follows. Section 2 presents the overview of intra prediction of HEVC. In Sect. 3, the proposed context-adaptive fast intra coding algorithm is presented. Section 4 shows the performance evaluation of the proposed algorithm, and Sect. 5 concludes with further research topics.

2 Overview intra prediction of HEVC

In this section, intra prediction of HEVC standard is briefly introduced. In order to achieve high coding efficiency, intra prediction is performed to spatially predict target pixels from the reconstructed neighboring pixels of the left and/or upper blocks.

As shown in Fig. 1a, the target pixels (a, b, …, and p) are predicted with the neighboring pixels (A, B, …, and Q). Furthermore, to precisely predict the target block, 35 prediction modes were designed in HEVC standard, consisting of planar, DC, and 33 angular prediction modes, as shown in Fig. 1b.

Figure 2 depicts block diagram of the adopted fast intra coding in HM. As shown in Fig. 2, the fast intra coding algorithm selects limited numbers, among 35 prediction modes, as the candidate modes in the RDO stage, depending on the CU sizes from 64 × 64 to 4 × 4. RD cost of each prediction mode is computed by the sum of

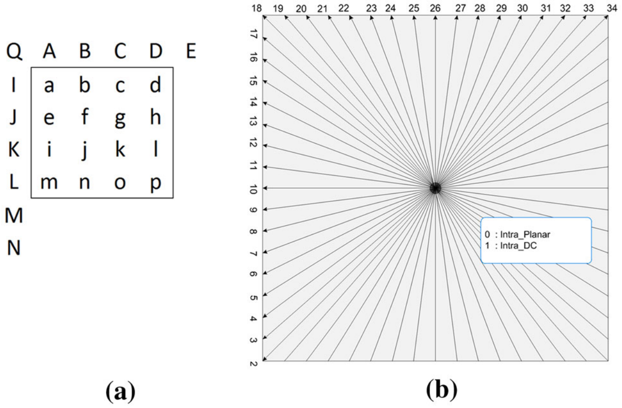


Fig. 1 Intra prediction: **a** A 4 × 4 block and its neighboring pixels and **b** 35 intra prediction modes of HEVC

absolute Hadamard transform difference (SATD) and its associated bit, and it is defined by:

$$J_{\mathrm{RMD}} = \mathrm{SATD} + \lambda.\mathrm{Bit}_{\mathrm{mode}} \quad (1)$$

The selected modes are sorted in ascending order based on its cost, and then the modes are evaluated in RDO stage with the zero depth of RQT to find the best mode for the current CU. Finally, RQT stage is performed to determine the best TU partition. In HM, the maximum depth of TU quad-tree for intra coding is set to 3 for coding efficiency improvement. Although, the fast intra coding already adopted for HEVC; however, further improvement is required due to it still remains the complexity load for encoding. Table 1 shows the complexity load observation of HM for intra coding. RDO stage is the most expensive process that occupies about 42 % in computation complexity, followed by RQT stage at 41 %. Note that RMD stage covers only about 17 % [24].

To decrease the computational load of intra coding, many works have been proposed based on HM reference software. In general, they can be categorized depending on the three stages of HM: fast intra prediction mode decision (consisting of RMD and RDO optimizations), fast block types (CU, PU, or TU) decision, and their combinations. Existing algorithms will be discussed in these three categorizations as follows.

2.1 Fast intra prediction mode decision

Zhao et al. [34] proposed a new intra prediction algorithm to reduce the intra mode prediction mode decision complexity. There are four configurations employed on HM-1.0 to decrease the number of candidate modes. These configurations are set according to PU size of 4 × 4 to 64 × 64, respectively. The experimental results show that this algorithm improves encoding time saving up to 20 % with BD-rate loss of 0.12 % in the high efficiency test condition. The similar idea was also proposed by Mengmeng et al. [36]. The proposed algorithm performs different treatments for each PU size based on the statistics of intra prediction mode probability on HM-4.0. It is claimed to achieve about 15 % of encoding time reduction with 0.64 % BD-rate increment for main intra configuration. According to these works and other similar works [35, 50, 52], acceleration of only RMD or RDO stage will not provide significant improvement in reducing the computational complexity. These conventional algorithms are based on the fact that coding efficiency and computational complexity are jointly optimized by adaptive number of coding modes depending on PU size, instead of a fixed number of coding modes.

Other works [31, 38, 42] optimize both RMD and RDO stages in order to achieve better encoding time reduction with minimal BD-rate increment. A local saliency

Fig. 2 Block diagram of intra prediction of HM

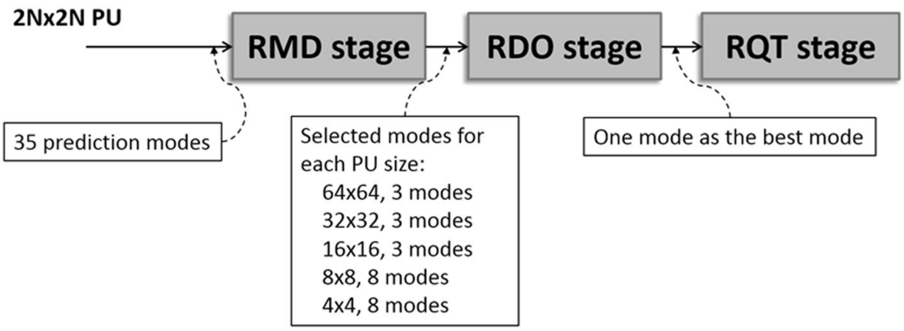


Table 1 Complexity observation of HM intra coding

Stage	Complexity ratio (%)
Rough mode decision (RMD)	17
Rate distortion optimization (RDO) calculation	42
Residual quad-tree (RQT) process	41

detection for intra prediction mode decision is proposed to optimize RMD and RDO processes [31]. It is reported to achieve about 19 % complexity reduction with 0.3 % BD-rate loss over HM-7.0. Hao's work [38] proposed a progressive intra prediction mode reduction for RMD stage. An early RDO termination is also proposed for further complexity reduction in RDO stage. However, this method still loses high BD-rate of 3 % with encoding time reduction of approximately 38 % over HM-6.0.

A fast intra coding for ultra-high definition (UHD) videos is also proposed by Dhollande et al. [42]. Although this work claims significant time saving of 67.3 % on average, the BD-rate increment is somehow high 5 %, thanks to the Hadamard transform for RMD and RDO stages. Note that performance evaluation is not clearly presented in this manuscript. Jamali et al. [49] utilize Sobel operator determine the dominant edge of PU block to optimize RMD and SATD cost is also obtained to skip RDO stage. It yields about 36 % encoding time reduction with 1.07 % BD-rate loss, compared the reference software of HM-15.0.

2.2 Fast block types (CU, PU, or TU) decision

Liu's et al. [43] proposed to reduce intra coding of HEVC by optimizing RDO stage in HM-12.1. This algorithm was designed by employing the support vector machine (SVM). The SVM is trained according to the average pixel difference on CU boundary, pixel variance of current CU, variance of the mean of sub-CUs, and the number of edge points. It is then used to decide the intra CU depth to improve encoding time of intra coding. This fast algorithm can accelerate encoding about 46.5 % on average with BD-rate lose over 2 %. An effective CU size decision for intra coding of HEVC is proposed by Shen et al. [44]. Two algorithms (adaptive thresholding and bypass algorithms for large CUs) are presented according to texture homogeneity and coding information of the adjacent coded CU blocks. The experimental results yield that about 47 % complexity reduction with BD-rate loss of 1.08 % on average against HM 10.0 are achieved.

Another CU size decision algorithm is introduced by Li et al. [45]. A fast CU splitting and pruning algorithms have been carried out to reduce the intra coding complexity over the reference software HM-10.0. In this work, each CU is classified depending on its size. Then, the statistics of RD cost and SATD cost are utilized to determine whether CU will be split or pruned further. This method reports around 38 % encoding time reduction on average with 1.05 % BD-rate loss. Other conventional works were also proposed by considering edge and texture characteristics inside of CU [47, 56, 58]. Min and Cheung [47] tried to analyze global and local edge complexities of each CU to determine its CU size decision. This algorithm shows about 52 % encoding time reduction with only 0.8 % BD-rate increment over HM-10.0. In HM-RExt platform, Huang et al. [56] and Zhang et al. [58] utilized CU texture complexities to decide CU size decision. It is reported that the presented algorithm [56] could achieve 39 % encoding time reduction with 0.6 % BD-rate increment over HM-Rext-13.0. In addition, reference [58] was proposed based on early decision of block partition, yielding about 33 % complexity reduction, compared HM-Rext-12.0.

2.3 Combination fast prediction mode decision and fast block types decision

Zhang's algorithm [39] is designed to improve the previous work [38] for fast intra coding of HEVC. Hadamard cost and RD cost are employed to early terminate CU and TU partition decision. The progressive rough mode search in RMD stage and the early termination for RDO stage presented in [38] are also still employed. It yields to achieve average encoding time saving of 32 % against HM software version 7.0 with only 1.1 % BD-rate loss. And then the algorithm [40] is proposed to improve algorithm [39]. Macro and micro operations are organized to speed up intra coding of HM-10.0. These proposed algorithms reduce intra coding complexity about 60 % with 1.0 % BD-rate loss.

The same approach is also introduced by Na et al. [41]. The algorithm employs pre-processing, Sobel edge operation, implemented on HM-10.0. Edge operation is used to compute the correlation between a local edge of the image block and an intra prediction mode to optimize RMD and RDO stage. However, this algorithm yields high BD-BR loss of 2.5 % with encoding time reduction by 56.8 %. Yang and Grecos [46] proposed a heuristic fast intra encoding algorithm introduced on HM-10.0. This algorithm consists of fast CU skip decision, CU early termination, fast PU mode decision, and fast TU size decision by considering both time reduction and BD-rate increment for intra coding. It reports to achieve approximately 65 % in encoding time on average with 1.3 % BD-rate lose. Hu and Yang [48] tried to optimize RMD and RDO stages and also proposed entropy coding refinement against intra coding of HM-15.0. These algorithms show approximately 50 % encoding time reduction with only 0.2 % BD-rate increment. Khan et al. [51] consider hardware implementation on FPGA to speed up intra prediction mode decision in RMD stage of HM-7.2. Mode decision is performed after early evaluating PU block size decision. Around 50 % complexity reduction is achieved for a few test sequences. In addition, it is reported that this work also reduces the energy consumption about 35 % over the HM.

Palomino et al. [53] proposed a fast intra coding algorithm to tackle RMD, RDO, and PU decision. The previously chosen intra modes of PU decision are employed to optimize RMD and RDO processes. The proposed algorithm could achieve 59 % encoding time reduction for intra case with 1.1 % BD-rate increment, compared HM-10.0. Coll et al. [54] utilize a machine learning approach to determine CU sizes for RDO process complexity reduction. The proposed algorithm reports only 28 % encoding time reduction over HM-10.0 with 0.6 % BD-rate loss. Other conventional algorithms were also introduced by Shi et al. [55] and Wang et al. [57]. Shi's algorithm [55] reduces encoding time of about 27 % with 0.8 % BD-rate loss over HM-12.1. It is achieved by considering edge information for bottom-up partition in PU decision and by optimizing the RMD stage on the largest PU size. Wang's algorithm [57] was designed for RMD, RDO, and CU optimizations. Around 54 % complexity reduction is reported with 1.0 % BD-rate loss over HM-10.0. In this algorithm, CTU depth range is predicted as a key process for CU size decision, RMD, and RDO optimization.

Although many intra coding algorithms have been proposed to speed up the fast intra coding, computation loads of the existing algorithms and HM still remain high for real-time video applications. In this paper, a context-adaptive fast intra coding algorithm is proposed based on intra coding mode decision optimization and early termination techniques.

3 Proposed context-adaptive fast intra coding algorithm

This paper proposes a context-adaptive fast intra coding algorithm of HEVC to reduce the computational complexity. Figure 3 is depicted to show the flowchart of the proposed context-adaptive fast intra mode selection algorithm.

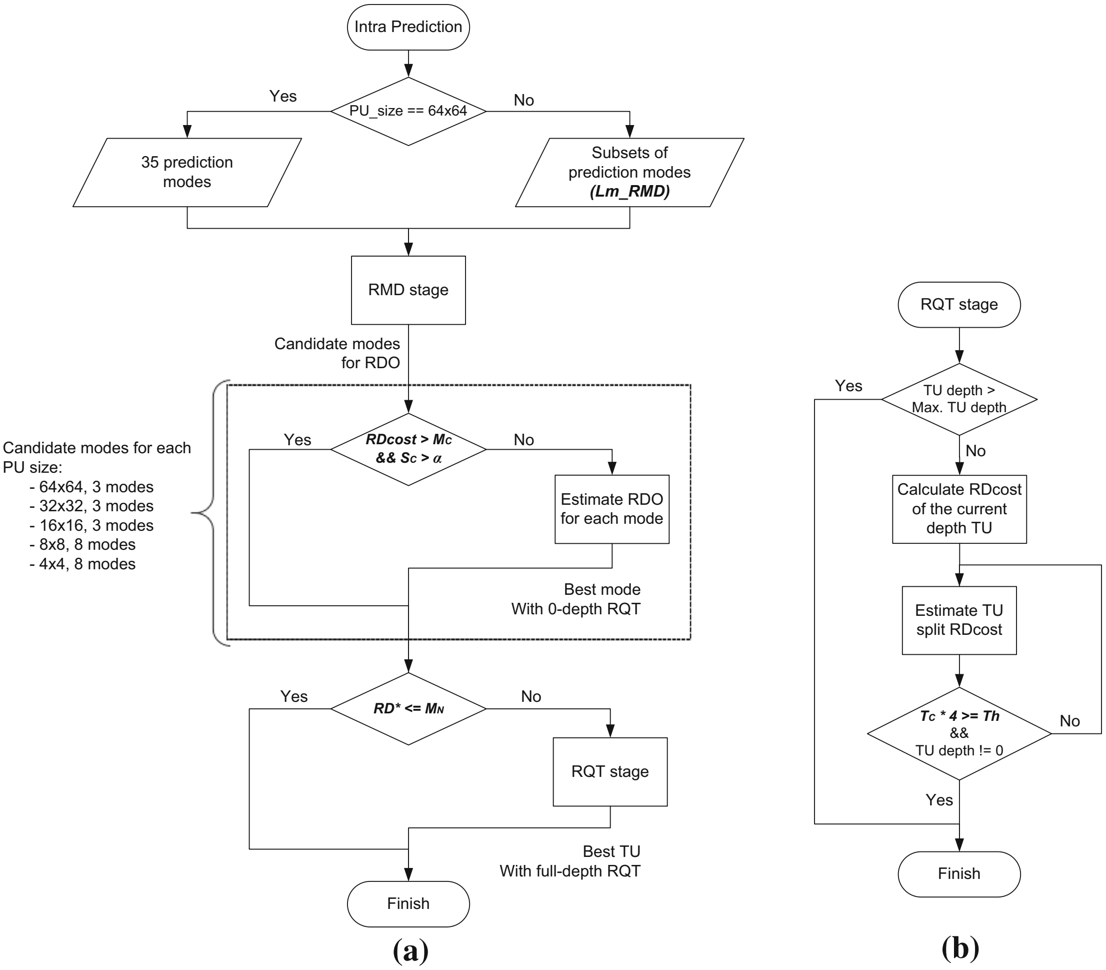


Fig. 3 Flowchart of the proposed fast intra decision algorithm: **a** proposed fast intra decision and **b** proposed early termination of TU split algorithm in RQT stage

Figure 3a shows the flowchart of the overall proposed algorithm. First, an adaptive rough mode selection is proposed to decrease the number of intra modes for 32×32 to 4×4 PUs in the RMD stage. The proposed algorithm organizes the modes for RMD from the prediction modes of upper layer CU and neighboring PUs. *Lm_RMD* represents the subset candidate prediction mode set organized by the adaptive rough mode selection. Furthermore, a simple threshold α is obtained to early terminate the RDO stage for all PU sizes. It is computed by considering the RD cost of prediction mode computed from the RMD stage, denoted as *RDcost* in Fig. 3a. The mean and the standard deviation values expressed by $\boldsymbol{M}_{\mathbf{C}}$ and $\boldsymbol{S}_{\mathbf{C}}$, respectively, are calculated to divide the RD cost into groups of RD cost. If a given condition is satisfied for this stage, then the modes that are provided for RDO stage are skipped.

In addition, the RQT skip and the early terminate TU split in quad-tree fashion are proposed to enhance encoding time saving. The average of RD-cost values for the reconstructed neighboring CUs M_{N} is calculated to skip the RQT stage. Then, it is compared with RD cost of the TU depth zero computed in the RDO stage referred as RD^{*}. Note that only the reconstructed CUs having the same depth with the current CU are evaluated to retain the BD-rate loss. Moreover, an adaptive threshold value Th is also set to early terminate the TU split cost T_{C} in the RQT stage, as shown in Fig. 3b. It is provided to give higher acceleration of the proposed context-adaptive fast intra mode selection algorithm.

To describe the propose algorithm, the proposed adaptive rough mode selection algorithm in RMD stage is firstly discussed. Then, the proposed algorithm for RDO stage is presented. Finally, the proposed algorithm for RQT stage with the skip RQT stage and the early terminate TU split of RQT are discussed.

3.1 Proposed adaptive rough mode selection

In RMD stage, the proposed algorithm carefully reduces the prediction modes to retain the bit-rate increment. The proposed algorithm organizes the intra prediction candidate list derived from the upper layer of a CU. First, we observe that the mode of the upper CU layer is likely to be the same as one of the current coded block.

Table 2 shows mode dependency between the current CU block and its upper CU layers. The investigation was performed for the first 30 frames of three sequences (‘Traffic’, ‘Kimono’, and ‘Video4’) with three quantization parameters (QP) of 27, 32, and 37. According to Table 2, it shows that the smaller block sizes are likely to have higher similarity in the best prediction mode than the larger blocks. In other words, the upper CU layer is likely to share the same prediction mode in predicting the lower CU layer to be encoded. In addition, we investigated the probability of the modes of a current PU in terms of its adjacent PU blocks. The investigation was performed for the first 30 frames of ‘BQTerrace’ sequence at quantization parameter (QP) of 32. Table 3 presents the co-occurrence of the current PU with the surrounded PU blocks, in terms of the PU depth. As shown in Table 3, each adjacent PU blocks have different co-occurrence percentages over the current PU. The high prediction dependency is observed from the top and the left neighboring PU blocks for all the PU depths.

Table 2 Probability of the current PU having the same mode with its upper PU layer

Sequence	QP	Depth	PU size	Not same (%)	Same (%)
Traffic	27	0	64 × 64	–	–
		1	32 × 32	75.87	24.13
		2	16 × 16	77.73	22.27
		3	8 × 8	59.25	40.95
	32	0	64 × 64	–	–
		1	32 × 32	51.66	48.34
		2	16 × 16	25.80	74.20
		3	8 × 8	16.84	83.16
	37	0	64 × 64	–	–
		1	32 × 32	40.36	59.64
		2	16 × 16	23.62	76.38
		3	8 × 8	13.99	86.01
Kimono	27	0	64 × 64	–	–
		1	32 × 32	67.60	32.40
		2	16 × 16	70.19	29.81
		3	8 × 8	50.34	49.66
	32	0	64 × 64	–	–
		1	32 × 32	66.23	33.77
		2	16 × 16	47.70	52.30
		3	8 × 8	36.85	63.15
	37	0	64 × 64	–	–
		1	32 × 32	67.02	32.98
		2	16 × 16	46.21	53.79
		3	8 × 8	24.91	75.09
Video4	27	0	64 × 64	–	–
		1	32 × 32	73.73	26.27
		2	16 × 16	72.46	27.54
		3	8 × 8	46.08	53.92
	32	0	64 × 64	–	–
		1	32 × 32	71.21	28.79
		2	16 × 16	40.06	59.94
		3	8 × 8	27.36	72.64
	37	0	64 × 64	–	–
		1	32 × 32	68.69	31.31
		2	16 × 16	33.95	66.05
		3	8 × 8	18.49	81.51

Table 3 Probability of the current PU having the same mode with neighboring PU layer

PU depth	Left-bottom PU		Left-PU		Left-top PU		Top PU		Right-top PU	
	Not same (%)	Same (%)	Not same (%)	Same (%)	Not same (%)	Same (%)	Not same (%)	Same (%)	Not same (%)	Same (%)
0	62.83	37.17	72.01	27.99	72.00	28.00	72.38	27.62	64.03	35.97
1	77.93	22.07	67.20	32.80	70.54	29.46	70.08	29.92	78.54	21.46
2	83.50	16.50	48.60	51.40	50.27	49.73	66.58	33.42	83.78	16.22
3	94.51	5.49	58.39	41.61	75.12	24.88	68.86	31.14	91.74	8.26

An adaptive rough mode selection algorithm is proposed for RMD stage to reduce the complexity with minimum bit-rate increment. The adaptive rough mode selection is performed with the PU block sizes from 4 × 4 up to 32 × 32. The candidate list modes generated by the proposed algorithm is called *Lm_RMD*. It is organized by involving the best mode (BM) and the second best mode (SBM) derived from the upper layer of CU, as illustrated in Fig. 4a. Furthermore, two modes from left and right sides of BM and SBM are also added into *Lm_RMD*, respectively. These additional modes are obtained in series from the prediction modes of the left and the right of BM to the modes of the left–right of SBM. Note that there should be no overlap modes listed in *Lm_RMD*. Otherwise, only the mode that is prior listed be retained. By default, the listed modes in *Lm_RMD* can contain 10 candidate prediction modes that will be used in RMD stage for the current CU without any overlap modes condition. However, if the prediction modes in *Lm_RMD* are less than the modes that should be processed for RDO stage, prediction mode from the neighboring PUs, as shown in Fig. 4b, is considered to be in the list. In addition, the modes from the RMD stage of the upper layer of CU are also added into the *Lm_RMD*. The proposed adaptive rough mode selection algorithm for RMD stage can be illustrated with an example as below:

1. First, BM and SBM from the upper CU are added into *Lm_RMD*. For example: BM → 10 and SBM → 26.
2. Then, add two modes from left and right of BM and SBM: two modes from BM's left → 8, 9; two modes from BM's right → 11, 12; two modes from SBM's left → 24, 25; and two modes from SBM's right → 27, 28.
3. If the candidate mode set for RMD is smaller than the candidate modes for RDO of current PU, then add modes derived from the neighboring PU blocks (top left, top, top right, left, and bottom left), and also add modes from the RMD stage of the upper layer of CU.

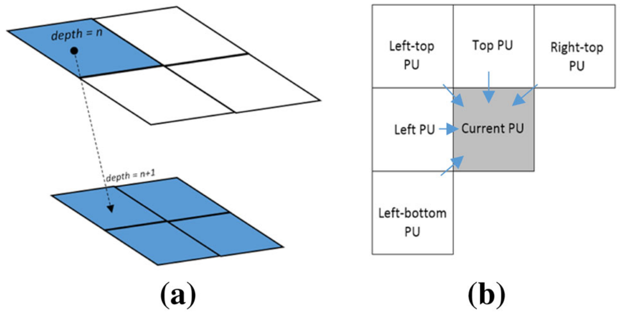


Fig. 4 The proposed rough mode selection for RMD stage: **a** the *upper layer* of CU *block*, **b** the neighboring PUs *block*

3.2 Proposed early RDO skip algorithm

An early RDO skip algorithm is proposed by employing an adaptive threshold to early terminate the RDO stage. The basic concept of this proposed algorithm is to divide the RD-cost modes, estimated from the RMD stage, into two groups. One group will be processed in RDO stage while another one will be considered to be skipped to achieve more encoding time saving. Firstly, the mean value of RD cost (referred as M_C) is set as the threshold to divide the two groups. Prediction modes with larger RD cost than M_C value will be skipped for the next RDO stage. However, it could not be efficient to set the M_C value as the only one boundary for all the cases. Note that only three modes are investigated for 64 × 64, 32 × 32, and 16 × 16 PUs for RDO stage. The pruning with only M_C value can lead somehow BD-rate lose. Therefore, we use the additional threshold to extend the boundary, *Sc* which is used to refer to the normalized standard deviation of RD-cost values. With the additional decision, more prediction modes can be fed into the RDO stage.

In general, early skip algorithms could make several optimal modes skipped as a side effect. The proposed algorithm is designed to improve the correct decision rate by using the additional thresholding. In our observations, we found that the block having a large M_C is likely to have larger variation for all the prediction modes for RDO stage and vice versa. Therefore, the additional thresholding is employed to reduce skipping of the best mode. In our proposed algorithm, the threshold *Sc* is calculated by normalizing the variance and it is denoted by

$$V_C = \frac{1}{(n \times M_C)} \sum_{i=1}^{n} (R_{Ci} - M_C)^2, \quad (2)$$

$$S_C = \sqrt{V_c}, \quad (3)$$

where $\boldsymbol{R}_{C_i}$ refers to the RD-cost value of the $\boldsymbol{i}$-th prediction mode and $\boldsymbol{n}$ represents the total number of prediction modes for the RDO stage. In our algorithm, the condition set by mean and normalized standard deviation value is designed as:

$$(\text{RDcost} > M_C)\ \&\&\ (S_C > \alpha) \quad (4)$$

where *RDcost* is computed from RMD stage with SATD and bit of intra modes; and α is determined by evaluating the average of BD-rate and encoding time of some video sequences with different resolutions. It should be noted that *RDcost* is already sorted in ascending order in the RMD stage. The normalized standard deviation formula can consider more modes that would be evaluated in RDO stage to maintain both BD-rate increment and encoding time saving.

The threshold value, **α** used in our proposed algorithm is determined by evaluating several video sequences: 'PeopleOnStreet' and 'NebutaFestival' (4 K), 'Cactus' and 'BQTerrace' (1080p), 'PartyScene' and 'BasketballDrill' (WVGA), 'RaceHorse' and 'BlowingBubbles' (WQVGA), and 'Video1' and 'Video4' (720p). These video sequences are evaluated with all QP ranges under the common test condition (CTC) defined in HEVC [59]. Each sequence is tested with the given threshold value as shown in Fig. 5. Then, the average of BD rate and encoding time corresponding to the sequences are evaluated. Figure 5a and b present BD-rate loss and encoding time in terms of threshold **α**, respectively. The figures show that the higher threshold value provides the better coding efficiency, but the lower encoding time saving and vice versa. In our proposed work, the threshold is set to 0.15 to achieve both encoding time reduction and BD-rate retained.

3.3 Proposed RQT skip and early termination of TU split

In this paper, RQT skip and early termination are proposed to accelerate the RQT stage. In HM, the quad-tree process is performed with depth-first search technique into 4 × 4 to 32 × 32 of TU partition. The maximum search depth of this process is set to 3 for both intra and inter coding. In order to determine the best TU of current depth (e.g. 16 × 16 TU), its cost will be compared with the sum of four sub-TU partitions (e.g. 8 × 8 TU). However, the RQT process for intra and inter coding are treated in a quite different way. RQT process for intra case is carried out for each PU. When it is performed to determine the best prediction mode in the RDO stage, the RQT search depth is set to 1 to save the encoding time. It is further called as TU of zero depth in this paper. However, the search depth is set to 3 when it is performed to decide the best TU split partition in RQT stage. In other words, this maximum search depth of RQT is conducted recursively with heavy computation to find the optimum TU for the coding efficiency reason. It should be noted that the RD-cost TU of zero depth is always compared with the RD-cost TU evaluated from RQT stage, and TU with the smaller RD cost will be decided as the optimum TU for intra case.

In this work, the selection percentages between zero-depth TU (denoted by *RD**) and full quad-tree TU partition from RQT stage (referred as R_{full}) were evaluated with HM-14.0. Several sequences from Class A, Class B, and Class C with QPs of 22, 27, 32, and 37 were used for investigation, as shown in Table 4. According to Table 4, the zero-depth TU partition (*RD**) is mostly selected as the best TU decision over the full quad-tree TU partition. If we can decide whether the best TU is determined from only *RD**, RQT stage can be skipped to achieve further time reduction for intra coding.

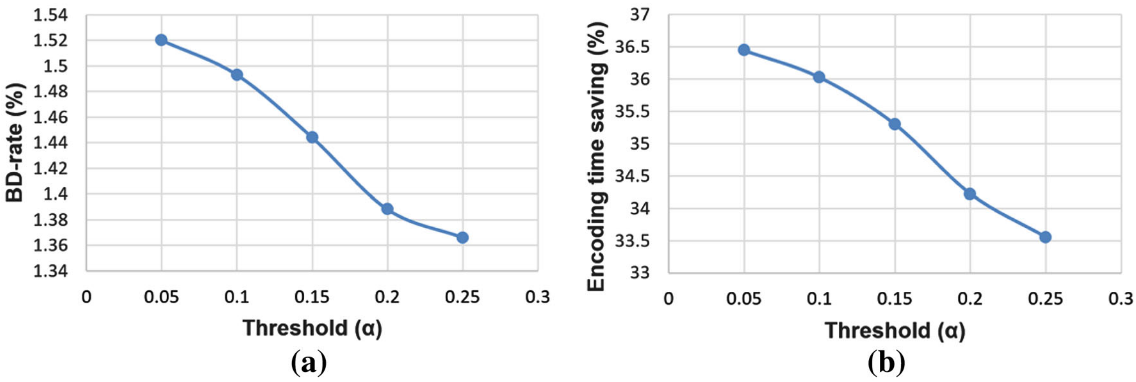


Fig. 5 **a** BD rate in terms of the threshold and **b** encoding time saving in terms of the threshold

Table 4 Selection ratios of TU decision between zero-depth TU and full quad-tree TU partitions

Sequence	QP	R_{full} (full quad-tree) (%)	*RD** (zero-depth) (%)
Nebuta festival	22	20.38	79.62
	27	15.07	84.93
	32	11.13	88.87
	37	8.21	91.79
	Avg	13.70	86.30
Steam locomotive	22	27.79	72.21
	27	21.14	78.86
	32	18.36	81.64
	37	14.43	85.57
	Average	20.43	79.57
Basketball drive	22	8.27	91.73
	27	8.04	91.96
	32	6.46	93.54
	37	5.01	94.99
	Average	6.95	93.05
BQ terrace	22	5.52	94.48
	27	5.29	94.71
	32	4.66	95.34
	37	3.18	96.82
	Average	4.66	95.34
BQ mall	22	57.42	42.58
	27	53.98	46.02
	32	50.54	49.46
	37	47.66	52.34
	Average	52.40	47.60
Party scene	22	12.30	87.70
	27	11.01	88.99
	32	8.67	91.33
	37	6.42	93.58
	Average	9.60	90.40

However, it is very difficult to fairly skip the RQT stage without increasing bit-rate. Therefore, after the RDO stage, the proposed RQT skip algorithm is designed in a very careful way. In the proposed algorithm, the mean value of RD cost of the reconstructed CU blocks, denoted as M_{N}, is adaptively used as a threshold. Then, each *RD** is compared with M_{N} to decide whether the RQT stage will be skipped or not. Note that only RD costs from the reconstructed CUs, having the same depth with the current CU, are evaluated for M_{N}. By applying this condition to skip the RQT stage, bit-rate increment can be avoided; however, the encoding time saving could be low. Therefore, to achieve more encoding time reduction, we need to exploit the TU split part in the RQT process. Thus, an early termination of TU split in the RQT process is then proposed to enhance the encoding time saving.

In this algorithm, a simple adaptive threshold is computed from average of neighboring RD costs and that of the current depth TU, and the threshold is defined as:

$$\mathrm{Th} = \frac{1}{n+1}\left(\sum_{k=1}^{n} \mathrm{RC}_k + Cd\right) \tag{5}$$

where n represents the number of the available neighboring CUs, C_{d} is expressed for RD cost of the current depth TU, and RC_k refers to the RD-cost value of the $\boldsymbol{k}$-th neighboring CU. In HM, each TU splits into four partitions, then RD cost of each single TU partition (T_{C}) is accumulated and the accumulated cost is compared with C_{d} to determine the optimum TU partition in RQT stage. In the proposed algorithm, we use the single sub-TC cost (T_{C}) multiplied by four as an approximation of the accumulation T_{C} to reduce computational complexity. If the T_C multiplied by four is larger than the threshold (Th), the rest of sub-TU partition is skipped for the proposed algorithm, as shown in Fig. 3b.

4 Experiment results

In this section, several performance evaluations of the proposed context-adaptive fast intra coding algorithm of HEVC are presented. These evaluations consist of coding efficiency, computational profiling, and power consumption analysis. BD-rate performance [60] and encoding time reduction were evaluated on HM-14.0 main profile to measure the coding efficiency. The profiling analysis is useful to determine which components are the most time consuming; hence, practically, the profiling is widely used [33]. In addition, power consumption evaluation of the proposed algorithm was performed against the reference software ones. We employed "Joulemeter" software [63] developed by Microsoft Research to assess power consumption of the proposed algorithm and existing algorithms. The software measures energy consumption for hardware resources, software applications, or individual software in every second. It monitors some resource usages, such as CPU, screen, memory, and storage power in Watts [64, 65].

In terms of BD-rate performance and complexity reduction, various video sequences were employed, such as those in Class A (4 K), Class B (1080p), Class C (WVGA), Class D (WQVGA), and Class E (720p). The evaluations

Table 5 BD-rate performance and encoding time reduction of individual proposed algorithm in rough mode decision (RMD), rate distortion optimization (RDO), and residual quad-tree (RQT)

Sequence	RMD				RDO				RQT			
	Y (%)	U (%)	V (%)	TR (%)	Y (%)	U (%)	V (%)	TR (%)	Y (%)	U (%)	V (%)	TR (%)
Class A (4 K)												
Traffic	0.0	0.0	−0.1	13	0.5	0.7	0.5	33	0.0	0.0	0.0	10
People on street	0.0	0.0	0.0	18	0.6	1.1	1.2	35	0.0	0.0	0.0	14
Nebuta festival	0.0	0.0	0.0	23	0.5	0.2	0.1	36	0.0	0.0	0.0	9
Steam locomotive	0.0	0.0	0.0	19	1.2	3.1	3.1	32	0.0	0.0	0.0	10
Class B (1080p)												
Kimono	0.0	0.0	0.0	17	1.5	1.5	1.6	38	0.0	0.0	−0.1	11
Park scene	0.0	0.0	0.0	22	0.3	0.3	0.4	38	0.0	−0.1	0.0	14
Cactus	0.1	0.1	0.1	24	1.3	1.3	1.6	36	0.0	0.0	0.0	17
Basket ball drive	0.0	0.0	0.1	15	4.1	3.6	3.6	33	0.0	0.0	0.1	11
BQ terrace	0.0	0.1	0.0	23	2.3	2.6	3.1	36	0.0	0.0	0.0	9
Class C (WVGA)												
Basket ball drill	0.4	0.3	0.3	24	2.6	1.9	1.4	34	0.2	0.2	0.2	11
BQ mall	0.2	0.2	0.1	22	0.4	1.0	1.1	34	0.0	0.0	0.0	8
Party scene	0.3	0.2	0.1	21	0.5	0.4	0.5	37	0.0	0.0	0.0	8
Race horses	0.1	0.0	0.0	21	0.4	1.0	1.0	33	0.0	−0.2	−0.2	5
Class D (WQVGA)												
Basket ball pass	0.5	0.4	0.5	18	1.1	0.9	0.9	34	0.1	−0.1	0.1	10
Flower vase	0.5	0.3	0.2	22	1.1	1.6	1.1	33	−0.1	0.0	−0.1	7
Blowing bubbles	0.5	0.2	0.3	19	0.8	0.5	0.4	37	−0.1	0.0	0.0	13
Race horses	0.4	0.3	0.2	17	0.7	0.6	0.8	35	0.0	0.1	0.0	10
Class E (720p)												
Video 1	0.0	0.0	0.2	15	2.3	3.8	3.9	32	0.0	0.0	0.1	5
Video 3	0.6	0.4	0.6	18	4.2	5.3	4.9	33	0.0	0.1	0.0	6
Video 4	0.0	−0.1	0.0	20	2.9	3.5	3.8	30	0.0	0.0	0.0	10
Class A	0.0	0.0	0.0	18	0.7	1.3	1.2	34	0.0	0.0	0.0	11
Class B	0.0	0.0	0.0	19	1.9	1.9	2.1	36	0.0	0.0	0.0	13
Class C	0.2	0.2	0.1	22	1.0	1.1	1.0	34	0.0	0.0	0.0	8
Class D	0.5	0.3	0.3	19	0.9	0.9	0.8	35	0.0	0.0	0.0	10
Class E	0.2	0.1	0.2	18	3.1	4.2	4.2	32	0.0	0.0	0.0	7
All	0.2	0.1	−0.2	20	1.5	1.8	1.8	34	0.0	0.0	−0.3	10

were conducted under ‘all intra configuration’, defined in the HEVC common test conditions. Four QP values at 22, 27, 32, and 37 are set to demonstrate the proposed algorithm performance. Furthermore, the proposed algorithm is tested on Intel(R) Core(TM) i7-4770 K CPU @ 3.50 GHz with memory of 32 GB, and Windows 7 Professional 64-bit. The overall performances of the integrated proposed algorithm are also presented against the HM software and the conventional algorithms. Encoding time reduction (*TR*) is estimated to measure the complexity comparison of the proposed algorithm over the anchor reference software, which is defined by:

$$\mathrm{TR} = \frac{T_{\mathrm{HM}-14.0} - T_{\mathrm{Proposed}}}{T_{\mathrm{HM}-14.0}} \times 100\,(\%) \qquad (6)$$

where $T_{\text{HM-14.0}}$ and T_{Proposed} represent the encoding time of HM-14.0 and the proposed algorithm, respectively.

As shown in Table 5, three modules of the proposed algorithm (RMD, RDO, and RQT) achieve different BD increments and time savings. The proposed algorithm in RMD, by decreasing the number of prediction modes, achieves 20 % of time reduction on average with 0.2 % BD-rate loss. The proposed algorithm in RDO, based on the early termination of RDO stage, decreases the encoding time by about 34 %, but increases BD-rate of 1.5 %. Moreover, the proposed algorithm in RQT process reduces the encoding time by approximately 10 % without any loss in BD-rate performance. The proposed algorithms are compared against the HM-14.0 reference software. Based on the experimental results reported in Table 5, the proposed RDO algorithm provides the highest BD-rate increment and encoding time reduction. The proposed RMD and RQT algorithms yield almost similar performance in BD-rate and encoding time over the reference software.

Additionally, the encoding performance of the proposed context-adaptive intra coding algorithm of HEVC is listed in Table 6. It is reported that the proposed algorithm can save encoding time by about 40 % with BD-rate increment of about 1.3 % on average, against the default HM-14.0. For ‘RaceHorse’ test sequence of Class D, the lowest performance is observed in term of the complexity reduction at about 36 % with slight distortion in BD-rate performance. However, about 42 % of encoding time reduction is favorable for ‘Cactus’ test sequence in Class B with BD-rate increment of 1.1 %.

For further evaluation of the proposed algorithm, several conventional algorithms of the fast intra coding HEVC are compared. In this paper, three conventional algorithms [34, 38, 39] elaborated in Sect. 2 were implemented on HM-14.0. Table 7 shows the overall

Table 6 BD-rate and time saving of the proposed algorithm

Sequence		BD rate			Time reduction (%)
		Y (%)	*U* (%)	*V* (%)	
Class A (4 K)	Traffic	0.4	0.4	0.4	39
	People on street	0.7	0.6	0.5	40
	Nebuta festival	0.6	0.4	0.4	39
	Steam locomotive	0.6	0.3	0.5	39
Class B (1080p)	Kimono	1.2	0.7	0.6	39
	Park scene	0.3	0.1	0.2	41
	Cactus	1.1	0.9	1.1	42
	Basket ball drive	0.8	0.5	0.6	40
	BQ terrace	0.4	0.6	0.3	41
Class C (WVGA)	Basket ball drill	2.3	2.4	2.6	40
	BQ mall	1.4	1.4	1.3	41
	Party scene	1.0	1.0	1.1	40
	Race horses	0.2	−0.1	−0.1	39
Class D (WQVGA)	Basket ball pass	2.5	2.4	2.5	39
	Flower vase	2.4	1.8	1.7	40
	Blowing bubbles	1.2	1.0	1.1	40
	Race horses	2.2	2.1	2.1	36
Class E (720p)	Video 1	1.5	1.0	1.2	40
	Video 3	2.8	1.9	2.5	41
	Video 4	1.6	1.4	1.2	41
Average		1.3	1.1	0.7	40

Table 7 Overall comparisons of RD performance (ΔR) and encoding time (ΔT) between the conventional algorithms and the proposed algorithm implemented in HM-14.0

Seq	RMD optimization				RMD + RDO optimization						RMD + RDO + RQT optimization				[39] RMD + RDO + RQT + CU termination	
	[34]		Proposed RMD		[38]		[39] RMD + RDO		Proposed RMD + RDO		[39] RMD + RDO + RQT		Proposed algorithms			
	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)	ΔR (%)	ΔT (%)
A	0.5	25	0.0	18	1.7	44	2.0	17	0.4	31	2.0	20	0.6	39	1.2	31
B	0.4	25	0.0	19	2.1	45	0.4	16	0.4	32	0.6	20	0.8	41	1.7	31
C	0.4	22	0.2	22	3.3	41	0.5	13	0.5	32	0.9	21	1.2	40	1.0	24
D	0.6	25	0.5	19	4.0	42	0.7	16	0.8	30	1.0	22	2.1	38	1.2	29
E	0.4	24	0.2	18	2.6	44	0.5	15	0.8	32	0.9	19	2.0	41	1.5	34
Average	0.4	24	0.2	20	2.7	43	0.5	16	0.5	31	0.8	20	1.3	40	1.3	30

comparisons of RD performance (ΔR) and encoding time (ΔT) of the three conventional and the proposed algorithms. Although they were implemented on the same software model, their performances were observed to be almost similar to those in the original literatures. As shown in Table 7, overall Zhao's algorithm [34] yields the least BD-rate increment of 0.4 % with only small encoding time reduction of 24 %. Hao's algorithm [38] shows the best performance with the encoding time reduction of 43 %, but it also presents the highest BD-rate loss of 2.7 %. Zhan's algorithm [39] gives similar BD-rate increment of 1.3 % with the proposed algorithm. However, the proposed algorithm reports with better encoding time reduction of 10 %, compared to Zhan's algorithm. Based on these experimental results, the proposed context-adaptive fast intra coding algorithm of HEVC provides better performance improvements with the trade-off in coding efficiency and the encoding complexity.

In terms of profiling analysis, Table 8 shows the most active functions in encoding time for HEVC intra coding under the 'all-intra configuration'. The first 30 frames of one test sequence of Class A, 'NebutaFestival', were employed to evaluate the profiling results with QP of 22. According to Table 8, the proposed works are able to reduce the CPU time usages for the same function by a factor of two. For "Reconstruction of Intra Coding QT" and "Intra Coding Luma Block" functions, the proposed algorithm achieves the highest speed gain in CPU time (ΔTime) against the anchor. The proposed system reduces the CPU usage about 52 % for both functions in intra coding. Other two functions, "Estimation Intra Prediction QT" and "Rate Distortion Optimize Quantization," also show the significant suppression of about 45 % in CPU usage over the anchor. Figure 6 shows a histogram of the total CPU elapsed time. In the figure, we can see that the proposed algorithm can reduce the encoder complexity almost two times against the reference software.

Regarding evaluation of power consumption, several test sequences in Class C, Class D, and Class E for QP = 22 by using "Joulemeter" were conducted as shown in Table 9. One notebook computer with a charged battery was utilized in this experiments. As presented in the table, the proposed algorithm can efficiently reduce power usages up to 42 % for two sequences, 'Flowervase' in Class D and 'Video3' in Class E. In addition, our proposed algorithm is observed to save approximately 38 % on average over the anchor ones. The fluctuation of powers for both anchor and proposed systems per second is also evaluated. Based on this evaluation, the proposed algorithm can save power consumption with minimal power fluctuation over time, compared to those of the reference software of HM-14.0. In Fig. 7, the power impact of the first 30 frames of

Table 8 Profiling analysis between the anchor software HM-14.0 and the proposed algorithm

Functions	CPU time		
	Anchor (s)	Proposed (s)	ΔTime (%)
Check RD-cost intra	797.091	492.450	38
Estimation intra pred. QT	674.232	368.882	45
Reconstruction of intra coding QT	533.057	257.733	52
Intra coding luma Block	406.966	193.753	52
Rate distortion opt. quant	242.874	133.685	45
Prediction intra angular	52.803	38.495	27
Calculate Hadamard transf	48.394	41.66	14
Memory copy	39.872	24.641	38
Mode bits intra	19.432	12.060	38
Fill reference samples	14.726	10.500	29
Others	488.334	318.584	35

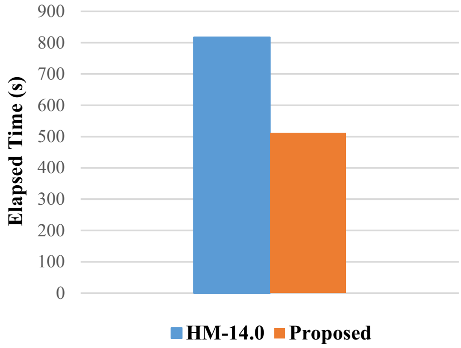


Fig. 6 Histogram of total CPU time usage

'NebutaFestival' test sequence in Class A with QP = 22 is depicted. The graph shows that the proposed algorithms are able to decrease CPU power utilization approximately 40 % over HM-14.0. The power consumption efficiency of our proposed algorithms is mainly provided aligned with the encoding time reduction. It is clearly illustrated that the proposed algorithm terminates earlier than the anchor. As consequence, it significantly saves the CPU power consumption almost two times. In addition, the proposed algorithm also shows that the power fluctuation over time is more static than that of anchor during the encoding process.

The proposed algorithm can be implemented in many platforms such as GPP, special purpose processors, and hardwired circuits. However, in targeting a real time for high resolution videos, many implementation issues should be also acknowledged depending on platforms. The hardwire implementation based on the proposed algorithm can be further studied to deal with the complexity of intra coding.

Table 9 Power consumption comparison between the anchor software HM-14.0 and the proposed algorithm

Sequence	Power consumption		Power difference (%)	Fluctuation (variance) of power consumption	
	Anchor (Watt)	Proposed (Watt)		Anchor	Proposed
Class C					
Basket ball drill	21208.8	12797.8	40	0.15	0.07
BQ mall	23831.9	14938.1	37	0.21	0.27
Party scene	25359.2	15856.1	37	0.30	0.30
Race horse C	12283.6	7890.1	36	0.21	0.07
Class D					
Basket ball pass	5042.4	3260.1	35	0.31	0.72
Blowing bubbles	6265.9	3957.2	37	0.25	0.26
Flower vase	2637.5	1531.4	42	0.25	0.27
Class E					
Video 1	43258.3	26795.1	38	0.16	0.21
Video 3	44586.2	26003.4	42	0.22	0.21
Video 4	42938.6	27125.5	37	0.61	0.38
Average			38	0.29	0.30

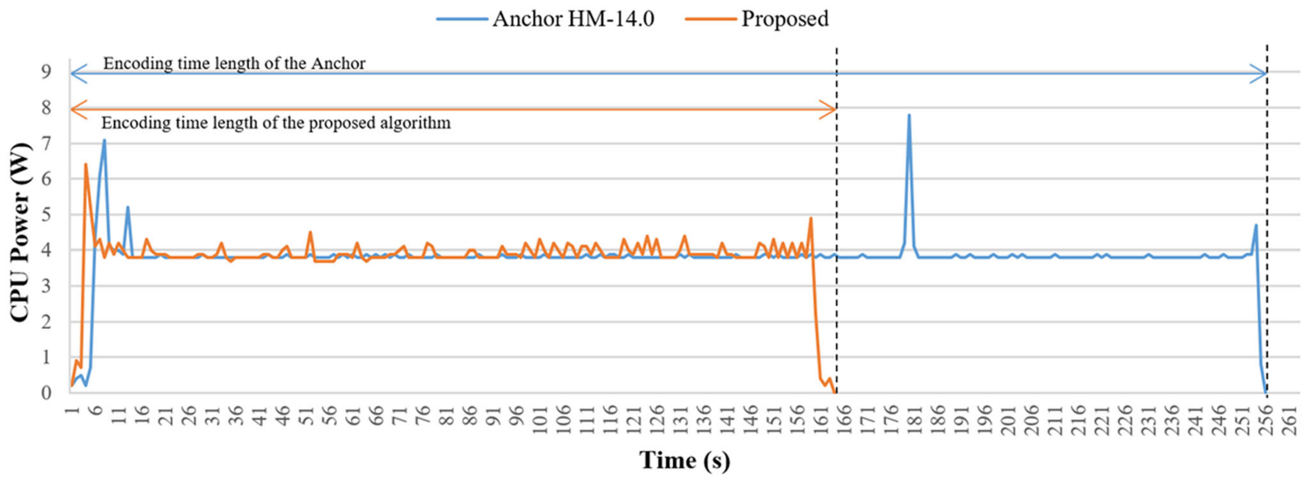


Fig. 7 CPU power consumption for the first 30 frames of 'NebutaFestival' test sequence

5 Conclusion

In this paper, a context-adaptive fast intra coding algorithm of HEVC is proposed to reduce the computational complexity of intra coding. An adaptive rough mode selection is proposed to reduce prediction modes in RMD stage, by utilizing the upper layer of CU and the neighboring PU blocks. The RDO skip is performed by estimating mean and normalizing standard deviation. The RQT skip and the early termination of TU split processed are conducted using a simple threshold derived from the average RD cost of the available neighboring coded CU blocks. The experimental results show that the proposed algorithm reduces encoding time on average about 40 % and BD-rate increases about 1.3 %, compared to HM-14.0. In addition, the proposed algorithm also reduces the CPU power usage in approximation up to 42 % over the reference software. For further work, the CU level optimization or the hardware implementation for intra coding might be considered to achieve better intra coding acceleration in HEVC.

Acknowledgments This research was partly supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (NRF-2014R1A2A1A11052210), the ICT R&D program of MSIP/IITP (B0101-15-295, Development of UHD Realistic Broadcasting, Digital Cinema, and Digital Signage Convergence Service Technology), and the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2015-H8501-15-1005) supervised by the IITP (Institute for Information & communications Technology Promotion).

References

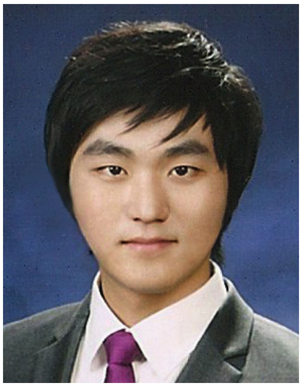
1. ITU-T SG16 Q6 and ISO/IEC JTC1/SC29/WG11 Coding of Moving Pictures and Audio: Joint call for proposals on videocompression technology. ITU-T SG16 Q6 document VCEG-AM91 and ISO/IEC JTC1/SC29/WG11 document N11113 (2010) [Online]. Available: http://www.itu.int/oth/T4601000002/en
2. Sullivan, G.J., Ohm, J.R.: Recent developments in standardization of high efficiency video coding (HEVC). In: Proc. 33rd SPIE Appl. Digital Image Process. 7798 (2010)
3. Wiegand, T., Ohm, J.R., Sullivan, G.J., Han, W.J., Joshi, R., Tan, T.K., Ugur, K.: Special section on the joint call for proposals on high Efficiency video coding (HEVC) standardization. IEEE Trans. Circuits Syst. Video Technol. **20**(12), 1661–1666 (2011)
4. Ahn, Y.J., Sim, D.: Square-type-first inter-CU tree search algorithm for acceleration of HEVC encoder. J. Real Time Image Process. (2015). doi:10.1007/s11554-015-0487-5
5. ITU-T Rec. H.265 and ISO/IEC 23008-2: High efficiency video coding. Final draft approval (2013) (formally published by ITU-T in June, 2013, and in ISO/IEC in Nov. 2013)
6. Choi, H., Yoo, J., Nam, J., Sim, D., Bajic, I.V.: Pixel-wise unified rate-quantization model for multi-level rate control. IEEE J. Sel. Topics Signal Process. **7**(6), 1112–1123 (2013)
7. Sullivan, G.J., Ohm, J.R., Han, W.J., Wiegand, T.: Overview of the high efficiency video coding (HEVC) standard. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1649–1668 (2012)
8. ITU-T, Recommendation H.265 (04/13): Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services–Coding of Moving Video, High Efficiency Video Coding. [Online]. Available: https://www.itu.int/rec/dologin_pub.asp?lang=f&id=T-REC-H.265-201410-I!!PDF-E&type=items
9. HEVC Reference Model-14th. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-14.0/
10. McCann, K., Bross, B., Han, W.J., Kim, I.K., Sugimoto, K., Sullivan, G.J.: High Efficiency Video Coding (HEVC) Test Model 14 (HM14) Encoder Description. Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-P1002 (2014)
11. Bossen, F., Drugeon, V., Francois, E., Jung, J., Kanumuri, S., Narroschke, M., Sasai, H., Sole, J., Suzuki, Y., Tan, T.K., Wedi, T., Wittmann, S., Yin, P., Zheng, Y.: Video coding using a simplified block structure and advanced coding techniques. IEEE Trans. Circuits Syst. Video Technol. **20**(12), 1667–1675 (2010)
12. Han, W.J., Min, J., Kim, I.K., Alshina, E., Alshin, A., Lee, T., Chen, J., Seregin, V., Lee, S., Hong, Y.M., Cheon, M.S., Shlyakhov, N., McCann, K., Davies, T., Park, J.H.: Improved video compression efficiency through flexible unit representation and corresponding extension of coding tools. IEEE Trans. Circuits Syst. Video Technol. **20**(12), 1709–1720 (2010)

13. Karczewicz, M., Chen, P., Joshi, R.L., Wang, X., Chien, W.J., Panchal, R., Reznik, Y., Coban, M., Chong, I.S.: A hybrid video coder based on extended macroblock sizes, improved interpolation, and flexible motion representation. IEEE Trans. Circuits Syst. Video Technol. **20**(12), 1698–1708 (2010)
14. Marpe, D., Schwarz, H., Bosse, S., Bross, B., Helle, P., Hinz, T., Kirchoffer, H., Lakshman, H., Nguyen, T., Oudin, S., Siekmann, M., Suhring, K., Winken, M., Wiegand, T.: Video compression using nested quadtree structures, leaf merging, and improved techniques for motion representation and entropy coding. IEEE Trans. Circuits Syst. Video Technol. **20**(12), 1676–1687 (2010)
15. Ugur, K., Andersson, K., Fuldseth, A., Bjontegaard, G., Endresen, L.P., Lainema, J., Hallapuro, A., Ridge, J., Rusanovskyy, D., Zhang, C., Norkin, A., Priddle, C., Rusert, T., Samuelsson, J., Sjoberg, R., Wu, Z.: High performance, low complexity video coding and the emerging HEVC standard. IEEE Trans. Circuits Syst. Video Technol. **20**(12), 1688–1697 (2010)
16. Sze, V., Madhukar, B., Sullivan, G.J.: High efficiency video coding (HEVC): algorithms and architectures. Springer International Publishing, Switzerland (2014)
17. Ahn, Y.J., Hwang, T.J., Sim, D.G., Han, W.J.: Implementation of fast HEVC encoder based on SIMD and data-level parallelism. EURASIP Journal on Image and Video Processing **2014**(1), 1–19 (2014)
18. Jo, H., Sim, D.G.: Bitstream decoding processor for fast entropy decoding of variable length coding–based multiformat videos. Opt. Eng. **53**(6), 063102 (2014). doi:10.1117/1.OE.53.6.063102
19. Ryu, H., Ahn, Y.J., Mok, J.S., Sim, D.: Performance analysis of HEVC parallelization methods for high-resolution videos. IEIE Trans. Smart Process. Comput. **4**(1), 28–34 (2015)
20. Lee, H., Kang, J.W., Lee, J., Choi, S., Kim, J., Sim, D.G.: Scalable extension of HEVC for flexible high-quality digital video content services. ETRI J. **35**(6), 990–1000 (2013)
21. Kim, I., Min, J., Lee, T., Han, W.J., Park, J.: Block partitioning structure in the HEVC standard. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1697–1706 (2012)
22. Yuan, Y., Kim, I.K., Zheng, X., Liu, L., Cao, X., Lee, S., Cheon, M.S., Lee, T., He, Y., Park, J.H.: Quadtree based nonsquare block structure for inter frame coding in high Efficiency video coding. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1707–1719 (2012)
23. Sole, J., Joshi, R., Nguyen, N., Ji, T., Karczewicz, M., Clare, G., Henry, F., Duenas, A.: Transform Coefficient coding in HEVC. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1765–1777 (2012)
24. Ismail, M., Jo, H., Sim, D.G.: Fast intra mode decision for HEVC intra coding. In: IEEE International Symposium on Consumer Electronics (ISCE), pp. 1–2 (2014)
25. Correa, G., Assuncao, P., Agostini, L., Cruz, L.A.S.: Complexity scalability for real-time HEVC encoders. J. Real Time Image Process. (2014). doi:10.1007/s11554-013-0392-8
26. Zhang, H., Ma, Z.: Fast intra mode decision for high efficiency video coding (HEVC). IEEE Trans. Circuits Syst. Video Technol. **24**(4), 660–668 (2014)
27. Lainema, J., Bossen, F., Han, W.J., Min, J., Ugur, K.: Intra coding of the HEVC standard. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1792–1801 (2012)
28. Ma Jonghyun., Sim, D.: Early termination of block vector search for fast encoding of HEVC screen content coding IEIE Trans Smart Process. Comput. **3**(6), 388–392 (2014)
29. Norkin, A., Bjontegaard, G., Fuldseth, A., Narroschke, M., Ikeda, M., Andersson, K., Zhou, M., Auwera, G.V.: HEVC deblocking filter. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1746–1754 (2012)
30. Fu, C.M., Alshina, E., Alshin, A., Huang, Y.W., Chen, C.Y., Tsai, C.Y., Hsu, C.W., Lei, S.M., Park, J.H., Han, W.J.: Sample adaptive offset in the HEVC standard. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1755–1764 (2012)
31. Shi, Y., Au, O.C., Zhang, H., Zhang, X., Jia, L., Dai, W., Zhu, W.: Local saliency detection based fast mode decision for HEVC intra coding. In: IEEE International Workshop on Multimedia Signal Processing (MMSP), pp. 429–433 (2013)
32. Guilherme, C., Pedro, A., Luciano, A., Cruz, L.A.S.: Performance and computational complexity assessment of high-Efficiency video encoders. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1899–1909 (2012)
33. Bossen, F., Bross, B.: Šuhring, K., Flynn, D.: HEVC complexity and implementation analysis. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1685–1696 (2012)
34. Zhao, L., Zhang, L., Ma, S., Zhao, D.: Fast mode decision algorithm for intra prediction in HEVC. In: IEEE Visual Communications and Image Processing (VCIP), pp. 1–4 (2011)
35. Alwani, M., Johar, S.: A method for fast rough mode decision in HEVC. In: Data Compression Conference (DCC), pp. 476 (2013)
36. Zhang, M., Zhao, C., Xu, J.: An adaptive fast intra mode decision in HEVC. In: IEEE International Conference on Image Processing (ICIP), pp. 221–224 (2012)
37. Ismail, M., Ma, J., Sim, D.G.: Full depth RQT after PU decision for fast encoding of HEVC. In: The IEEE International Symposium on Consumer Electronics (ISCE), pp.1–2 (2014)
38. Zhang, H., Ma, Z.: Fast intra prediction for high efficiency video coding. Advances in Multimedia Information Processing-PCM **7674**, 568–577 (2012)
39. Zhang, H., Ma, Z.: Early termination schemes for fast intra mode decision in high Efficiency video coding. In: IEEE International Symposium on Circuits and Systems (ISCAS), pp. 45–48 (2013)
40. Zhu, S., Zhao, D.: Fast intra-prediction mode decision algorithm for high efficiency video coding. In: IEEE International Conference on Industrial Electronics and Applications (ICIEA), pp. 936–939 (2014)
41. Na, S., Lee, W., Yoo, K.: Edge-based fast mode decision algorithm for intra prediction in HEVC. In: IEEE International Conference on Consumer Electronics (ICCE), pp. 11–14 (2014)
42. Dhollande, N., Meur, O., Guillemot, C.: HEVC intra coding of ultra hd video with reduced complexity. In: IEEE International Conference on Image Processing (ICIP), pp. 41122–4126 (2014)
43. Liu, Y., Chen, Z., Fang, J., Chang, P.: SVM-based fast intra CU depth decision for HEVC. In: Data Compression Conference (DCC), pp. 458 (2015)
44. Shen, L., Zhang, Z., Liu, Z.: Effective CU size decision for HEVC intra coding. IEEE Trans. Image Process. **23**(10), 4232–4241 (2014)
45. Li, Y., Liu, Y., Yang, H., Yang, D.: Fast CU splitting and pruning method based on online learning for intra coding in HEVC. In: IEEE Visual Communications and Image Processing Conference (VCIP), pp. 450–453 (2014.)
46. Yang, M., Grecos, C.: Fast intra encoding decisions for high efficiency video coding standard. J. Real-Time Image process. (2014). doi:10.1007/s11554-014-0445-7
47. Min, B., Cheung, R.C.C.: A fast CU size decision algorithm for the HEVC intra encoder. IEEE Trans. Circuits Syst. Video Technol. **25**(5), 892–896 (2015)
48. Hu, N., Yang, E.H.: Fast mode selection for HEVC intra frame coding with entropy coding refinement based on transparent composite model. IEEE Trans. Circuits Syst. Video Technol. **25**(9), 1521–1532 (2015)
49. Jamali, M., Coulombe, S., Caron, F.: Fast HEVC intra mode decision based on edge detection and SATD costs classification. In: Data Compression Conference (DCC), pp. 43–52 (2015)
50. Khan, M.U.K., Shafique, M., Henkel, J.: Fast hierarchical intra angular mode selection for high efficiency video coding. In: IEEE International Conference on Image Processing (ICIP), pp. 3681–3685 (2014)

51. Khan, M.U.K., Shafique, M., Grellert, M., Mateus, Henkel, J.: Hardware-software collaborative complexity reduction scheme for the emerging HEVC intra encoder. In: Design, Automation and Test in Europe Conference & Exhibition (DATE), pp. 125–128 (2013)
52. Thisa. L.A.S., Cruz, L.A.S., Agostini, L.V.: HEVC intra mode decision acceleration based on tree depth levels relationship. In: Picture Coding Symposium (PCS), pp. 277–280 (2013)
53. Palomino, D., Cavichioli, E., Susin, A., Agostini, L., Shafique, M., Henkel, J.: Fast HEVC intra mode decision algorithm based on new evaluation order in the coding tree block. In: Picture Coding Symposium (PCS), pp. 209–212 (2013)
54. Coll, D.R, Adzic, V., Escribano, G.F, Kalva, H., Martinez, J.L., Cuenca, P.: Fast partitioning algorithm for HEVC Intra frame coding using machine learning. In: IEEE International Conference on Image Processing (ICIP), pp. 4112–4116 (2014)
55. Shi, W., Jiang, X., Song, T., Shimamoto, T.: Edge information based fast selection algorithm for intra prediction of HEVC. In: IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), pp. 17–20 (2014)
56. Huang, X., Jia, H., Wei, K., Liu, J., Zhu, C., Lv, Z., Xie, D.: Fast algorithm of coding unit depth decision for HEVC intra coding. In: IEEE Visual Communications and Image Processing Conference, pp. 458–461 (2014)
57. Wang, Y., Fan, X., Zhao, L., Ma, S., Zhao, D., Gao, W.: A fast intra coding algorithm for HEVC. In: IEEE International Conference on Image Processing (ICIP), pp. 4117-4121 (2014)
58. Zhang, M., Bai, H., Lin, C., Zhang, M., Zhao, Y.: Texture characteristics based fast coding unit partition in HEVC intra coding. In: Data Compression Conference (DCC), pp. 477 (2015)
59. Bossen, F.: Common HM test conditions and software reference configurations. Joint Collaborative Team on Video Coding (JCT-VC), JCT-VC-L1100 (2013)
60. Bjontegaard, G.: Improvements of the bd-psnr model. ITU-T Q.6/SG16 Document, VCEG-AI11 (2008)
61. Hanhart, P., Rerabek, M., Korshunov, P., Ebrahimi, T.: Subjective evaluation of HEVC intra coding for still image compression. Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-L0380 (2013)
62. Ugur, K., Lainema, J.: Updated results on HEVC still picture coding performance. Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-M0041 (2013)
63. Microsof Research, Joulemeter version 1.2 (2011). [Online]. Available: http://research.microsoft.com/en-us/downloads/fe9e10c5-5c5b-450c-a674-daf55565f794/default.aspx
64. Akramullah, S.: Digital video concepts, method, and metrics: quality, compression, performance, and power trade-off analysis. Apress (2014). [Online]. Available: http://link.springer.com/book/10.1007%2F978-1-4302-6713-3
65. Kansal, A., Zhao, F., Kothari, N., Bhattacharya, A.A.: Virtual machine power metering and provisioning. In: Proceedings of the 1st ACM Symposium on Cloud computing (SOCC'10), pp. 39–50, (2010)

Ismail Marzuki received the B.S. degree in Informatics from UIN Sultan Syarif Kasim Riau—Indonesia in 2011. And he entered to the Computer Engineering Department and joint to Image Processing Systems Lab of Kwangwoon University in 2013. His current research interests are high-efficiency video compression and next-generation video coding.

Jonghyun Ma received the B.S. degree in Computer Engineering from Kwangwoon University, Seoul, Republic of Korea, in 2013. Currently, he is working toward M.S degree in computer engineering at the same university. His current research interests are video coding and computer vision.



Yong-Jo Ahn received the B.S. and M.S. degree in Computer Engineering from Kwangwoon University, Seoul, Korea, in 2010 and 2012, respectively. He is working toward Ph.D. candidate at the same University. His current research interests are high-efficiency video compression, parallel processing for video coding and multimedia system.

Donggyu Sim received the B.S. and M.S. degrees in Electronic Engineering from Sogang University, Korea, in 1993 and 1995, respectively. He also received the Ph.D. degree at the same University, in 1999. He was with the Hyundai Electronics Co., Ltd. From 1999 to 2000, he involved in MPEG-7 standardization. He was a senior research engineer at Varo Vision Co., Ltd, working on MPEG-4 wireless applications from 2000 to 2002. He worked for the Image Computing Systems Lab. (ICSL) at the University of Washington as a senior research engineer from 2002 to 2005. He researched on ultrasound image analysis and parametric video coding. Since 2005, he has been with the Department of Computer Engineering at Kwangwoon University, Seoul, Korea. In 2011, he joined the Simon Frasier University, as a visiting scholar. He was elevated to an IEEE Senior Member in 2004. He is one of the main inventors in many essential patents licensed to MPEG-LA for HEVC standard. His current research interests are video coding, video processing, computer vision, and video communication.