

## Research Article

# Novel Intermode Prediction Algorithm for High Efficiency Video Coding Encoder

**Chan-seob Park, Gwang-Soo Hong, and Byung-Gyu Kim**

*Department of Computer Engineering, Sun Moon University, 100 Kalsan-ri, Tangjeong-myeon, Chungnam, Asan-si 336-708, Republic of Korea*

Correspondence should be addressed to Byung-Gyu Kim; [bg.kim@mpcl.sunmoon.ac.kr](mailto:bg.kim@mpcl.sunmoon.ac.kr)

Received 30 April 2014; Revised 18 June 2014; Accepted 18 June 2014; Published 30 June 2014

Academic Editor: Martin Reisslein

Copyright © 2014 Chan-seob Park et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The joint collaborative team on video coding (JCT-VC) is developing the next-generation video coding standard which is called high efficiency video coding (HEVC). In the HEVC, there are three units in block structure: coding unit (CU), prediction unit (PU), and transform unit (TU). The CU is the basic unit of region splitting like macroblock (MB). Each CU performs recursive splitting into four blocks with equal size, starting from the tree block. In this paper, we propose a fast CU depth decision algorithm for HEVC technology to reduce its computational complexity. In  $2N \times 2N$  PU, the proposed method compares the rate-distortion (RD) cost and determines the depth using the compared information. Moreover, in order to speed up the encoding time, the efficient merge SKIP detection method is developed additionally based on the contextual mode information of neighboring CUs. Experimental result shows that the proposed algorithm achieves the average time-saving factor of 44.84% in the random access (RA) at Main profile configuration with the HEVC test model (HM) 10.0 reference software. Compared to HM 10.0 encoder, a small BD-bitrate loss of 0.17% is also observed without significant loss of image quality.

## 1. Introduction

Recently, high efficiency video coding (HEVC) has been developed as a new video coding standard mainly focusing on the coding of ultrahigh definition (UHD) videos as the high resolution and high quality videos are getting more popular. HEVC is a draft video compression standard, a successor to H.264/MPEG-4 AVC [1]. Market and user demand for advanced video services with high quality standards is increasing. UHD has a resolution of 4 K to 8 K (such as  $3840 \times 2160$  and  $7680 \times 4320$ ), while HD is  $1920 \times 1080$ . The data rate for UHD is up to 4~16 times more than for HD video. This kind of data rate increase will put additional pressure on all types of networks and services. HEVC emerges to meet the new demand in UHD TV which has large resolution, high frame rates, and limited bandwidth. In particular, the variable block size prediction and compensation in H.264/AVC are a key factor which contributes to significant bit reduction with the same image quality. To improve performance of video coding efficiency, the ITU-T video coding experts group (VCEG) and the ISO/IEC moving picture experts

group (MPEG) recently formed the joint collaborative team on video coding (JCT-VC) [2]. The JCT-VC is finalizing the next-generation video coding standard, called high efficiency video coding.

Usually, adoption of large block types in video compression enables us to compress high resolution 3D video sequences effectively and it helps us to service video sequences on mobile and network economically [3–5].

The video encoding and decoding processes in HEVC are composed of three units: (a) a coding unit (CU) for the root of the transform quadtree, as well as a prediction mode for the INTER/SKIP/INTRA prediction, (b) a prediction unit (PU) for coding the mode decision, including motion estimation (ME) and rate-distortion optimization, and (c) a transform unit (TU) for transform coding and entropy coding. Initially, a frame is divided into a sequence of the nonoverlapped largest coding units, called a coding tree unit (CTU). A CTU can be recursively divided into smaller coding units (CU) and made flexible using quadtree partitioning, which is called a coding tree block (CTB). It is clear that the new CTB structure with larger coding block size in HEVC greatly increases the

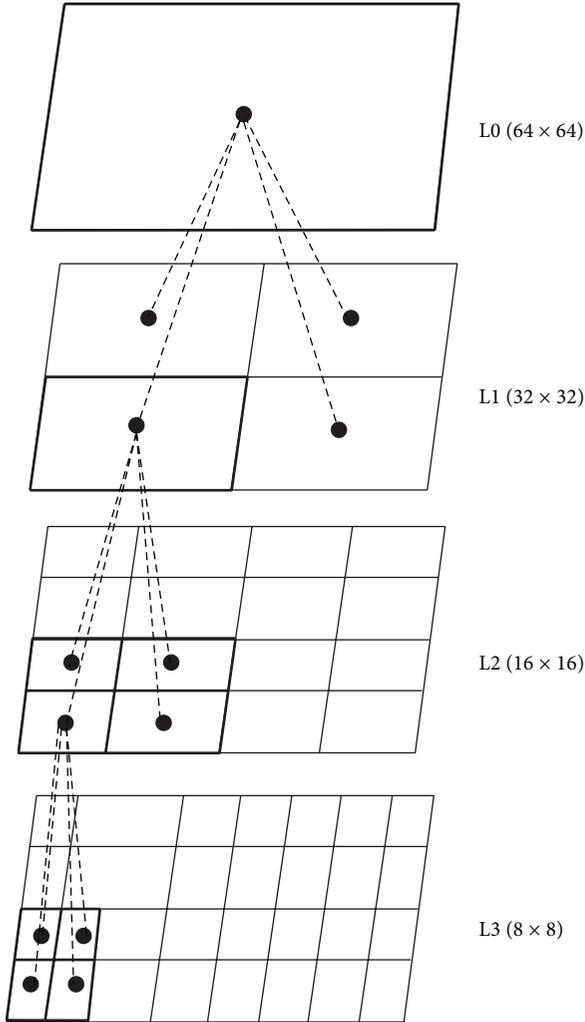


FIGURE 1: Hierarchical structure of HEVC.

computational complexity to achieve the high efficiency of coding gain in HEVC standard, comparing to the H.264/AVC video standard.

In Figure 1, it is shown that a  $64 \times 64$  size of CTU is divided into smaller dimensions of CUs. A CTU has the dimension of  $64 \times 64$  which can be decomposed into four  $32 \times 32$  CUs and further each  $32 \times 32$  CU can be divided into 4 CUs of  $16 \times 16$  dimension. This decomposition process can go up to  $8 \times 8$  dimension of CU. It means the size of  $8 \times 8$  is the lowest dimension that a CU can have. Moreover, for the different combinations of CU structures, different CTBs are generated for a single CTU. For each CTB rate-distortion (RD) cost value is calculated.

To reduce the computational complexity of the HEVC, there are several algorithms with high speed-up factor keeping negligible losses of BD-bitrate. In [6], an early termination scheme called coded block flags (CBFs) fast mode (CFM) decision used the CBF of luminance and 6 chrominance and in order to reduce the complexity of the intermode decision. If the CBFs of luminance and chrominance are both zeros, the

search process for the next PU modes in the current depth level is not performed.

Kiho and Jang [7] proposed a tree-pruning algorithm that makes an early determination of CU. To reduce computational complexity, it uses mode information of the current CU. When the best PU mode of the current CU selects the SKIP mode, the current CU is not divided into sub-CUs in the subdepth level of the current CU. This process was adopted in HEVC test model 4.0 reference software [8].

Zhang et al. [9] proposed algorithm by reducing the depth search range. Based on the depth information correlation between spatiotemporal adjacent CUTs and the current CTU, some depth can be adaptively excluded from the depth search process in advance. Similar to [6, 7], Yang et al. [10] proposed an early detection algorithm for the SKIP mode. Their motivation was early determination of skip conditions from fast method decision schemes in the H.264/AVC [11–20]. They utilized differential motion vectors (DMV) and coded block flags (CBFs) of the inter  $2N \times 2N$  mode as skip conditions. In [21], an adaptive coding unit selection method has been proposed based on early SKIP detection technique.

Yang et al. [22] have proposed the advanced motion vector prediction (AMVP) mode skip decision method utilizing the correlated tendency of PU mode. This method checks whether the best prediction mode of the inter  $2N \times 2N$  is skip mode or not. If it is the skip mode, the rest of subprediction (i.e.,  $N \times 2N$ ,  $2N \times N$ ,  $N \times N$ , and asymmetric subpartitions) units omit the AMVP process.

Vanne et al. [23] proposed a method based on three techniques: symmetric motion partition (SMP) mode decision, asymmetric motion partition (AMP) mode decision, and quantization parameter (QP) specific mode decision. This algorithm examined the rate-distortion-complexity (RDC) characteristics of the HEVC interprediction and used the obtained RDC results to optimize the mode decision and associated block partition.

To reduce the computational complexity of HEVC encoding system, we propose an effective CU selection algorithm for HEVC based on RD cost of  $2N \times 2N$ . In the proposed algorithm, we use also early merge SKIP mode detection technique based on correlation of neighboring CUs including depth level.

The rest of the paper is organized as follows. In Section 2, we present the suggested algorithm including merge SKIP detection. Simulation results and some discussion will be given in Section 3. Concluding comments are given in Section 4.

## 2. Proposed Fast Intermode Prediction Algorithm

*2.1. Adaptive CU Depth Decision Method.* In order to enhance the encoding speed, the depth information of CTU is found by using  $2N \times 2N$  PU information. Table 1 indicates the probability that  $2N \times 2N$  PU is determined as the best mode. In this table, you can see the probability of  $2N \times 2N$  PU determined is very high. It shows almost 90% in Class A and 79% in Class B case. This means that  $2N \times 2N$  PU decision

TABLE 1: The probability of  $2N \times 2N$  PU as the best mode.

Class A		Class B		Class C		Class D	
Traffic	85.7%	Kimono	82.1%	BasketballDrill	86.3%	BasketballPass	86.5%
PeopleOnStreet	87.3%	ParkScene	76.4%	BQMall	89.6%	BQSquare	78.7%
Nebuta	92.1%	Cactus	82.3%	PartyScene	77.8%	BlowingBubbles	81.2%
SteamLocomotive	91.2%	BasketballDrive	73.7%	RaceHorses	79.1%	RaceHorses	74.8%
—	—	BQTerrace	80.9%	—	—	—	—
Average	89%	Average	79%	Average	83.2%	Average	80.3%

has large portion. From this result, if we find a fast scheme to decide it early, then the overall consumed time for encoding may be decreased effectively.

In this study, the experiment environment to calculate  $2N \times 2N$  PU selection probability was set as the sequences from Class A for Class D and 50 frames for each sequence. Also, the selection probability has been computed by averaging of various QP values (22, 27, 32, 37).

As shown in Table 1, the maximum 91.2% and minimum 73.7% of  $2N \times 2N$  PU portion are shown. By using the depth information selected from  $2N \times 2N$  PU, we are able to predict the proper CU in the relevant depth. When the  $2N \times 2N$  is given as input in the current frame, depth information of each CU is calculated in CTU [2]. The depth is defined as CTU in the quadtree structure that represents the split of the CU.

After finding the residual with the best motion vector obtained through  $2N \times 2N$  PU, finally, the RD is calculated through the full RQT. By using this depth information, the depth of CTU is determined through the RD cost competition. The current depth ( $x$ ) with the best RD cost will be determined as the current depth level. If the RD cost of depth ( $x + 1$ ) is smaller than that of the current depth ( $x$ ), the calculation to find the minimum RD cost is repeated recursively for upper depth. After that, the best subpartitioned mode is determined by calculating and comparing the RD cost from the depths  $2N \times 2N$ ,  $2N \times N$ , and  $N \times 2N$  to go to the detailed partition search.

Figure 2 illustrates the overall procedure of the proposed CU depth decision. The proposed algorithm is performed as follows. Firstly, as input, if the  $2N \times 2N$  PU size is given at the current depth ( $x$ ), our algorithm calculates the RD cost from the given  $2N \times 2N$  PU size. After that, the best RD cost of the current depth ( $x$ ) is compared with the accumulated RD cost of next depth ( $x + 1$ ). The accumulated RD cost can be calculated from the previous encoded modes (PUs). If the given condition is satisfied, the current CU depth is selected as the best depth level. Otherwise, go to next CU depth ( $2N \times 2N$ ) and perform in the same manner.

In the flowchart, after selecting the best CU depth, the detailed partition mode is determined by calculating and comparing the RD cost from the depths  $2N \times 2N$ ,  $2N \times N$ , and  $N \times 2N$  as the detailed partition search.

**2.2. Early Merge SKIP Decision Method.** We also develop an early merge SKIP decision to increase the encoding. According to the SKIP of HEVC standard, the merge SKIP has been adopted [2] for providing more coding efficiency.

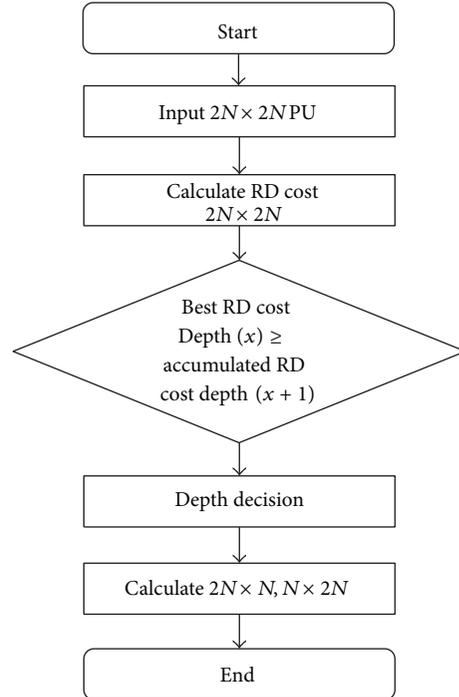


FIGURE 2: Flowchart of the proposed CU depth decision.

The proposed merge SKIP detection method utilizes the information of neighboring blocks.

Figure 2 shows the position of adjacent CUs relative to the current CU. Neighboring CUs such as CU1, CU2, and CU3 (above-left, above, and left CU from the current CU) have a high degree of spatial correlation. CU4 and CU5 are used as CUs for temporal correlations, and CU6 and CU7 are used for depth correlations. Each encoding unit has the information of the flag generated in  $2N \times 2N$ , in  $2N \times 2N$ . After checking the merge flags, merge SKIP mode is decided.

When encoding the current block, the spatial and temporal neighboring blocks can provide useful information, because they have much similarity in terms of texture and motion. In HEVC, depth concept has been introduced as described in Section 2. So, the optimal block mode of the current block can also be deduced from the neighboring blocks which is composed of the spatial, temporal, and depth relationship.

In Figure 4, the proposed early CU SKIP decision method is displayed. When  $2N \times 2N$  is the current block in usual

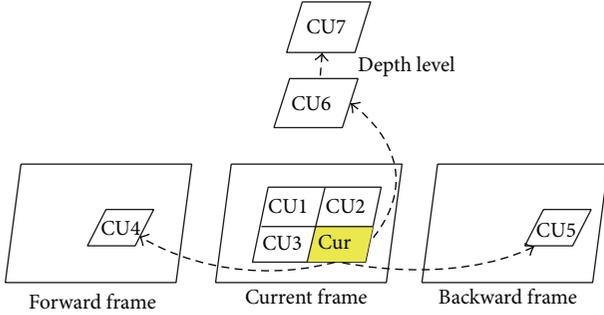


FIGURE 3: Relationship between the current and adjacent blocks.

HEVC coding, the merge process of motion information is performed to achieve more coding gain. In our previous work [24], an early merge skip decision method has been proposed using only  $2N \times 2N$  block. In addition to this, we develop the extended version based on the RD cost and comparing with CBF from  $2N \times N$  and  $N \times 2N$  partition types.

In the merge process, the proposed method checks on the mode types of spatial, temporal, and depth neighboring blocks (as shown in Figure 3) in the first. If all modes of neighboring blocks are SKIPS, then SKIP is selected as the best mode for merge process. The remaining mode search is omitted directly. Otherwise, a partition mode is determined by calculating the RD cost and comparing with CBF (coded block flag) from  $2N \times N$  and  $N \times 2N$  partition types. With the merge SKIP detection technique, the complexity of the motion estimation can be more reduced while keeping similar image quality.

### 3. Experimental Results and Discussion

To verify the performance of the proposed algorithm, the experiment was performed with various standard video sequences. The proposed algorithm was implemented on HM 10.0 (HEVC reference software) [25]. Test condition was random access using RA-Main. Standard sequences with 100 frames were used from Classes A to D [26] with various QP values (22, 27, 32, 37). All frames of each sequence are selected from the 0th frame frame to the 99th frame like other works. Sequences of each class (from A to D) are illustrated in Table 2.

To evaluate performance, we defined the measurement of  $\Delta\text{Bit}$ ,  $\Delta\text{PSNR}_Y$ , and  $\Delta\text{Time}$  as

$$\Delta\text{Bit} = \text{Bit}_{\text{proposed}} - \text{Bit}_{\text{original}} \quad (1)$$

$$\Delta\text{PSNR}_Y = \Delta\text{PSNR}_{Y\text{proposed}} - \text{PSNR}_{Y\text{original}} \quad (2)$$

$\Delta\text{Bit}$  is the total bit difference between the compared methods (see (1)).  $\Delta\text{PSNR}_Y$  represents the difference in quality variation between the average PSNR of the proposed method and the corresponding values of an anchor (HEVC reference software) (see (2)).

TABLE 2: Test sequences.

Class	Resolution	Name	Frame-rate
A	2560 × 1600	Traffic	30 fps
		PeopleOnStreet	30 fps
		Nebuta	60 fps
		SteamLocomotive	60 fps
		Kimono	24 fps
B	1920 × 1080	ParkScene	24 fps
		Cactus	50 fps
		BasketballDrive	50 fps
		BQTerrace	60 fps
		BasketballDrill	30 fps
C	832 × 480	BQMall	60 fps
		PartyScene	50 fps
		RaceHorses	50 fps
D	416 × 240	BasketballPass	30 fps
		BQSquare	60 fps
		BlowingBubbles	50 fps
		RaceHorses	50 fps

$\Delta\text{Time}$  is a complexity comparison factor to indicate the amount of total encoding time saving as follows:

$$\Delta\text{Time} = \frac{\text{Time}_{\text{Proposed}} - \text{Time}_{\text{Anchor}}}{\text{Time}_{\text{Anchor}}} \times 100, \quad (3)$$

where  $\text{Time}_{\text{proposed}}$  and  $\text{Time}_{\text{Anchor}}$  represent the time of the proposed method and the original method, respectively.

The results in Table 3 show the performance of our algorithm when comparing it to the original HM 10.0 encoder. The proposed algorithm achieves 44.84% of time-saving factor on average with only 0.04 (dB) loss in PSNR and a 0.17% increment in total bits. For Class A, about 46% of time-saving factor was observed with very small loss of quality. For smaller size of image (Class D), the speed-up gain is slightly decreased, but the quality loss is still negligible.

The proposed algorithm increases the speed of the HEVC encoding system up to 48.54% in the Traffic sequence, compared to the full mode search. Compared to Li's method, the proposed algorithm achieved speed-up gain of up to 13% with a smaller bit increment for the ICE sequence. By using the proposed algorithm, we can see that the average speed-up gain of over 44% was obtained comparing to the full mode search while suffering less quality loss and negligible bit rate increment.

In terms of bit rate, the proposed method achieved small increment of BDBR bit number (almost 0.17%). For the SteamLocomotive sequence, bit increment of up to 0.23% was obtained. From this result, the proposed algorithm is efficient to make real-time encoding system.

Regarding the quality of image, the proposed method gave very small loss of quality. But very large loss of image quality was observed from the original HM encoder (0.04 (dB) in average value). In the SteamLocomotive and Cactus sequences, maximum loss of quality was given by 0.07 (dB) with the proposed method. This means that the

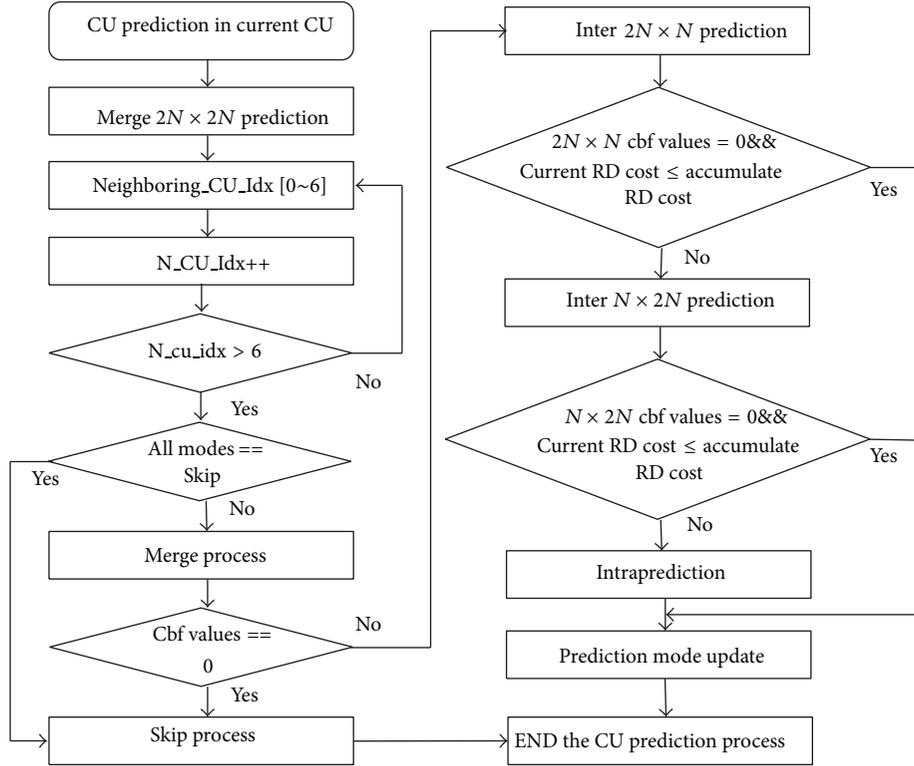


FIGURE 4: Flowchart for the fast CU detection method.

TABLE 3: The performance of the proposed fast scheme.

Test Seqs.	$\Delta\text{Bit}$ (%)	$\Delta\text{PSNR}_Y$ (dB)	$\Delta\text{Time}$ (%)
Traffic	0.15%	-0.02	48.54%
A PeopleOnStreet	0.21%	-0.03	44.74%
Nebuta	0.12%	-0.04	45.47%
StramLocomotive	0.23%	-0.07	47.62%
Kimono	0.19%	-0.04	45.81%
ParkScene	0.15%	-0.05	42.86%
B Cactus	0.22%	-0.07	40.11%
BasketballDrive	0.17%	-0.03	43.84%
BQTerrace	0.18%	-0.03	41.21%
BasketballDrill	0.14%	-0.05	47.35%
C BQMall	0.17%	-0.04	48.22%
PartyScene	0.12%	-0.07	42.67%
RaceHorses	0.21%	-0.02	43.43%
BasketballPass	0.13%	-0.03	46.92%
D BQSquare	0.10%	-0.02	43.26%
BlowingBubbles	0.08%	-0.04	45.38%
RaceHorses	0.15%	-0.02	41.87%
Average	0.17%	-0.04	-44.84%

proposed algorithm can provide credible quality with large speed-up factor.

Table 4 shows results for the performance comparison between Yang's algorithm [22] and the proposed algorithm.

Both algorithms have been implemented on HM 10.0 with same condition. The Yang's algorithm achieves 20.92% of time-saving factor on average with only 0.02 (dB) loss in  $\text{PSNR}_Y$  and 0.15% of decrement in total bits. The proposed algorithm gave about 45% of time-saving factor while keeping small loss of quality. When compared with the Yang's algorithm, bit loss of the proposed algorithm is slightly high but it is negligible. The speed-up factor was faster by amount of over 24% than Yang's method.

For all classes, the proposed method achieves over 20% of speed-up factor comparing to Yang's method. In terms of quality, just 0.04 (dB) of the quality loss was observed on the average value. These results mean that the proposed algorithm is credible in terms of the speed-up factor and the quality loss.

Figure 5 shows the rate-distortion (RD) performance [27]. The performance of the proposed method is very similar to the HM 10.0 software. There is negligible loss of quality. For the Traffic sequence and BasketballDrive sequences, the proposed method gives almost similar performance to the original HM encoder. This means that the suggested algorithm can keep a reliable video quality with speeding up the HM encoder by about 44.84%. Up to 15 Mbps, we can see that the performance of the proposed algorithm is credible in terms of bitrate and image quality.

To check on the subjective evaluation of the original image and proposed algorithm, the decoded picture has been shown in Figures 6 and 7. In terms of visual quality, we can see there is almost no difference between two images. From this

TABLE 4: The performance comparison with Yang et al.'s method [22] on HM reference software 10.0.

	Yang et al.'s [22]			Proposed		
	$\Delta\text{Bit}$ (%)	$\Delta\text{PSNR}_Y$ (dB)	$\Delta\text{Time}$ (%)	$\Delta\text{Bit}$ (%)	$\Delta\text{PSNR}_Y$ (dB)	$\Delta\text{Time}$ (%)
Class A	-0.19%	-0.01	20.71%	0.17%	-0.04	46.59%
Class B	-0.14%	-0.02	23.18%	0.18%	-0.04	42.74%
Class C	-0.21%	-0.02	21.47%	0.16%	-0.04	45.41%
Class D	-0.08%	-0.02	18.35%	0.11%	-0.02	44.35%
AVG	-0.15%	-0.02	20.92%	0.17%	-0.04	44.84%

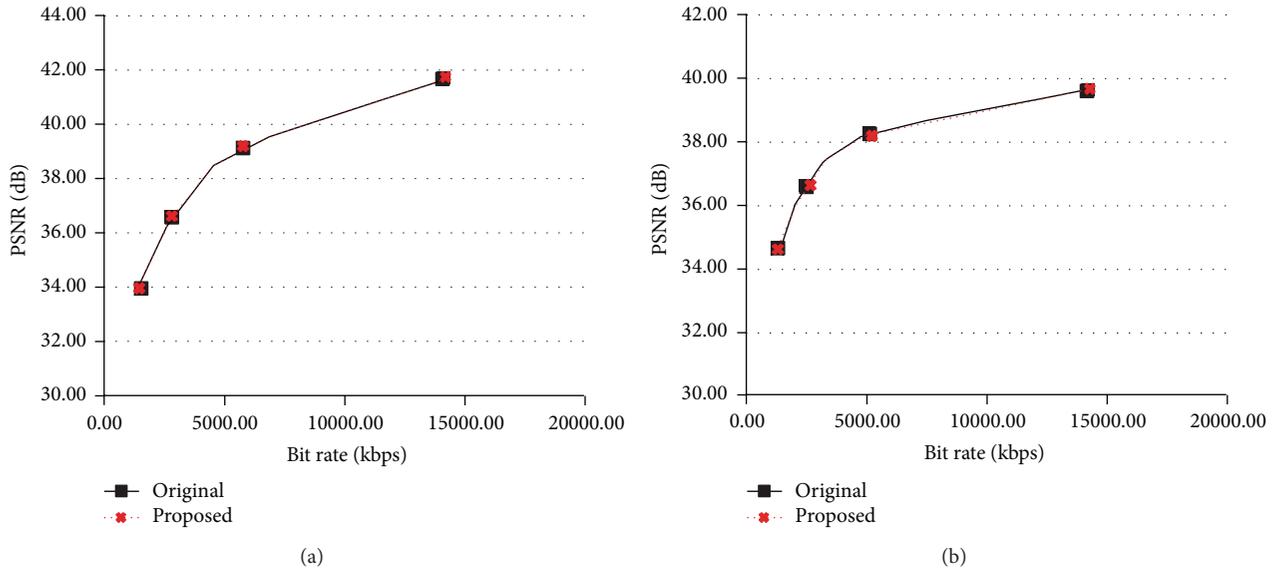


FIGURE 5: Rate-distortion (RD) curves for Class A: (a) Traffic and (b) BasketballDrive sequences.



FIGURE 6: Class B (BQTerrace-QP 27, the 30th frame): (a) the original picture and (b) the decoded picture by the proposed algorithm.

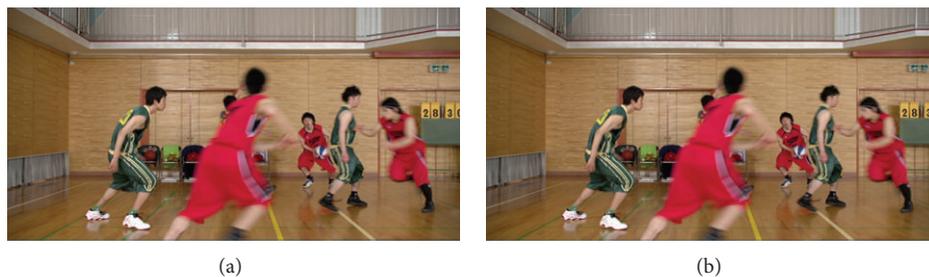


FIGURE 7: Class B (BasketballDrive-QP 22, the 75th frame): (a) the original picture and (b) the decoded picture by the proposed algorithm.

result, we can deduce that the proposed algorithm is able to keep very high quality for very large scale video service (Full or Ultra HD video).

#### 4. Conclusions

In this paper, we have proposed a fast CU depth decision algorithm based on the RD cost comparison for high efficiency video coding (HEVC) technology to reduce its computational complexity. In addition, merge SKIP extraction method was developed and integrated with CU depth decision algorithm. Experimental result shows that the proposed algorithm achieves the average time-saving factor of 44.84% in the random access (RA) at Main profile configuration with HM 10.0 reference software while keeping small loss of quality. From experimental results, we can make a conclusion that the suggested algorithm can be a useful way to make real-time video encoding system for large scale multimedia service.

#### Conflict of Interests

The authors declare that there is no conflict of interests.

#### Acknowledgment

This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2010-0024786).

#### References

- [1] T. Wiegand and G. J. Sullivan, "The H.264/AVC video coding standard," *IEEE Signal Processing Magazine*, vol. 24, no. 2, pp. 148–153, 2007.
- [2] G. J. Sullivan, J. R. Ohm, W. J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.
- [3] E. Song, H. Kim, and Y. Jeong, "Visual monitoring system of multi-hosts behavior for trustworthiness with mobile cloud," *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 347–358, 2012.
- [4] J. C. Tsai and N. Y. Yen, "Cloud-empowered multimedia service: an automatic video storytelling tool," *Journal of Convergence*, vol. 4, no. 3, pp. 13–19, 2013.
- [5] Y.-S. Ho, "Challenging technical issues of 3D video processing," *Journal of Convergence*, vol. 4, no. 1, pp. 1–6, 2013.
- [6] Y.-L. Lee and J. Lim, "Early termination of CU encoding to reduce HEVC complexity," in *Proceedings of the JCTVC-F045, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 6th Meeting*, JCTVC, Torino, Italy, July 2011.
- [7] C. Kiho and E. S. Jang, "Fast coding unit decision method based on coding tree pruning for high efficiency video coding," *Optical Engineering Letters*, vol. 51, no. 3, Article ID 030502, 2012.
- [8] K. Choi, S.-H. Park, and E. S. Jang, "Euee S.Jang, Coding tree pruning based CU early termination, presented at," in *Proceedings of the Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 6th Meeting*, JCTVC-F092, pp. 14–22, JCTVC, Torino, Italy, July 2011.
- [9] Y. Zhang, H. Wang, and Z. Li, "Fast coding unit depth decision algorithm for interframe coding in HEVC," in *Proceedings of the Data Compression Conference (DCC '13)*, pp. 53–62, March 2013.
- [10] J. Yang, J. Kim, K. Won, H. Lee, and B. Jeon, "Early SKIP detection for HEVC," Tech. Rep. JCTVC-G543, JCT-VC, 2011.
- [11] B. Jeon and J. Lee, "Fast method decision for H.264," JVT-J033, Hawaii, USA, 2003.
- [12] J. Lee and B. Jeon, "Fast mode decision for H.264," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '04)*, pp. 1131–1134, June 2004.
- [13] I. Choi, J. Lee, and B. Jeon, "Fast coding mode selection with rate-distortion optimization for MPEG-4 Part-10 AVC/H.264," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 12, pp. 1557–1561, 2006.
- [14] S. Liquan, L. Zhi, Z. Zhaoyang, and S. Xuli, "Fast inter mode decision using spatial property of motion field," *IEEE Transactions on Multimedia*, vol. 10, no. 6, pp. 1208–1214, 2008.
- [15] M. Paul, W. Lin, C. T. Lau, and B. Lee, "Direct intermode selection for H.264 video coding using phase correlation," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 461–473, 2011.
- [16] H. Zeng, C. Cai, and K.-K. Ma, "Fast mode decision for H.264/AVC based on macroblock motion activity," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 4, pp. 491–499, 2009.
- [17] J. Kim, B. Kim, and C. Cho, "A fast block mode decision algorithm using a block correlation in H.264/AVC," in *Proceedings of the IEEE International Symposium on Consumer Electronics (ISCE '11)*, pp. 1–5, June 2011.
- [18] D. Wu, F. Pan, K. P. Lim et al., "Fast intermode decision in H.264/AVC video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 6, pp. 953–958, 2005.
- [19] B. Feng, G. Zhu, and W. Liu, "Fast adaptive inter-prediction mode decision method for H.264 based on spatial correlation," in *Proceedings of the International Symposium on Circuits and Systems (ISCAS '06)*, pp. 1804–1807, IEEE, May 2006.
- [20] M. Yang and W. Wang, "Fast macroblock mode selection based ON motion content classification in H.264/AVC," in *Proceedings of the International Conference on Image Processing (ICIP '04)*, vol. 2, pp. 741–744, IEEE, October 2004.
- [21] J. Kim, S. Jeong, S. Cho, and J. S. Choi, "Adaptive Coding Unit early termination algorithm for HEVC," in *Proceedings of the IEEE International Conference on Consumer Electronics (ICCE '12)*, pp. 261–262, Las Vegas, Nev, USA, January 2012.
- [22] S. Yang, H. Lee, H. J. Shim, and B. Jeon, "Fast inter mode decision process for HEVC encoder," in *Proceedings of the IEEE 11th Image, Video, and Multidimensional Signal Processing Technical Committee Workshop (IVMSP '13)*, pp. 1–4, Seoul, Republic of Korea, June 2013.
- [23] J. Vanne, M. Viitanen, and T. Hamalainen, "Efficient mode decision schemes for HEVC intra prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, 2014.
- [24] C.-S. Park, B.-G. Kim, G.-S. Hong, and S.-K. Kim, "Fast Coding Unit (CU) depth decision algorithm for High Efficiency Video Coding (HEVC)," in *Proceedings of the 5th FTRA International Conference on Computer Science and Its Applications (CSA '13)*, Lecture Notes in Electrical Engineering, pp. 293–299, Danang, Vietnam, 2013.

- [25] K. McCann, B. Bross, W.-J. Han, I.-K. Kim, K. Sugimoto, and G. J. Sullivan, "High Efficiency Video Coding (HEVC) test model 10 (HM10) encoder description," in *Proceedings of the Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 12th Meeting*, Geneva, Switzerland, January 2013.
- [26] F. Bossen, "Common test conditions and software reference configurations," Joint Collaborative Team on Video Coding (JCTVC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Document: JCTVC-B300, 2nd Meeting, Geneva, Switzerland, 2010.
- [27] G. Bjontegaard, "Calculation of average PSNR differences between RD Curves," ITU-T SG16 Q.6 Document, VCEG-M33, Austin, Tex, USA, April 2001.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

