

Research Article

An Effective Transform Unit Size Decision Method for High Efficiency Video Coding

Chou-Chen Wang,¹ Chi-Wei Tung,¹ and Jing-Wein Wang²

¹ Department of Electronic Engineering, I-Shou University, Kaohsiung 84001, Taiwan

² Institute of Photonics and Communications, National Kaohsiung University of Applied Sciences, Kaohsiung 80778, Taiwan

Correspondence should be addressed to Chou-Chen Wang; chchwang@isu.edu.tw

Received 25 February 2014; Accepted 13 April 2014; Published 8 May 2014

Academic Editor: Her-Terng Yau

Copyright © 2014 Chou-Chen Wang 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.

High efficiency video coding (HEVC) is the latest video coding standard. HEVC can achieve higher compression performance than previous standards, such as MPEG-4, H.263, and H.264/AVC. However, HEVC requires enormous computational complexity in encoding process due to quadtree structure. In order to reduce the computational burden of HEVC encoder, an early transform unit (TU) decision algorithm (ETDA) is adopted to pruning the residual quadtree (RQT) at early stage based on the number of nonzero DCT coefficients (called NNZ-ETDA) to accelerate the encoding process. However, the NNZ-ETDA cannot effectively reduce the computational load for sequences with active motion or rich texture. Therefore, in order to further improve the performance of NNZ-ETDA, we propose an adaptive RQT-depth decision for NNZ-ETDA (called ARD-NNZ-ETDA) by exploiting the characteristics of high temporal-spatial correlation that exist in nature video sequences. Simulation results show that the proposed method can achieve time improving ratio (TIR) about 61.26%~81.48% when compared to the HEVC test model 8.1 (HM 8.1) with insignificant loss of image quality. Compared with the NNZ-ETDA, the proposed method can further achieve an average TIR about 8.29%~17.92%.

1. Introduction

With the rapid development of electronic technology, the panels of $4K \times 2K$ (or $8K \times 4K$) high-resolution will become the main specification of large size digital TV in the future. However, the currently state-of-the-art video coding standard H.264/AVC is difficult to support the video applications of high definition (HD) and ultrahigh definition (UHD) resolution [1]. Therefore, the ITU-T video coding experts group (VCEG) and ISO/IEC moving pictures expert group (MPEG) through their joint collaborative team on video coding (JCT-VC) have developed the newest high efficiency video coding (HEVC) for video compression standard to satisfy the UHD requirement in 2010 [2], and the first version of HEVC was approved as ITU-T H.265 and ISO/IEC 23008-2 by JCT-VC in January 2013 [3–5].

HEVC can achieve an average bit rate decrease of 50% in comparison with H.264/AVC high profile while still maintaining the same subjective video quality [6]. This is

because HEVC adopts some new coding structures including coding unit (CU), prediction unit (PU), and transform unit (TU). The CU is the basic unit of region splitting used for inter-/intraprediction, which allows recursive subdividing into four equally sized blocks. The CU can be split by coding quadtree structure of 4 level depths, in which CU size ranges from largest CU (LCU: 64×64) to the smallest CU (SCU: 8×8) pixels. The PU is the basic unit used for carrying the information related to the prediction processes, and the TU can be split by residual quadtree (RQT) at maximally 3 level depths which vary from 32×32 to 4×4 pixels. The relationship among the CU, PU, and TU is shown in Figure 1.

In general, intra CUs have two types of PUs including $2N \times 2N$ partition and $N \times N$ partition, but inter CUs have four types of PUs including $2N \times 2N$, $2N \times N$, $N \times 2N$, and $N \times N$. Therefore, HEVC encoder enables 7 different partition modes including SKIP, inter $2N \times 2N$, inter $2N \times N$, inter $N \times 2N$, inter $N \times N$, intra $2N \times 2N$, and intra $N \times N$ for interslice as shown in Figure 2. The rate distortion (RD) cost

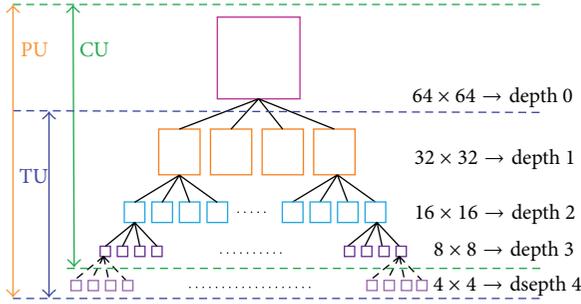


FIGURE 1: The relationship among the CU, PU, and TU.

under all partition mode and all CU sizes has to be calculated by performing the PUs and TUs to select the optimal CU size and partition mode. For an LCU, all the PUs and available TUs listed in Table 1 have to be exhaustively searched by rate-distortion optimization (RDO) process and this causes dramatically increased computational complexity compared with H.264/AVC. This “try all and select the best” method will result in the high computational complexity and limit the use of HEVC encoders in real-time applications.

To reduce the computational burden of HEVC encoder, there have been many fast encoding methods proposed to reduce the number of CUs and PUs to be tested [7–12]. Most existing fast HEVC encoding approaches are to check all-zero block (AZB) or motion homogeneity or RD cost or coding tree pruning to skip motion estimation (ME) on unnecessary CU sizes. In HEVC, every possible CU size is tested in order to estimate the coding performance of each CU, and the coding performance is determined by the CU size compared with the corresponding PUs and TUs. Among these coding units, TU is the basic unit used for the transform and quantization processes. The TU process recursively partitions the structure of a given prediction block from PU into transform blocks that are represented by the residual quadtree (RQT). Therefore, in addition to above-mentioned fast CU size decision methods, the early TU decision algorithm (ETDA) is another method selected to reduce the encoding complexity of HEVC [13, 14]. Recently, a new ETDA is proposed by pruning the RQT at early stage based on the number of nonzero DCT coefficients (called NNZ-EDTA) to accelerate the encoding process of TU module [15]. However, the NNZ-ETDA cannot effectively reduce the computational load for sequences with active motion or rich texture. Therefore, in order to further improve the performance of NNZ-ETDA, we propose an adaptive RQT-depth decision for NNZ-ETDA (called ARD-NNZ-ETDA) by exploiting the characteristics of high temporal-spatial correlation that exist in nature video sequences. Firstly, we analyse and calculate the temporal and spatial correlation values from the temporal (colocated) and spatial (left, upper, and left upper) neighbouring blocks of the current TU, respectively. And then the dynamic depth level range of RQT for the current TU is predicted by using the correlation weight and maximum depth levels of its neighbouring blocks. Finally, we combine the proposed

TABLE 1: PU and TU sizes in HEVC.

Depth	Inter PU	Intra PU	TU
0	$64 \times 64, 64 \times 32, 32 \times 64$	64×64	$32 \times 32, 16 \times 16$
1	$32 \times 32, 32 \times 16, 16 \times 32$	32×32	$32 \times 32, 16 \times 16, 8 \times 8$
2	$16 \times 16, 16 \times 8, 8 \times 16$	16×16	$16 \times 16, 8 \times 8, 4 \times 4$
3	$8 \times 8, 8 \times 4, 4 \times 8$	$8 \times 8, 4 \times 4$	$8 \times 8, 4 \times 4$

adaptive RQT depth and the NNZ-ETDA to further reduce the computational load of TU module.

The rest of the paper is organized as follows. Section 2 introduces the RQT structure of HEVC. In Section 3 we briefly review the method of NNZ-EDTA. In Section 4 we describe the proposed effective TU size decision method. The results of experiment are shown in Section 5. Finally, Section 6 shows the conclusions of this study.

2. RQT Structure of HEVC

2.1. Integer Transform and Quantization of HEVC. The forward integer transform in HEVC is an approximation of DCT specified as a matrix multiplication. The integer DCT is

$$\mathbf{Y} = \mathbf{C}\mathbf{X}\mathbf{C}^T, \quad (1)$$

where \mathbf{X} is the residual block, \mathbf{Y} is the transformed coefficient matrix, and \mathbf{C} is a core transformation matrix [16]. Equation (2) shows a general form of an $N \times N$ block transformed by integer DCT in HEVC. Consider

$$\mathbf{Y}_{N \times N} = \mathbf{C}(\mathbf{X}_{N \times N} \mathbf{C}^T \otimes \text{PF}_1) \otimes \text{PF}_2, \quad (2)$$

where all of the above matrices are with the size of $N \times N$. The symbol \otimes stands for a scalar multiplication. PF_1 and PF_2 are postscaling factors defined as follows:

$$\begin{aligned} \text{PF}_1: y &= (x + (1 \ll (M + B - 10))) \gg (M + B - 9), \\ \text{PF}_2: y &= (x + (1 \ll (M + 5))) \gg (M + 6), \end{aligned} \quad (3)$$

where y and x are the intermediate sample value in \mathbf{Y} and \mathbf{X} , respectively, $M = \log_2 N$, and B indicates the internal bit depth of video signals [17]. All computations in (2) can be implemented by only using addition and shift operations. The main reason is that transformed coefficients can be represented with 16 bits to avoid the computation burden in hardware. After integer DCT, the transformed coefficient is quantized by the operations defined as follows:

$$\mathbf{Q} = (\mathbf{Y} \times \text{QM}(\text{QP}\%6) + n \ll (\text{Qbits} - 9)) \gg \text{Qbits}, \quad (4)$$

where \mathbf{Q} is the quantized coefficient matrix, QP is a quantization parameter ranging from 0 to 51, and \ll and \gg are the binary left-shift and right-shift operators, respectively. The constant n is an offset for quantization, which is 171 and 85 for intra- and intercoding, respectively. The quantization matrix (QM) is defined as follows:

$$\text{QM}(\text{QP}\%6) = \{26214, 23302, 20560, 18396, 16384, 14564\}, \quad (5)$$

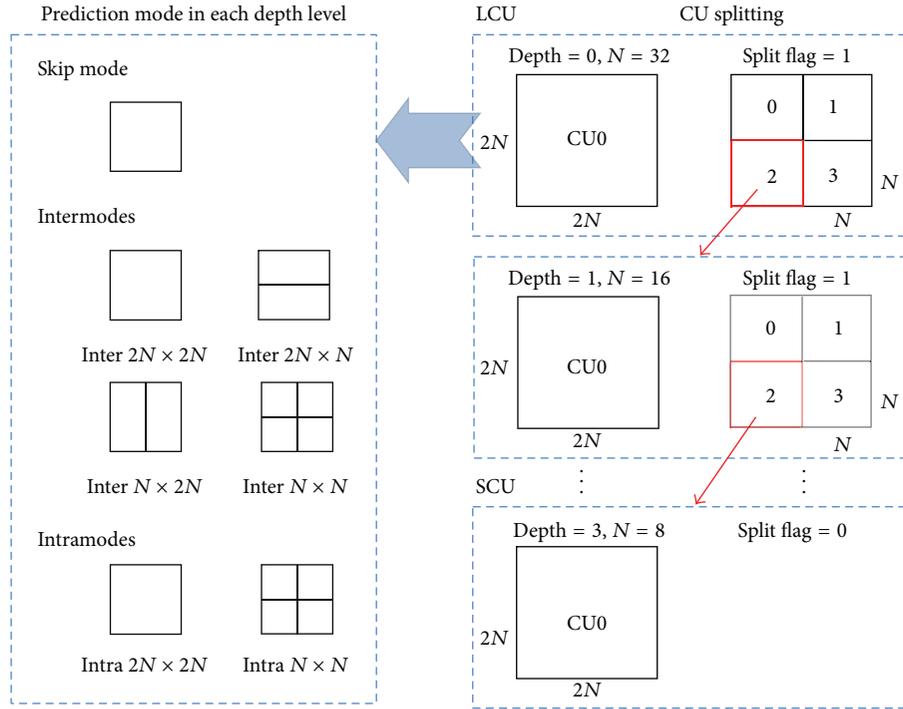


FIGURE 2: Recursive CU structure, PU splitting for skip, and inter- and intramodes.

where % is the mod operation. The relationship between Qbits and QP can be derived as follows:

$$Qbits = 21 + \left\lceil \frac{QP}{6} \right\rceil - \log_2 N, \quad (6)$$

where $\lceil \cdot \rceil$ denotes rounding to the nearest integer.

2.2. Best RQT Structure of HEVC. Similar to an LCU, a TU is also recursively partitioned into smaller TUs using a quadtree structure. The residual samples corresponding to a CU are to be subdivided into smaller units using a RQT and the leaf nodes of the RQT is referred to as TUs. In order to achieve the optimal coding performance, the full RQT needs to be pruned to obtain best RQT structure of TUs by comparing the RD cost between the upper layer TU and the lower layer sub-TUs from bottom to top. The minimization process of the RD cost is the well-known RDO measurement. The RD cost function is defined as follows:

$$J = D + \lambda R, \quad (7)$$

where D is the distortion, R is the bit rates, and λ is the Lagrangian multiplier depending on QP. When the prediction errors are transmitted to the transform and quantization steps, the bit rates can be estimated through entropy coding, and the distortion can be estimated after the inverse quantization and inverse transform steps. RD cost is the main evaluation indicator for achieving the optimal performance. For example, the optimal decision-making is described in Figure 3 when given a CU size of 32×32 . The leaf nodes ($a \sim j$) of the RQT represent the final mode decision of the chosen size of each TU.

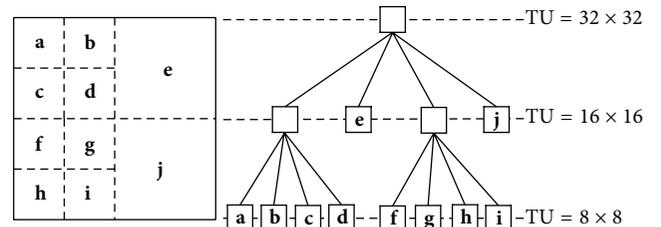


FIGURE 3: Example of RQT for dividing given coding tree block.

Although the coding efficiency in HEVC can be improved by using varying transform block sizes (from 32×32 to 4×4), the computational complexity increased dramatically in terms of the transform kernel size and the transform coding structure [3]. This is because the RQT selects the best TU partition mode by checking the RD cost of all possible TUs in all the RQT depths. From the example shown in Figure 3, we can find that the RD cost evaluation is performed a number of times within each RQT structure: once for the TU size of 32×32 , four times for the TU size of 16×16 , and 16 times for the TU size of 8×8 .

3. Brief Review of NNZ-EDTA

In order to further speed up the encoding process of HEVC, Chio and Jang proposed an early TU decision method for fast video encoding by pruning the TUs at an early stage based on the number of nonzero DCT coefficients (NNZ-EDTA) [15]. They find that if a reasonable cost is found at an early stage, the HEVC encoder is not able to skip the remaining

TABLE 2: $P_r(X_N | NZ_N = k)$ of determined TU sizes according to NNZ in traffic.

QP	Conditional probability $P_r(X_N NZ_N = k)$ (%)											
	$N = 32$				$N = 16$				$N = 8$			
	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 0$	$k = 1$	$k = 2$	$k = 3$
22	98.5	62.2	49.1	40.1	90.9	74.5	68.9	63.3	94.4	84.8	75.4	71.9
27	94.0	65.0	58.2	44.8	95.5	85.2	75.0	68.6	98.0	88.8	81.0	77.2
32	96.5	81.0	68.9	54.7	97.6	87.1	78.5	73.5	99.1	90.7	82.0	79.3
37	97.8	80.8	65.9	57.6	98.5	88.9	79.3	74.8	99.6	94.3	85.8	80.3
Average	96.7	72.3	60.5	49.3	95.6	83.9	75.4	70.0	97.8	89.7	81.1	77.2

TABLE 3: $P_r(X_N | NZ_N = k)$ of determined TU sizes according to NNZ in people on street.

QP	Conditional probability $P_r(X_N NZ_N = k)$ (%)											
	$N = 32$				$N = 16$				$N = 8$			
	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 0$	$k = 1$	$k = 2$	$k = 3$	$k = 0$	$k = 1$	$k = 2$	$k = 3$
22	81.5	62.5	53.5	40.4	84.3	80.7	71.2	67.5	92.1	90.1	83.6	81.0
27	84.5	63.3	49.8	38.6	88.5	81.5	71.0	65.2	95.1	92.0	83.8	80.3
32	85.8	77.1	53.9	40.6	91.5	86.6	75.8	70.1	97.2	94.1	86.4	81.6
37	88.7	73.7	54.0	41.3	94.1	88.9	78.7	72.9	98.6	94.8	87.4	81.2
Average	85.1	69.2	52.8	40.2	89.6	84.4	74.2	68.9	95.8	92.8	85.3	81.0

RQT processes by using subtree computations. For example, if the TU size of the root node (i.e., $TU = 32 \times 32$ in Figure 3) is determined to be the best TU size after the RQT process, the computations related to subtree RQT processes for finding the best TU size (i.e., $TU = 16 \times 16$ and 8×8 in Figure 3) would be unnecessary. On the other hand, they also find that the determined TU size and the number of nonzero DCT coefficients (NNZ) are strongly correlated. Based on two observations, they proposed an efficient way to stop the RD cost evaluation below the root node without sacrificing the coding efficiency.

To evaluate and analyze the performance of the new EDTA, we first implement the NNZ-EDTA in the HEVC test model 8.1 (HM 8.1) [18]. Tables 2 and 3 show the conditional probability of traffic and people on street sequences as shown in Figure 4, respectively. In these tables, $P_r(X_N | NZ_N = k)$ denotes the conditional probability of the selection of a root node (X_N) given by NNZ ($NZ_N = k$), where N is the TU size of the root node. NNZ is selected as a threshold to stop further RD cost evaluation based on balancing the computational complexity reduction and the compression efficiency loss. From Tables 2 and 3, we find that the correlation between the determined TU size and NNZ is very low when the threshold of NNZ is set to $k = 3$. Moreover, from the experimental results, we further find that the NNZ-EDTA cannot effectively reduce the computational load for sequences with fast motion or rich texture.

4. Proposed Fast TU Decision Method

HM usually adopts the maximum TU size equal to 32×32 and maximally 3 depth levels whose depth range is from 1 to 4. From the statistical analysis in [3], we can find that there are different depth level distributions for sequences. For sequences that contain a large area with high activities,

the possibility of selecting depth level “1” is very low. For sequences containing a large area with low activities, more than 70% blocks select the depth level “1” as the optimal levels. These results show that small depth levels are always selected at TUs in the homogeneous region, and large depth levels are selected at TUs with active motion or rich texture. The depth range should be adaptively determined based on the residual block property.

Natural video sequences have strongly spatial and temporal correlations, especially in the homogeneous regions. The optimal RQT depth level of a certain TU is the same or very close to the depth level of its spatially adjacent blocks due to the high correlation between adjacent blocks. Therefore, we firstly analyze and calculate the temporal and spatial correlation values of maximum RQT depth from the temporal (colocated: Col) and spatial (left: L, upper: U, and left upper: L-U) neighboring blocks of the current TU as shown in Figure 5. Table 4 shows the probability of the same maximum RQT depth among the temporal (Col) and spatial (L, U, and L-U) neighboring blocks of the current TU.

To speed up RQT pruning, we make use of spatial and temporal correlations to analyse region properties and skip unnecessary TU sizes. Specifically, the optimal RQT depth level of a block is predicted using spatial neighbouring blocks and the colocated block at the previously coded frame is as follows:

$$\text{Depth}_{\text{pred}} = \sum_{i=0}^{N-1} \omega_i \cdot d_i, \quad (8)$$

where N is the number of blocks equal to 4, d_i is the value of depth level, and ω_i is the weight determined based on correlations between the current block and its neighbouring blocks. The four weights are normalized to have $\sum_{i=0}^{N-1} \omega_i = 1$. Table 4 shows the statistical analysis of the same maximum



FIGURE 4: Test sequences (2560 × 1600). (a) Traffic and (b) people on street.

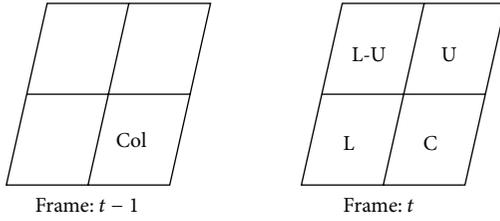


FIGURE 5: Temporal and spatial correlations of TUs.

RQT depth under different sequences. From Table 4, we can observe that the colocated block in the previously coded frame and spatial neighbouring blocks have almost the same correlation of the maximum RQT depth for the current TU after normalization. Therefore, the weights for four neighbouring blocks are set to $\omega_i = 0.25$.

According to the predicted value of the optimal RQT depth, each block is divided into five types as follows.

- (1) If $\text{depth}_{\text{pred}} \leq 1$, its optimal RQT depth is chosen to “1” and the TU size is 32×32 . The dynamic depth range (DR) of current TU is classified as Type 0.
- (2) If $1 < \text{depth}_{\text{pred}} \leq 1.5$, its optimal RQT depth is chosen to “2” and the TU size is $32 \times 32 \sim 16 \times 16$. The dynamic DR of current TU is classified as Type 1.
- (3) If $1.5 < \text{depth}_{\text{pred}} \leq 2.5$, its optimal RQT depth is chosen to “3” and the TU size is $32 \times 32 \sim 8 \times 8$. The dynamic DR of current TU is classified as Type 2.
- (4) If $2.5 < \text{depth}_{\text{pred}} \leq 3.5$, its optimal RQT depth is chosen to “4” and the TU size is $16 \times 16 \sim 4 \times 4$. The dynamic DR of current TU is classified as Type 3.
- (5) If $\text{depth}_{\text{pred}} > 3.5$, its optimal RQT depth is chosen to “4” and the TU size is $8 \times 8 \sim 4 \times 4$. The dynamic DR of current TU is classified as Type 4.

Based on the above analysis, the candidate depth levels that will be tested using RDO for each TU are summarized in Table 5. Therefore, the dynamic depth level range of RQT of the current TU can be predicted by using the correlation weight and maximum depth levels of its neighboring blocks. The flowchart of the proposed fast TU decision method is shown in Figure 6. Finally, we combine the proposed adaptive depth of RQT and the NNZ-ETDA to further reduce the computational load of TU. In other words, we propose an

TABLE 4: Probability of the same maximum RQT depth (%).

Sequences	Col	L	L-U	U
Traffic	56.24	60.17	53.21	56.53
People on street	72.87	68.08	63.40	66.70
Kimono	40.93	43.51	34.80	39.71
Park scene	59.03	60.69	54.38	62.71
Cactus	52.49	62.39	55.58	60.78
Basketball drive	49.20	58.50	44.96	49.98
BQ terrace	60.22	66.38	56.84	60.66
Average (%)	55.85	59.96	51.88	56.72
Normalized	0.25	0.26	0.24	0.25

adaptive RQT depth to set the candidate depth levels before performing the NNZ-ETDA.

The flowchart of proposed adaptive RQT-depth decision for NNZ-EDTA (ARD-NNZ-EDTA) implemented in HEVC HM 8.1 is shown in Figure 7. The proposed ARD-NNZ-EDTA firstly utilizes an adaptive RQT-depth decision algorithm to limit the depth range of RQT and then employs the NNZ-ETDA to exclude the unnecessary calculations of RDO for TUs in RQT.

5. Experiment Results

For the performance evaluation, we assess the total execution time of the proposed method in comparison to those of the HEVC HM 8.1 [18, 19] and the NNZ-ETDA (threshold $k = 3$) in order to confirm the reduction in computational complexity. The coding performance is evaluated based on $\Delta\text{Bitrate}$, ΔPSNR , and ΔTime , which are defined as follows:

$$\Delta\text{Bitrate} = \frac{\text{Bitrate}_{\text{method}} - \text{Bitrate}_{\text{HM8.1}}}{\text{Bitrate}_{\text{HM8.1}}} \times 100\%,$$

$$\Delta\text{PSNR} = \text{PSNR}_{\text{method}} - \text{PSNR}_{\text{HM8.1}}, \quad (9)$$

$$\Delta\text{Time} = \frac{\text{Time}_{\text{HM8.1}} - \text{Time}_{\text{method}}}{\text{Time}_{\text{HM8.1}}} \times 100\%.$$

The system hardware is Intel (R) Core (TW) CPU i5-3350P at 3.10 GHz, 8.0 GB memory, and Windows XP 64-bit O/S. Additional details of the encoding environment are described in Table 6.

Table 7 shows the TU processing time ($\text{QP} = 22$) of the proposed method and the HEVC HM 8.1, respectively.

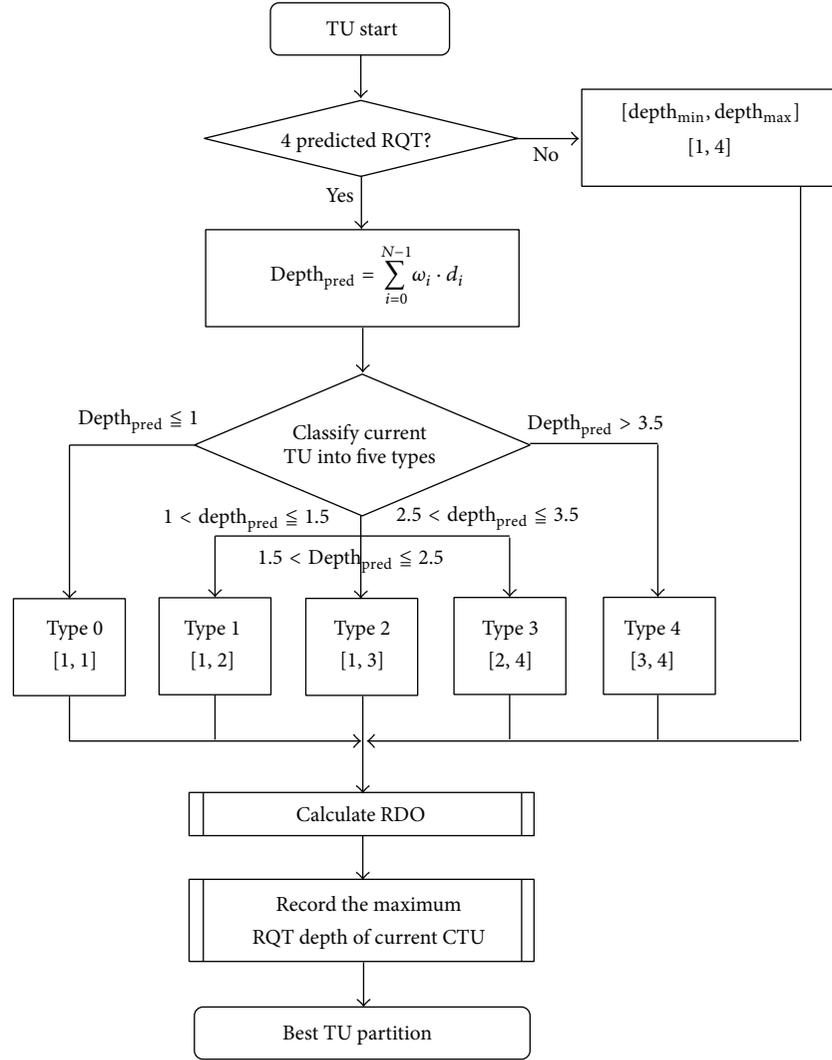


FIGURE 6: Flowchart of the proposed fast TU decision method.

TABLE 5: Candidate depth levels for each RQT type.

RQT type	Depth _{pred}	Depth range [depth _{min} , depth _{max}]	TU size
Type 0	Depth _{pred} ≤ 1	[1, 1]	32 × 32
Type 1	1 < depth _{pred} ≤ 1.5	[1, 2]	32 × 32~16 × 16
Type 2	1.5 < depth _{pred} ≤ 2.5	[1, 3]	32 × 32~8 × 8
Type 3	2.5 < depth _{pred} ≤ 3.5	[2, 4]	16 × 16~4 × 4
Type 4	Depth _{pred} > 3.5	[3, 4]	8 × 8~4 × 4

As shown in Table 7, the TU processing time was effectively improved to $\Delta T_{im} = 61.26\%$ compared with that of HM 8.1 on average with insignificant loss of image quality. On the other hand, Table 8 also shows the TU processing time of the proposed method and the NNZ-EDTA with the same scenario and QP value, respectively. We also find that our method can further achieve time improving ratio (ΔT_{im}) to 17.92% as compared with the NNZ-EDTA. It is clear that the proposed method indeed efficiently reduces the

computational complexity in TU module with insignificant loss of image quality. In addition, the average time improving ratio (TIR) using different QP values is shown in Figure 8. Figures 8(a) and 8(b) show the TIR varying of the proposed method compared with HM 8.1 and NNZ-EDTA, respectively. From Figure 8(a), we can observe that the encoding time improving is more efficient when the value of QP increases. However, from Figure 8(b), we can find that the encoding time improving is decreasing when the value of QP increases.

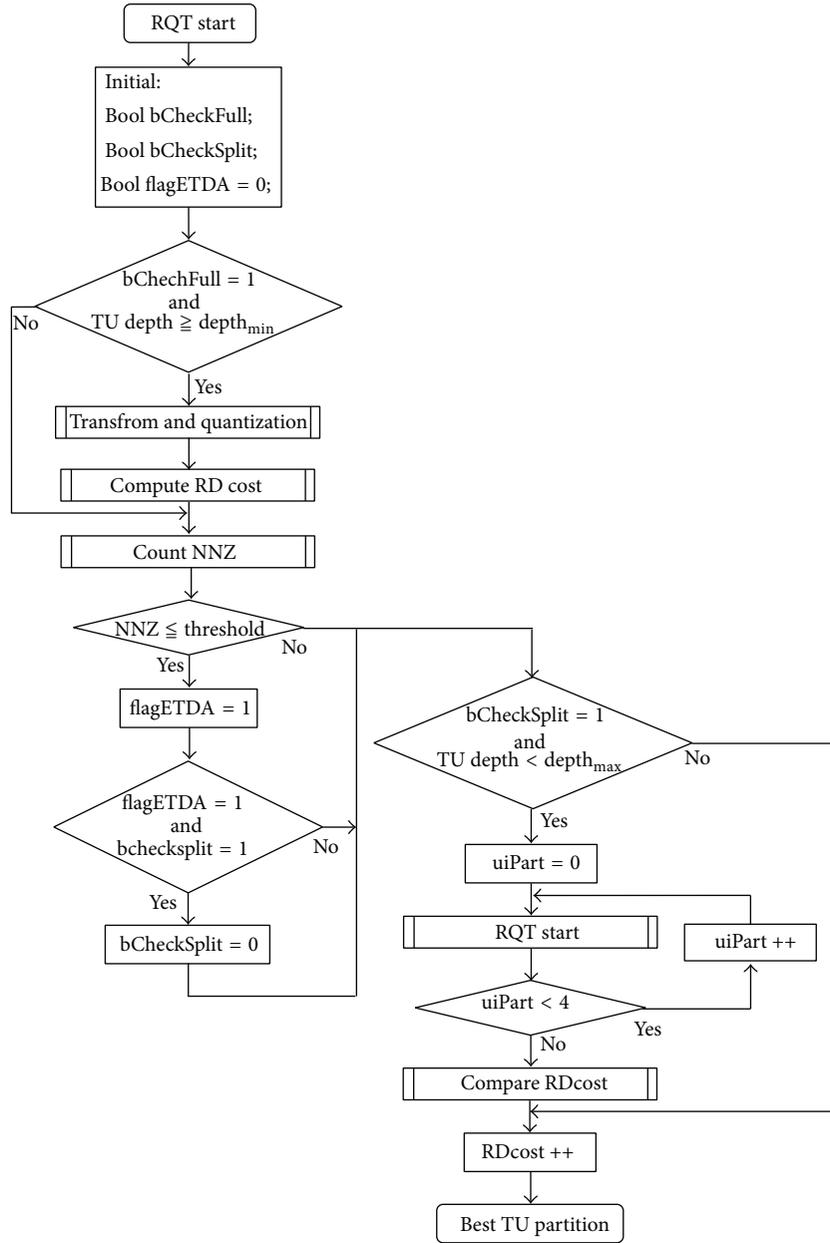


FIGURE 7: Flowchart of the proposed ARD-NNZ-EDTA in HEVC HM 8.1.

TABLE 6: Test conditions and software reference configurations.

Test sequences	(i) Class A (2560 × 1600): traffic and people on street (ii) Class B (1920 × 1080): kimono, park scene, cactus, basketball drive, and BQ terrace
Total frames	100 frames
Quantization parameter	22, 27, 32, and 37
Software	HM 8.1
Scenario	Random access (RA) at main profile (MP)

Figures 9(a) and 9(b) show the TU processing time for traffic and people on street sequences tested by different

methods with different QP values, respectively. As shown in Figure 9, the proposed method can improve the TU encoding time under different QPs. However, the encoding time is close to that of NNZ-ETDA when the value of QP increases. This is because the quantization error is too large that it results in the lower temporal-spatial correlation.

6. Conclusions and Further Works

We propose an adaptive RQT depth for NNZ-ETDA by exploiting the characteristics of high temporal-spatial correlation that exists in nature video sequences. An adaptive depth of RQT is employed to the NNZ-ETDA to further reduce the computational load of TU. The proposed method

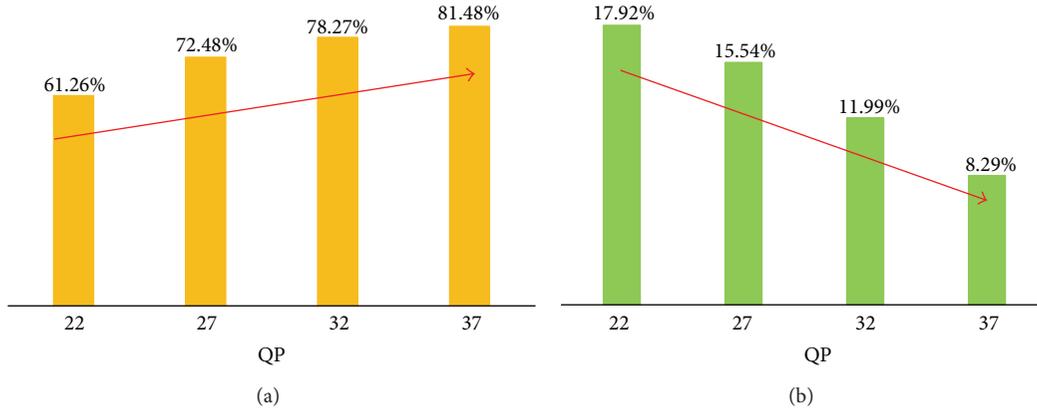


FIGURE 8: Average time improving ratio using different QP values: (a) proposed versus HM 8.1 and (b) proposed versus NNZ-EDTA.

TABLE 7: Results of comparison between HM 8.1 and proposed method.

Sequence (QP = 22)	HM 8.1			Proposed			Comparison		
	Bit rate (kb/s)	PSNR (dB)	Time (sec)	Bit rate (kb/s)	PSNR (dB)	Time (sec)	Δ Bit rate (%)	Δ PSNR (dB)	Δ Time (%)
Traffic	15823	41.75	10329	15644	41.66	3152	-1.13	-0.09	69.49
People on street	34367	40.29	14881	34491	40.19	7201	0.36	-0.10	51.61
Kimono	6174	41.88	7511	6233	41.86	2860	0.96	-0.02	61.93
Park scene	8724	40.08	5465	8669	40.01	1936	-0.63	-0.07	64.58
Cactus	23165	38.49	5997	23195	38.46	2306	0.13	-0.03	61.56
Basketball drive	13719	39.82	6072	14175	39.80	2134	3.32	-0.02	64.85
BQ terrace	44713	38.09	6478	44540	38.03	2926	-0.39	-0.06	54.83
Average	20955	40.06	8105	20992	40.00	3216	0.37	-0.06	61.26

TABLE 8: Results of comparison between NNZ-ETDA and proposed method.

Sequence (QP = 22)	NNZ-ETDA			Proposed			Comparison		
	Bit rate (kb/s)	PSNR (dB)	Time (sec)	Bit rate (kb/s)	PSNR (dB)	Time (sec)	Δ Bit rate (%)	Δ PSNR (dB)	Δ Time (%)
Traffic	15525	41.68	3671	15644	41.66	3152	0.77	-0.02	14.15
People on street	34076	40.23	9024	34491	40.19	7201	1.22	-0.04	20.20
Kimono	6152	41.87	3848	6233	41.86	2860	1.32	-0.01	25.69
Park scene	8603	40.02	2245	8669	40.01	1936	0.77	-0.01	13.78
Cactus	22833	38.47	2725	23195	38.46	2306	1.59	-0.01	15.39
Basketball drive	13667	39.81	2608	14175	39.80	2134	3.72	-0.01	18.19
BQ terrace	44175	38.04	3571	44540	38.03	2926	0.83	-0.01	18.05
Average	20719	40.02	3956	20992	40.00	3216	1.46	-0.01	17.92

can achieve time improving ratio about 61.26%~81.48% as compared to HM 8.1 encoder with insignificant loss of image quality. When compared with the NNZ-ETDA, the proposed method can further achieve time improving ratio about 8.29%~17.92%. In addition, the proposed method also can be equally implemented with or be considered in design of a fast HEVC encoder.

To support 3D video format applications and efficient rate control at low bit rate, HEVC extensions for the efficient compression of stereo and multiview video are being developed by JCT-3V [20]. However, the higher encoding complexity of 3D-HEVC is a key problem in real-time applications [21, 22]. Thus, it gives an opportunity to develop fast 3D video encoding algorithms for its practical implementations.

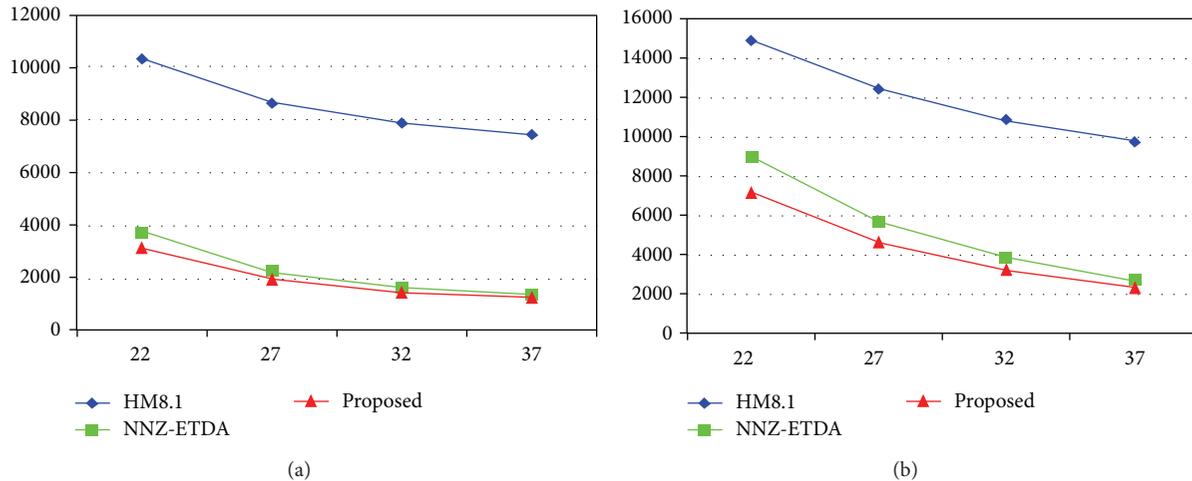


FIGURE 9: The TU processing time with different QP: (a) traffic and (b) people on street.

We will further exploit the characteristics of high temporal-spatial correlation existing in 3D video to develop an effective 3D-HEVC encoding method in our further works.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Science Council, Taiwan, under Contract no. NSC 100-2221-E-214-048 and I-Shou University, Taiwan, under Contract no. ISU 103-01-05A.

References

- [1] "Advanced video coding for generic audiovisual services," ITU-T Rec. H. 264 and ISO/IEC, 14496 10, ITU-T and ISO/IEC, 2010.
- [2] "Joint call for proposals on video compression technology," Kyoto, Japan, document VCEG-AM91 of ITU-T Q6/16 and N1113 of JTC1/SC29/WG11, January 2010.
- [3] J. 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.
- [4] B. Bross, W. J. Han, J. R. Ohm, G. J. Sullivan, and T. Weingand, "High efficiency video coding (HEVC) text specification draft 8," JCT-VC Document, JCTVC-J1003, July 2012.
- [5] *High Efficiency Video Coding*, Rec. ITU-T H. 265 and ISO/IEC, 23008-2, January 2013.
- [6] B. Bross, W. J. Han, J. R. Ohm, G. J. Sullivan, Y. K. Wang, and T. Wiegand, "High efficiency video coding (HEVC) text specification draft 10 (for FDIS & Consent)," Geneva, Switzerland, document JCTVC-L1003 of JCT-VC, July 2013.
- [7] X. L. Shen, L. Yu, and J. Chen, "Fast coding unit size selection for HEVC based on Bayesian decision rule," in *Proceedings of the Picture Coding Symposium (PCS '12)*, pp. 453–456, May 2012.
- [8] 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*, pp. 53–62, March 2013.
- [9] H. S. Lee, K. Y. Kim, T. R. Kim, and G. H. Park, "Fast encoding algorithm based on depth of coding-unit for high efficiency video coding," *Optical Engineering*, vol. 51, no. 6, 2013.
- [10] 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, January 2012.
- [11] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 465–470, 2013.
- [12] K. Lee, H. J. Lee, J. Kim, and Y. Choi, "A novel algorithm for zero block detection in high efficiency video coding," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 1124–1134, 2013.
- [13] G. Tian and S. Goto, "An optimization scheme for quadtree-structured prediction and residual encoding in HEVC," in *Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems*, pp. 547–550, December 2012.
- [14] Y. Shi, Z. Gao, and X. Zhang, "Early TU split termination in HEVC based on quasi-zero-block," in *Proceedings of the 3rd International Conference on Electric and Electronics*, pp. 450–454, January 2013.
- [15] K. Choi and E. S. Jang, "Early TU decision method for fast video encoding in high efficiency video coding," *Electronics Letters*, vol. 48, no. 12, pp. 971–973, 2012.
- [16] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.
- [17] E. G. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding For Next-Generation Multimedia*, John Wiley & Sons, 2003.
- [18] Reference software HM8.1, https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/branches/.
- [19] "Common test conditions and software reference configurations," JCTVC-E700, Joint Collaborative Team on Video Coding (JCT-VC) 5th Meeting, Geneva, Switzerland, March 2011.

- [20] G. Tech, K. Wegner, Y. Chen, and S. Yea, "3D-HEVC draft text 1," Joint Collaborative Team on 3D Video Coding Extensions (JCT-3V) Document JCT3V-E1001, 5th Meeting, Vienna, Austria, August 2013.
- [21] Q. Zhang, L. Tian, L. Huang, X. Wang, and H. Zhu, "Rendering distortion estimation model for 3D high efficiency depth coding," *Mathematical Problems in Engineering*, vol. 2014, Article ID 940737, 7 pages, 2014.
- [22] Y. Wu and S.-W. Ko, "A simple and high performing rate control initialization method for H.264 AVC coding based on motion vector map and spatial complexity at low bitrate," *Mathematical Problems in Engineering*, vol. 2014, Article ID 595383, 11 pages, 2014.



Hindawi

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

