www.ijaceeonline.com ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2, Issue: 7, July 2017, pp. 11–20

Fast Edge Detection and Early Depth Decision for Intra Coding of 3D-HEVC

Takafumi Katayama
Research Scholar, Department of Electrical and Electronic Engineering, Tokushima University, Minami-Josanjima 2-1, Tokushima City, 770-8506, Japan
Email: katayama@ee.tokushima-u.ac.jp

Dr. Tian Song
Associate Professor, Department of Electrical and Electronic Engineering, Graduate School of Advanced Technology and Science, Tokushima University, Minami-Josanjima 2-1, Tokushima City, Japan
Email: katayama@ee.tokushima-u.ac.jp

Wen Shi
Research Scholar, Department of Electrical and Electronic Engineering, Tokushima University, Minami-Josanjima 2-1, Tokushima City, 770-8506, Japan
Email: shiwen@ee.tokushima-u.ac.jp

Dr. Xiantao Jiang
Assistant Professor, Department of Information Engineering, Shanghai Maritime University, Shanghai, China
Email: xtjiang@shmtu.edu.cn

Dr. Takashi Shimamoto
Professor, Department of Electrical and Electronic Engineering, Graduate School of Advanced Technology and Science, Tokushima University, Minami-Josanjima 2-1, Tokushima City, Japan
Email: katayama@ee.tokushima-u.ac.jp

Abstract: *In this paper, an improved low complexity algorithm for 3D-HEVC is proposed to reduce the computational complexity. The basic idea of the method is to utilize the boundary homogeneity of texture and depth map which is generated by Laplacian filter. Firstly, based on the evaluation result of the boundary homogeneity, an early coding unit (CU) decision method is proposed to achieve the low computational complexity. Secondly, an efficient edge detection algorithm using Laplacian filter provide fast intra depth coding with the reduction of the number in intra prediction mode. The simulation results show that the proposed algorithm can achieve the average of 56.9% computation complexity reduction comparing to the original HTM algorithm.*

Keyword: *HEVC; 3D video coding (3D-HEVC); Intra depth coding; Intra prediction.*

1. INTRODUCTION

With the development of the technology of 3D television (3DTV) and free viewpoint television (FTV), 3D video coding (3D-HEVC) attracts more attention. The typical 3D video is represented using the multi-view video and depth format [1], in which few captured texture videos as well as associated depth maps are used. The depth maps provide per-pixel with the depth corresponding to the texture video that can be used to render arbitrary virtual views by using depth image based rendering [2] [3]. Recently, 3D-HEVC technology based on high efficiency video coding (HEVC) is now being standardized by joint collaborative team on 3D video coding (JCT-3V) as an extension to HEVC[4][5][6]. From the JCT-3V meetings, the developed coding schemes for 3D-HEVC mainly use HEVC together with exploiting temporal and interview correlation. Thus, many coding tools applied in 3D-HEVC are based on the hybrid coding scheme and highly related to HEVC.

Considering the intra depth coding, the traditional HEVC prediction modes result in distinct coding artifacts at sharp edges. To represent the depth information

Cite this paper:

T. Katayama, T. Song, W. Shi, X. Jiang, and T. Shimamoto, "Fast Edge Detection and Early Depth Decision for Intra Coding of 3D-HEVC", International Journal of Advances in Computer and Electronics Engineering, Vol. 2, No. 7, pp. 11-20, July 2017.

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

in a better way, two depth modeling modes (DMMs), named wedgelet partition mode (DMM1) and contour partition mode (DMM4), have been added to the intra coding of 3D-HEVC. DMMs can largely contribute to the depth coding, but much computational complexity is induced. Moreover, the decisions of the coding units (CUs) and modes have the most of the computational complexity for HEVC encoder. The 3D-HEVC also adopts the quad-tree structure coding, which supports the coding units (CUs) varying from 64x 64 to 8x8 (to 4x4, if considering PU partition). The traditional process of CU decision includes very high computational complexity. For the mode decision, it takes more computation time due to DMMs. The complexity of intra depth coding takes about 5 times more than that of texture coding and contributes to the coding efficiency for 3D-HEVC [7]. [8] also shows the detail of the complexity distribution according to the CU size, and it is clear that two DMMs occupies a large proportion among the complexity distribution of encoding. Therefore, the low complexity algorithms for intra depth coding are required.

Some previous works have proposed the complexity reduction for the intra depth coding in 3D-HEVC [9]-[17]. In [9] and [10], Gu et al. used the evaluation cost of intra traditional mode as the skip signal to avoid some DMMs calculation. Park et al. proposed an algorithm which performs a simple edge classification in the Hadamard transform to omit unnecessary DMMs [11]. In [12], Silva et al. proposed an algorithm to reduce the number of modes to be evaluated in the mode decision. All these methods focus on the mode decision of DMMs. Thereby, improvement of DMMs is discussed positively. Regarding fast CU decision algorithm, [13] and [14] proposed the selection of the adaptively CU and intra prediction mode from rate-distortion (RD) cost. However, further improvement is required for the low complexity algorithm because the complexity reduction with the high bit-rate tend to be low complexity reduction rate. Moreover, Sanchez et al. proposed an aggressive and lightweight complexity reduction technique using the simplified edge detector (SED) algorithm [15][16]. The one presented in [16] is capable of performing bipartition modes evaluation without data dependencies in HEVC intra-prediction mode, which is a desired characteristic in a hardware design. Additionally, [16] presents the development of the SED hardware design. This related work contributes to a hardware oriented algorithm. However, SED algorithm has to perform the filter process for all pixel information in a CTU. Therefore, considering hardware implementation of pre-processing, the hardware architecture area has been increased in the method of [16]. Therefore, the efficient edge detection which can determine the optimal CU depth is required for hardware implementation of pre-encoding. As a particularly new technique, H. Zheng et al. proposed [17] a low complexity depth intra coding method for 3D-HEVC based on the depth classification. For optimal CU decision, the classifier trained by support vector machine (SVM) is applied to determine its depth complexity class for only checking the corresponding intra prediction modes. [17] is also very effective for the reduction of recursive rate-distortion optimization (RDO) process. However, [17] consume the high computational time by performing the depth classification. Therefore, it is difficult to apply the depth classification using SVM to real-time encoding processing.

Our research target is on the complexity reduction for RDO process in 3D-HEVC. Additionally, to achieve the hardware oriented parallel algorithm without data dependencies in RDO process, high efficient pre-processing is proposed.

The remainder of this paper is organized as follows. Section 2 introduces the intra texture and depth coding of 3D-HEVC. Section 3 presents the proposed fast algorithm for the CU size decision and the efficient edge detection. Section 4 addresses the simulation results and Section 5 concludes our work.

2. OVERVIEW OF INTRA CODING IN 3D-HEVC

The 3D-HEVC standard inherits the well-known block based hybrid coding architecture of HEVC. It employs a flexible quad-tree coding block partitioning structure that enables the usage of large and multiple sizes of CU. One of the frames is divided into a sequence of coding tree units (CTUs) and the maximum size allowed for the luma block in a CTU is specified to be 64x64. Each 2Nx2N CUs which shares the same prediction mode can be divided into four smaller NxN CUs recursively until the minimum CU size is reached. The sizes of CU range from 64x64 to 8x8. The number of the prediction modes in intra depth coding for each CU is also increased to 37, including 35 conventional modes in HEVC, and two DMMs. In the DMMs, two different types of partitioning patterns named wedge-

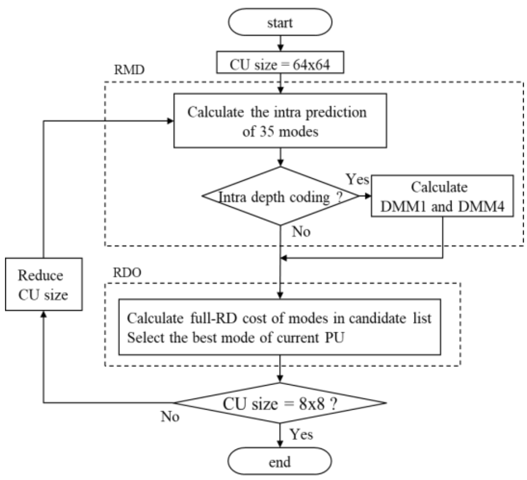


Figure 1 The processing flow of intra coding in HTM16.0

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

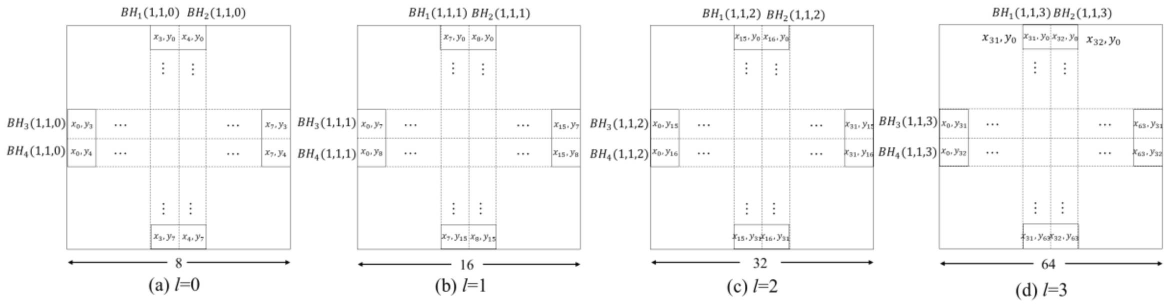


Figure 2 Overview for calculating the boundary homogeneity

lets and contours are applied to satisfy the characteristics of depth maps. They are different in the segmentation of the depth block. Each pattern of partition divides the area of the block into two nonrectangular regions, where each region is represented by a constant value. The flowchart of intra texture and depth coding in the reference software 3D-HEVC Test Model (HTM16.0) is introduced [4][5], as shown in Figure 1. For each iteration, it starts from trying the CUs size of 64x64. The CU size ranges from 64x64 to 8x8. For each CU, the mode decision could be divided into rough mode decision (RMD) and rate-distortion optimization (RDO) stages. In the RMD process, 35 HEVC intra prediction modes are searched with the sum of absolute transformed difference. Three or eight modes are selected as the candidate modes.

If intra depth coding is performed, DMMs with all partitions are searched with the sum of squared error. The best DMMs are put into candidate modes list as well. In the RDO process, the candidate modes are searched with the complex full RD cost function, where the evaluation cost is defined as J_{RDO}. J_{RDO} highly depends on the quantization parameter (QP). The quantizer of HEVC is similar to that of H.264/AVC where QP is defined in the range of [0, 51]. An optimized bit-rate can be generated by adaptively selecting the QP. The best mode is decided with the smallest J_{RDO} and this cost is regarded as the current CU cost. For the CU decision, the cost of each CU is always compared with the total cost of its four sub-CUs to decide whether the CU is to be split or not.

3. PROPOSED ALGORITHM

Firstly, the proposed algorithm for early CU depth decision in the intra texture and the depth coding which used boundary homogeneity are represented. Secondly, the proposed fast intra depth prediction mode decision is represented with the edge detection technique. Finally, the proposed algorithms are combined and explained by the overall process.

3.1 Early CU Size Decision by Boundary Homo Geneity for Intra Texture and Depth Coding

As shown in the previous section, to determine the optimal CU size, the recursive RDO calculation of 3D-HEVC encoder is required. The redundant calculation of RDO processing can be reduced by early determining the best CU size. Therefore, our approach predicts the optimal CU size using the boundary homogeneity (*BH*) with the input texture and depth information.

To measure the *BH* of the CTU area, our approach uses the boundary pixels in a CU from 8x8 to 64x64 blocks which are shown in Figure 2. In the case an 8x8 CU is selected, a parameter l is set to 0. The pixels which are used to calculate the *BH* are shown in Figure 1 (a). Similarly, the boundary pixels from 16x16 to 64x64 block are represented by (b), (c), and (d). The average value of *BH* in each CUs is calculated by

$$BH_1(k,m,l) = \sum_{j=0}^{2^{3+l}-1} P(x_{2^{2+l}-1+2^{3+l}(k-1)}, y_{j+2^{3+l}(m-1)}) >> 3+l \quad (1)$$

$$BH_2(k,m,l) = \sum_{j=0}^{2^{3+l}-1} P(x_{2^{2+l}+2^{3+l}(k-1)}, y_{j+2^{3+l}(m-1)}) >> 3+l \quad (2)$$

$$BH_3(k,m,l) = \sum_{j=0}^{2^{3+l}-1} P(x_{j+2^{3+l}(k-1)}, y_{2^{2+l}-1+2^{3+l}(m-1)}) >> 3+l \quad (3)$$

$$BH_4(k,m,l) = \sum_{j=0}^{2^{3+l}-1} P(x_{j+2^{3+l}(k-1)}, y_{2^{2+l}+2^{3+l}(m-1)}) >> 3+l \quad (4)$$

where $P(x, y)$ denotes the pixel position in a CTU. k and m represent the number of divided CUs. In this work, when calculating the *BH* for 8x8, 16x16, 32x32, and 64x64, the maximum number of k and m are equal to 8, 4, 2, and 1, respectively. The horizontal and vertical difference value of $BH_1(k, m, l)$, $BH_2(k, m, l)$,

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

$BH_3(k, m, l)$, and $BH_4(k, m, l)$ are defined as

$$BH_V(k,m,l) = \left|BH_1(k,m,l) - BH_2(k,m,l)\right| \quad (5)$$

$$BH_H(k,m,l) = \left|BH_3(k,m,l) - BH_4(k,m,l)\right| \quad (6)$$

where $BH_V(k, m, l)$ and $BH_H(k, m, l)$ indicate the homogeneous relation for each CU using the pixel values of vertical line and horizontal line in a CU boundary. These values realize the optimal CU depth selection with few pixel information and simple calculation. In contrast, the edge detection in [15] has to perform the simplified edge processing for all pixel information into CTU. Thereby, compare with [15], our edge detection approach can efficiently reduce the computational complexity. Moreover, the recursive RDO calculation of 3D-HEVC encoder is reduced by early CU depth decision (ECDD) before encoding.

TABLE I CONFIGRATION OF ENCODED TEST SEQUENCES

.

Resolution	Sequences	Frame Length
1024x768	Ballons	100
	Newspaper	
	Kendo	
1920x1088	PoznanHall2	
	PoznanStree	
	Shark	
	UndoDancer	
	GTFly	

To achieve the reduction of recursive RDO calculation, our proposed ECDD algorithm determine the optimal CU depth by *BH*. The boundary strength correlation is represented by $BH_V(k, m, l)$ and $BH_H(k, m, l)$ of the previous paragraph. When $BH_V(k, m, l)$ and $BH_H(k, m, l)$ are small, the *BH* is high. Because the reference pixel for IPM use the neighboring pixel, the reference pixel for IPM can be easily predicted when $BH_V(k, m, l)$ and $BH_H(k, m, l)$ are small. In other words, the optimal CU size can be encoded by large CU size. However, the threshold (TH) value using $BH_V(k, m, l)$ and $BH_H(k, m, l)$ depend on QP because the optimal CU depth selection in 3D-HEVC is determined by RDO. Therefore, an adaptive TH value is required to determine the optimal CU size. From many simulation and analysis, we found that the adaptive *TH* value based on the relationship of CU depth and QP is calculated by

$$TH = 5 + 2 \times depth + (QP - 25) \quad (7)$$

Moreover, we confirm the prediction performance by different *TH*s using different QPs with the test sequences of Table I. Table II show the CU depth prediction accuracy (*Acc*) for every QP when the optimal CU depth is determined by using different *TH* values. *Acc* was computed using the following equation.

$$Acc(\%) = \sum_{f=1}^{F} \sum_{j=1}^{J} \frac{\dfrac{\sum_{i=1} SCU(i,j,f) \times CorrectCU(i,j,f)}{\sum_{i=1} SCU(i,j,f)}}{J \times F} \times 100 \quad (8)$$

where $CU(i, j, f)$ indicates the *i*th CU of the *j*th CTU of the *f*th frame of the test sequence. $SCU(i, j, f)$ represents the area (number of pixels in width × number of pixels in height) of the $SCU(i, j, f)$. $CorrectCU(i, j, f)$ is equal to one if the CU depth of $CU(i, j, f)$ is predicted correctly ($CorrectCU(i, j, f)$). *J* and *F* are the total number of CTUs in each frame and the total number of frames of the test video, respectively.

TABLE II *ACC* ACCORDING TO QP (%).

```
1:  For l = 0 to 3 do
2:    X = 64/2^(3+l)
3:    For m = 1 to X do
4:      For k = 1 to X do
5:        -Calculate BH1(k, m, l), BH2(k, m, l), BH3(k, m, l),
           and BH4(k, m, l).
6:        -Perform the efficient edge detection algorithm
           (Table V).
7:         BHV(k, m, l) = |BH1(k, m, l) - BH2(k, m, l)|
8:         BHH(k, m, l) = |BH3(k, m, l) - BH4(k, m, l)|
9:         BH = max{BHV(k, m, l), BHH(k, m, l)}
10:        If BH ≥ TH then
11:          OptPU_size = 2^(2+l)x2^(2+l)
12:        End if
13:     End for
14:   End for
15: End For
```

As shown in Table II, *Acc* will be very different when QP changes. Generally, the value of the optimal *TH* increased when QP is increasing. It's also clear that the proposed algorithm can archive high prediction accuracy.

TABLE III EARLY CU DEPTH DECISION ALGORITHM

	QP=22	QP=27	QP=32	QP=37
TH=10	83	83	55	45
TH=20	63	86	81	68
TH=30	55	68	82	79
TH=50	45	67	73	88

In summary, the proposed ECDD algorithm with analyzing the block from 8x8 pixels to 64x64 pixels is represented in Table III. Firstly, the number of the process in ECDD is calculated from *X* using l (lines 1-2). The boundary homogeneity of the current CU size are calculated from $BH_1(k, m, l)$, $BH_2(k, m, l)$, $BH_3(k, m, l)$, and $BH_4(k, m, l)$ (line 5). Similarly, a

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

proposed efficient edge detection algorithm is performed (lines 6). The detail of the proposed method is introduced in next section. $BH_V(k, m, l)$ and $BH_H(k, m, l)$ are calculated by in $BH_1(k, m, l)$, $BH_2(k, m, l)$, $BH_3(k, m, l)$, and $BH_4(k, m, l)$, and BH is designed as max value of $BH_V(k, m, l)$ and $BH_H(k, m, l)$ (lines 7-9).

In the next stage, *BH* is compared with TH (line 10). If BH is larger than *TH*, the CU size of 2^{2+l}x2^{2+l} is determined as the optimal CU size (line 11). For example, when l is equal to 0, all of the boundary correlation of 8x8 pixels in a CTU is evaluated (lines 3-14). Similarly, the block from 16x16 pixels to 64x64 pixels are performed with the same process.

3.2 Efficient Edge Detection by Laplacian Filter and Edge Classification For Intra Depth Prediction Mode

The edge detection in [11] and [15] need to perform Hadamard transform and the simplified edge for all pixel information in a CTU. On other words, these edge detection processes induce the recursive computational complexity. To achieve more computation complexity reduction than other edge detection algorithm, we propose an efficient edge detection algorithm for fast intra depth prediction mode. In our approach, the adaptive Laplacian filter (LF) processing is performed while calculating *BH*s of ECDD.

In the previous section, the strength of the boundary homogeneity of the horizontal line and the vertical line are used as $BH(k, m, l)$. Additionally, by calculating $DiffBH(k, m, l, i)$, the boundary position of depth image can be obtained. The detail of the boundary position is calculated by following equation.

$$DiffBH_1(k,m,l,i) = |P(x_{2^{2+l}-1+2^{3+l}(k-1)}, y_{i+2^{3+l}(m-1)}) - P(x_{2^{2+l}-1+2^{3+l}(k-1)}, y_{i+1+2^{3+l}(m-1)})| \quad (9)$$

$$DiffBH_2(k,m,l,i) = |P(x_{2^{2+l}+2^{3+l}(k-1)}, y_{i+2^{3+l}(m-1)}) - P(x_{2^{2+l}+2^{3+l}(k-1)}, y_{i+1+2^{3+l}(m-1)})| \quad (10)$$

$$DiffBH_3(k,m,l,i) = |P(x_{i+2^{3+l}(k-1)}, y_{2^{2+l}-1+2^{3+l}(m-1)}) - P(x_{i+1+2^{3+l}(k-1)}, y_{2^{2+l}-1+2^{3+l}(m-1)})| \quad (11)$$

$$DiffBH_4(k,m,l,i) = |P(x_{i+2^{3+l}(k-1)}, y_{2^{2+l}+2^{3+l}(m-1)}) - P(x_{i+1+2^{3+l}(k-1)}, y_{2^{2+l}+2^{3+l}(m-1)})| \quad (12)$$

$DiffBH_1$, $DiffBH_2$, $DiffBH_3$, and $DiffBH_4$ represent the difference value of the neighboring pixel. i represent the position of *DiffBH*. In our approach, while the $BH(k, m, l)$ are obtained, the calculation of *DiffBH* are performed. Therefore, the proposed algorithm can detect the edge information efficiently. The detail of the edge detection method in the case of {k, m, l} = {0} is introduced by Figure 3 and Figure 4.

Figure 3 represents the position of $DiffBH(k, m, l, i)$. $DiffBH(k, m, l, i)$ is used to judge whether to apply LF. Therefore, our proposed edge detection algorithm achieves an adaptive LF selection. From some verification result, it is clear that the edge direction can be detected by LF when the difference value of the pixel is larger than 5. Accordingly, If the value of $DiffBH(k, m, l, i)$ is larger than 5, which is defined as edge point (EP), the coefficients are calculated around the EP by using the LF. For example, Figure 4 shows the position where LF is applied when EP is $DiffBH_3(1,1,0,2)$. By using the calculated coefficients, the candidate mode list in $DiffBH_3(1,1,0,2)$ is selected from Table IV. Additionally, the candidate mode list in $DiffBH_3(1,1,0,2)$ is stored to $BH\_Edge(1,1,0,2)$. Similarly, $BH\_Edge(1, 1, 0, i)$ from $i = 0$ to $i = 6$ is calculated. Figure 5 shows the detail position of $BH_1\_Edge(k, m, l, i)$, $BH_2\_Edge(k, m, l, i)$, $BH_3\_Edge(k, m, l, i)$, and $BH_4\_Edge(k, m, l, i)$. As shown in Figure 5, *BH_Edge* represent the candidate modes of boundary region. The candidate mode list of 4x4 block ($\mathbf{X}_{4x4}$) is represented by following.

$$\mathbf{X}^0_{4x4} = \begin{cases} BH_1\_Edge(k,m,l,i) & (0 \le i \le 3) \\ BH_3\_Edge(k,m,l,i) & (0 \le i \le 3) \end{cases} \quad (13)$$

$$\mathbf{X}^1_{4x4} = \begin{cases} BH_2\_Edge(k,m,l,i) & (0 \le i \le 3) \\ BH_3\_Edge(k,m,l,i) & (3 \le i \le 6) \end{cases} \quad (14)$$

$$\mathbf{X}^2_{4x4} = \begin{cases} BH_1\_Edge(k,m,l,i) & (3 \le i \le 6) \\ BH_4\_Edge(k,m,l,i) & (0 \le i \le 3) \end{cases} \quad (15)$$

$$\mathbf{X}^3_{4x4} = \begin{cases} BH_2\_Edge(k,m,l,i) & (3 \le i \le 6) \\ BH_4\_Edge(k,m,l,i) & (3 \le i \le 6) \end{cases} \quad (16)$$

In the case of 4x4 block, the edge classification values of eight which is selected from Table IV are stored to $\mathbf{X}_{4x4}$. To decide the edge classification value used for the intra depth prediction mode, the most selected edge classification value in $\mathbf{X}^0_{4x4}$, $\mathbf{X}^1_{4x4}$, $\mathbf{X}^2_{4x4}$, and $\mathbf{X}^3_{4x4}$ are substituted for X^0_{4x4}, X^1_{4x4}, X^2_{4x4}, and X^3_{4x4}. Moreover, for calculating X_{8x8}, X_{16x16}, X_{32x32}, and X_{32x32}, the combinational method is introduced.

In the proposed algorithm, if more than two of the edge classification value in X^0_{4x4}, X^1_{4x4}, X^2_{4x4}, and X^3_{4x4} are same, the edge classification value is used as the prediction mode of X^0_{8x8}. Figure 6 represents the combinational case of X^0_{2Nx2N}. Similarly, the edge classification value of 16x16, 32x32, and 64x64 block are calculated. The detail of the procedure is described in Table V and Table VI.

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

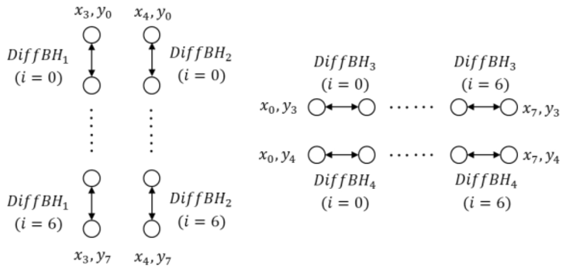


Figure 3 The representation of DiffBH(k, m, l, i) from i=0 to i=6 in the case of k=0, m=0, and l=0.

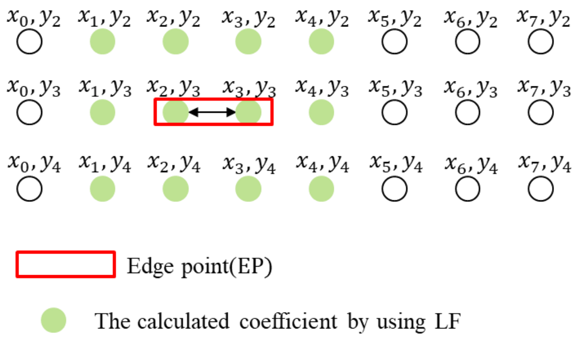


Figure 4 The positon where LF is applied when EP is $DiffBH_3(1,1,0,2)$.

TABLE IV CANDIDATE LIST OF THE MODE NUMBER

Edge classification	Mode number
DC, Planar	0, 1
Vertical	22, 23, 24, 25, 26, 27, 28, 29, 30
Horizontal	6, 7, 8, 9, 10, 11, 12, 13, 14
Left diagonal	2, 3, 4, 5, 30, 31, 32, 33, 34
Right diagonal	13, 14, 15, 16, 17, 18, 19, 20, 21

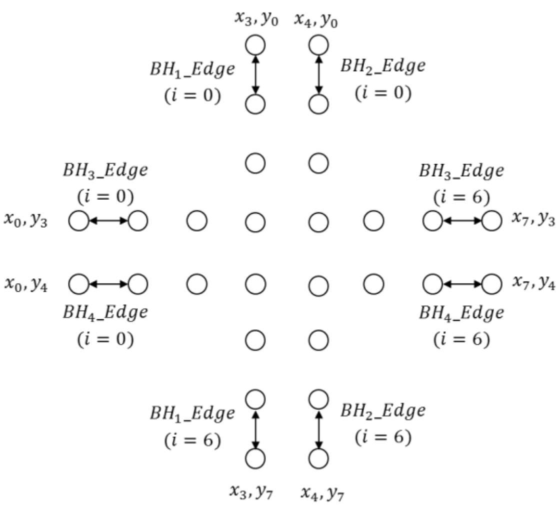


Figure 5 The representation of BH_Edge(k, m, l, i) in the case of k=0, m=0, and l=0

Table V show the efficient edge detection and the edge classification algorithm. The efficient edge detection is achieved by *Edge_flag*(*i*) (line 3-7). If *Edge_flag*(*i*) is equal to 1, Laplacian coefficients is calculated and the candidate list of the mode number is selected (line 8-10). On the other case, the candidate list of DC and Planar mode is selected (line 11-14). After calculating all of *BH_Edge*(*k*, *m*, *l*, *i*), the edge classification (X_{4x4}) is obtained (line 16-17).

As shown in Table VI, The combinational algorithm is performed. X^0_{2Nx2N} is decided by X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, and X^3_{NxN} (line 1). The combinational case is explained by the previous paragraph (line 2-4). On the other case, X^0_{2Nx2N} is selected in order of vertical, horizontal, left diagonal, and right diagonal (line 5-15). This is because it is clear that this order is the most probably selected from previous work [14].

3.3 OVERALL PROCESSING

The proposed algorithm is composed of two parts intra texture and depth coding. Firstly, the boundary

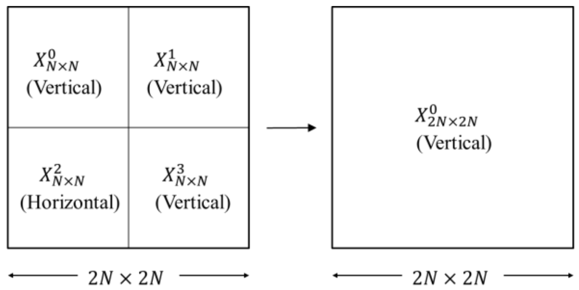


Figure 6 Graphical explanation of combinational case.

TABLE V EFFICIENT EDGE DETECTION AND EDGE CLASSIFICATION ALGORITHM.

```
1:  If l = 0 then
2:   For i = 0 to 2^(3+l)-2 do
3:     If DiffBH(k, m, l, i) > 5 then
4:        Edge_flag(i) = 1
5:     Else
6:        Edge_flag(i) = 0
7:     End If
8:     If Edge_flag(i) == 1 then
9:     -Laplacian coefficients are calculated
        as the center of DiffBH(k, m, l, i).
10:    -Using Laplacian coefficient,
        the candidate list of the mode number is selected,
        and stored to BH_Edge(k, m, l, i)
11:    Else
12:    BH_Edge(k, m, l, i) = 0
13:    -DC and Planar mode are stored.
14:    End If
15:  End For
16:  -Based on BH_Edge(k, m, l, i), the edge classification
      values of eight which is selected from Table IV are stored
      to X_4x4
17:  -X_4x4 is calculated from X_4x4
18: End If
```

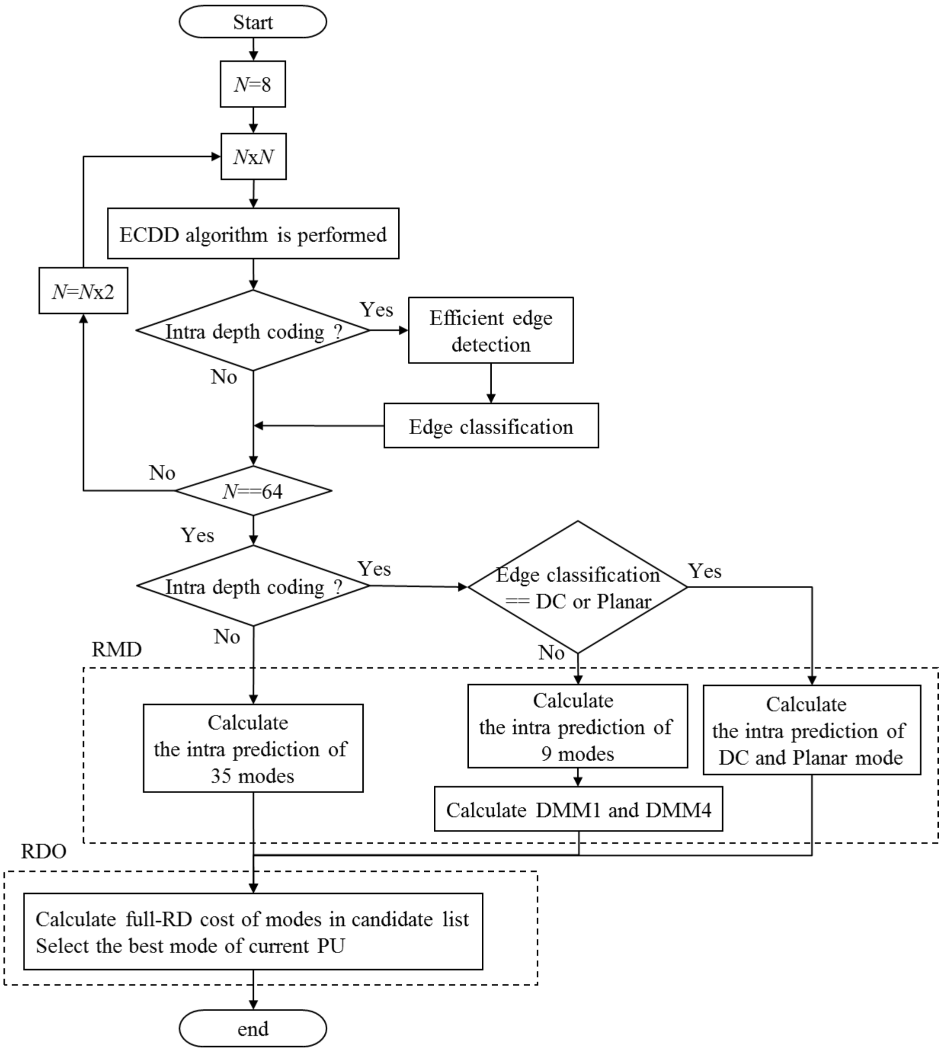


Figure 6. The processing flow of intra coding in proposed algorithm

TABLE VI COMBINATIONAL ALGORITHM

$X^0_{2Nx2N} = \{X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, X^3_{NxN}\}$ $(N = 4,8,16,32)$
If Combinational case **then**
-More than two of the edge classification value in X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, and X^3_{NxN} are same
$X^0_{2Nx2N} = X_{NxN}$
Else
If X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, and X^3_{NxN} include vertical **then**
X^0_{2Nx2N} = Vertical
Else If X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, and X^3_{NxN} include horizontal **then**
X^0_{2Nx2N} = Horizontal
Else If X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, and X^3_{NxN} include left diagonal **then**
X^0_{2Nx2N} = Left diagonal
Else If X^0_{NxN}, X^1_{NxN}, X^2_{NxN}, and X^3_{NxN} include right diagonal **then**
X^0_{2Nx2N} = Right diagonal
End If
End if

homogeneity with intra texture and depth coding respectively is calculated for optimal CU size decision.

Compared with HTM16.0, the proposed early CU size decision provide the reduction of RDO iteration to decide the optimal CU size.

The intra texture coding is performed with optimal CU size which is decided by early CU size decision algorithm, and RMD calculate the intra prediction mode of 35 mode. On the other hand, if the intra depth coding is performed, the efficient edge detection and the edge classification are calculated. Because the complexity of intra depth coding takes about 5 times more than that of texture coding, the fast decision of the candidate mode number greatly contributes to the complexity reduction of intra depth coding. In particular, compared with previous works, the intra depth prediction mode is decided by a little edge information. The edge classification reduces the RMD process from 35 mode to 9 mode. On the other hand, when the edge classification is DC or Planar, the current CU show the simple depth map.

In contrast to the intra directional mode, DC and Planar mode also is an important factor for complexity reduction. In current HTM16.0, a fast skip algorithm for DMMs has been adopted. When the best mode in

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

RMD is Planar mode and the pixel variance of current CU is smaller than a threshold, this indicates the current CU is homogeneous and DMMs should be skipped from the mode decision. Therefore, the Planar and DC mode could be a very important factor in skipping DMMs. In previous work, when DMMs skipped conditions are met, the CU has more than 99% probability to choose Planar and DC mode as the best mode. Therefore, when Planar and DC mode are selected, DMMs are not performed.

Finally, for deciding the best mode of current CU, full-RD cost is calculated from candidate list which is decided by RMD.

4. SIMULATION RESULT

The proposed algorithm is implemented in reference software HTM16.0 [5] [6]. it follows the common test condition (CTC) [18]. The simulation environment is Intel(R) Core(TM) i7-4770 CPU@3.40GHz with 4 cores, RAM 8.00 GB and Windows 10 Home Edition 64-bit. All test sequences are coded using the all intra configuration for three-view cases. The texture and depth map use the QPs setting to (QP texture, QP depth)={(25,34), (30,39), (35,42) , (40,45)}. Other encoding parameters remain the same as the CTC. Coding efficiency is measured by BDBR and BDPSNR[19]. Time saving (*TS*) represents the reduced total encoding time, including the texture video coding and the depth video coding. *TS* is defined as

$$TS(\%) = \frac{T_{HTM} - T_{proposed}}{T_{HTM}} \tag{17}$$

where T_{HTM} is the encoding time of the original reference software and $T_{proposed}$ is that of the proposed algorithm. The computation complexity reduction (*CR*) is evaluated with different QP. *CR* is defined as

$$CR(\%) = \sum_{f=1}^{F} \sum_{j=1}^{J} \frac{\sum_{i=1} BestCU(i,j,f)}{J \times F \times 85} \times 100 \tag{18}$$

where *BestCU*(*i*, *j*, *f*) indicates the *i*th best *CU* of the *j*th CTU of the *f*th frame of the test sequence. *J* and *F* are the total number of CTUs in each frame and the total number of frames of the test video, respectively. 85 represents the number of iteration which is required for the best CU size decision from 64x64 to 8x8 in reference software HTM16.0. Our proposed algorithm reduce the computational complexity in intra texture coding and intra depth coding, respectively. Table VII shows the simulation result of the proposed algorithm compared to HTM16.0. Fast intra texture coding reduce the computational complexity up to 84.1% with a

TABLE VII SIMULATION RESULT OF PROPOSED ALGORITHM COMPARED TO HTM16.0

Resolution	Sequences	Proposed Fast Intra Texture Coding			Proposed Fast Intra Depth Coding			Combined Fast Intra Texture and Depth Coding		
		CR (%)	BDBR (%)	BDPSNR (dB)	*CR* (%)	BDBR (%)	BDPSNR (dB)	*CR* (%)	BDBR (%)	BDPSNR (dB)
1024x768	Ballons	82.6	0.7	-0.04	85.4	-0.1	0.05	84.0	0.3	0.01
	Newspaper	82.3	1.9	-0.09	83.3	-0.1	0.01	82.8	0.9	-0.04
	Kendo	85.1	2.1	-0.09	85.9	-0.3	-0.09	85.5	0.9	-0.09
1920x1088	PoznanHall2	86.1	1.4	-0.06	85.8	0.8	-0.01	85.9	1.1	-0.04
	PoznanStreet	83.6	1.2	-0.04	86.3	0.9	-0.01	85.0	1.1	-0.03
	Shark	84.2	1.8	-0.1	85.2	0.2	-0.02	84.7	1.0	-0.06
	UndoDancer	83.9	2.5	-0.09	86.8	0.8	-0.03	85.4	1.7	-0.06
	GTFly	84.7	1.9	-0.07	87.6	1.9	-0.09	86.1	1.9	-0.08
	Average	84.1	1.7	-0.07	85.8	0.5	-0.02	84.9	1.1	-0.05

TABLE VIII COMPARISON WITH PREVIOUS WORKS BY IN TS AND BDBR

Resolution	Sequences	[14]		[15]		[16]		[17]		Proposed	
		BD-rate (%)	*TS* (%)	BD-rate (%)	*TS* (%)	BD-rate (%)	*TS* (%)	BD-rate (%)	*TS* (%)	BD-rate (%)	*TS* (%)
1024x768	Ballons	1.2	27.6	0.2	26.2	1.1	32.6	0.0	36.7	-0.1	56.3
	Newspaper	1.1	26.6	-2.0	28.9	2.4	45.9	0.1	35.0	-0.1	53.2
	Kendo	0.7	26.3	0.1	25.2	1.1	47.9	0.1	33.7	-0.3	59.7
1920x1088	PoznanHall2	1.3	43.7	0.4	32.7	1.8	22.9	0.2	38.4	0.8	47.4
	PoznanStreet	1.40	49.1	0.2	33.7	0.4	39.0	0.1	34.5	0.9	56.6
	Shark	0.1	33.7	NA	NA	0.4	34.9	0.1	33.6	0.2	55.6
	UndoDancer	0.6	49.1	0.3	31.0	0.4	40.6	0.1	32.2	0.8	57.4
	GTFly	0.3	45.1	0.1	30.1	0.4	39.2	0.0	34.3	1.3	69.4
	Average	0.8	37.6	-0.06	29.7	0.9	37.9	0.1	34.8	0.5	56.9

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

bit-rate increase of 1.7%. Fast intra depth coding reduce the computational complexity up to 85.8% with a bit-rate increase of 0.5%. Moreover, our proposed algorithm which combined texture and depth coding is examined. The combination algorithm reduces the total *CR* by 84.9% at the cost of 1.1% bit-rate increase, on average. BD-PSNR decrease also becomes tiny at -0.05dB. Our proposed algorithm shows very high performance.

The results given in Table VIII indicate BD-rate comparison by our proposed algorithm and previous works[14]-[17][20]. Table VIII shows that the proposed algorithm reduces *TS* by about 20% better than in [14]-[17] whereas it achieves better BD-rate as well. In particularly, [16] reduce the total TS by 37.9% at the cost of 0.90% BD-rate increase, on average. [16] can perform CU size decision without data dependencies in HEVC intra-prediction mode, which is a desired characteristic in a hardware design. Because the complexity reduction of the intra depth prediction mode is achieved by our proposed algorithm, our proposed algorithm reduces *TS* 19.0% compared with the proposed algorithm in [16]. [17] proposed fast CU size and prediction mode decision by using SVM. Compared with our proposed algorithm, the new approach which is proposed by [17] achieves better BD-rate. On the other hand, *TS* reduction of [17] is lower than our proposed algorithm. Because we consider that DMMs could not be reduced efficiently in [17], our proposed intra depth prediction mode decision algorithm have high efficacy for DMMs.

TABLE IX COMPARISON WITH PREVIOUS WORK BY TS OF PROPOSED INTRA DEPTH CODING UNDER DIFFERENT QPS

Sequences	Proposed			[14]		
	QP34	QP39	QP34	QP34	QP39	QP43
	TS(%)			*TS*(%)		
Ballons	68.5	51.2	52.9	20.4	20.5	28.2
Newspaper	39.6	50.3	50.4	17.2	15.9	26.1
Kendo	81.3	50.3	53.9	NA	NA	NA
PoznanHall2	66.3	44.1	30.4	NA	NA	NA
PoznanStree	52.3	46.6	72.7	NA	NA	NA
Shark	57.3	60.2	55.4	NA	NA	NA
UndoDancer	48.7	40.6	68.6	43.9	42.5	44.6
GTFly	49.2	72.9	81.8	35.1	33.3	47
Average	57.9	52.1	58.3	29.2	28.1	36.5

Moreover, in Table IX, the performance of our proposed algorithm is compared with the proposed algorithm in [14] every QP value. The test condition of proposed algorithm is the same as the compared algorithm. The results given in Table IX indicate that the proposed algorithm reduces *TS* up to 28.7 % better than in [14] and our proposed algorithm achieves the complexity reduction for every QP value. According to the above results, we confirm that the computational complexity reduction is achieved with almost no video quality loss.

5. CONCLUSION

The focus of this paper is on developing a complexity reduction scheme for 3D-HEVC encoder. The proposed algorithms use fast intra texture and depth coding. Our scheme utilizes the boundary homogeneity to predict the CU sizes of the CTUs of texture and depth coding. To realize the low complexity of CU size decision, our approach notices the boundary homogeneity of every CU size. Moreover, for complexity reduction in the intra depth prediction mode, we proposed the edge classification by using Laplacian filter. The performance of the proposed algorithm was tested on a representative set of video sequences and was compared against the unmodified HTM encoder as well as two of the art complexity reduction schemes and combinations. Performance evaluations show that our proposed algorithms reduce encoding time on average 56.9% and increases BD-rate about 0.5%, compared with HTM 16.0.

6. ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI Grant Numbers 24500117, 26330066, 17K00157.

REFERENCES

[1] K. Müller, H. Schwarz, D. Marpe et al., (2013), "3D high-efficiency video coding for multi-view video and depth data", *IEEE Transactions on Image Processing*, Vol. 22, Issue. 9, pp. 3366-3378.

[2] P. Kauff, N. Atzpadin, C. Fehn et al., (2007), "Depth map creation and image-based rendering for advanced 3DTV services providing interoperability and scalability", *Signal Processing: Image Communication*, vol. 22, Issue. 2, pp. 217-234.

[3] Q. Zhang, P. An, Y. Zhang, L. Shen, and Z. Zhang, (2013), "Efficient depth map compression for view rendering in 3D video", *Imaging Science Journal*, Vol. 61, Issue. 4, pp. 385-395.

[4] G. Tech, Y. Chen, K. Müller et al., (2016), "Overview of the Multiview and 3D Extensions of High Efficiency Video Coding", *IEEE Transactions on circuits and systems for video technology*, Vol. 26, Issue. 1, pp.35-49.

[5] Y. Chen, G. Tech, K. Wegner, and S. Yea, (2015), "Test Model 11 of 3DHEVC and MV-HEVC", *JCT3V-K1003 ISO/IEC JTC1/SC29/WG11, MPEG/m36142*.

[6] K. Suehring and K. Sharman, HEVC reference software, date:[2013],online available at:

https://hevc.hhi.fraunhofer.de/trac/3dhevc/browser/3DVCSoftware/tags/HTM-16.0.

[7] C. Fu, W. Su, and S. Tsang, (2015), "Fast wedelet pattern decision for DMM in 3D-HEVC", *Proceedings* of *IEEE International Conference on Digital Signal Processing(DSP)*, pp.477-481.

ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume: 2 Issue: 7, July 2017, pp. 11–20

[8] G. Sanchez, R. Cataldo, R. Fernandes et al., (2016), "3D-HEVC depth maps intra prediction complexity analysis", *Proceedings of IEEE International Conference on Electronics Circuits and Systems (ICECS)*, pp. 348-351.

[9] Z. Gu, J. Zheng, N. Ling, and P. Zhang, (2014), "Simplified depth intra mode selection for 3D video compression", *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, pp.1110-1113.

[10] Z. Gu, J. Zheng, N. Ling, and P. Zhang, (2014), "Fast bipartition mode selection for 3D HEVC depth intra coding", *IEEE International Conference on Multimedia and Expo (ICME)*, pp.1-6.

[11] C. Park, (2015), "Edge-based intramode selection for depth-map coding in 3D-HEVC", *IEEE Transactions on Image Processing*, Vol.24, Issue. 1, pp.155-162.

[12] T.L.da Silva, L.V.Agostini, L.A.da Silva Cruz, (2014), "Complexity reduction of depth intra coding for 3D video extension of HEVC", *Proceedings of IEEE Visual Communications and Image Processing Conference (VCIP)*, pp.229-232.

[13] R. Guo, G. He, Y. Li, and K. Wang, (2016), "Fast algorithm for prediction unit and mode decisions of intra depth coding in 3D-HEVC", *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp.1121-1125.

[14] K. Peng, J. Chiang and W. Lie, (2016), "Low complexity depth intra coding combining fast intra mode and fast CU size decision in 3D-HEVC", *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp.2381-8549.

[15] G. Sanchez, M. Saldanha, G. Balota, B. Zatt, M. Porto, and L. Agostini, (2014), "Complexity reduction for 3D-HEVC depth maps intra-frame prediction using simplified edge detector algorithm", *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 3209-3213.

[16] G. Sanchez, M. Saldanha, M. Porto et al., (2016), "Real-time simplified edge detector architecture for 3D-HEVC depth maps coding", *Proceedings of IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pp. 352-355.

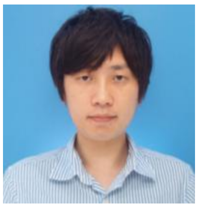
[17] H. Zheng, J. Zhu, H. Zeng et al., (2016), "Low complexity depth intra coding in 3D-HEVC based on depth classification", *Visual Communications and Image Processing (VCIP)*, pp. 1-4.

[18] K. Müller and A.Vetro, (2014), "Common test conditions of 3DV core experiments", *JCT3V-G1100, ITU-T SG 16 WP 3 and ISO/IEC JTC1/SC 29/WG 11*.

[19] G.Bjontegaard, (2001), "Calculation of Average PSNR Differences Between RD Curves", *VCEG-M33 ITU-T Q6/16*.

[20] F. Bossen (2013) "Common test conditions and software reference configurations", *JCTVCL1100, ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*.

Authors Biography



Takafumi Katayama, received his B.E. and M.E. degrees in Electrical Engineering from Tokushima University. He belonged to the Renesas Electronics Corporation from 2012 to 2014. Now he is currently proceeding to the Ph.D. degree in Electrical Engineering of Tokushima University from 2015. His current research interests include video coding algorithms and hardware design.

Tian Song, received his B.E. degree from Dalian University of Technology, China, in 1995, and his M.E. and Dr.E. degrees from Osaka University in 2001 and 2004, respectively. He joined Tokushima University in 2004 as an Assistant Professor. Presently, he is an Associate Professor of the Department of Electrical and Electronic Engineering, Graduate School of Advanced Technology and Science, Tokushima University. He is a member of IEICE, IEEE and RISP. His current research interests include video coding algorithms, VLSI architectures, and system design methodology.

Wen Shi, received his B.E. degree in electrical engineering and automation from Harbin Institute of Technology, China, in 2012, and his M.E. degree from Tokushima University, Japan, in 2015. Currently, He is pursuing the Ph.D. degree in Electrical and Electronic Engineering at the Graduate School of Advanced Technology and Science, Tokushima University, Japan. His research interests include video coding algorithms and VLSI design.

Xiantao Jiang, received the M.E. and Dr. E. degrees from Shanghai Maritime University and Tokushima University in 2012 and 2016. He joined Shanghai Maritime University in 2016 as an Assistant Professor. He is a member of IEEE. His interested research areas are the video coding, computer vision, machine learning and parallel processing algorithm.

Takashi Shimamoto, received his B.E. and M.E. degrees in Electrical Engineering from Tokushima University, and his Dr. E. degree from Osaka University in 1982, 1984, and 1992, respectively. He joined Tokushima University in 1984 as an Assistant Professor. Presently, he is a Professor of the Department of Electrical and Electronic Engineering, Graduate School of Advanced Technology and Science, Tokushima University. He is a member of IEEE, IEICE and RISP. His research interests include heuristic algorithms for VLSI CAD.

Cite this paper:

T. Katayama, T. Song, W. Shi, X. Jiang, and T. Shimamoto, "Fast Edge Detection and Early Depth Decision for Intra Coding of 3D-HEVC", International Journal of Advances in Computer and Electronics Engineering, Vol. 2, No. 7, pp. 11-20, July 2017.