Bayesian-theory-based Fast CU Size and Mode Decision Algorithm for 3D-HEVC Depth Video Inter-coding

Fen Chen, Sheng Liu, Zongju Peng, Qingqing Hu, Gangyi Jiang and Mei Yu
Faculty of Information Science and Engineering, Ningbo University, Ningbo 315211, China
[e-mail: pengzongju@nbu.edu.cn]
*Corresponding author: Zongju Peng

Abstract

Multi-view video plus depth (MVD) is a mainstream format of 3D scene representation in free viewpoint video systems. The advanced 3D extension of the high efficiency video coding (3D-HEVC) standard introduces new prediction tools to improve the coding performance of depth video. However, the depth video in 3D-HEVC is time consuming. To reduce the complexity of the depth video inter coding, we propose a fast coding unit (CU) size and mode decision algorithm. First, an off-line trained Bayesian model is built which the feature vector contains the depth levels of the corresponding spatial, temporal, and inter-component (texture-depth) neighboring largest CUs (LCUs). Then, the model is used to predict the depth level of the current LCU, and terminate the CU recursive splitting process. Finally, the CU mode search process is early terminated by making use of the mode correlation of spatial, inter-component (texture-depth), and inter-view neighboring CUs. Compared to the 3D-HEVC reference software HTM-10.0, the proposed algorithm reduces the encoding time of depth video and the total encoding time by 65.03% and 41.04% on average, respectively, with negligible quality degradation of the synthesized virtual view.

Keywords: Depth video coding; 3D-HEVC; Inter-coding; CU size decision; Mode decision

1. Introduction

In free viewpoint video (FVV) systems, users can interactively select the different angle of view, and immerse in the three dimensional (3D) scene[1]. With these characters, FVV systems have a broad prospect of many potential applications, such as, television, cinema, games, education, and mobiles phones. So far, multi-view video plus depth (MVD) is a mainstream format of 3D scene representation in FVV systems [2]. MVD signal includes color videos of multiple viewpoints and the associated depth videos which represent the geometric information of the scene. Using depth image based rendering technique (DIBR), the intermediate views can be generated for continuously view switch [3]. Efficient storage and transmission of depth video are important for realizing an FVV system.

Depth video is the auxiliary information in DIBR, and the characteristics of depth maps are different from those of the corresponding color video. Therefore, the joint collaborative team on 3D video coding extension development (JCT-3V) drafted a 3D video coding standard based on the high efficiency video coding (HEVC) standard which is called 3D-HEVC [4] [5]. Besides the spatial and temporal correlations exploited by HEVC, the correlation among different views is further exploited by 3D-HEVC. JCT-3V launches a test model of 3D-HEVC, namely HTM [6], as a common platform for research on MVD encoding. In HTM, hierarchical quadtree-based largest coding unit (LCU) structure, flexibly adapts to various motion and texture characteristics of video signal, is used. The LCU is the basic unit of 3D-HEVC, the size of which is predefined [7][8]. Each CTU is recursively divided into four coding units (CUs). The largest CU size is 64×64, and the smallest CU size is 8×8. The depth of a quadtree coding structure ranges from 0 (CU size of 64×64) to 3 (CU size of 8×8). For each CU, Skip, Merge, Inter-2N×2N, Inter-2N×N, Inter-N×2N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, Inter-nR×2N, Intra-2N×2N, and Intra-N×N is probed among all temporal and inter-view frames to get the best one with minimal rate distortion (RD) cost. Although this scheme can achieve optimal RD performance, its computational complexity is extremely high. Moreover, depth videos are not for watching, but for virtual view rendering. Hence, unlike color video coding, the distortion of depth video coding causes rendering distortion during the DIBR process, and indirectly deteriorated the quality of intermediate virtual view. Correspondingly, the distortion measure in mode decision process of depth video coding is modified as a weighted average of the synthesized view distortion and the depth map distortion. The RD optimization criterion is improved for depth coding [6]. Although the HEVC intra-prediction and transform coding are well suited for the regions with nearly constant depth, they result in significant coding artifacts at sharp edges, which are visible in the synthesized intermediate views. To improve the coding performance in the regions with sharp edges, depth model modes are introduced for depth video coding, and appended into the HTM platform. Therefore, the coding complexity of depth video coding in 3D-HEVC is higher than texture video coding, which hampers the practical application of 3D-HEVC [9]. Therefore, low complexity depth video is hot topic in the fields of FVV systems.

To reduce the computational complexity of depth video, the straightforward way is directly using the fast algorithm of color video coding. In [10], an early CU termination algorithm (ECU) has been proposed to determine whether to stop the partition process if the best mode of current CU is Skip. A coded block flag-based fast mode (CFM) decision algorithm has been proposed [11]. The mode selection is terminated if the coded block flag of both luma and two chromas are zero. The methods in [10] and [11] were adopted in the test model of 3D-HEVC.

Recently, several fast algorithms [12]-[20] have been proposed to reduce the computational complexity of depth video inter coding. These algorithms can be classified into two types. The first type adjusts the CU size and mode decision strategies on the basis of appropriate thresholds [12]-[16], and the second type directly utilizes the coding information of the corresponding color video to accelerate the depth video coding [17]-[20]. A fast CU size decision scheme is proposed based on a depth no-synthesis-error model [12], and it hardly degrades the quality of the virtual views. Zhang et al. exploited the correlation between the current CU and its spatial-temporal neighboring CUs to early terminate the mode decision, and reduce the computational complexity of motion and disparity estimation [13]. Lei et al. proposed an early termination algorithm for DMM in depth video coding [14]. Zhao et al. proposed an efficient DMM decision algorithm for depth video coding [15]. Yang et al. proposed a hybrid algorithm, includes a estimation scheme of virtual view distortion, a QP adjustment scheme, and a fast CU partition scheme, to improve the encoding performance of depth video [16]. However, the complicated threshold selection in the fast algorithms is the overhead of depth video coding. Mora et al. utilized the phenomenon that the CU depth is generally less than the depth of the co-located CU in the corresponding color video, and proposed an algorithm to speed up the quadtree construction process [17]. The algorithm has been adopted in both the 3D-HEVC working draft and the HTM software. Lei et al. proposed a depth video coding algorithm that utilizes the depth-texture motion and structure similarities [18]. The encoding speed and RD performance are improved. Shen et al. proposed a fast mode decision for depth video coding by exploiting the inter-view mode correlation, the texture-depth correlation and the inter-level correlation [19]. Zhang et al. proposed a fast depth mode decision based on texture-depth correlation and edge classification [20].

Currently, machine learning, which is widely used in signal processing and big data analysis, is a hot research topic. Many fast algorithms based on machine learning have been proposed to reduce the complexity of color video coding [21]-[25]. Zhang et al. proposed an machine learning-based fast CU depth decision method to optimize the complexity allocation with RD cost constraints [21]. Shen et al. proposed an early termination algorithm of transform unit size decision based on Bayesian decision theory and correlation between the variance of a block and the transform unit size [22]. Tohidypour et al. proposed an online learning-based method to speed up the encoding process by adjusting the motion search range and complexity reduction of inter-/intra-prediction mode search [23]. Zhu et al. proposed a multi-class support vector machine-based fast algorithm for HEVC [24]. Kim et al. proposed a fast for HEVC algorithm that CU portioning is early terminated based on online leaning Bayesian decision rule [25]. However, the machine learning-based methods were proposed for color video, and cannot be efficiently applied to depth video coding because of the different statistics and characteristics between color video and depth video.

Inspired by the machine learning-fast algorithms in color video, we propose and realize a fast depth video coding algorithm in the 3D-HEVC encoder. The algorithm contains two parts, Bayesian-theory-based fast CU size decision and CU mode decision. In the first part, an off-line trained Bayesian model is built to predict the depth level of the current LCU. During the training process, the optimal depths of the corresponding LCU in color video, the LCUs in reference frames, and the neighboring LCUs are used as feature vector. In the second part, the mode decision is accelerated by jointly using the mode correlation of the spatial, inter-component (texture-depth), and inter-view neighboring CUs. The major contribution of our work is the utilization of off-line trained Bayesian model, and the depth level can be precisely predicted without threshold selection.

The remainder of this paper is organized as follows. In Section 2, we investigate the depth level and the prediction mode distribution in depth video. The proposed algorithm is described in detail in Section 3. Experimental results are given in Section 4, and the conclusions are drawn in Section 5.

2. Observation and Statistical Analysis

The basic encoding structure of 3D-HEVC is shown in **Fig. 1**. In general, the input signal for the 3D-HEVC encoder consists of multi-view color videos and associated depth videos. One of the multi-view color videos is selected as the independent view and it is encoded along with its corresponding depth video by using an unmodified HEVC encoder. The remaining views, known as dependent views, are coded using a modified HEVC encoder [26][27]. New coding tools, including disparity-compensated prediction, motion parameter inheritance, and inter-view motion prediction, achieve the maximum encoding efficiency. However, the computational complexity is increased.

Fig. 2 shows the possible partition of an LCU and the possible prediction mode of a CU. To find the optimal prediction mode of a CU and the optimal partition depth level of an LCU, a full search of all possible CU sizes and modes is performed. Therefore, CU size and CU mode decision in 3D-HEVC are time consuming.
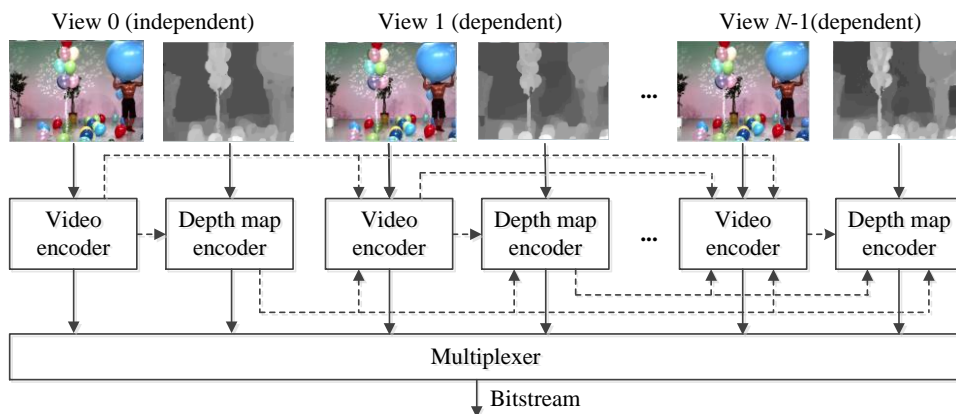

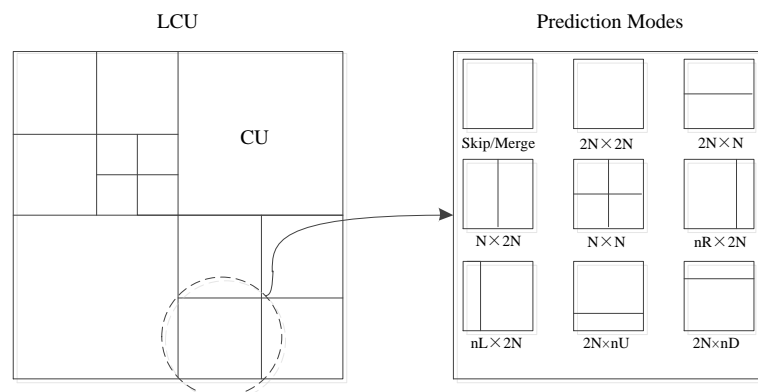**Fig. 1.** Encoding structure of 3D-HEVC [26]

Fig. 2. Possible partition of an LCU and possible prediction mode of a CU

To investigate the depth level and mode distribution of depth video, we conduct explorative experiments under 4 pairs of QPs, (25, 34), (30, 39), (35, 42), and (40, 45). In each QP pair, the former and the latter one are QPs for color video and the corresponding depth video. **Table 1** lists the proportion of LCUs in depth video with four depth levels of optimal CTU partition, 0, 1, 2 and 3. The average distribution of LCUs with depth level 0 is up to 88.14%. The proportion of LCUs, depth level ranging from 0 to 2, is 93.49% on average. Specially, the quadtree of CTU partition is simple in low-motion sequences, such as GhostTownFly and PoznanStreet. Hence, the distribution of LCUs with depth level 0 or 1 is more than 97%. Thus, the quadtree construction strategy in 3D-HEVC, trying all depth levels to get the optimal one, cannot further improve the encoding performance of the LCUs with depth level 0, 1, or 2. If we can predict the depth level of LCU partition in advance, unnecessary search process can be avoided. Consequently, the coding complexity of the 3D-HEVC encoder can be significantly reduced.

Table 1. Proportion of LCUs in depth video with four depth levels (%)

Sequences	0	1	2	3
Balloons	80.82	4.42	3.21	11.55
Kendo	87.07	4.62	2.54	5.77
Newspaper	82.09	3.38	2.83	11.69
GhostTownFly	96.14	1.43	0.86	1.58
Shark	86.86	4.14	2.47	6.53
PoznanStreet	95.84	1.29	0.91	1.96
Average	88.14	3.21	2.13	6.51

Table 2. Proportion of predition modes of depth video in 3D-HEVC encoder (%)

Sequences	Skip	Merge	Inter-2N×2N	Inter-Others	Intra
Balloons	91.93	2.17	3.03	0.46	2.43
Kendo	88.30	2.25	1.97	0.36	7.12
Newspaper	93.42	1.44	2.56	0.39	2.18
GhostTownFly	93.87	4.45	0.65	0.05	0.98
Shark	92.35	2.52	2.47	0.58	2.08
PoznanStreet	98.31	0.49	0.48	0.09	0.64
Average	93.03	2.22	1.86	0.32	2.57

Inter-Others include Inter-N×2N, Inter-2N×N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, and Inter-nR×2N;
Intra include Intra-2N×2N and Intra-N×N.

Table 2 lists the proportion of prediction modes of depth video in 3D-HEVC encoder. The CUs that the optimal prediction mode is Skip is more than 93.03%. On the contrary, for the remainder 6.97% of CUs, the optimal prediction mode is Merge, Inter-2N×2N, Inter-N×2N, Inter-2N×N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, Inter-nR×2N, Intra-2N×2N, and Intra-N×N. If we can predict the optimal prediction mode of a CU in advance, and skip the search process of other inter modes, the computational complexity of the depth video coding in 3D-HEVC encoder will be reduced significantly.

3. Bayesian-theory-based Fast CU Size and Mode Decision Algorithm

We proposed a fast depth video coding based on machine learning. In the proposed algorithm, we model the CTU partition process as a multi-class classification problem and solve it using Bayesian theory. **Fig. 3** shows the flowchart of the proposed algorithm in which *dcur* is the current partition depth, *pre_d* is the predicted depth level, CU_{ct} and CU_{cv} are the co-located CU in the texture video and the co-located CU in the depth video of the independent view. The proposed algorithm consists of two parts, fast CU size decision based on Bayesian theory and fast CU mode decision. In the CU size decision part, an off-line trained Bayesian model is built based on the encoding information of depth and video coding in 3D-HEVC encoder. When an CTU is encoded, the features are extracted, and an off-line trained Bayesian model is loaded to predict the depth level of the current LCU. The predicted depth level decides whether to early terminate CU splitting. In the mode decision part, the depth level is used as one kind of the auxiliary information. If the depth level is 0, the Skip, Merge, Inter-2N×2N, and Intra modes are searched, and one of them is selected as the prediction mode. If the mode of the parent CU is Skip, only Skip mode is selected as the optimal mode. For the CUs in dependent view, if both the modes of CU_{ct} and CU_{cv} are Skip, Skip is directly selected as the optimal mode.
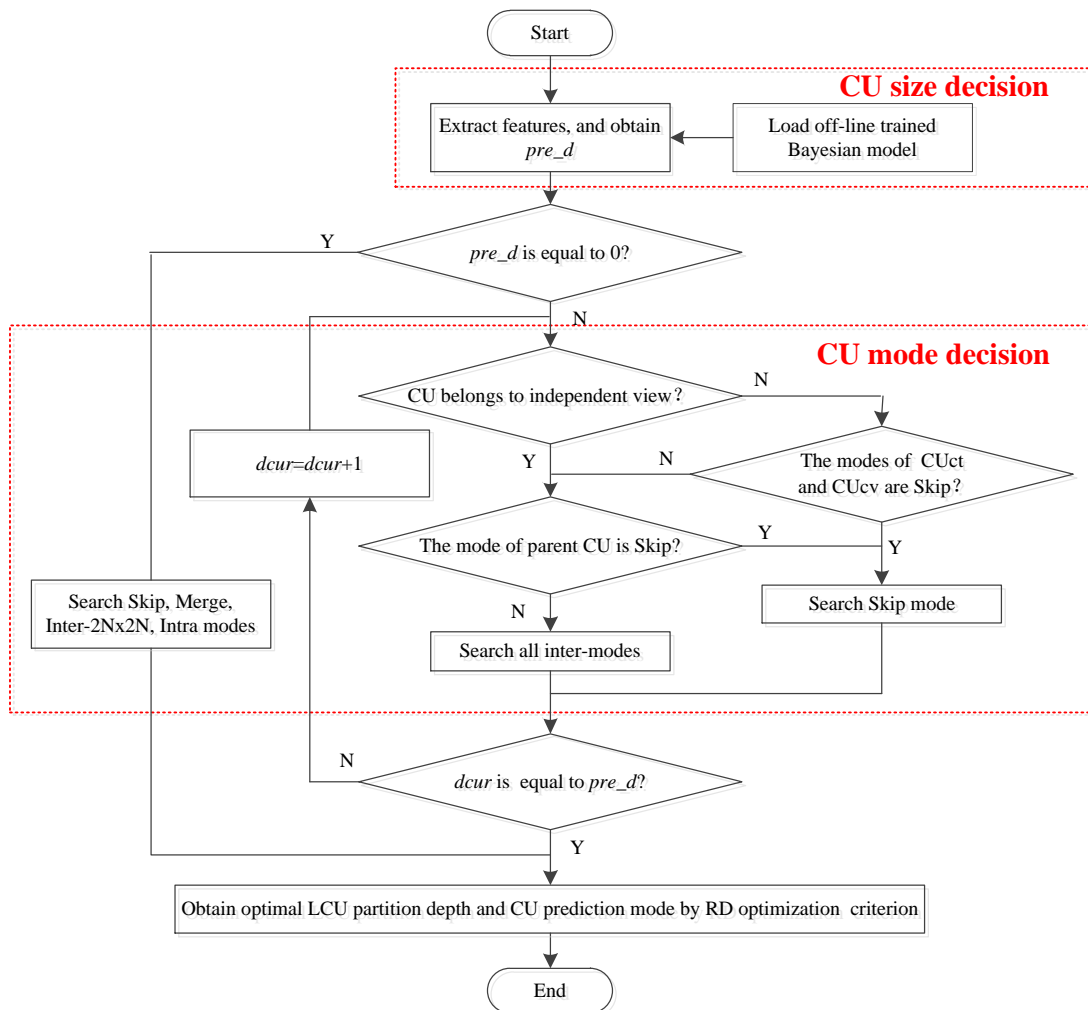

Fig. 3. Flowchart of the proposed fast depth inter-coding algorithm

3.1 Fast CU Size Decision

In the proposed algorithm, CU size decision is accelerated by off-line trained Bayesian model. This subsection describes the model building process in detail, and then analyzes the robustness of the Bayesian model.

3.1.1 Fast CU Size Decision Based on Bayesian Theory

The statistical results of depth levels are shown in **Fig. 4**. *Ddepth* denotes the depth level of LCUs to be encoded, *Dtexture* denotes the depth level of the co-located LCU in the texture video, $Dref_1$ and $Dref_2$ respectively denote the depth levels of the co-located LCUs in the forward and backward reference frames. Further, *Dabove* and *Dleft* denote the depth levels of the above and left LCUs, respectively. As far as the distribution trend of depth level is concerned, the LCUs to be encoded are similar to that of the co-located and spatial neighbor LCUs. Thus, we can use the correlation to predict the optimal partition depth of the current LCU.
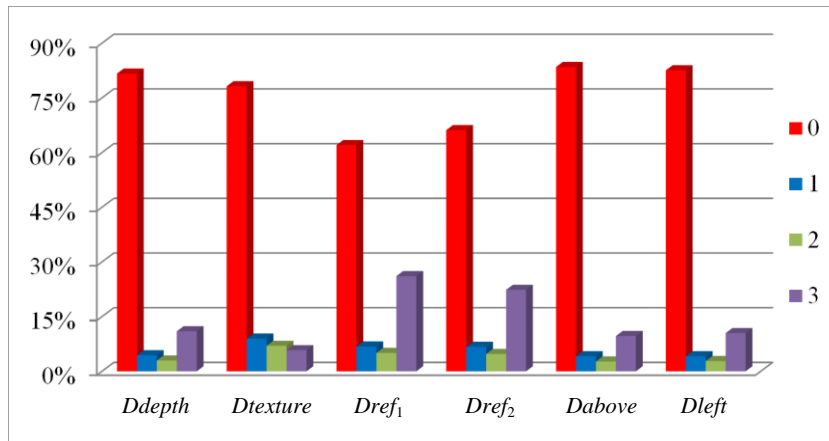

Fig. 4. Optimal partition depth distribution of the current LCU and its neighboring LCUs

Let d_i be the depth levels of optimal partition which have 4 states, 0, 1, 2, and 3, and **x** be the feature vector that consists of five variables, *Dtexture*, $Dref_1$, $Dref_2$, *Dabove*, and *Dleft*, the solution of the multi-class classification problem can be given by

$$pre\_d = \arg\max_{d} P(d \mid \mathbf{x}) \tag{1}$$

where *pre_d* is the class label and also denotes the depth level of optimal partition of the LCU. For a given **x**, $P(d|\mathbf{x})$ denotes the posterior probability of d. The d with maximal probability is predicted value of depth level. Since Bayesian theory is a generated learning algorithm, we should calculate $P(\mathbf{x}|d)$ and $P(d)$. Hence,

$$\arg\max_{d} P(d \mid \mathbf{x}) = \arg\max_{d} \frac{P(\mathbf{x} \mid d) P(d)}{P(\mathbf{x})} \tag{2}$$

where $P(\mathbf{x})$ denotes the probability of the feature vector **x** in the LCU training samples. For the parameter $P(\mathbf{x}|d)$, we suppose that the elements in the feature vector **x** are independent each other. We have

$$P(\mathbf{x} \mid d) = \prod_{j=1}^{5} P(x_j \mid d) \tag{3}$$

where $P(x_j|d)$ is the occurrence probability of x_j (the jth element in the feature vector **x**) for a given d, and it can be obtained using the LCU training samples,

$$P(x_j \mid d) = \frac{\sum_{i=1}^{k} \Gamma\{x_j = z_i\}\Gamma\{d_i = d\}}{\sum_{i=1}^{k} \Gamma\{d_i = d\}} \qquad j = 1,2,3,4,5 \tag{4}$$

where z_i denotes the optimal partition depth of the ith LCU, k is the total number of training samples, $\Gamma\{true\}$ equals 1, and $\Gamma\{false\}$ equals 0 (i.e., $\Gamma\{0 = 0\} = 1$, $\Gamma\{1 = 3\} = 0$). The parameter $P(d)$ denotes the probability of d in the LCU training samples. Because d has four kinds of values, we can use polynomial distribution to model $P(d)$, which is defined as

$$P(d) = \prod_{d=0}^{3} P_d^{\Gamma\{d_i = d\}} \qquad s.t. \ \sum_{d=1}^{3} P_d = 1 \tag{5}$$

where P_d denotes the probability of d in the training samples.

Table 3. Predicted depth level and prediction accuracy of Bayesian model (%)

Sequences	0	1	2	3	*pre_acc*
Balloons	62.42	14.89	4.32	18.38	97.59
Kendo	73.63	13.58	3.76	9.03	96.65
Newspaper	62.96	13.95	4.28	18.83	97.83
GhostTownFly	87.41	7.94	1.24	3.42	99.47
Shark	67.91	17.66	3.69	10.75	97.24
PoznanStreet	86.64	7.78	1.36	4.23	99.28
Average	73.49	12.63	3.10	10.77	98.01

3.1.2 Robustness of Model Established by Bayesian Theory

The off-line training process contains three steps, encoding, training sample selection, and model training. We performed encoding experiments on original 3D-HEVC encoding platform HTM-10.0. Multiple video sequences, with different scene feature, are encoded under 4 pairs of QPs, (25, 34), (30, 39), (35, 42), and (40, 45). For each sequence, the first 25 frames are used for training, and succeeding 80 frames, except the I and P slices, are used for test. We randomly select 30% of the data in the training frames as the training samples to establish the Bayesian model. **Table 3** lists prediction accuracy of the Bayesian model. In this table, 0, 1, 2, and 3 represent the predicted depth levels of depth video. The average proportion of LCUs with predicted level 0 is 73.49%. It is noted that the statistical results in **Table 1** is based on the same LCUs as **Table 3**. Compared with **Table 1**, the proportion of LCUs with predicted depth level 0 is less than the proportion of LCUs with depth level 0 in original 3D-HEVC encoder. In other words, the prediction depth level of many LCUs is slightly larger than their original depth level. The increment of prediction depth level does not change encoding performance. Only the decrement of prediction level, the predicted depth level is less than its original depth level, deteriorates the encoding performance. Hence, we estimated the robustness of the model by the prediction accuracy *pre_acc*, and computed by

$$pre\_acc = \frac{\sum_{i=1}^{p} \Gamma\{d_i \leq pre\_d_i\}}{p} \times 100\% \tag{6}$$

where p denotes the total number of test samples. We can see that the prediction accuracy ranges from 96.65% to 99.47% for various test sequences, and it is 98.01% on average. Therefore, we can conclude that the Bayesian model is sufficiently efficient to predict the depth level of an LCU.

3.2 CU Mode Decision

The decision process of the CU optimal prediction mode is quite complex. Based on statistical analysis, we propose fast CU mode decision method that can be divided into two stages, LCU mode decision and early Skip mode decision.

3.2.1 LCU Mode Decision

Table 4 lists the statistical results of the optimal prediction modes of LCUs in depth video when the depth level equals 0. Clearly, most LCUs in inter frames choose the Skip mode as the optimal mode. The total proportion of LCUs that choose Inter-N×2N, Inter-2N×N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, or Inter-nR×2N as the optimal mode is only 0.18%. Hence, if the depth level of an LCU is 0, only Skip, Merge, Inter-2N×2N, and Intra modes are searched. Unnecessary mode search can be avoided without significant encoding performance degradation.

Table 4. Statistical results of the optimal mode of depth video when depth level equals 0 (%)

Sequences	Skip	Merge	Inter-2N×2N	Inter-Others	Intra
Balloons	94.75	1.56	2.39	0.25	1.05
Kendo	88.93	2.23	1.91	0.27	6.65
Newspaper	97.09	0.68	1.69	0.15	0.39
GhostTownFly	93.82	4.89	0.40	0.03	0.87
Shark	95.30	1.82	1.82	0.32	0.73
PoznanStreet	99.05	0.34	0.29	0.04	0.29
Average	94.82	1.92	1.42	0.18	1.67

Inter-Others includes Inter-N×2N, Inter-2N×N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, Inter-nR×2N；
Intra include Intra-2N×2N and Intra-N×N.

Table 5. Statistical results of the optimal mode of CUs if optimal modes of their parent CUs are Skip (%)

Sequences	Skip	Merge	Inter-2N×2N	Inter-Others	Intra
Balloons	99.52	0.15	0.17	0.02	0.14
Kendo	99.70	0.08	0.07	0.01	0.14
Newspaper	99.41	0.20	0.16	0.03	0.20
GhostTownFly	99.86	0.08	0.04	0.00	0.02
Shark	99.68	0.11	0.13	0.04	0.04
PoznanStreet	99.87	0.06	0.03	0.01	0.03
Average	99.67	0.11	0.10	0.02	0.10

Inter-Others include Inter-N×2N, Inter-2N×N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, Inter-nR×2N；
Intra include Intra-2N×2N and Intra-N×N.

3.2.2 Early Skip Mode Decision

Based on observation and statistical analysis, we found the phenomena that the optimal modes of almost all the CUs are Skip under two scenarios, 1) the optimal modes of their parent

CUs are Skip, and 2) the CUs are in a dependent view and the optimal modes of the corresponding CU_{cv} and CU_{ct} are Skip. **Table 5** lists the statistical results of the optimal prediction modes of CUs in depth video if the optimal prediction modes of their parent CUs are Skip. Skip is the optimal mode for most CUs. The proportion of CUs that choose other modes as the optimal mode amounts to 0.33%. **Table 6** lists the statistical results of the optimal prediction mode of CUs in a dependent view if the optimal modes of CU_{cv} and CU_{ct} are Skip. We can conclude that most CUs choose the Skip mode as the best mode. The total proportion of CUs that choose other modes as the optimal mode is 0.86%. Hence, only Skip is searched for current CU under the two scenarios. The average miss rates are less than 0.33% and 0.86 respectively, and negligible.

Table 6. Statistical results of the optimal mode of CUs in a dependent view if optimal modes of CU_{cv} and CU_{ct} are Skip (%)

Sequences	Skip	Merge	Inter-$2N \times 2N$	Inter-Others	Intra
Balloons	98.53	0.46	0.46	0.06	0.50
Kendo	98.46	0.29	0.25	0.06	0.93
Newspaper	98.91	0.32	0.35	0.07	0.35
GhostTownFly	99.49	0.46	0.03	0.00	0.02
Shark	99.57	0.13	0.21	0.05	0.03
PoznanStreet	99.88	0.06	0.03	0.01	0.03
Average	99.14	0.29	0.22	0.04	0.31

Inter-Others include Inter-N×2N, Inter-2N×N, Inter-N×N, Inter-2N×nU, Inter-2N×nD, Inter-nL×2N, and Inter-nR×2N；
Intra include Intra-2N×2N and Intra-N×N.

4. Experimental Results and Analysis

4.1 Experimental conditions and sequences

To evaluate the performance of the proposed algorithm, we implemented it on the 3D-HEVC reference software HTM-10.0. The test environment recommend by JCT-3V [28] is described as follows. The prediction structure is the hierarchical B pictures. The group of pictures (GOP) is 8, and the period of the I slice is 24. Both the ranges of motion and disparity search are 64. The LCU size is 64×64, and the depth level of an LCU ranges from 0 to 3. Four pairs of QPs, (25,34), (30,39), (35,42), and (40,45), are tested.

As shown in **Table 7**, six sequences recommended by JCT-3V were tested. The resolution of Baloons, Kendo, and Newspaper is 1024×768, whileas, the resolution of GhostTownFly, Shark, and PoznanStreet is 1920×1088. The features of these sequences are different. The encoded views, virtual view and frames are listed in **Table 7**.

Table 7. Test sequences

Sequences	Resolution	Encoded Views	Virtual View	Frames
Balloons	1024×768	3,5	4	300
Kendo	1024×768	3,5	4	300
Newspaper	1024×768	4,6	5	300
GhostTownFly	1920×1088	1,9	5	250
Shark	1920×1088	1,9	5	300
PoznanStreet	1920×1088	3,5	4	250

The performance of the proposed algorithm is measured in terms of PSNR, computational complexity, and bitrate. PSNR is the quality of synthesized virtual view, and the bitrate represents the total bitrate of texture video and depth video. The bitrate variation is evaluated in terms of the Bjontegaard metric *BD-Rate* [29]. In order to illustrate the effectiveness of the proposed algorithm, the fast algorithms in [10], [11], and [17], embedded into the reference software HTM, are used for comprehensive comparison. *Mora*, *ECU*, and *CFM* respectively denote the algorithms in [17], [10], and [11]. We also compared the state-of-the-art algorithms in terms of speedup and RD performance.

4.2 Speedup performance of the proposed algorithm

Table 8 lists the time saving of various fast algorithms under different *QP*s that is calculated by using original HTM-10.0 as benchmark. Negative sign means time reduction. DT_x is the time saving of depth and color video coding of algorithm x,

$$DT_x = (T_x - T_{ori}) / T_{ori} \times 100\% \qquad x \in \{Mora, CFM, ECU, CUSD, CUMD, Pro\} \quad (9)$$

where T_x denotes the entire encoding time of algorithm x, T_{ori} denotes the entire encoding time of the original HTM-10.0. *CUSD* denotes the first part of our algorithm, i.e., CU size decision based on Bayesian theory. *CUMD* denotes the second part of our algorithm, i.e., CU mode decision. *Pro* denotes the proposed algorithm that is the combination of *CUSD* and *CUMD*. Dt_x is the time saving of depth video of algorithm x, calculated by

$$Dt_x = (t_x - t_{ori}) / t_{ori} \times 100\% \qquad x \in \{Mora, CFM, ECU, CUSD, CUMD, Pro\} \quad (10)$$

where t_x denotes encoding time of depth video of algorithm x, t_{ori} denotes encoding time of depth video of the original HTM-10.0.

Table 8. Time saving comparison of various fast algorithms (%)

Sequences	QP	DT_{Mora}	Dt_{Mora}	DT_{CFM}	Dt_{CFM}	DT_{ECU}	Dt_{ECU}	DT_{CUSD}	Dt_{CUSD}	DT_{CUMD}	Dt_{CUMD}	DT_{Pro}	Dt_{Pro}
Balloons	25	-30.76	-51.86	-13.97	-23.24	-23.71	-39.75	-12.26	-20.34	-14.94	-24.74	**-30.56**	**-51.09**
	30	-35.16	-57.62	-16.20	-26.45	-29.87	-48.75	-19.41	-31.70	-16.25	-26.37	**-35.81**	**-58.49**
	35	-38.71	-61.16	-18.10	-28.39	-34.39	-54.12	-26.03	-40.95	-17.37	-27.22	**-40.54**	**-63.80**
	40	-41.68	-63.71	-19.21	-29.17	-38.86	-59.03	-31.20	-47.41	-18.44	-27.92	**-44.80**	**-68.17**
Kendo	25	-33.19	-54.72	-12.44	-20.29	-20.81	-34.15	-19.26	-31.44	-13.44	-22.00	**-33.56**	**-55.14**
	30	-35.53	-58.23	-15.32	-24.99	-28.57	-46.75	-25.23	-41.38	-15.09	-25.18	**-38.02**	**-62.45**
	35	-37.96	-60.93	-17.10	-27.21	-33.24	-53.30	-28.97	-46.21	-16.41	-26.43	**-40.83**	**-65.72**
	40	-40.36	-63.24	-18.24	-28.44	-37.48	-58.65	-31.10	-49.16	-17.32	-27.39	**-43.49**	**-68.18**
Newspaper	25	-35.81	-56.83	-14.63	-23.12	-28.18	-44.73	-13.46	-21.18	-15.12	-23.76	**-31.79**	**-50.37**
	30	-38.97	-60.19	-16.85	-25.95	-32.53	-50.09	-20.78	-31.80	-16.49	-25.14	**-36.99**	**-57.00**
	35	-41.33	-62.28	-17.82	-26.74	-35.81	-53.78	-26.51	-39.79	-17.19	-25.74	**-40.95**	**-61.38**
	40	-43.57	-64.32	-18.81	-27.89	-39.37	-58.01	-31.51	-46.26	-17.79	-26.34	**-44.30**	**-65.25**
GhostTownFly	25	-34.40	-57.84	-16.36	-27.25	-28.25	-47.52	-27.92	-46.77	-15.59	-26.29	**-40.56**	**-68.04**
	30	-39.44	-62.92	-19.04	-30.22	-34.91	-55.87	-33.22	-52.83	-15.69	-24.80	**-44.90**	**-71.57**
	35	-43.23	-65.37	-19.19	-28.93	-40.00	-60.73	-36.13	-54.57	-17.33	-25.99	**-48.34**	**-72.49**
	40	-46.01	-67.64	-20.88	-30.61	-43.69	-64.18	-38.47	-56.40	-18.98	-27.87	**-50.93**	**-74.70**
Shark	25	-25.85	-46.79	-13.36	-24.27	-24.16	-43.77	-15.94	-28.44	-13.35	-24.13	**-31.28**	**-56.62**
	30	-31.10	-54.72	-15.88	-27.84	-30.55	-53.77	-24.64	-42.83	-15.17	-26.20	**-37.65**	**-66.28**
	35	-36.27	-60.08	-17.84	-29.38	-36.47	-60.56	-30.57	-50.17	-16.50	-27.03	**-42.91**	**-71.01**
	40	-41.26	-64.38	-19.60	-30.51	-41.21	-64.50	-34.84	-54.13	-17.80	-27.60	**-47.06**	**-73.40**
PoznanStreet	25	-33.07	-56.97	-15.89	-27.32	-32.95	-56.72	-24.64	-41.85	-15.49	-26.23	**-36.68**	**-63.09**
	30	-39.43	-62.91	-18.46	-29.41	-38.94	-62.28	-32.30	-51.10	-17.35	-27.51	**-43.82**	**-69.92**
	35	-43.49	-65.28	-20.09	-29.98	-42.92	-64.43	-35.86	-53.68	-18.37	-27.54	**-48.27**	**-72.51**
	40	-46.06	-67.05	-20.57	-29.85	-46.14	-67.17	-38.70	-56.14	-18.95	-27.44	**-50.88**	**-74.13**
Average		-38.03	-60.29	-17.33	-27.39	-34.29	-54.28	-27.46	-43.19	-16.52	-26.12	**-41.04**	**-65.03**

Compared with the original HTM-10.0 encoder, the proposed algorithm can reduce the entire encoding time and depth video encoding time by 41.04% and 65.03%, respectively. The proposed algorithm is efficient for the sequences with slight motion sequences and higher resolution, such as GhostTownFly and PonzanStreet. The main reason is that more CUs can satisfy the early termination condition. The fast algorithm *CFM* reduces the entire encoding time and depth video encoding time by 17.33% and 27.39%, respectively. The fast algorithm *ECU* reduces the entire encoding time and depth video encoding time by 34.29% and 54.28%, respectively. The algorithms *CFM* and *ECU* are initially designed for color video. They are directly applied into 3D-HEVC, and have not taken the feature of depth video into consideration. Hence, the time saving of the algorithms is less than the proposed algorithm. The fast algorithm *Mora* reduces the entire encoding time and depth video encoding time by 38.03% and 60.29%, respectively. In terms of time saving, the proposed algorithm is equivalent to the algorithm *Mora* for sequences with resolution 1024×768, and is superior to the algorithm *Mora* for other sequences with resolution 1920×1088. As a whole, the speedup performance of the proposed algorithm is better than the algorithms *Mora*, *CFM* and *ECU*.

The underlying reason of the better speedup performance is that the contribution on both CU size decision and CU mode decision. *CUSD* reduces the depth video encoding time by 43.19% on average. For the sequences GhostTownFly and PonzanStreet, *CUSD* can efficiently skip unnecessary CU partition levels, and respectively saves the encoding time ranging from 46.77% to 56.40%, and from 41.85% to 56.14%. *CUMD* reduces the depth video encoding time by 26.12% on average. The time saving of *CUMD* is nearly the same for all sequences, because the proportion of CUs that meet the early termination condition is nearly the same for all sequences.

Table 9 lists the encoding performance comparison with Lei's [14], Zhao's [15], Yang's [16], Shen's [19], and Zhang's [20] algorithms. The proposed algorithm outperforms Lei's and Zhao's algorithms because the algorithms are optimization for DMM. Yang's algorithm improves the encoding speed by a CU partition scheme, while Shen's, Zhang's algorithms optimize the mode selection process. Hence, the algorithms cannot overall improve the speedup performance.

Table 9. Encoding performance comparison with the state-of-the-art algorithms

Algorithms	HTM Version	BD-Rate (%)	Time Saving (%)
Lei [14]	15.1	0.0	20
Zhao [15]	16.0	0.6	31
Yang [16]	14.0	-12.4	47
Shen [19]	9.0	1.3	34
Zhang [20]	16.0	0.8	43
Proposed	10.0	-0.4	65

4.3 Rate distortion performance of the proposed algorithm

Table 10 tabulates the quality variation of the synthesized view of various fast algorithms under different *QP*s that is calculated by using original HTM-10.0 encoder as benchmark. Negative sign means quality decrease. $DPSNR_x$ is the PSNR difference between algorithm x and the original HTM-10.0 encoder,

$$DPSNR_x = PSNR_x - PSNR_{ori} \qquad x \in \{Mora, CFM, ECU, CUSD, CUMD, Pro\} \quad (11)$$

where $PSNR_x$ and $PSNR_{ori}$ in the synthesized view are measured with respect to the view synthesized using a compressed view.

For the proposed algorithm, the synthesized view quality variation ranges from -0.01dB to -0.17dB. For the algorithms *Mora*, *CFM*, and *ECU*, the synthesized view quality variation ranges from -0.01dB to -0.23dB, -0.02dB to 0.01dB, and -0.01dB to -0.26dB, respectively. Compared with the algorithms *ECU* and *Mora*, the proposed algorithm can get the virtual view with nearly the same quality on average. The average quality of the proposed algorithm is slight lower than that of the *CFM*. The average quality variation of *CUSD* and *CUMD* are -0.01dB and 0.00dB. The proposed algorithm, combination of *CUSD* and *CUMD*, slight degrades the quality of synthesized virtual view. The depth videos of sequences Baloons and Kendo are estimated by software. There are many errors in the depth videos of these sequences. These sequences can tolerate more distortion than the computer-generated sequences. Thus, coding errors introduced by the fast algorithms slightly impacts on the quality of the synthesized view. By contrast, the coding errors of depth videos in sequence Shark leads to a severe decline in the quality of the synthesized view.

Table 10. Quality variation of synthesized view of various fast algorithms (dB)

Sequences	*QP*	*DPSNR$_{Mora}$*	*DPSNR$_{CFM}$*	*DPSNR$_{ECU}$*	*DPSNR$_{CUSD}$*	*DPSNR$_{CUMD}$*	*DPSNR$_{Pro}$*
Balloons	25	-0.03	0.00	-0.01	0.01	0.00	**0.00**
	30	-0.01	0.01	-0.01	0.02	0.01	**0.01**
	35	-0.04	-0.02	-0.04	-0.01	-0.01	**-0.01**
	40	-0.04	0.00	-0.02	-0.01	0.00	**-0.01**
Kendo	25	-0.08	0.00	-0.04	-0.01	-0.01	**-0.03**
	30	-0.06	0.01	-0.05	-0.01	0.00	**-0.04**
	35	-0.05	0.00	-0.04	-0.01	-0.01	**-0.05**
	40	-0.02	0.01	-0.02	0.00	0.01	**-0.03**
Newspaper	25	-0.03	-0.01	-0.02	0.00	0.00	**0.00**
	30	-0.01	0.00	-0.02	0.00	0.00	**0.00**
	35	-0.02	0.01	-0.01	0.00	0.01	**0.00**
	40	-0.03	-0.01	-0.01	-0.01	0.00	**-0.02**
GhostTownFly	25	-0.09	-0.02	-0.09	-0.06	-0.03	**-0.07**
	30	-0.02	-0.01	-0.06	-0.01	0.00	**-0.04**
	35	-0.03	0.00	-0.03	0.00	0.00	**-0.03**
	40	-0.01	0.00	-0.01	0.00	-0.01	**-0.01**
Shark	25	-0.23	-0.02	-0.26	-0.04	-0.02	**-0.13**
	30	-0.12	-0.01	-0.19	-0.05	-0.01	**-0.17**
	35	-0.06	-0.01	-0.13	-0.06	-0.01	**-0.13**
	40	-0.02	0.01	-0.08	-0.02	-0.01	**-0.08**
PoznanStreet	25	-0.02	0.00	-0.02	-0.01	0.00	**0.00**
	30	-0.01	0.00	-0.01	-0.01	-0.01	**0.00**
	35	-0.01	0.00	-0.03	0.00	0.00	**-0.02**
	40	-0.01	0.00	-0.02	-0.01	0.00	**-0.02**
Average		-0.04	0.00	-0.05	-0.01	0.00	**-0.04**

Fig. 5 shows the results of distortion comparison of various fast algorithms. The distortion is the difference of synthesized virutal images between the fast algorithm and the original HTM-10.0. For visualization, the distortion part of the virtual view synthesized image is enlarged by a factor of 6. White means distortion. The virtual view is synthesized using the color video and the associated depth video. The depth value generates the pixel displacement between the synthesized virtual view and the real view during the DIBR process. The depth at boundaries is sensitive to the distortion as far as the quality of synthesized virtual view is concerned. The depth distortion results in quality deterioration at boundaries. Consequently, as illustrated in **Fig. 5**, the distortion in synthesized virtual view mainly locates at boundaries. Among the fast algorithms, *CUSD* can get synthesized virtual view with minimal distortion, whereas the algorithm *Mora* generates virtual view with maximal distortion. It is consistent
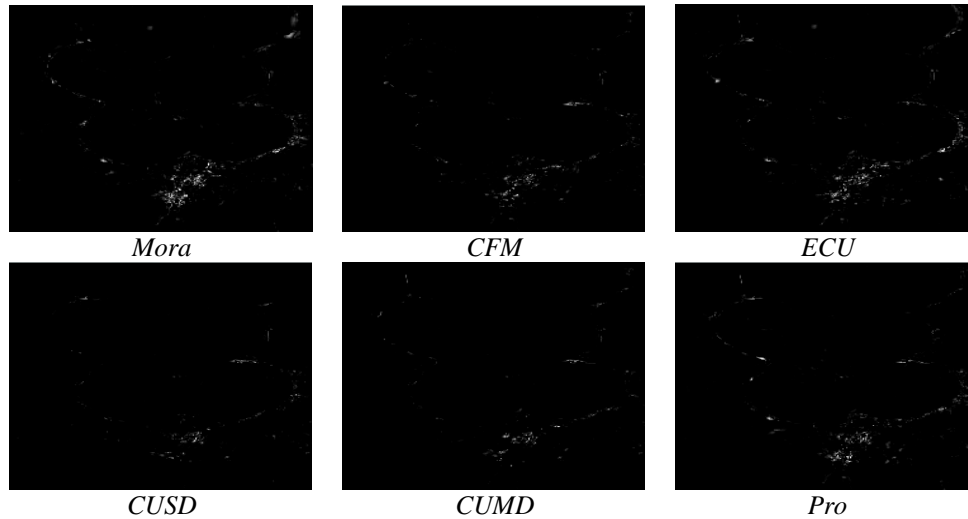
with the the *PSNR* variation in **Table 10**.

Fig. 5. Synthesized image distortion of various algorithms of 6th frame in 4th view in Balloons sequence when *QP* equals to 25.

Table 11 lists the *BD-Rate* variation of various fast algorithms that is calculated by using original HTM-10.0 encoder as benchmark. Negative sign means bitrate saving. The fast algorithms *Mora*, *CFM*, and *ECU* achieve *BD-Rate* variation of -3.45%, -0.10%, and -1.40% on average, respectively. The *BD-Rate* variation of the proposed algorithm is -0.37% on average. It indicates that the bitrate saving of the proposed algorithm is 0.37% on average under same PSNR when compare to original HTM-10.0. The *BD-Rate* variation of *CUSD* and *CUMD* are -0.41% and 0.10%, respectively.

For many sequences, The BD-rate is negative which means the bitrate saving under the same quality. PSNR increase and bitrate decrease, individually or collectively, will leads to negative BD-rate. As shown in **Table 10**, the fast algorithms decrease the PSNR for most of the sequences. Hence, the reason of the negative BD-rate is bitrate decrease. In sequences GhostTownFly and PozanStreet, the bitrate decrease cannot counterbalance the PSNR decrease, and the BD-rate is positive for all fast algorithms. The bitrate saving of these algorithms is contributed by large CU size and more Skip mode. In the proposed algorithm, more CTUs are encoded with large CU size, and more CUs selected Skip as optimal mode. In algorithm *Mora*, CU depth limitation and mode pre-decision are utilized, and the bitrate reduction of depth video mainly caused by the cost of transmitting split flags and partition sizes.

For the proposed algorithm, the *BD-Rate* variation of Newspaper and Shark is -3.67% and 2.44%, respectively. The large difference is caused by PSNR variance. The PSNR variation of Newspaper and Shark are ranging from -0.02dB to 0.00dB, and from -0.17dB to -0.08dB. Hence, the PSNR decrease of Shark is more than that the Newspaper.

Compared with other methods, the algorithm *CFM* slightly changes the quality of the synthesized view, the *BD-Rate* is ranging from -0.64% to 0.34%, and -0.10% on average. It indicates that the fast mode selection strategy in the algorithm *CFM* does not work for many CUs. It is coincide with analysis of time saving.

Table 9 also tablutes the RD performance comparson with the state-of-the-art algorithm. The proposed algorithm is equivalent to the Lei's, Zhao's, Shen's, and Zhang's algorithms.

Because a region-based QP adjustment scheme is adopted in Yang's algorithm, the bitrate is great reduced, and significantly less than other than algorithms.

Table 11. *BD-Rate* variation of various fast algorithms (%)

Sequences	$BDBR_{Mora}$	$BDBR_{CFM}$	$BDBR_{ECU}$	$BDBR_{CUSD}$	$BDBR_{CUMD}$	$BDBR_{Pro}$
Balloons	-8.00	**-0.23**	-4.32	-1.36	-0.34	-3.28
Kendo	-2.66	-0.10	-1.37	-1.38	**0.67**	0.25
Newspaper	-10.90	**-0.64**	-6.48	-1.08	-0.72	-3.67
GhostTownFly	0.55	0.34	1.18	0.45	0.46	**1.41**
Shark	0.17	-0.05	1.78	0.47	0.18	**2.44**
PoznanStreet	0.15	0.11	**0.81**	0.43	0.32	0.64
Average	-3.45	-0.10	-1.40	-0.41	**0.10**	-0.37

5. Conclusion

The encoding process of the 3D-HEVC encoder includes color video encoding and depth video encoding. However, compared with color video encoding, depth video encoding consumes more encoding time under the same configuration. In this paper, we proposed a fast CU size decision and CU mode decision algorithm to reduce the computational complexity of depth video coding by exploiting correlations among MVD signals. The experimental results showed that the proposed algorithm significantly reduces the computational complexity of the 3D-HEVC encoder with negligible quality deterioration of synthesized view. Moreover, the proposed algorithm outperforms the state-of-the-art algorithms in terms of time saving. The *BD-Rate* variation indicates that the proposed algorithm can nearly maintain the RD performance. In the future, we will further investigate the feature of depth video coding, and realize fast algorithm based on machine learning to achieve better coding performance.

References

[1] Tanimoto M., "FTV: Free-viewpoint television," *Signal Processing: Image Commutation*, vol. 27, no.6, pp. 555-570, 2012. Article (CrossRef Link).

[2] Chen Y., Hannuksela M.M., Suzuki T., et al., "Overview of the MVC+ D 3D video coding standard," *Journal of Visual Communication and Image Representation*, vol. 25, no. 4, pp. 679-688, 2014. Article (CrossRef Link).

[3] Fehn C., "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *Proceedings of SPIE - The International Society for Optical Engineering*, pp. 93-104, 2004. Article (CrossRef Link).

[4] Sullivan G.J., Ohm J.R., Han W.J., et al., "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. Article (CrossRef Link).

[5] Hannuksela M.M., Rusanovskyy D., Su W., et al., "Multi view-video-plus-depth coding based on the advanced video coding standard," *IEEE Transactions on Image Processing*, vol. 22, no.9, pp.3449-3458, 2013. Article (CrossRef Link).

[6] Chen Y., Tech G., Wegner K., et al., "Test model 10 of 3D-HEVC and MV-HEVC," *Document of Joint Collaborative Team on 3D Video Coding Extension Development*, JCT3V JI003, 2014.

[7] Helle P., Oudin S., Bross B., et al., "Block merging for quadtree-based partitioning in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no.12, pp.1720-1731, 2012. Article (CrossRef Link).

[8] Bross B., Han W.J., Sullivan G.J., et al., "High efficiency video coding (HEVC) text specification draft 10 (JCTVCL1003)," *JCT-VC Meeting (Joint Collaborative Team of ISO/IEC MPEG & ITU-T VCEG)*, 2013.

[9] Zhang Q., Wang X., Huang X., et al., "Fast mode decision algorithm for 3D-HEVC encoding optimization based on depth information," *Digital Signal Processing*, vol. 44, no. 1, pp.37-46, 2015. Article (CrossRef Link).

[10] Gweon R., LEE Y.-L., "Early termination of CU encoding to reduce HEVC complexity," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 95, no. 7, 1215-1218, 2012.Article (CrossRef Link).

[11] Choi K., Park S.H., Jang E.S., "Coding tree pruning based CU early termination," *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 6th Meeting*, JCTVC, Torino, Italy 14-22 , 2011.

[12] Chung K. L., Huang Y. H., Lin C.H., et al., "Novel bitrate-saving and fast coding for depth videos in 3D-HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no.10, pp.1859-1869, 2016. Article (CrossRef Link).

[13] Zhang Q., Chen M., Huang X., et al., "Low-complexity depth map compression in HEVC-based 3D video coding," *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 2, pp. 1-14, 2015. Article (CrossRef Link).

[14] Lei H., Wang A., Liu W., et al., "Early termination algorithm for the depth modeling mode in three-dimensional extension of high efficiency video coding," *Journal of Electronic Imaging*, vol. 25, no. 5, pp. 050503-1 – 050503-4, 2016. Article (CrossRef Link).

[15] Zhao J., Zhao X., Zhang W., et al., "An efficient depth modeling mode decision algorithm for 3D-HEVC depth map coding," *Optik*, vol. 127, no. 24, pp. 12048-12055, 2016. Article (CrossRef Link).

[16] Yang C., An P., Shen L., et al., "Fast depth map coding based on virtual view quality," *Signal Processing: Image Communication*, vol. 47, no. c, pp. 183-192, 2016. Article (CrossRef Link).

[17] Mora E.G., Jung J., Cagnazzo M., et al., "Initialization, limitation, and predictive coding of the depth and texture quadtree in 3D-HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no.9, pp.1554-1565, 2014. Article (CrossRef Link).

[18] Lei J., Li S., Zhu C., et al., "Depth coding based on depth-texture motion and structure similarities," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no.2, pp.275-286, 2015. Article (CrossRef Link).

[19] Shen L., An P., Zhang Z., et al., "A 3D-HEVC Fast Mode Decision Algorithm for Real-Time Applications," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 11, no.3, pp.1-23, 2015. Article (CrossRef Link).

[20] Zhang Q., Zhang N., Wei T., et al., "Fast depth map mode decision based on depth–texture correlation and edge classification for 3D-HEVC," *Journal of Visual Communication and Image Representation*, vol. 45, no.c, pp.170-180, 2017. Article (CrossRef Link).

[21] Zhang Y., Kwong S., Wang X., et al., "Machine learning-based coding unit depth decisions for flexible complexity allocation in high efficiency video coding," *IEEE Transactions on Image Processing*, vol. 24, no.7, pp.2225-2238, 2015. Article (CrossRef Link).

[22] Shen L., Zhang Z., Zhang X., et al., "Fast TU size decision algorithm for HEVC encoders using Bayesian theory detection," *Signal Process:Image Communication*, vol. 32, pp.121-128, 2015. Article (CrossRef Link).

[23] Tohidypour H. R., Pourazad M. T., Nasiopoulos P., "Online learning-based complexity reduction scheme for 3D-HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 10, pp.1870-1883, 2016. Article (CrossRef Link).

[24] Zhu L., Zhang Y., Pan Z., et al., "Binary and Multi-Class Learning Based Low Complexity Optimization for HEVC Encoding," *IEEE Transactions on Broadcasting*, vol. 63, no. 3, pp.547-561, 2017. Article (CrossRef Link).

[25] Kim. H.-S., Park R.-H., "Fast CU Partitioning Algorithm for HEVC Using an Online-Learning-Based Bayesian Decision Rule," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp.130-138, 2016. Article (CrossRef Link).

[26] Muller K., Schwarz H., Marpe D., et al., "3D high-efficiency video coding for multi-view video and depth data," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp.3366-3378, 2013. Article (CrossRef Link).

[27] Tech G., Chen Y., Muller K., et al., "Overview of the multiview and 3D extensions of high efficiency video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp.35-49, 2016. Article (CrossRef Link).

[28] Müller K., Vetro A., "Common test conditions of 3DV core experiments," *JCT3V meeting*, JCT3VG1100, 2014.

[29] Bjontegaard G., "Calculation of Average PSNR Differences between RD-Curves," *in Proc. of Austin: 13th Video Coding Experts Group (VCEG) Meeting*, VCEG-M33, 2001.

Fen Chen received her B.S. degree from Sichuan Normal College, China, in 1996, and M.S. degree from Institute of Optics and Electronics, Chinese Academy of Science, in 1999, and received her Ph.D. degree from Ningbo University in 2015. She is now an associate professor in Faculty of Information Science and Engineering, Ningbo University, China. Her research interests mainly include digital signal processing, image processing and communications, multiview video processing.

Sheng Liu received his B.S. degree in Shenyang University of Chemical Technology, China, in 2014, and M.S. degree from Ningbo University, China, in 2017. His research interests concentrate on image/video compression.

Zongju Peng received his B.S. degree from Sichuan Normal College, China, in 1995, and M.S. degree from Sichuan University, China, in 1998, and received his Ph.D. degree from Institute of Computing Technology, Chinese Academy of Science in 2010. He is now a professor in Faculty of Information Science and Engineering, Ningbo University, China. His research interests mainly include image/video compression, multi-view video coding and video perception.

Qingqing Hu is currently a M.S. candidate with the College of Information Science and Engineering, Ningbo University, China. Her research interests mainly include digital video compression and communications, screen content video coding.

Gangyi Jiang received his M.S. degree from Hangzhou University in 1992, and received his Ph.D. degree from Ajou University, Korea, in 2000. He is now a professor in Faculty of Information Science and Engineering, Ningbo University, China. His research interests mainly include digital video compression and communications, multi-view video coding and image processing.

Mei Yu received her M.S. degree from Hangzhou Institute of Electronics Engineering, China, in 1993, and Ph.D. degree form Ajou University, Korea, in 2000. She is now a professor in Faculty of Information Science and Engineering, Ningbo University, China. Her research interests include image/video coding and video perception.