Journal of Computers Vol. 30 No. 1, 2019, pp. 166-173
doi:10.3966/199115992019023001016

Gray Level Co-occurrence Matrix Based Fast Depth Intra Coding for 3D-HEVC

Jie Liao[1,2], Jing Chen[1,2*], Huan-Qiang Zeng[1,2], Can-Hui Cai[3]

[1] School of Information Science and Engineering, Huaqiao University, Xiamen 361021, China

[2] Xiamen Key Laboratory of Mobile Multimedia Communications, Xiamen 361021, China

[3] College of Engineering, Huaqiao University, Xiamen 361021, China
liaojieahao@163.com, chengjing8005@gmail.com

Received 15 September 2017; Revised 29 November 2017; Accepted 10 January 2018

Abstract. 3D-HEVC is a newly developed video coding standard to efficiently encode the 3D video and free view video, which consists not only the texture but also the depth map sequence for each view. To reduce the computational complexity of depth map coding, a fast algorithm for depth intra coding based on the Gray Level Co-occurrence Matrix (GLCM) is proposed in this paper. By studying the correlation of nonzero number in GLCM with CTU numbers in different partitioning depth, the CU size of intra coding can be prejudged, which means the intra prediction process of CU in other depth can be selectively skipped. Experiments evaluating the proposed algorithm reduce 19.1% encoding time saving with a small synthesized total bitrate loss of 0.0591% in the all-intra mode.

Keywords: 3D-HEVC, depth intra prediction, Gray-level Co-occurrence Matrix

1 Introduction

A three-dimensional (3D) video sequence is composed of two or more videos captured by diverse view cameras collecting the same scene. With additional information compared to the traditional 2D video, 3D video sequence can provide a more live sense to the audience. The more information needs more effective coding strategy of 3D video. Therefore, 3D-HEVC standard was established by the Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) to meet the requirements of 3D video systems and applications [1]. The video format of 3D-HEVC is multi-view video plus depth (MVD) [2], which consists of two sequences, one is the texture; the other is the depth map. However, both texture and depth map sequences follow the quadtree coding structure and prediction mode traversal process in HEVC, which increase the coding complexity. Therefore, it is necessary to optimize the coding complexity.

Algorithms for improving the coding efficiency while maintaining the quality of constructed 3D video is highly demanded. Studies for 3D-HEVC optimization are mainly on the depth map sequence, because of the large amount of flat areas in the depth map. Early determination of intra prediction pattern and the coding unit (CU) segmentation size prejudge are two ways to reduce the computational complexity of video coding.

In the case of reducing the intra-frame candidate mode, the depth modeling modes (DMM) are mainly skipped by designing the preconditions. [3] has classified the PU blocks with vertical and horizontal edges to skip DMM1. In [4], a fast depth mode decision algorithms was proposed to early terminate the unnecessary prediction modes with full rate-distortion cost (RdCost) calculation in 3D-HEVC. [5] proposed a fast mode decision algorithm based on the grayscale similarity and inter-view correlation, which reduce the complexity of depth map coding by skipping the unnecessary mode checking during the mode decision process. Two fast algorithms including the squared Euclidean distance of variances

* Corresponding Author

(SEDV) and probability-based early depth intra mode decision (PBED) are presented in [6] to speed up the most time-consuming intra mode processes in 3D-HEVC depth map coding. In [7], the DMM1 fast pattern selector (DFPS) algorithm was proposed, which includes lightweight and medium weight DMM1 pattern predictors to simplify the process of DMM1. In [8], sum of gradient is presented to determine whether the current block belongs to the flat region so as to skip unnecessary checking of DMMs and smaller partitioning sizes, which save about 21.8% coding time.

In the case of CU size segmentation prejudging, [9] proposed an efficient depth CU prediction strategy by studing the Corner-Point (CP) and the co-located texture of CU, which can reduce the search range of the current depth CU. [10] skipped the split type with little possibility, and several possible prediction modes by using the segmentation depth of space adjacent CU. In [11], the correlation between diverse viewpoints is adopted to determine the CU depth of each nonindependent viewpoint based on the inter-view prediction and the quadtree structure constraint. As a result, the unnecessary process of rate-distortion optimization (RDO) is deceased. [12] proposed two techniques, early termination is performed if the minimum RdCost of test candidate modes is smaller than the threshold computed from full mode search for the fast intra mode decision. And smaller CU will be evaluated if the current CU presents some desired properties for the fast CU size decision. It has reduced the 32% of the encoding time, but there was a big distortion for synthetic viewpoint.

Considering that the depth information can be used to predict the segmentation size of the CU during the depth coding process, a fast depth intra coding algorithm using GLCM [13] is proposed to prejudge the CU size without traversing every possible sizes. The experimental results show that the proposed algorithm can effectively reduce the complexity of intra prediction coding while provide well virtual views rendering quality.

The paper is organized as follows: the segmentation process of CTU intra-coding in 3D-HEVC is introduced in Section 2. In Section 3, the generation process of GLCM is illustrated first, and then the relationship between the CTU separationsize and the nonzero numbers of GLCM is analyzed. The idea of the proposed algorithm and the process is descripted in Section 4. The Section 5 is the experimental results and analysis. Finally, Section 6 comes the conclusion.

2 The Process of Depth Intramode Selection

The same as HEVC, the size of 64×64, as known as LCU, is the maximum coding unit in 3D-HEVC depth map coding. Choosing the best coding size for a CTU requires traversing all 64×64, to 8×8, divisions and selecting the best segmentation mode for this CTU by calculating RdCosts of different CUs function of RdCost.

Fig. 1 lists the process of a CTU divided into multiple CUs by a quadtree structure. During the prediction process, it is necessary to divide the CTU into one CU of size 64×64, four CUs size of 32×32, sixteen CUs size of 16×16, and 64 CUs sizes of 8×8 respectively. Each CU is independently predicted in each split mode and calculate the RdCost for each CU. As a result, a tree structure is established. After that, from the direction of the leaf node to the root node, compares the sum of RdCost of the four child nodes with their parent nodes. If RdCost of the parent node is small, its child nodes are removed; on the contrary, its child nodes are preserved. Until the comparison to the root node, the final tree structure is the best way to split the current CTU.

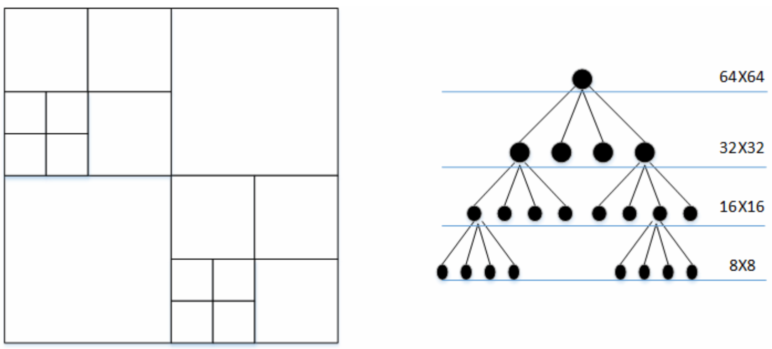


Fig. 1. The selection process of depth coding in CTU

According to the above analysis, it can be noticed that the CU segmentation and the prediction mode determination processes have a high computational complexity. It needs to traverse all sizes of 64×64 to 8×8 segmentations, that is from depth0 to depth 3, to determine the best CU coding depth of a CTU. Since the depth map in 3D-HEVC has a large number of flat areas and some sharp edge information, the probability for the flat area using the size of 64×64 coding block, then comes the probability of using 32×32, 16×16 and 8×8 sizes. If the segmentation size of CTU can be prejudged, the complexity of depth coding can be reduced.

3 Calculation of Gray Level Co-occurrence Matrix in Depth Map Coding

The GLCM is a common method for describing texture using the spatial correlation with image gray, which can reflect the comprehensive information about image grayscale in direction, adjacent interval, and variation amplitude [14]. Therefore, we consider calculating the GLCM of CTU in the depth video to roughly analyze the complexity of texture in 3D-HEVC depth maps.

For an image with gray level n, GLCM is an estimate of the second-order joint probability $p(i, j \mid d, n, \theta)$ of two pixels separated by a distance of d along direction with gray level i and $j (0 \le i, j < n)$ respectively [13]. It is a two-dimensional dependence matrix square with n, formed by calculating how many times a pixel $pair(i, j)$ occurs and divided by the total number of such comparisons made on the image. An example of the matrix is illustrated in Fig. 2. The original gray value of the original image is shown in Fig. 2(a). We get a point $A(x, y)$ and another point $B(x+a, y+b)$ (a and b are the distances from A which are integers), which constitute a pixel $pair(i, j)$. Each element in Fig. 2(b) shows the sum of the occurrence for a pixel $pair(i, j)$. Without loss of generality, let distance a equals to 1 and b equals to 0. The angle equals θ equals to 0° and the gray level n equals to 4. The element (0, 1) in (a) obtains the value 2 in (b) because there are three occurrences where the horizontal adjacent pixels have values of 0 and 1 in (a). Similarly there is only one occurrence of horizontal adjacent pixels with values 2 and 3, so the element (2, 3) in the GLCM obtains the value 1. The whole matrix is created in this way. Therefore, nonzero numbers in GLCM has a great correlation with the CU optimal coding size.

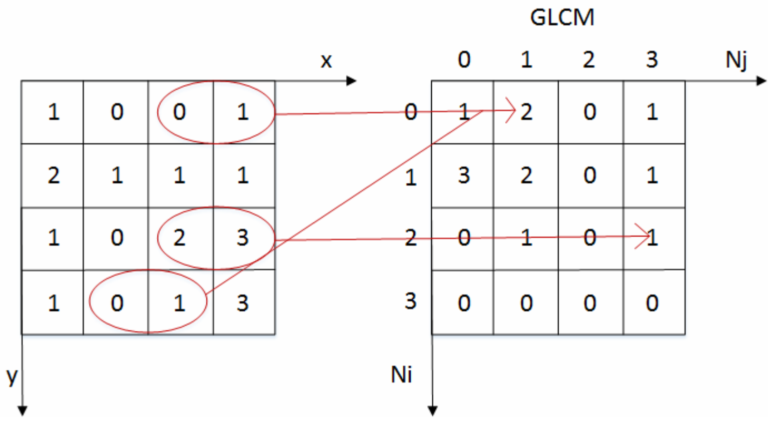


Fig. 2. An example of GLCM generation:
(a) Gray value of the original image; (b) Gray level co-occurrence matrix of (a)

Due to the large number of flat areas in 3D-HEVC, nonzero numbers generated by GLCM is limited. The more nonzero numbers in GLCM, the more detail information in the depth sequence, which leads to the smaller CU size and more time consuming. Experimental results have verified the deduction. The sequences for our experiments are listed in Table 1. Fig. 3 shows one set of experiment results.

Table 1. The parameters of test sequences

Number	Resolution	Sequence	Frames
1	1024×768	Balloons	300
2	1024×768	Kendo	300
3	1024×768	Newspaper1	300
4	1920×1088	GT_Fly	250
5	1920×1088	Poznan_Hall2	200
6	1920×1088	Poznan_Street	250
7	1920×1088	Undo_Dancer	250
8	1920×1088	Shark	300

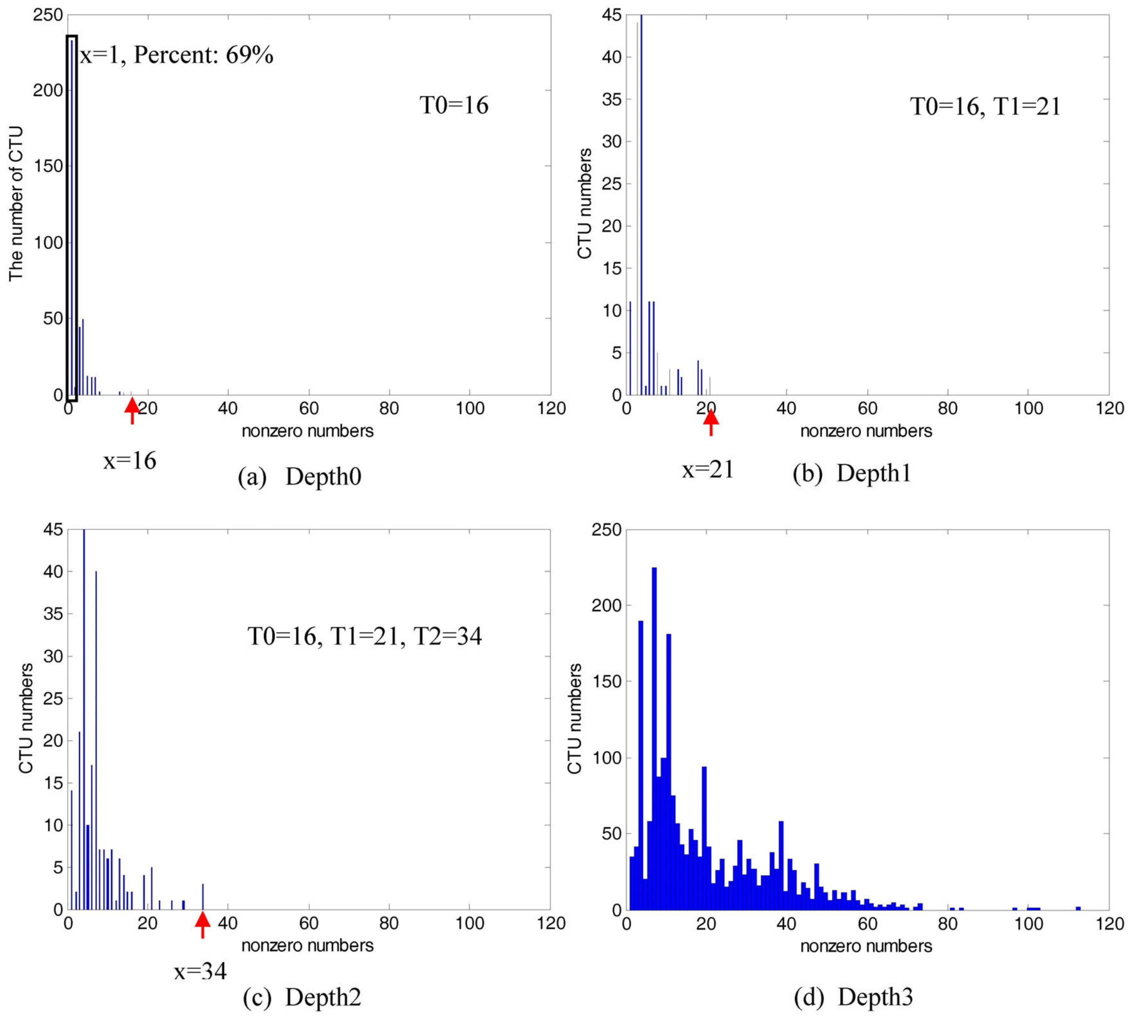


Fig. 3. The statistical results of different GLCM CTU segmentation depth of nonzero numbers for "Newspaper1"

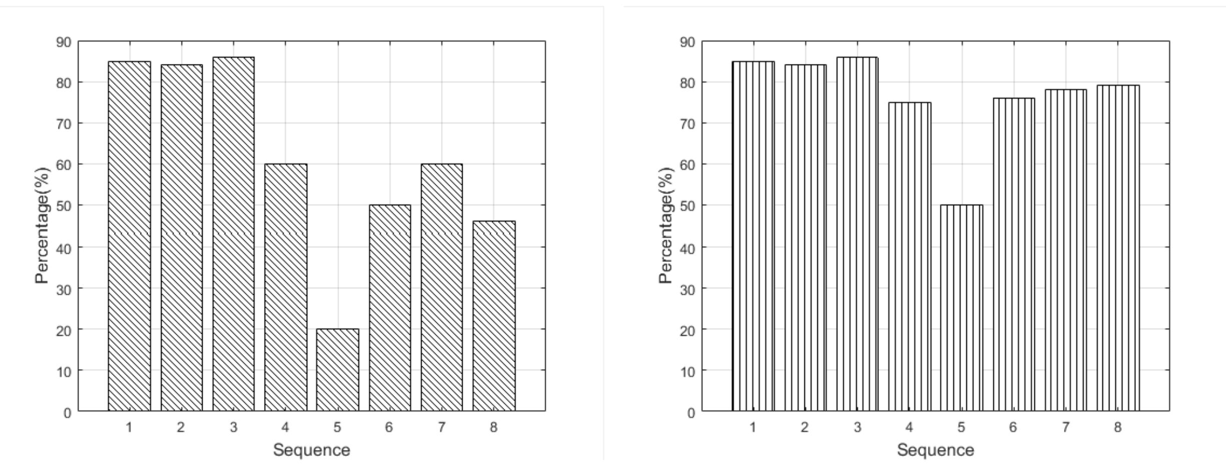
The test sequence is the standard Newspaper1 (1024×768) sequence of 5 frames which is tested in the reference software of 3D-HEVC (HTM-16.0) and tested using the common test condition [15]. The quantization parameters were set to $\{(25, 34), (30, 39), (35, 42) \text{ and } (40, 45)\}$. The first frame did not use the fast algorithm. Let distance a equals to 1 and b equals to 0, The angle equals θ equals to 0°, and the gray level n equals to 5. The GLCM is calculated for the depth map video.

As shown in Fig. 3(a), Fig. 3(b), Fig. 3(c) and Fig. 3(d) corresponds to four different CTU segmentation depths, respectively. The horizontal axis represents nonzero numbers generated at the corresponding split depth. The vertical axis represents the number of each CTU which selects the current split depth as the best split depth. From the statistical results, the percent of CTU is about 70% when nonzero numbers equals to 1, which indicates that a large number of flat areas in the depth map have a very good probability of coding with the best depth 0 in (a). In order to explain the details, we intercepted the distribution of CTU when nonzero numbers equals to 0 to 20 in the right part of (a). As we can see, on the one hand, nonzero numbers that equals to 1 in depth0 is much larger than that of in depth1, depth2 and depth3. This indicates that when nonzero numbers is small in GLCM, the area of CTU tends to be an image smoothing area. It can predict the CU segmentation depth of 0 in advance for this part of the CTU which can reduce a lot of computational complexity. On the other hand, a quite part of nonzero numbers in GLCM do not appear in depth1, depth2 and depth3. When depth3 is the best segmentation depth. So that we can predict in advance that the optimal depth of the current block is 3 and skip the intra prediction of the depth1, depth2 and depth3, including the process of pattern prediction and RdCost, thereby reduce the computational complexity.

According to the GLCM generated by four different CTUs, three suitable thresholds to divide the interval in depth0, depth1, depth2 and depth3 are denoted as T0, T1 and T2, respectively. The value of each threshold is decided by diverse sequence under the experiment. For example, as illustrated in Fig. 3, when depth0 is the optimal split depth in the Newspaper1(1024×768) sequence, the corresponding nonzero maximum value is $T0=16$, the same way to get $T1=21$, $T2=34$, which meets the law of $T0<T1<T2$, and the nonzero numbers can be discriminated in different depths. By this way, the depth of CTU segmentation can be determined in advance.

In order to verify the rationality of the law, the remaining seven sequences were tested under the same experimental conditions. The probability distribution of the law is shown in the Fig. 4. The X-axis indicates the test sequence number, and the Y-axis represents the percentage of correctness which satisfying the threshold distribution law. It found that there is quite a part threshold is not suitable for the law about Poznan_Hall2, Poznan_Street, Undo_Dancer, GT_Fly and Shark (1920×1088). Therefore, the following threshold algorithm is used to satisfy the law for the high-definition sequences.

$$Max\{Depth[x][y]\}-mean[x]>10 \quad \textbf{(1)}$$



(a) Before optimization (b) After optimization

Fig. 4. The probability distribution of the law

As shown in formula (1), x=1, 2, 3, 4, which stands for different optimal depth of segmentation. Different nonzero numbers at different depths are represented by y. mean[x] signifies the mean of nonzero numbers in current depth. After that, the probability distribution satisfying the regularity is shown in Fig. 4, which increases the percentages conform to the law from 44% to 60% for the high-definition sequences.

4 Proposed Algorithm

Based on the above discussion, a fast coding algorithm for 3D-HEVC depth map using GLCM for depth prediction is proposed. The algorithm flow chart is shown in Fig. 5. It calculate the GLCM of CTU in the first frame. Considering the temporal correlation in the video and the subtle motion difference of each frame, we choose to update the threshold every 8 frames. Because the Group of Picture (GOP) in 3D-HEVC is 8, which can guaranteed the smaller coding distortion. After the determination of the threshold, the corresponding GLCM of CTU is calculated for each of the remaining frames, and the CTU segmentation depth is determined in advance according to the distribution of its nonzero numbers. And the process of intra prediction is skipped selectively.

The overall steps of the algorithm are as follows:

(1) Calculate nonzero numbers (N_nonzeros) in each 64×64 CTU for the first frame using the GLCM according to each segmentation depth. Respectively, get the maximum threshold T0, T1, T2 in depth0, depth1, depth2. The thresholds are generated from different for different test sequences, and the threshold is recalculated every 8 frames.

(2) For the next 7 frames, we also calculate the nonzero number of each CTU segmentation depth, and the statistical results are entered into the fast algorithm.

(3) The fast algorithm: If $N\_nonzeros > T2$, then skip the process of intra prediction in depth0, depth1, depth2, except depth3, which eliminates the complexity of a large number of RdCost calculation; If $T1 < N\_nonzeros < T2$, then skip the process of intra prediction in depth0, depth1, except depth2, depth3; If $T0 < N\_nonzeros < T1$, then skip the process of intra prediction in depth0, depth1, depth2, depth3; If $0 < N\_nonzeros < T0$, turn to step (4), otherwise turn to step (5).

(4) If $N\_nonzeros < 1$, the prediction in depth0 is carried out directly.

(5) Start an intra prediction as usual.

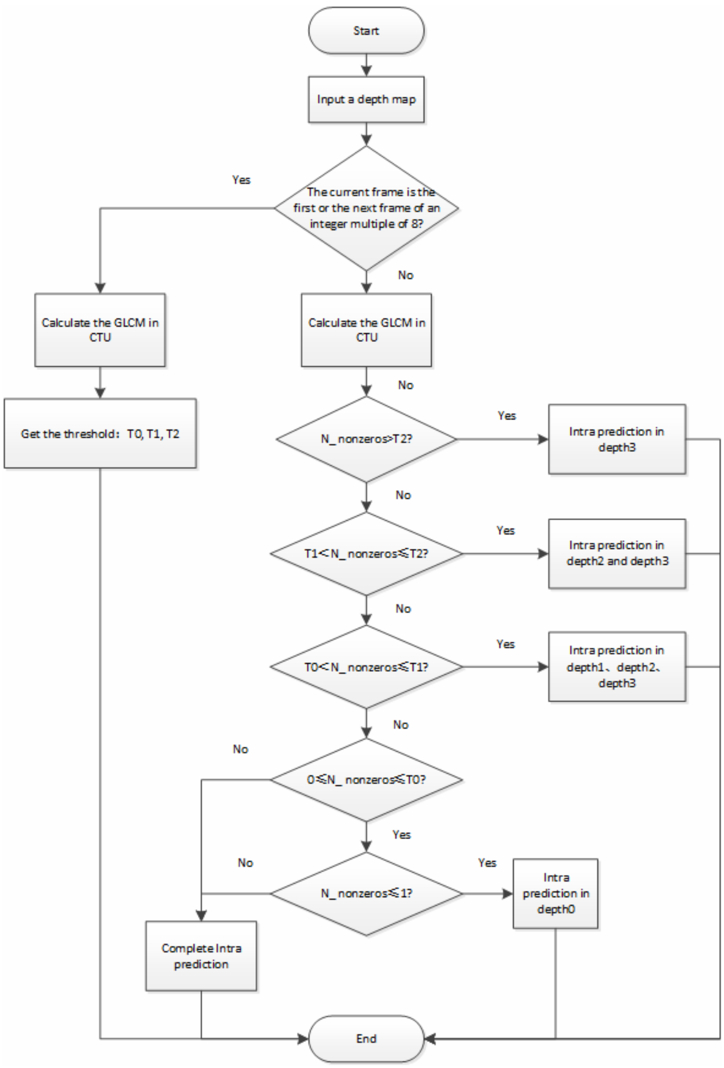


Fig. 5. The probability distribution of the law

5 Experimental Results

In order to verify the feasibility and effectiveness of the algorithm, the test circumstance of the fast algorithm we proposed is implemented as same as before. Respectively, the test sequence is shown in Table 1. We compare the algorithm with the [7]. The algorithms are tested on the same platform HTM-16.0. Table 2 shows the experimental results of the four algorithms for synthesizing the viewpoint $\Delta BDBR$, ΔT , where $\Delta BDBR$ and ΔT are defined as (2) (3).

Table 2. Simulation results of state-of-the-art algorithms and proposed algorithms

Sequence	G. Sanchez's [7]		GLSM	
	$\Delta BDBR$(%)	ΔT(%)	$\Delta BDBR$(%)	ΔT(%)
Kendo	0.1366	-12.1	0.1251	-24.0
Balloons	0.1249	-12.6	0.1395	-21.9
Newspaper1	0.3547	-12.9	0.1879	-18.5
1024×768	**0.2054**	**-12.5**	**0.1508**	**-21.4**
Undo_Dancer	0.0346	-14.0	0.1337	-24.2
GT_Fly	0.0281	-10.5	-0.5556	-4.2
Poznan_Hall2	0.4898	-14.3	0.2811	-31.3
Poznan_Street	0.1540	-17.0	0.1246	-12.7
Shark	0.0488	-13.8	0.0366	-8.0
1920×1088	**0.1511**	**-13.9**	**0.0041**	**-16.1**
Average	**0.1714**	**-13.4**	**0.0591**	**-19.1**

$$\Delta BDBR = \frac{BDBR_{proposed} - BDBR_{ori}}{BDBR_{ori}} \times 100\% . \tag{2}$$

$$\Delta T = \frac{T_{proposed} - T_{ori}}{T_{ori}} \times 100\% . \tag{3}$$

Where $BDBR_{ori}$, represents that the original coding of the average synthesized bitrate, T_{ori} denotes the coding time under the all-intra mode; $BDBR_{proposed}$ stands for the fast algorithm of the average synthesized bitrate and $T_{proposed}$ is the coding time under the all-intra mode.

We can see that both the algorithms tend to accelerate the encoding process without sacrificing the coding efficiency. For all sequences, [7] algorithm achieves a 13.4% reduction in encoding time with BD-rate increasing by 0.1714%. For the sequence of Newspaper1 and Poznan_Hall2, there is a greater distortion in the synthesized views, reach to 0.3547% and 0.4898%. The proposed algorithm of this paper can reduce 19.1%encoding time saving with a small synthesized total bitrate loss of 0.0591%. Especially for the sequence Poznan_Hall2, there are 31.3% time saving, because of a large number of similar areas.

6 Conclusion

A fast algorithm for depth intra coding based on the gray level co-occurrence matrix was presented in this paper. The nonzero numbers in GLCM formed by different CTUs are counted, and different thresholds are observed according to the distribution rule. The CU traversal of different depths is selectively skipped, and reduce the calculation of the intra prediction the complexity. The experimental results show that the algorithm reduces the coding time by 19.1% when the bit rate of synthesis increases by only 0.0591%, which superior to the compared 3D-HEVC depth fast algorithm. Compared with other new 3D-HEVC depth fast algorithm, the proposed has a certain advantages.

Acknowledgements

This work is partially supported by National Natural Science Foundation of China (Grant Nos. 61372107 and 61401167), Natural Science Foundation of Fujian Province (Grant Nos. 2017J05103 and 2016J01308), and Opening Project of State Key Laboratory of Digital Publishing Technology of China (Grant No. FZDP2015-B-001), Research Foundation for Talented Scholars of Huaqiao University (Grant No. 16BS709), and Promotion Program for Young and Middle-aged Teacher in Science and Technology Research of Huaqiao University (Grant No. ZQN-YX403).

References

[1] Y. Chen, G. Tech, K. Wegner, S. Yea, Test model 8 of 3D-HEVC and MV-HEVC. <https://www.researchgate.net/publication/271851757_Test_Model_8_of_3D-HEVC_and_MV-HEVC>, 2014.

[2] K. Muller, P. Merkle, G. Tech, T. Wiegand, 3D video formats and coding methods, in: Proc. IEEE International Conference on Image Processing, 2010.

[3] C.S. Park, Edge-based intramode selection for depth map coding in 3D-HEVC, IEEE Transactions on Image Processing 24(1)(2014) 155-162.

[4] Z.Y. Gu, J.H. Zheng, N. Ling, P. Zhang, Low complexity bi-partition mode selection for 3D video depth intra coding, Display 40(2015) 2-8.

[5] J.J. Lei, J.H. Duan, F. Wu, Ling. N, C.P. Hou, Fast mode decision based on grayscale similarity and inter-view correlation for depth map coding in 3D-HEVC, in: Proc. IEEE Transactions on Circuits & Systems for Video Technology, 2016.

[6] H.B. Zhang, C.H. Fu, Y.L. Chan, S.H. Tsang, W.C. Siu, Probability-based depth intra mode skipping strategy and novel VSO metric for DMM decision in 3D-HEVC, in: Proc. IEEE Transactions on Circuits & Systems for Video Technology, 2016.

[7] G. Sanchez, L. Jordani, L. Agostini, C. Marcon, DFPS: a fast pattern selector for depth modeling mode 1 in three-dimensional high-efficiency video coding standard, in: Proc. IEEE Transactions on Circuits & Systems for Video Technology, 2016.

[8] J. Chen, B.H. Wang, H.Q. Zeng, C.H. Cai, K.-K. Ma, Sum-of-Gradient based fast intra coding in 3D-HEVC for depth map sequence (SOG-FDIC), Journal of Visual Communication & Image Representation 48(C)(2017) 329-339.

[9] H.B. Zhang, C.H. Fu, W.M. Su, R.L. Cheng, W.C. Siu, Fast coding unit decision algorithm for depth intra coding in 3D-HEVC, Journal of Electronics & Information Technology 38(10)(2016) 2523-2530.

[10] L.Q. Shen, Z.Y. Zhang, P. An, Fast CU size decision and mode decision algorithm for HEVC intra coding, IEEE Transactions on Consumer Electronics 59(1)(2013) 207-213.

[11] G.S. Chi, Xin Jin, Q.H. Dai, A quadtree and statistics based fast CU depth decision algorithm for 3D-HEVC, in: Proc. IEEE International Conference on Multimedia and Expo Workshops, 2014.

[12] K.K. Peng, J.C. Chiang, W.N. Lie, Low complexity depth intra coding combining fast intra mode and fast CU size decision in 3D-HEVC, in: Proc. IEEE International Conference on Image Processing, 2016.

[13] Z.Y. Gao, X.M. Yang, J.M. Gong, H. Jin, Research on image complexity description methods, Journal of Image & Graphics 15(1)(2010) 130-135.

[14] J.H. Feng, Y.J. Yang, Study of texture images extraction based on Gray Level Co-occurence Matrix, Beijing Surveying & Mapping 2007(3)(2007) 19-22.

[15] F. Bossen, 3D-HEVC software HTM-16.0, Beijing Surveying & Mapping 2015.