Journal of Electronic Imaging

JElectronicImaging.org

Efficient wedgelet pattern decision for depth modeling modes in three-dimensional high-efficiency video coding

Hong-Bin Zhang
Chang-Hong Fu
Yui-Lam Chan
Sik-Ho Tsang
Wan-Chi Siu
Wei-Min Su

Hong-Bin Zhang, Chang-Hong Fu, Yui-Lam Chan, Sik-Ho Tsang, Wan-Chi Siu, Wei-Min Su, "Efficient wedgelet pattern decision for depth modeling modes in three-dimensional high-efficiency video coding," *J. Electron. Imaging* **25**(3), 033023 (2016), doi: 10.1117/1.JEI.25.3.033023.

Efficient wedgelet pattern decision for depth modeling modes in three-dimensional high-efficiency video coding

Hong-Bin Zhang,[a,b] Chang-Hong Fu,[a,*] Yui-Lam Chan,[b] Sik-Ho Tsang,[b] Wan-Chi Siu,[b] and Wei-Min Su[a]
[a]Nanjing University of Science and Technology, School of Electronic and Optical Engineering, 200 Xiao Ling Wei, Nanjing, Jiangsu 210094, China
[b]The Hong Kong Polytechnic University, Department of Electronic and Information Engineering, Hung Hom, Kowloon, Hong Kong

Abstract. The three-dimensional (3-D) video extension of high-efficiency video coding is an emerging coding standard for multiple-view-plus-depth that allows view synthesis for multiple displays with depth information. In order to avoid mixing between the foreground and background, the depth discontinuities defined at the object boundary should be retained. To solve this issue, a depth intramode, i.e., the depth-modeling mode (DMM), is introduced in 3-D-high-efficiency video coding as an edge predictor. The test model HTM 8.1 includes DMM1 and DMM3. However, the mode-decision strategy of DMM increases the complexity drastically. Therefore, we propose a fast DMM1 decision algorithm that estimates sharp edges by a subregional search method. The optimal wedgelet pattern of DMM1 is then searched only in the most probable region. Additionally, another fast method is raised to skip DMM3 when mismatch occurs between the depth prediction unit (PU) and its colocated texture PU. Simulation results show that the proposed algorithm has slightly better performance in terms of complexity reduction compared with the wedgelet-pattern-reducing algorithm from the literature while better maintaining the coding performance. In addition, the proposed algorithm has a performance similar to that of an existing DMM-skipping algorithm. Moreover, it could be integrated with that category of algorithms for additional time savings. *© 2016 SPIE and IS&T* [DOI: 10.1117/1.JEI.25.3.033023]

Keywords: video coding; multiview video-plus-depth; depth map; wedgelet pattern; depth modeling mode.
Paper 15930 received Jan. 6, 2016; accepted for publication Jun. 2, 2016; published online Jun. 23, 2016.

1 Introduction

In the past decade, the field of video coding has developed significantly after the creation of the H.264 video coding standard. Owing to the limitations in the bandwidth constraints of high-resolution video, a new generation of video coding standards, referred to as high-efficiency video coding (HEVC),[1] was developed by the VCEG and MPEG in January 2013. Since then, it has attracted much attention from industry and academia. A significant amount of research work has been performed on HEVC, including codec optimization and extension establishment. The most groundbreaking research is arguably three-dimensional (3-D) video extension of the high-efficiency video coding standard, which is referred to as 3-D-HEVC.[2,3] It adopts multiple-view-plus-depth (MVD)[4] as the main video format because of its capability to provide depth perception and the flexible manner in which it employs virtual-view generation via depth image-based render technology.[5] Utilizing auxiliary depth information, the advanced MVD format can achieve free-viewpoint video without transmitting all the desired views. In MVD, the depth map is essentially a grayscale image, the level of which describes the distance between the camera and the actual object. Generally, the characteristics of the depth map are quite different from those of the texture image. The major differences are that (i) the depth image has mostly smooth regions delimited by a discontinuity edge; (ii) the distortion of the discontinuity edge can make the disparity unreliable, incurring erroneous pixel mapping and mixing of the background and foreground;[6] and (iii) the depth map is used as geometrical information for synthesis rather than being watched directly by the audience.

Consequently, a series of additional depth-coding techniques has developed, which includes the depth-modeling mode (DMM)[7] and view synthesis optimization (VSO). DMM is defined as a set of new intramodes to preserve the sharp edge and suppress the disturbance caused by the edge loss. It is also a complement of the conventional HEVC intramode. In DMM, the current prediction unit (PU) can be divided into two nonrectangular segments by a wedgelet[8] or contour method. Since each segment has its own prediction, the flexibility partition can give a fine-grain edge description. As a result, DMM can save ~5% of the bitrate of the entire transmitted bitstream while maintaining the visual quality of the synthesized view. Simultaneously, DMM introduces a huge number of mode candidates. Additionally, the time-consuming VSO[9,10] is a new rate-distortion (RD) metric to determine the best DMM partition, since the depth quality cannot guarantee the synthesized view quality. In order to determine the best DMM partition, a huge number of mode candidates have to be checked by evaluating the complicated VSO. The complexity of depth intracoding increases drastically. Therefore, a fast algorithm is desired to preserve the sharp edge with relatively low computational complexity.

In light of the above discussion, much research work has been proposed to reduce the complexity caused by the DMM decision, which can be classified into two categories: skipping the DMM decision and reducing the number of DMM candidates. The algorithm of Merkle et al.[11,12] seeks the suboptimal wedgelet pattern in the double-pixel domain following

*Address all correspondence to: Chang-Hong Fu, E-mail: enchfu@njust.edu.cn

the refinement process. It achieves a time reduction of more than 60% with a negligible loss of coding performance. The algorithm of Zhang et al.[13] finds the direction of the current PU from the rough-mode decision (RMD) and omits the wedgelet patterns from the DMM decision if they have a different direction.

Another group of studies has been proposed to skip the DMM decision entirely if the current PU is labeled as smooth. Gu et al.[14] proposed skipping the DMM process if the planar mode in the RMD has the minimum rough RD cost. It can provide a time reduction of ~30% for depth intracoding, with the bitrate increasing by 0.49%. Zhang et al.[15] made full use of the coding information from the spatial neighboring depth-map treeblock and the colocated texture to predict the depth intramode treeblock and terminate its mode decision process early. Park et al.[16] rearranged the DMM decision process and minimized the number of calculations for DMM distortion by an RD-cost method.

The complexity analysis of the 3-D-HEVC test model[17] shows that the DMM decision still occupies ~40% of the encoding time, even when algorithms[11,12,14] are adopted. Moreover, the depth intracoding consumes more time than texture intra- or intercoding. Therefore, expediting the DMM decision is one of the important tasks for 3-D video applications.

All the previous algorithms have shortcomings. The algorithms in Refs. 14–16 attempt to determine whether the current PU is smooth in order to skip all the DMMs for these smooth regions. If the current PU makes it worth applying the DMMs, as determined by Refs. 14–16, these algorithms calculate all the wedgelet-pattern candidates. In the other category, the algorithms in Refs. 11–13 narrow the scope of the wedgelet patterns that need to be searched. Although these algorithms reduce the number of wedgelet patterns in a different way, the number of wedgelet-pattern candidates is still constant for each PU in Refs. 11 and 12. However, the optimal wedgelet pattern is highly related to the distribution of sharp edges. Clearly, a content-based algorithm is required that can reduce the number of wedgelet patterns adaptively according to the distribution of sharp edges.

In our previous work,[18] a content-dependent approach was proposed for DMM1. Regions corresponding to a subset of wedgelet patterns are predefined. The region that includes a sharp edge is determined early on by comparing the regional variance with a threshold. Finally, the wedgelet pattern with the minimum distortion is sought only inside the corresponding subset. In this paper, we provide more experimental evidence for the high complexity associated with the DMM decision, give a detailed discussion of the threshold determination, and analyze the influence on performance of the different thresholds. In addition, another fast algorithm is proposed for DMM3, which is motivated by the fact that there is no need to search the wedgelet pattern when there is a structural mismatch between the depth block and its colocated texture block. Meanwhile, the compatibility of the proposed combined algorithm is also shown. It is compared and integrated with the DMM-skipping algorithms.[15,16]

It is noted that although DMM3 has been removed from the later version of HTM, we believe that it is still a versatile algorithm that is worth investigating for possible further use. In addition, the proposed checking similarity between texture and depth can be extended to many other research areas, e.g., mode inheriting[19] or probability updating in entropy coding. This is also a reason for us to select an older version of HTM.

The rest of this paper is organized as follows. Section 2 reviews DMMs and the depth intramode decision in brief and analyzes the complexity of the default wedgelet-pattern decision. The preparatory work, including analysis of the characteristics of PUs coded by DMM1 and the mismatch between the depth and its corresponding texture, is described in Sec. 3.1. Then, the proposed wedgelet-pattern-decision algorithms are represented for DMM1 and DMM3 in Secs. 3.2 and 3.3, respectively. Simulation results and conclusions are given in Secs. 4 and 5, respectively.

2 Depth-Modeling Modes

There are up to 35 conventional intramodes, of which 33 are directional modes that present the texture information, while the DC and planar modes describe smooth regions, as shown in Fig. 1(a). Angular(N) modes are the directional intramodes indexed by 2 to 34. In addition, four DMMs are employed in HTM 8.1 to retain the sharp edges of the depth map, including DMM1, DMM3, and DMM4, and region boundary chain (RBC) coding. The DMMs can divide each PU into two nonrectangular segments, and only one pixel value is used to represent each segment. Coding efficiency is improved by the DMMs. At the same time, the

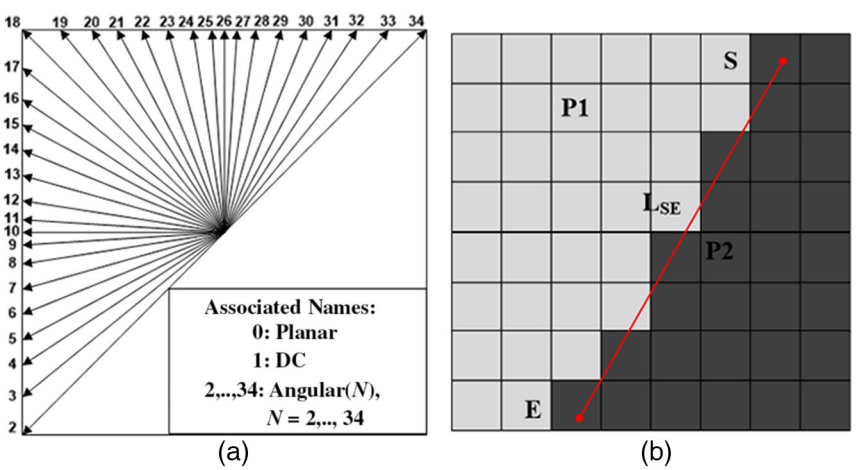

Fig. 1 Depth intramodes: (a) 35 regular intra modes in HEVC and (b) a wedgelet partition in DMM1.[18]

Table 1 Number of wedgelet patterns for DMM1 in full algorithm and fast algorithm.[11]

PU size	Full	Method[11]
4×4	86	66
8×8	766	318
16×16	1350	346
32×32	1503	376

computational complexity increases dramatically due to the numerous wedgelet-pattern candidates. Since DMM4 and RBC have much lower complexity than DMM1 and DMM3, we concentrate on the wedgelet-pattern decision for DMM1 and DMM3 in this paper to accelerate depth intracoding.

2.1 *Wedgelet Partition*

As mentioned above, when one PU is coded by DMMs, it is divided into two regions (P1 and P2) by a straight line L_{SE}, as shown in Fig. 1(b). A binary matrix, called a wedgelet pattern, is used to store the partition information. Each sample in the binary matrix represents whether it belongs to region P1 or P2. The same set of matrices is indexed and stored in both the encoder and decoder. The optimal wedgelet pattern is determined by calculating the distortion of all wedgelet-pattern candidates in the encoder. The corresponding index is then written explicitly into the bitstream. There are two ways to generate the optimal partition information or index of a wedgelet pattern, namely explicit wedgelet signaling (DMM1) and intercomponent prediction of wedgelet partitions (DMM3).

2.2 *Explicit Wedgelet Signaling*

A variety of wedgelet patterns are shown in this section to demonstrate the computational complexity in DMM1. In small PUs of sizes 4×4 and 8×8, the resolutions of the start and end points [S and E shown in Fig. 1(b)] are set to a half-pixel domain for a more precise partition. The half-pixel resolution results in more partition candidates. On the contrary, in large PUs of sizes 16×16 and 32×32, full-pixel and double-pixel resolutions are used for complexity control. However, the number of candidates is more than 1000 for the larger PUs shown in Table 1. Although the number of patterns has been reduced significantly by the two-stage coarse-to-fine method[11] adopted in HTM8.1, DMM1 is still time intensive due to the large number of remaining candidates to be searched in each PU.

2.3 *Intercomponent Prediction of Wedgelet Partitions*

In the early stage of 3-D-HEVC,[20] the best wedgelet pattern of DMM3 was calculated by using the colocated texture

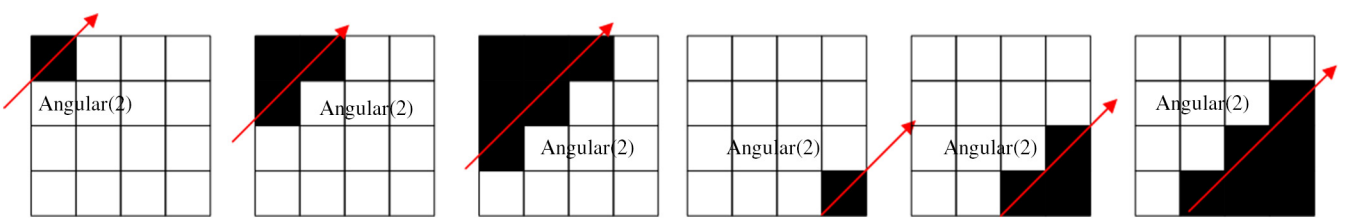

Fig. 2 Example of wedgelet-pattern subset in DMM3 when the intramode of texture PU is Angular(2).

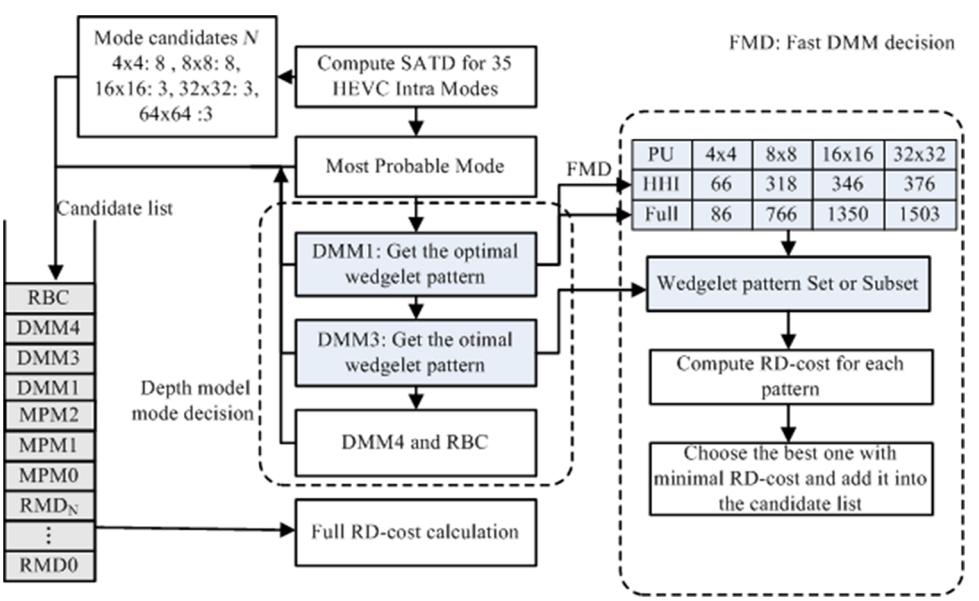

Fig. 3 Flowchart of depth intramode decision. FMD is an option for selecting the fast wedgelet-pattern decision of DMM1 in Ref. 11.

Table 2 Number of wedgelet patterns in each subset corresponding to Angular(i) from 2 to 34 in DMM3.

PU	Number of wedgelet pattern candidates in subsets corresponding to Angular(i) (I from 2 to 34) for different sizes of PU																																
Angular(i)	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	1
4×4	10	13	9	7	8	4	4	11	11	11	4	4	9	8	9	8	7	9	9	7	8	4	4	11	11	11	4	4	9	8	9	4	1	23
8×8	46	80	87	79	79	79	61	58	35	58	62	78	81	83	86	76	65	76	84	79	79	82	61	58	35	58	62	75	81	83	89	60	23	39
16×16	89	134	117	119	123	137	120	142	119	142	120	137	123	119	117	130	124	130	117	119	123	137	120	142	119	142	120	137	123	119	117	84	39	45
32×32	104	158	135	137	138	155	135	155	128	152	130	149	133	132	129	146	138	146	129	131	133	149	130	152	128	155	135	155	138	137	135	99	45	

signal as a reference. Hence, it is not necessary to transmit partition information (wedgelet pattern index) that can be calculated by the decoder. However, DMM3 results in high complexity at both the encoder and decoder sides. In order to reduce the heavy burden on the decoder, an algorithm referred to as the restricted signaling and intercomponent prediction of wedgelet partition was proposed to speed up the process of DMM3.[17] In a later version of DMM3 in HTM8.1, the wedgelet-pattern subset corresponding to the colocated texture was used instead of the full set in DMM1. Figure 2 shows an example in which the intramode of the corresponding 4×4 texture PU is Angular(2). The optimal wedgelet pattern of DMM3 will also be decided using the distortion, with the original depth PU as a reference. Consequently, the complexity at both the decoder and encoder can be reduced significantly. However, the final optimal wedgelet pattern cannot work well all the time, since it is dependent on the coherence of the direction between the depth PU and its colocated texture PU. The mismatch case between the depth PU and its colocated texture PU is investigated in Sec. 3 to reduce the number of candidates further in the wedgelet-pattern subset for DMM3.

2.4 *Depth Intramode Decision*

There are four procedures for the depth intramode decision in 3-D-HEVC, as shown in Fig. 3. First, RMD is performed to choose N intraprediction modes (from RMD_0 to RMD_N) using the sum of the absolute transform difference. Rough modes are added to the candidate list. Second, the three most probable modes (MPMs) are appended as candidates; these are predicted from the left and upper neighboring PUs. Third, the optimal wedgelet patterns of both DMM1 and DMM3 are searched and added to the candidate list. Finally, the full complexity RD-cost is calculated for all candidates in the list, and the candidate with the minimum RD-cost is selected as the final optimal intramode.

In the above procedures, DMM1 and DMM3 have numerous wedgelet-pattern candidates. For DMM1, the numbers of wedgelet patterns for different sizes of PU in the full and fast search algorithms[11] are shown in Table 1. Although the fast algorithm[11] is integrated into the test model, there are more than 300 remaining candidates for most sizes of PU, as illustrated in the rightmost column of Table 1. In addition, Table 2 shows that there are at least 100 wedgelet patterns for DMM3 in most of the wedgelet-pattern subsets. Therefore, the selection of the optimal wedgelet pattern is still the most time-consuming process in the depth intramode decision.

3 Proposed Fast Wedgelet-Pattern Algorithm

In this section, we investigate the characteristics of the PUs coded by DMM1 and the mismatch between the depth PU and its colocated texture PU. The proposed algorithms for wedgelet-pattern decisions in DMM1 and DMM3 are then presented based on the analysis.

3.1 *Observation and Analysis*

As shown in Fig. 4, there are six orientations of wedgelet pattern in DMM1. The numbers of wedgelet patterns in each orientation are shown in Tables 3 and 4 for the full and fast search algorithms, respectively, in Ref. 11. It is clear that the wedgelet patterns are distributed evenly in the six orientations. If we could determine the orientation of the optimal

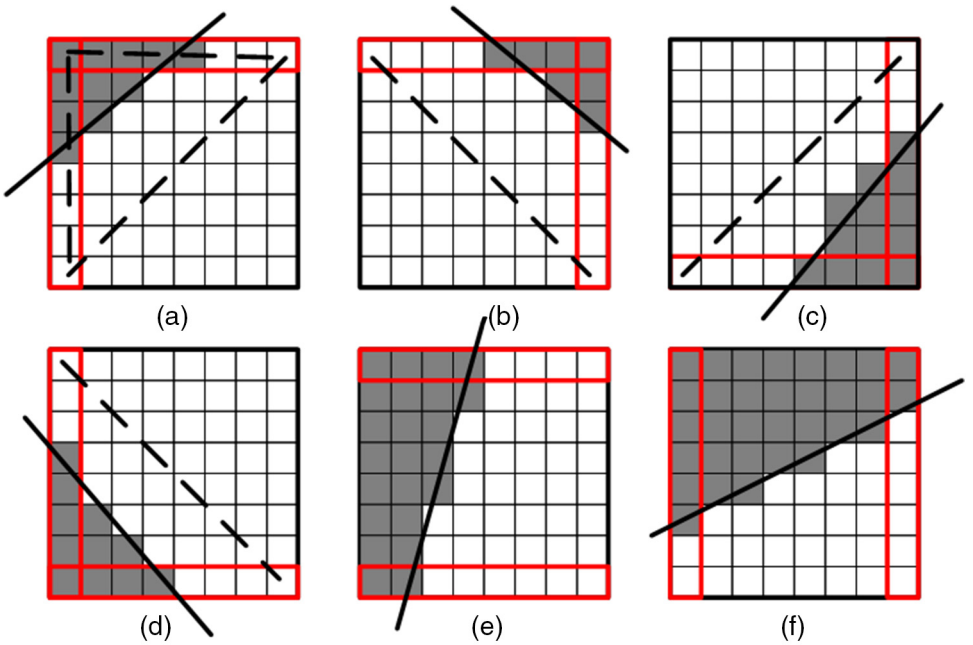

Fig. 4 Six orientations of wedgelet patterns,[18] e.g., in (a), the orientation 0 is from top to left and the partition line is located in the upper-left region in the PU. The rectangles on boundaries provide the range of start and end points for each orientation. (b) top to right, (c) bottom to right, (d) bottom to left, (e) top to bottom, and (f) left to right.

wedgelet pattern early on, a substantial fractional time reduction of up to 5/6 could be achieved. This motivates us to design a wedgelet-pattern-selection method for DMM1 based on an early decision of orientation. Since the pixel values segmented by the sharp edge usually change significantly in the depth map, the variance is an important criterion to estimate the location of the partition line. For the examples shown in Fig. 4(a), the optimal partition line is located in the upper-left region of the PU. The variance of the pixel values in that region is much larger than that of the bottom-right region. Therefore, we can predict the orientation of the partition line by evaluating the partial smoothness of the PU.

Since the optimal wedgelet pattern of DMM3 is searched for in the subset of the directional intramode corresponding to the colocated texture, the performance of DMM3 depends on the structural consistency between the depth map and its colocated texture. However, a mismatch commonly exists between them. There are two categories of mismatch between the texture and depth map, as shown in Fig. 5. First, the texture video has a stronger texture inside objects. On the contrary, the depth map has only gradual changes in the corresponding regions. This phenomenon is shown more or less by the embedded figures bounded by the upper-right rectangles in Figs. 5(a) and 5(b). The texture has a clear direction, related diagonally to Angular(2); the depth map is much smoother, with horizontal [Angular(10)], vertical [Angular(26)], and opposite [Angular(34)] diagonal directions. Another case of mismatch is shown by the lower-left rectangles in Figs. 5(a) and 5(b). This type of misalignment between the depth map and texture is common at the edge regions, more details of which are described in Ref. 21. In both cases, the colocated texture blocks have distinctly different directional information compared with the depth map. Hence, the irrelevant wedgelet-pattern subset corresponding to the colocated texture is decided for DMM3. In this way, the possibility of adopting the optimal DMM3 to be the final optimal intramode is relatively low after the full RD-cost calculation.

3.2 *Proposed Wedgelet-Pattern Decision for DMM1*

In order to investigate the partial smoothness of the PU, we classify most of the wedgelet patterns into four classes of what is known as a partition-line candidate subset (PCS): PCS_{D45}, PCS_{D135}, PCS_{Ver}, and PCS_{Hor}. Each class contains

Table 3 Number of wedgelet patterns in each orientation for full search.

PU\Ori	0	1	2	3	4	5
4 × 4	24	23	15	14	5	5
8 × 8	146	142	131	133	107	107
16 × 16	256	255	251	250	169	169
32 × 32	256	256	255	256	240	240

Table 4 Number of wedgelet patterns in each orientation for fast method.[11]

PU\Ori.	0	1	2	3	4	5
4 × 4	16	15	11	10	3	3
8 × 8	64	63	59	58	33	33
16 × 16	64	64	63	63	42	42
32 × 32	64	64	64	64	56	56

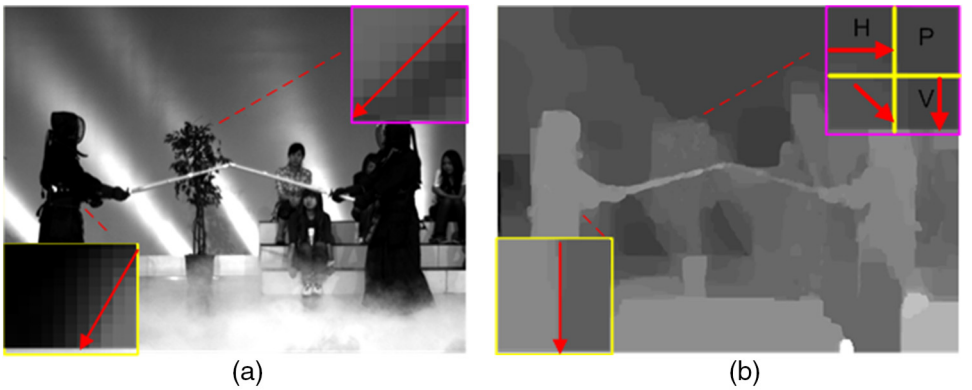

Fig. 5 Illustration of the mismatch between depth map and its corresponding texture: (a) luma component of texture image and (b) depth map.

two subsets, whose partition-line candidates are covered by a pair of complementary regions A and B, as shown in Fig. 6. For example, the symbol $PCS_{D45(X)}$, where X can be either A or B, stands for the subset containing all the partition-line candidates covered by the gray or white regions in Fig. 6(a). In addition, all the wedgelet patterns that do not belong to any of the four predefined classes in Fig. 6 are added into a remaining subset PCS_{NR} (PCS corresponding to nonregions). Also, $PCS_{(MP)}$ is defined as the most probable set of PCS. In the proposed method, only the patterns in $PCS_{(MP)}$ are traversed for each PU. Our task is then to find a small $PCS_{(MP)}$ that includes fewer wedgelet patterns. Based on the analysis in the previous section, the proposed fast wedgelet-pattern decision of DMM1 is described as follows, and is also illustrated in Ref. 18.

First, the current class is initialized as PCS_{D45}. The partial smoothness of the PU is evaluated by calculating the variance of each region X in the current class PCS_{D45}:

$$\mathrm{Var}_X = \frac{1}{N_X}\sum [p(x,y) - \mu_X]^2, \tag{1}$$

where $p(x,y)$ is the value of the depth pixel, N_X is the number of pixels in region X, μ_X is the average value of pixels in region X of the current class, and X is either A or B. Four typical cases of Var_A and Var_B in the current class are described as follows.

Case 1: $\mathrm{Var}_A \le T$ and $\mathrm{Var}_B \le T$. This case indicates that all the pixels in each region are relatively similar. In this case, we know that there are no sharp edges in region A or B. It is assumed that the partition line L_i itself in Fig. 6 is the sharp edge in the current PU. Therefore, the corresponding wedgelet pattern in Fig. 6 is included in the $PCS_{(MP)}$. Due to there being only one candidate in $PCS_{(MP)}$, the distortion calculation is skipped and the proposed algorithm for the wedgelet-pattern decision is terminated.

Case 2: $\mathrm{Var}_A > T$ and $\mathrm{Var}_B \le T$. If case 1 is not satisfied, case 2 is considered. If this condition is true, the current PU belongs to case 2. In this case, the PU is partially smooth in region B, which indicates that there are no sharp edges in region B. In contrast, Var_A is greater than T, which means that the pixel values in region A change markedly. Therefore, it can be inferred that the sharp edge would most likely be in region A. Hence, the best wedgelet pattern can probably be found in the subset of region A in this case. The $PCS_{(MP)}$ is set to the PCS of region A ($PCS_{D45(A)}$).

Case 3: $\mathrm{Var}_A \le T$ and $\mathrm{Var}_B > T$. If both of cases 1 and 2 are not satisfied, case 3 is considered. This case is similar to case 2. In this case, the sharp edge is considered to be inside region B due to the fact that Var_B is greater than T; meanwhile, Var_A is less than T. Therefore, the $PCS_{(MP)}$ is set to the PCS of region B ($PCS_{D45(B)}$).

Case 4: $\mathrm{Var}_A > T$ and $\mathrm{Var}_B > T$. If all the abovementioned cases are not fulfilled, this case will be considered. This case indicates that the sharp edge goes through both regions A and B. As such, we cannot decide which region covers the sharp edge by the current class (PCS_{D45}).

We now repeat the above steps for the three remaining classes (PCS_{D135}, PCS_{Ver}, and PCS_{Hor}) until $PCS_{(MP)}$ is

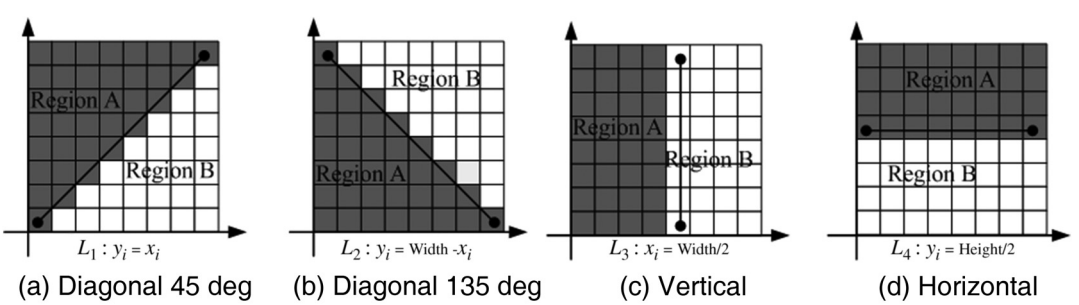

Fig. 6 Definition of four classes.[18] Each class has a pair of regions A and B, and its corresponding partition line.

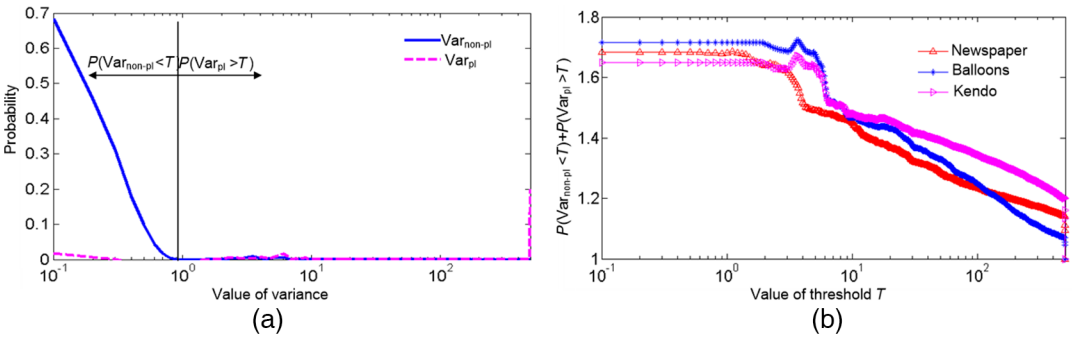

Fig. 7 Decision threshold T: (a) probability distribution of variances and (b) distribution of C_T.

obtained. If $PCS_{(MP)}$ is still empty after all classes have been considered, it is set to PCS_{NR}. This contains all the remaining wedgelet patterns that are not covered by the four classes. In PCS_{NR}, the partition line goes through both regions A and B in four classes at the same time. It is also noted that the order of the four classes has almost no influence on the performance of the scheme. Therefore, the results will not be affected by which class is chosen for the initialization.

After checking all four classes, the most probable wedgelet-pattern subset is obtained. The distortion is computed for all the wedgelet patterns in the subset $PCS_{(MP)}$ to find the optimal wedgelet pattern of DMM1. As the threshold T is the only important criterion in the classification of the PUs into four cases, it is selected carefully by observing the statistics as follows. First, the original reference software encodes sequences with an exhaustive search of DMM1, resulting in the optimal location of the partition lines. Each PU is then segmented into two equal regions, as shown in Fig. 6, according to the optimal partition line. The region containing the partition line that is found by the exhaustive search is called R_{pl}, while the complementary one is marked as $R_{non\text{-}pl}$. The variance of R_{pl} is recorded as Var_{pl}, while the variance of $R_{non\text{-}pl}$ is marked as $Var_{non\text{-}pl}$. The probability distributions of Var_{pl} and $Var_{non\text{-}pl}$ are shown in Fig. 7(a). It can be seen that $Var_{non\text{-}pl}$ is extremely zero-biased, and that 95% of the distribution is less than 1. This is another way of saying that most of the regions without partition lines are constant or contain only slight changes. In contrast, Var_{pl} is evenly distributed in a long range from 0.1 to 100, with marginally higher probabilities at larger values. The peak of Var_{pl} at the right bound of Fig. 7(a) is only for simplicity of display. All probabilities exceeding the maximum value on the horizontal axis are plotted together at that point. As shown by the real examples extracted from the *Balloons* sequence in Fig. 8, the difference between the ends of the partition line varies significantly. For example, although both of the regions marked by red rectangles are classified as R_{pl}, the value of Var_{pl} is 42.3916 in the upper-right region of Fig. 8(a), while it is only 0.4784 in the upper-left region of Fig. 8(b). This phenomenon results in a somewhat different distribution of Var_{pl} from $Var_{non\text{-}pl}$.

Fortunately, although Var_{pl} has no obviously biased distribution, the mean value is relatively large and the main distribution is far from that of $Var_{non\text{-}pl}$. It also demonstrates that it is promising to separate the two different types of regions (R_{pl} and $R_{non\text{-}pl}$) by variance in the proposed method. In order to find the threshold T for the proper classification of regions, a probability function is used as the criterion

$$C_T = P(\mathrm{Var}_{\text{non-pl}} < T) + P(\mathrm{Var}_{\text{pl}} > T), \tag{2}$$

where $P(\mathrm{Var}_{\text{non-pl}} < T)$ is the percentage of $R_{non\text{-}pl}$ with a variance less than the threshold T, and $P(\mathrm{Var}_{\text{pl}} > T)$ represents the percentage of R_{pl} with a variance greater than T. Since Var_{pl} is distributed evenly over a relatively wide range, as mentioned above, it is difficult to find the threshold T by using typical clustering schemes. We try to determine C_T by observing the probability distributions of $Var_{non\text{-}pl}$ and Var_{pl}. Since $Var_{non\text{-}pl}$ is extremely zero-biased, it is reasonable to classify most of the regions with variance less than T as $R_{non\text{-}pl}$. The value of $P(\mathrm{Var}_{\text{non-pl}} < T)$ is used to apply that condition. On the contrary, most of the regions R_{pl} should be separated from $R_{non\text{-}pl}$ with variance greater than T. As a result, $P(\mathrm{Var}_{\text{pl}} > T)$ is appended as another constraint. By maximizing this function, an approximate threshold is selected to separate these two types of region. The trend of C_T is similar in all tested sequences, three of which are plotted in Fig. 7(b) for illustration. Extensive simulations based on different sequences are then performed to refine the threshold T. The influence of different values of T on the performance of the proposed algorithm will also be shown later in the simulation results.

The proposed wedgelet-pattern decision is illustrated by the pseudo-code Algorithm 1. By considering the information of the pixel values for a given PU, this algorithm can decide the location of the partition line roughly by a simple but efficient variance calculation. As a result, the number of candidate wedgelet patterns is reduced significantly. The number of wedgelet patterns in the nine partition-line-candidate subsets is shown in Table 5. It can easily be found that the number of wedgelet patterns in the proposed algorithm is

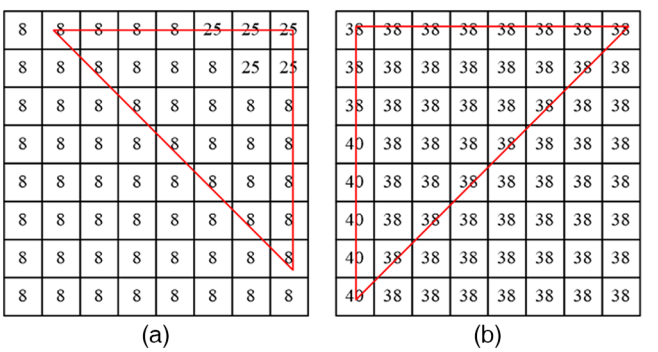

Fig. 8 Examples of R_{pl} with different levels of variance: (a) large variance and (b) small variance.

Algorithm 1 Wedgelet pattern decision for DMM1.[18]

Input: Threshold T and The current PU (PU_{curr})

for PCS_{curr}: PCS_{D45}, PCS_{D135}, PCS_{Ver}, PCS_{Hor} **do**

if $Var_A \leq T$ &&$Var_B \leq T$ then

$PCS_{(MP)}$ = Wedgelet pattern in Fig. 6 and Terminate algorithm

else if $Var_A > T$ &&$Var_B \leq T$

$PCS_{(MP)} = PCS_{cur(A)}$

else if $Var_A \leq T$ &&$Var_B > T$

$PCS_{(MP)} = PCS_{cur(B)}$

else

Go to the next class

end if

if $PCS_{(MP)}$ == Null then

$PCS_{(MP)} = PCS_{NR}$

end if

end for

Calculate the distortion for each wedgelet pattern in $PCS_{(MP)}$

Output: The minimum distortion wedgelet pattern

only approximately one quarter of that in method.[11] In the worst case, the proposed algorithm can reduce 37 patterns for 4×4, 233 for 8×8, 258 for 16×16, and 274 for 32×32 PUs, compared with the method in Ref. 11.

3.3 *Proposed Wedgelet-Pattern Decision for DMM3*

As discussed in Sec. 3.1, the mismatch between depth and texture occurs in both smooth and edge regions, where DMM3 should be skipped. Hence, our main task in achieving a fast DMM3 decision is to find the mismatch positions between the current depth PU and its colocated texture PU. Intuitively, the comparison of the intramodes of the depth PU and the colocated texture PU can be used to detect the mismatch. During the mode decision, the intramode of the current depth PU is unavailable. However, the MPMs estimated from the neighboring PUs can predict the directional information of the current PU, which usually has a similar direction and angular mode to those of the current depth PU. Therefore, the comparison between the MPMs from neighboring depth PUs and the directional mode of the colocated texture PU is employed as the criterion to find the mismatch. In order to ensure that the MPMs can make sufficiently good predictions, the statistics of the correlation between the final optimal directional mode and the MPMs estimated from the neighboring PUs are examined. As shown in Fig. 9, the optimal angular mode of the current depth PU is included in its MPMs for between 50% and 90% of the PUs for various sequences. The percentage is not as remarkable for some other sequences. In order to make a better prediction, we consider more of the neighboring angular modes as MPMs. For example, assume that one MPM is Angular(2), and that the four neighboring angular modes {Angular(33), Angular (34), Angular(3), and Angular(4)} are also regarded as MPMs. Let us define the symbol MPM_{SetNei} as the extended subset of mode candidates including MPM neighbors. By extending MPM to MPM_{SetNei}, we find that the percentage of PUs where the optimal angular mode for the current PU is included in its MPM_{SetNei} exceeds 90%. In other words, MPM_{SetNei} can represent the direction of the current depth PU better than MPM. Hence, if the angular mode of the colocated texture is not included in MPM_{SetNei}, it is highly probable that the directional information of the depth map and texture is different. Here, the corresponding PU is labeled as the mismatch case.

To verify that it is unnecessary to perform DMM3 in the mismatch case, some experiments are performed to classify all PUs into mismatch and match cases as illustrated above. Table 6 shows the average statistics. Among all the PUs employing DMM3 as the final optimal mode, only 4.41% of them are marked as mismatch cases. On the other hand, for all mismatched PUs, only 2.28% of them select DMM3 as the optimal mode.

Based on the analysis of the mismatch between the depth map and texture, the proposed wedgelet-pattern decision of DMM3 is described as follows.

First, the MPMs of the current depth PU are estimated from the neighboring PUs. If all the MPMs belong to the planar or DC modes (smooth representation), the DMM3 is skipped. Otherwise, MPMs are extended to MPM_{SetNei} and the existence of mismatch is decided by whether the

Table 5 Number of wedgelet patterns in each PCS for different sizes of PU in the proposed algorithm.[18]

PU\PCS	PCS_{D45}		PCS_{D135}		PCS_{Hor}		PCS_{Ver}		PCS_{NR}
Region	A	B	A	B	A	B	A	B	–
4×4	21	8	12	5	1	5	1	2	1
8×8	77	48	59	41	3	22	6	15	37
16×16	80	63	55	55	1	15	1	15	54
32×32	94	64	56	56	1	16	1	16	66

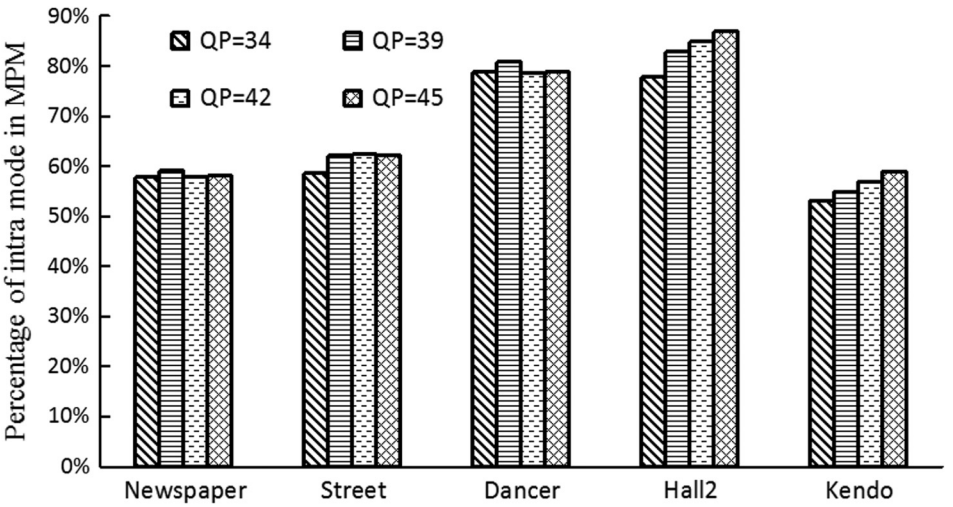

Fig. 9 Percentage of PUs, where the optimal directional intramode is found in MPMs.

Table 6 Distribution of DMM3 and non-DMM3 for mismatch cases.

PUs with DMM3 as the optimal mode		Mismatch PUs	
Match (%)	Mismatch (%)	DMM3 (%)	Non-DMM3 (%)
95.59	4.41	2.28	97.72

directional mode of the colocated texture mode is covered by $\mathrm{MPM}_{\mathrm{SetNei}}$. If the colocated texture mode is included in $\mathrm{MPM}_{\mathrm{SetNei}}$, the optimal wedgelet pattern will be searched in the subset corresponding to the Angular(i) mode of texture. The full RD calculation will then be carried out for the selected optimal wedgelet pattern of DMM3. Otherwise, DMM3 can be skipped safely to reduce the complexity for mismatch cases.

The flowchart of the proposed algorithm of DMM3 is summarized in the pseudocode Algorithm 2. The distortion calculations of ~100 unnecessary wedgelet patterns are skipped in most mismatch cases, as well as the full RD calculation in DMM3 also being avoided.

Algorithm 2 Wedgelet pattern decision for DMM3.

Get the intramode of colocated texture $Mode_{Tex}$

if $Mode_{Tex}$ is *Planar or DC*

skip the DMM3 and Terminate algorithm

else

Expand the MPMs from Neighboring CUs to be MPM_{SetNei}

if $Mode_{\mathrm{Tex}}$ is not included in MPM_{SetNei}

Skip the DMM3 and Terminate algorithm

else

Search the optimal wedgelet pattern for DMM3

end if

end if

4 Simulation Results and Discussions

The 3-D-HEVC reference software HTM-8.1[22] was used to evaluate the proposed wedgelet-pattern-decision algorithms. The fast algorithms from Refs. 11 and 14 are enabled in the test model. Six MVD sequences recommended by JCT-3DV are used, including Poznan_Hall2 (1920×1088), Poznan_Street (1920×1088), Undo_Dancer (1920×1088), Kendo (1024×768), Balloons (1024×768), and Newspaper (1024×768). The proposed fast algorithms encoded three views, and the six intermediate views were rendered using the decoded views. Meanwhile, the other six intermediate views generated by the uncompressed depth and texture were used as an anchor for the PSNR calculation. A common test condition of all the intraconfigurations is used for evaluations.[23] The approaches[13,15,16] mentioned in Sec. 1 are implemented on the HTM-8.1[22] for comparison.

The time-saving ΔT of the overall depth-encoding process is used to evaluate the complexity reduction. For an objective evaluation of video quality, we measure Bjontegaard delta bit rate (BDBR) and Bjontegaard delta peak signal-to-noise ratio (BDPSNR), which come from Bjontegaard metrics[24] presenting the average differences in the RD performance of the synthesized view quality and the total bitrate including texture and depth videos. They represent essentially the same meaning, and the variation in terms of BDBR is more noticeable than that in terms of BDPSNR. For the sake of simplicity, we include only BDBR as the measurement metric in the simulation results. The experiments are performed in 64-bit MS Windows 7 OS running on an Intel Xeon(R) E3-1230 CPU of 3.3 GHz and 16.0 GB RAM. The same clear environment is provided for each simulation.

4.1 *Performance of Depth-Modeling Modes*

In order to verify the high computational complexity of DMMs and the motivation of this paper, the performance of

Table 7 Performance of the anchor when disabling DMMs and disabling DMM1+DMM3.

	Disable DMMs		Disable DMM1 + DMM3	
Sequence	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)
Kendo	−33.25	2.94	−27.80	2.05
Newspaper	−35.15	6.05	−28.17	3.14
Balloons	−34.00	2.85	−28.22	2.18
Undo_Dancer	−27.67	6.48	−23.32	3.77
Poznan_Hall2	−28.30	5.58	−22.70	4.10
Poznan_Street	−30.82	2.03	−25.20	1.23
Average	−31.50	4.33	−25.93	2.75

DMMs is shown first. The original depth intracoding is executed with DMMs disabled, and the performance in terms of the coding performance and encoding time is shown in the left part of Table 7. It is clear that DMM is an effective technology that provides an improvement of 4.33% in BDBR on average. At the same time, it is a relatively time-consuming process, occupying 31.5% of the entire coding process on average. To demonstrate further the necessity for our fast wedgelet decision, we also test the anchor with only DMM1 and DMM3 disabled. The right part of Table 7 shows that the wedgelet partition-based modes (DMM1 and DMM3) are still the most time-consuming processes inside DMM, accounting for 25.9% of the overall intracoding encoding time.

4.2 *Performance of the Proposed Schemes Individually*

The effects of the proposed schemes for DMM1 and DMM3 are demonstrated separately in this section. As mentioned in Sec. 3.2, in the proposed fast DMM1 scheme, the threshold T should be selected carefully. Extensive simulations were conducted to refine the previously obtained approximate threshold. The influence of different values of T on the performance of the proposed algorithm is shown in Table 8. It is verified in the table that too low a threshold does not separate R_{pl} properly from $R_{non\text{-}pl}$, and results in a degradation of performance. This is consistent with the probability distribution shown in Fig. 7. On the other hand, larger values of T allow a greater variation in pixel intensity in each partition region. Consequently, the PUs as shown in Fig. 8(b) are classified incorrectly as case 1 (only one candidate). Hence, the time savings is marginally better while the quality drops significantly. Since the impact of the sequences and quantization factors is not significant in the threshold selection, a fixed threshold value of 1.2 is set based on the simulation results, instead of an adaptive threshold. This already represents an acceptable tradeoff between the quality and complexity.

Table 8 Impact of the selection of threshold T for proposed DMM1 algorithm.

Value of T	0.1	0.8	1.2	1.6	2
BDBR (%)	0.94	0.67	0.49	1.24	1.45
ΔT (%)	−12.7	−10.4	−11.3	−12.8	−15.5

Table 9 Performance of the proposed methods for DMM1 and DMM3.

	Proposed method in DMM1		Proposed method in DMM3	
Sequence	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)
Kendo	−12.33	0.51	−6.21	−0.16
Newspaper	−12.01	0.90	−7.92	0.18
Balloons	−11.75	0.64	−5.53	0.07
Undo_Dancer	−9.63	0.31	−3.61	0.08
Poznan_Hall2	−11.34	0.33	−4.73	0.16
Poznan_Street	−10.89	0.26	−5.31	0.07
Average	−11.33	0.49	−5.56	0.07

In addition, the results in Table 9 show that our proposed DMM1 decision method can save 11.33% of the overall encoding time, with the BDBR increasing by only 0.49%. Table 9 also shows that the proposed DMM3 decision method can achieve a savings of 5.56% of the depth encoding time, with almost no performance degradation.

4.3 *Performance Compared with the Wedgelet-Pattern-Reducing Algorithm*

The results of the comparison between the proposed combined algorithm (considering both of the methods for DMM1 and DMM3) and the wedgelet-pattern-reducing method of Zhang[13] are tabulated in Table 10. In Ref. 13, the direction of the rough mode is used to determine the wedgelet-pattern subset. This method brings about a 1.2%

Table 10 Comparison of encoding performance between the proposed algorithm and the wedgelet-pattern-reducing methods.

	Zhang's[13]		Proposed	
Sequence	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)
Kendo	−15.9	1.19	−16.5	0.52
Newspaper	−15.7	2.10	−17.0	0.93
Balloons	−15.8	1.28	−16.5	0.64
Undo_Dancer	−12.0	0.84	−13.9	0.34
Poznan_Hall2	−12.9	1.16	−14.4	0.37
Poznan_Street	−14.0	0.66	−15.6	0.30
Average	−14.3	1.21	−15.7	0.52

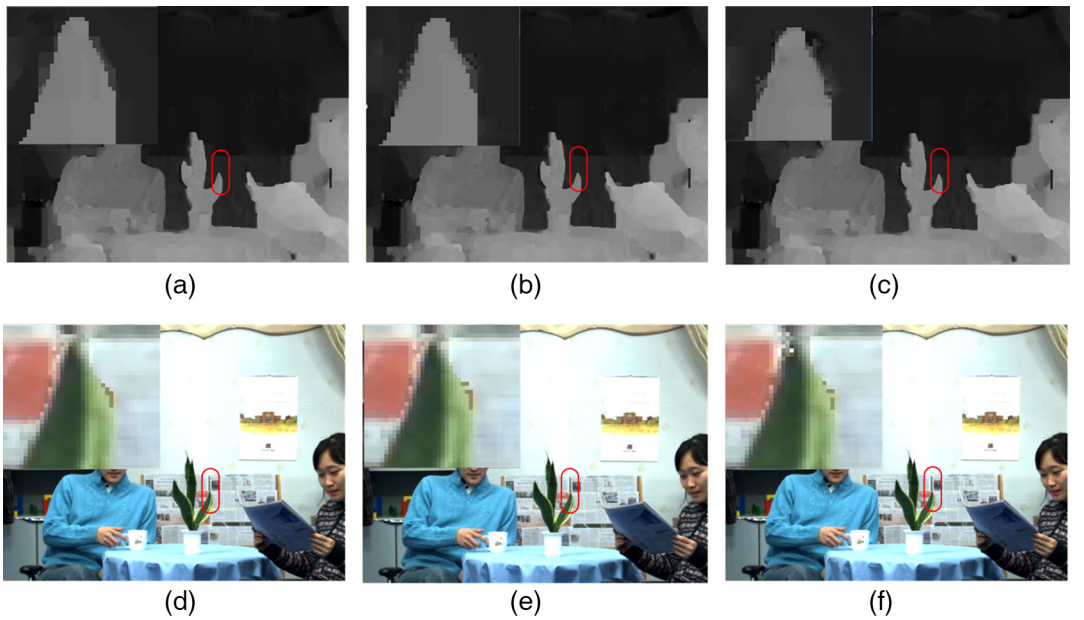

Fig. 10 Comparison of the subjective quality between the proposed algorithm and Zhang's method:[13] (a) depth map encoded by HTM8.1, (b) depth map encoded by proposed algorithm, (c) depth map encoded by Zhang's method,[13] (d) synthesized view by HTM8.1, (e) synthesized view by proposed algorithm, and (f) synthesized view by Zhang's method.

increase in the BD-rate, which counteracts approximately half the benefit produced by the two wedgelet-pattern-based modes (DMM1 and DMM3). In addition, Fig. 10 shows the comparison of subjective quality for the depth map and the synthesized view. The red rounded regions show that our method (the middle one) can better retain the sharp edges of the depth map. It is also noted that several artifacts (e.g., white pixels) exist, and that the left edge of the leaf is blurred in the synthesized view by Zhang's method.[13] On the contrary, the proposed algorithm provides visual effects that are similar to those of the original HTM. The reason for this is that our algorithm considers the pixel information of the current PU to determine the distribution of sharp edges. Zhang's method[13] uses only the directional mode with minimum rough cost from RMD. However, the rough mode cannot reflect the distribution of sharp edges in some cases. With respect to time savings, the proposed fast approach can provide a 15.7% reduction in the complexity of the total depth intracoding. Zhang's method[13] achieves a 14.3% time reduction. More significantly, it can be concluded that the proposed algorithm achieves a higher depth-map quality by better retaining the sharp edges, with a marginally smaller complexity compared with Zhang's algorithm.[13]

4.4 *Performance Compared and Integrated with the DMM-Skipping Algorithms*

In another type of fast DMM schemes, such as Zhang's[15] and Park's,[16] the entire DMM can be skipped for some PUs by using different features, e.g., the depth level of the quadtree in Ref. 15, and the RD-cost in Ref. 16. Additionally, the technique adopted in HTM[14] can skip some of the DMMs. These three methods share the similar idea that DMMs are designed for regions containing sharp edges and can be skipped for smooth regions; it is an idea that works particularly well. Since Ref. 14 is adopted in the HTM8.1, while Refs. 15 and 16 are implemented for comparison; the performance of the complexity reduction is concealed partially by the anchor with Ref. 14 enabled. The simulation results of Refs. 15 and 16 and the proposed combined algorithm are compared with the anchor in Table 11. Among the three algorithms, the time savings of Zhang's[15] and Park's[16] methods are 17.6% and 18.3% on average. Nevertheless, the coding performance is slightly worse than that of the proposed method. Especially for Zhang's[15] approach, the depth level of the quadtree may not be a stable feature to describe DMMs, since a DMM allows blocks with sizes from 4×4 to 32×32, which is different from the range of the quadtree level. Therefore, the BDBR increase of Ref. 15 is higher than that of the other methods.

In addition, the algorithms in Refs. 15 and 16 perform better for a smaller size of PUs by skipping the DMM process entirely for smooth cases. A larger PU contains more texture information in general, which is difficult to handle using the features employed in Refs. 15 and 16. Additionally, the number of wedgelet patterns is relatively huge for large PUs, as shown in Table 1. In such PUs, the proposed algorithm performs better by reducing the number of wedgelet patterns in the DMM decision. Therefore, the proposed algorithm is a good supplement to the algorithms of Refs. 15 and 16. When the proposed algorithm is integrated with Refs. 15 and 16, a further complexity reduction is achieved, as shown in Table 12.

4.5 *Comparison of Average Performance in Terms of Other Metrics*

In order to verify the complexity reduction by the proposed combined algorithm, additional internal metrics are checked besides the encoder runtimes. All the methods mentioned above are measured, including the anchor HTM8.1[11] with Ref. 14, the methods of Zhang,[13] Zhang,[15] and Park,[16] and the proposed one. Since the compared algorithms belong to different categories, the average number of wedgelet

Table 11 Comparison of encoding performance between the proposed algorithm and the DMM-skipping methods.

	Park's[16] (%)		Zhang's[15] (%)		Proposed (%)	
Sequence	ΔT	BDBR	ΔT	BDBR	ΔT	BDBR
Kendo	–18.9	0.57	–19.2	1.56	–16.5	0.52
Newspaper	–18.3	0.93	–18.5	2.31	–17.0	0.93
Balloons	–18.6	0.50	–18.9	1.33	–16.5	0.64
Undo_Dancer	–15.6	0.52	–17.2	0.48	–13.9	0.34
Poznan_Hall2	–16.5	0.63	–17.7	2.00	–14.4	0.37
Poznan_Street	–17.4	0.30	–18.8	0.70	–15.6	0.30
Average	–17.6	0.58	–18.3	1.40	–15.7	0.52

Table 12 Coding performance of proposed algorithm combined with the DMM-skipping methods.

	Proposed + Park's[16] (%)		Proposed + Zhang's[15] (%)	
Sequence	ΔT	BDBR	ΔT	BDBR
Kendo	–24.4	0.82	–23.9	1.76
Newspaper	–24.1	1.47	–23.7	2.61
Balloons	–23.7	0.76	–23.7	1.60
Undo_Dancer	–20.2	0.89	–20.6	0.55
Poznan_Hall2	–20.7	0.97	–19.6	2.08
Poznan_Street	–21.3	0.44	–20.4	0.71
Average	–21.3	0.92	–20.6	2.00

patterns checked in DMM1 for each frame is used as the internal metric, as shown in Table 13. Of the other algorithms, Zhang's[15] gives the most significant savings, and the proposed algorithm also performs well. This is due mainly to the fact that both of these algorithms reduce the number of wedgelet patterns in each PU. The percentage of PUs where DMM3 is tested is also recorded in the table. In this measurement, Refs. 15 and 16 perform better, since they skip the DMM3 for the smooth PUs. The proposed consistency-checking scheme also saves some computational time for DMM3 compared to the anchor. Generally, Refs. 15 and 16 perform better in the overall time savings, which is also consistent with the previous results in terms of runtimes. On the other hand, the advantages of our proposed combined algorithm are the better coding performance and the fact that it could be integrated with Refs. 15 and 16 for further time reduction, as illustrated above.

The average performance of all the above algorithms is also summarized in Table 14. It is shown that Park's[16] and the proposed method have the best performance in terms of the ratio between encoder-runtime reduction and bit-rate increase. On the other hand, the combined methods perform worse than the individual methods in that measurement. However, as is generally well known, all fast methods in video coding try to make a tradeoff between coding speed and performance. The choice of schemes depends greatly on the preference of applications. The proposed method is favored for applications more concerned about coding performance. In contrast, for low-bitrate applications where the runtime is more important, the combined solutions may be preferred due to the more significant time reduction.

5 Conclusion

In this paper, we present a fast wedgelet-pattern-decision approach to reduce the computational complexity of the 3-D-HEVC encoder by exploiting two methods, including an edge-based wedgelet-pattern decision in DMM1 and a DMM3-skipping strategy. In the edge-based wedgelet-pattern decision in DMM1, we divide each PU into complementary regions. Partial smoothness is evaluated to estimate the most probable region covering the partition line. The

Table 13 Comparison of the complexity reduction in terms of other metrics.

Metrics	Anchor HTM 8.1[11] with[14]	Zhang's[15]	Park's[16]	Zhang's[13]	Proposed
No. of wedgelet patterns tested in DMM1 per frame (% of anchor)	7,452,421 (100%)	990,810 (13.3%)	900,893 (12.1%)	516,438 (6.9%)	683,823 (9.17%)
Percentage of PUs testing DMM3	27%	12.7%	12.5%	27%	17%

Table 14 Comparison of the average performance of all algorithms.

Metrics	Zhang's[13]	Zhang's[15]	Park's[16]	Proposed	Proposed + Zhang's[15]	Proposed + Park's[16]
ΔT (%)	–14.3	–18.3	–17.6	–15.7	–20.6	–21.3
BDBR (%)	1.21	1.4	0.58	0.52	2.00	0.92
ΔT/BDBR	11.8	13.1	30.3	30.2	10.3	23.2

minimum distortion wedgelet pattern is then searched only in the most probable region. In the DMM3-skipping strategy, the MPMs estimated from the neighboring PUs are extended to evaluate the mismatch between the depth PU and its colocated texture PU. When mismatch occurs, DMM3 is skipped to speed up the encoding process. The experimental results show that the proposed algorithm has better performance compared with the wedgelet-pattern-reducing algorithm[13] and the DMM-skipping algorithm[15] in terms of the ratio between time-reduction and BDBR. In addition, the proposed algorithm has a similar performance to that of another DMM-skipping algorithm,[16] in the sense that it achieves a marginally better coding performance with a marginally smaller complexity reduction. In addition, our algorithm is a good supplement to the DMM-skipping algorithms.[15,16] Integration with Ref. 16 could provide a complexity reduction of up to 21%, while the BDBR is increased by 0.92%.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61301109) and by a grant from the Research Grants Council of the HKSAR, China (Grant No. PolyU 5119/13E).

References

1. G. J. Sullivan et al., "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.* **22**(12), 1649–1668 (2012).
2. ISO/IEC MPEG and ITU-T VCEG, "Call for proposals on 3D video coding technology," Standard ISO/IECJTC1/SC29/WG11, Daegu, Korea (2011).
3. K. Muller et al., "3D high-efficiency video coding for multi-view video and depth data," *IEEE Trans. Image Proces.* **22**(9), 3366–3378 (2013).
4. A. Smolic et al., "Multi-view video plus depth (MVD) format for advanced 3D video systems," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, document JVT-W100, San Jose (2007).
5. P. Kauff et al., "Depth map creation and image based rendering for advanced 3DTV services providing interoperability and scalability," *Signal Process. Image Commun.* **22**(2), 217–234 (2007).
6. K. Müller, P. Merkle, and T. Wiegand, "3D video representation using depth maps," *Proc. IEEE* **99**(4), 643–656 (2011).
7. H. Schwarz et al., "3D video coding using advanced prediction, depth modeling, and encoder control methods," in *Picture Coding Symp.*, pp. 1–4, Krakow, Poland (2012).
8. P. Merkle et al., "The effects of multiview depth video compression on multi-view rendering," *Signal Process. Image Commun.* **24**(1–2), 73–88 (2009).
9. W. S. Kim et al., "Depth map coding with distortion estimation of rendered view," *Proc. SPIE* **7543**, 75430B (2010).
10. B. T. Oh, J. Lee, and D. S. Park, "Depth map coding based on synthesized view distortion function," *IEEE J. Sel. Topics Signal Process.* **5**(7), 1344–1352 (2011).
11. P. Merkle, K. Muller, and T. Wiegand, "Coding of depth signals for 3D video using wedgelet block segmentation with residual adaptation," in *IEEE Int. Conf. on Multimedia and Expo (ICME '13)*, pp. 1–6, San Jose (2013).
12. P. Merkle et al., "Simplified wedgelet search for DMM modes 1 and 3," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Document, JCT3V-B0039, Shanghai, China (2012).
13. M. M. Zhang et al., "A fast depth map wedgelet partitioning scheme for intra prediction in 3D video coding," in *IEEE Int. Symp. on Circuits and Systems (ISCAS '13)*, pp. 2852–2855, Beijing, China (2013).
14. Z. Y. Gu et al., "Fast depth modeling mode selection for 3D HEVC depth intra coding," in *IEEE Int. Conf. on Multimedia and Expo Workshops (ICMEW '13)*, pp. 1–4, San Jose, California (2013).
15. Q. W. Zhang et al., "Effective early termination algorithm for depth map intra coding in 3D-HEVC," *IET Electron. Lett.* **50**(14), 994–996 (2014).
16. C. S. Park, "Efficient intra-mode decision algorithm skipping unnecessary depth-modelling modes in 3D-HEVC," *IET Electron. Lett.* **51**(10), 756–758 (2015).
17. L. Zhang et al., "3D-HEVC test model 5," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JCT3V-E1005, 5th Meeting: Vienna, Austria, 27 July–2 August (2013).
18. C.-H. Fu et al., "Fast wedgelet pattern decision for DMM in 3D-HEVC," in *IEEE Int. Conf. on Digital Signal Processing (DSP '15)*, pp. 477–481, Singapore (2015).
19. E. G. Mora et al., "Depth video coding based on intra mode inheritance from texture," *APSIPA Trans. Signal Inf. Process.* **3**, e1 (2014).
20. G. Tech et al., "3D-HEVC test model 1," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/ SC 29/ WG 11, JCT3V-A1005, 1st Meeting: Stockholm, Sweden, 16–20 July (2012).
21. Y. Zhao et al., "Boundary artifact reduction in view synthesis of 3D video: from perspective of texture-depth alignment," *IEEE Trans. Broadcast.* **57**(2), 510–522 (2011).
22. F. Bossen et al., "MPEG 3D-HTM software V8.1," ITU-T SG16 WP3 and ISO/IECJTC1/SC29/WG11, https://hevc.hhi.fraunhoferdesvn/svn3DVCSoftware/tags/8.1 (8 May 2013).
23. F. Jäger, "Description of core experiment 5 (CE5) on depth intra modes," ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JCT3V-E1105, 5th Meeting: Vienna, Austria, 27 July–2 August (2013).
24. G. Bjontegaard, "Calculation of average PSNR differences between RD curves," 2001: ITU-T SG16 Q.6 Document, VCEG-M33, Austin, April (2001).

Hong-Bin Zhang received his BSc degree from Northeast Agriculture University, Ha'erbin, China, in 2011. Now, he is currently pursuing a PhD degree in Nanjing University of Science and Technology. His research interests include 3D video coding, multiple-view video, image signal processing, video transcoding and video summarization.

Chang-Hong Fu received his BEng with a first-class honors degree and PhD degrees from the Hong Kong Polytechnic University in 2002 and 2008, respectively. He joined Nanjing University of Science and Technology in 2011 and is now an associate professor in the School of Electronic and Optical Engineering. He has published over 30 research papers in various international journals and conferences. His research and technical interests mainly include video compression and image processing.

Yui-Lam Chan received his BEng with a first class honors degree and his PhD degree from the Hong Kong Polytechnic University in 1993 and 1997, respectively. He is now an associate professor in the Department of Electronic and Information Engineering. He has published over 90 research papers in various international journals and conferences. His research and technical interests include video compression, video transcoding, digital TV/HDTV, 3DTV, multiview-video coding, future video coding standards and error-resilient coding.

Biographies for the other authors are not available.