RESEARCH

CrossMark

Fast inter-prediction algorithm based on motion vector information for high efficiency video coding

Kao-Min Lin[1], Jie-Ru Lin[1], Mei-Juan Chen[1], Chia-Hung Yeh[2,3*] and Cheng-An Lee[1]

Abstract

High Efficiency Video Coding (HEVC/H.265) is the latest international video coding standard, which achieves better compression ratio and supports higher resolution than Advanced Video Coding (H.264/AVC). However, HEVC/H.265 increases the computational burden. To reduce the coding complexity of the HEVC encoder, this paper proposes a fast inter-prediction algorithm to speed up coding time. We collect the average rate-distortion costs (RD-cost) of Skip modes and Merge modes to accelerate prediction unit (PU) mode decisions. In addition, we also acquire and analyze the motion vector range from Merge modes and Inter $2N \times 2N$ modes to decide whether to execute Merge and advanced motion vector prediction (AMVP) of other PUs. The experimental results show that the proposed algorithm provides 48.54% time saving on average in random-access configuration and maintains good rate-distortion performance and video quality at the same time. The proposed algorithm also outperforms previous works.

Keywords: High efficiency video coding, Fast algorithm, Motion vector range, Merge mode, Advanced motion vector prediction (AMVP), Inter-prediction

1 Introduction

With the advances in video technology, such as video streams, computer games, and TV shows, the video applications are everywhere in our life. The increasing demands of video quality and video resolution also bring about the growing data amount. Considering the future development of video applications, ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) established the Joint Collaborative Team on Video Coding (JCT-VC) and cooperated to develop the video coding standard for the next generation. The newest coding standard, High efficiency Video Coding (HEVC/H.265) [1], not only improves the compression efficiency but also supports the coding of ultra-high-definition (UHD) resolution up to $8K \times 4K$. Moreover, the required coding bitrate of HEVC/H.265 is almost only half of that of Advanced Video Coding

(H.264/AVC) [2]. However, the trade-off is the dramatic increase in coding complexity.

The development of HEVC/H.265 is based on the framework of H.264/AVC, which the residual from inter- or intra-prediction can be transformed by Discrete Cosine Transform (DCT) and quantized before entropy coding. In addition, there are many novel coding techniques in HEVC/H.265 encoder to provide more powerful coding efficiency. The coding structure of HEVC consists of Coding Units (CUs), Prediction Units (PUs), and Transform Units (TUs). CU is based on a quad-tree partition structure with the depth varying from 0 to 4 and the corresponding size varying from 64×64 to 8×8 as shown in Fig. 1. In addition, a CU with size 64×64 is referred to a Coding Tree Unit (CTU). PU is the prediction unit. There are also several partition types for the PU to perform mode prediction. As shown in Fig. 2, the inter-prediction modes comprise Merge/Skip $2N \times 2N$, Inter $2N \times 2N$, Symmetric Motion Partition (SMP, including Inter $2N \times N$ and Inter $N \times 2N$), Asymmetric Motion Partition (AMP, including Inter $2N \times nU$, Inter $2N \times nD$, Inter $nL \times 2N$, and Inter $nR \times 2N$), and Inter $N \times N$, while the intra-prediction modes involve Intra $2N \times 2N$ and

* Correspondence: chyeh@ntnu.edu.tw; yeh@mail.ee.nsysu.edu.tw
[2]Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan
[3]Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan
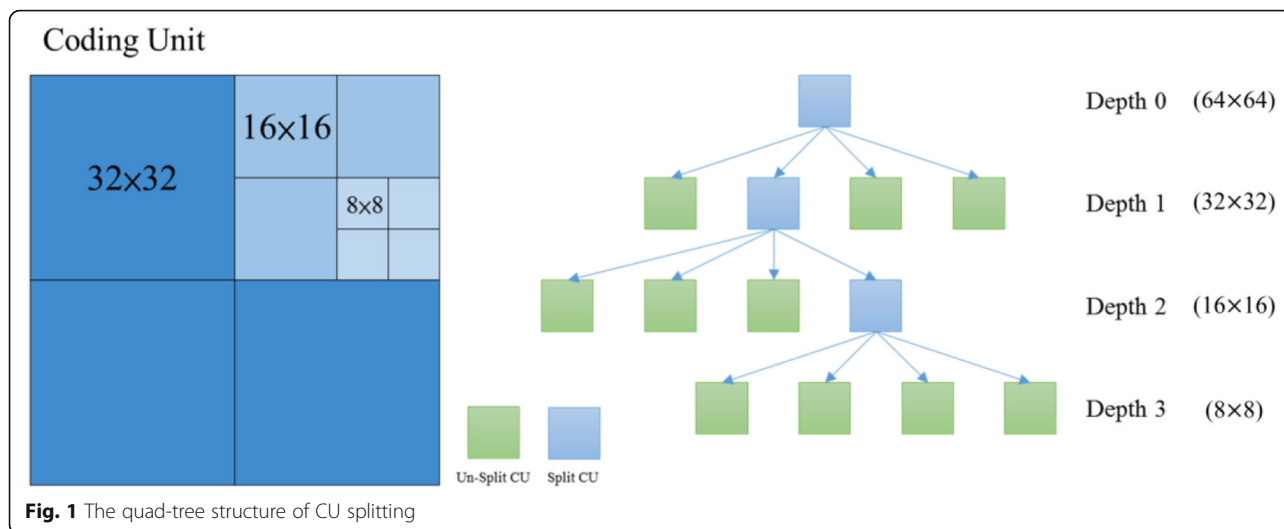Full list of author information is available at the end of the article

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 2 of 17

Fig. 1 The quad-tree structure of CU splitting

Intra N×N. The best PU mode is selected according to the minimum rate-distortion cost (RD-cost) of all modes. The flowchart in Fig. 3 shows the PU prediction procedure in the encoder side of HEVC reference software (HM).

If the information of the current PU is similar to a neighboring block, the motion vector (MV) may be fairly alike and the MV of the current PU can be predicted from the neighboring coded block. Compared to motion vector prediction (MVP) in H.264/AVC, advanced motion vector prediction (AMVP) [3] in HEVC/H.265 adds more candidates from spatial and temporal domains to select the MV predictor more precisely and to reduce bitrate. The reference neighboring PUs for Merge and AMVP candidates are denoted in Fig. 4. AMVP adds MVs of the first two reference PUs with valid MVs in the order of (A0 or A1), (B0, B1, or B2), and (T0 or T1) into candidate list, where A0, A1, B0, B1, and B2 are spatially neighboring PUs, while T0 and T1 are temporally neighboring PUs. Then, the best MV predictor is

selected from MVs in candidate list, which is with the minimum RD-cost. Finally, the index of the best MV predictor, residual, and motion vector difference (MVD) will be transmitted.

To further diminish the required data, Merge mode [4] is a new coding tool designed in HEVC, which only the index of the best MV candidate and residual will be transmitted if Merge mode is selected as the best mode. During the prediction, Merge mode adds MVs of the first four reference PUs with valid MVs from spatially neighboring PUs in the order of (A1, B1, B0, A0, B2) and the first one of the valid MV candidates from temporal PUs in the order of (T0, T1) into candidate list. The reference PUs are also shown in Fig. 4. Finally, the Merge MV is selected from MVs in a candidate list, which is with the minimum RD-cost. Furthermore, if coded block flag (CBF) is 0 during the prediction of Merge 2N×2N, it means that the residual is 0. The situation will be defined as Skip mode and the residual will not be transmitted.

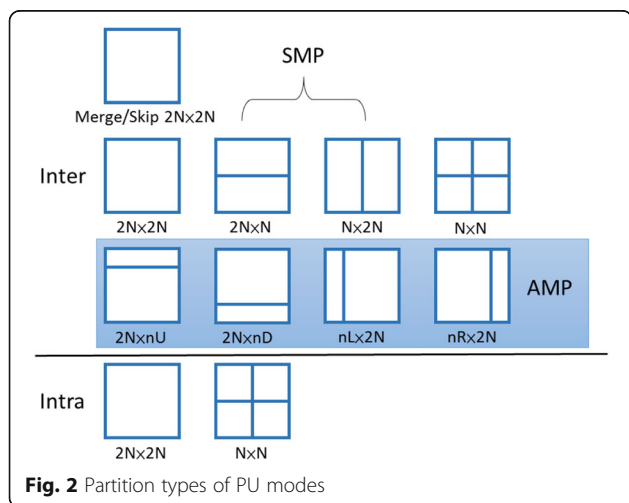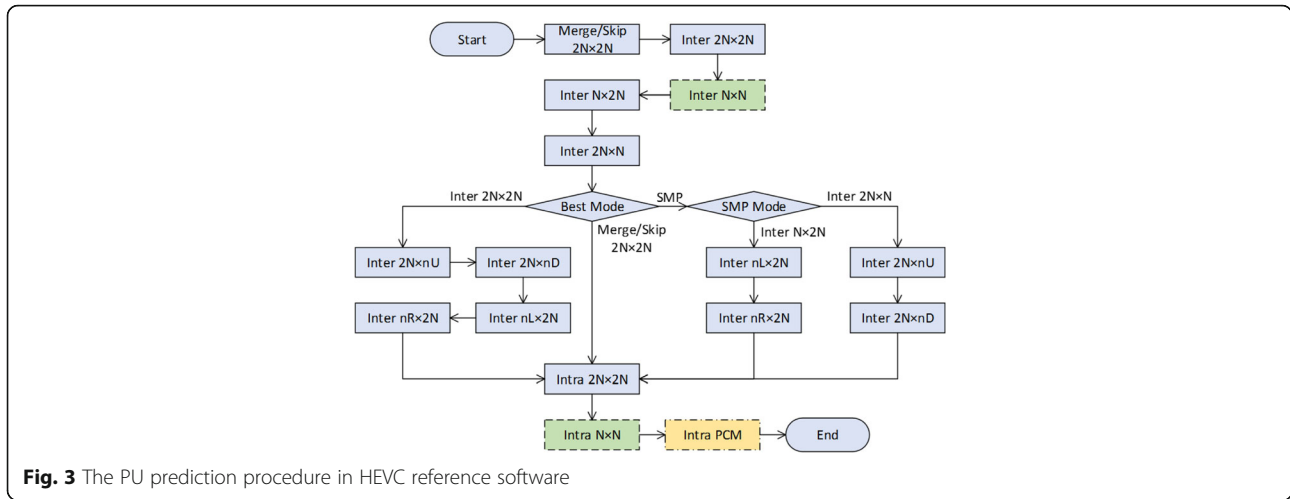## 2 Background and related works

More advanced coding techniques and flexible block sizes for both CU and PU in HEVC contribute to higher compression efficiency and reduce half of the bit-rate compared to H.264/AVC. However, they also result in increasing coding complexity and boosting coding time. There have been several optional fast coding algorithms involved in HEVC standard. In coding flag mode (CFM) [5], once the CBF of current PU is 0, the following PU mode decision in the current CU depth will be bypassed. Early CU (ECU) termination [6] avoids further CU splitting if the best prediction mode of the current CU depth is Skip mode. Early SKIP detection (ESD) [7] checks Inter 2N×2N before Merge/Skip 2N×2N. At that time, the subsequent PU mode decision will be bypassed if both of MVD and CBF are 0.

Fig. 2 Partition types of PU modes

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 3 of 17

Fig. 3 The PU prediction procedure in HEVC reference software

In recent years, numerous fast algorithms are also proposed to further accelerate and improve the encoding process of HEVC. BenHajyoussef et al. [8] detect the gradient information and speed up the intra modes searching. Zhang et al. [9] shrink unnecessary intra modes in Rate-distortion Optimization (RDO) process and early terminate CU partition according to the coding bits of the current CU. In [10], the CU splitting optimization is viewed as a classification task and solved by the weighted Support Vector Machine (SVM). Huang et al. [11] efficiently switch AMP by posterior probability analysis and adaptively adjust the search range by motion degree. Huang et al. [12] predict the CU depth range from neighboring CTUs. In addition, the coding information of Merge/Skip 2N×2N and CBF is utilized for the decisions of early split and early termination. For the work in [13], Yoo et al. check the RD-costs and CBF to Skip Inter and Intra PU modes. Yang et al. [14] disable AMVP if the best mode after Inter 2N×2N is Skip mode. A parallel merge estimation region (MER) is proposed by Jiang et al. [15] to remove the dependency of Merge MV candidates. Yang et al. [16] expedite the coding time by justifying the accuracy of AMVP and confining the reference frames. Shen et al. [17] propose an adaptive inter mode decision by the correlations of the reference CUs as well as the PU mode complexities. From the study of Tan et al. [18], the CU quad-tree is pruned by the prediction residual to decline the high encoding time. In the work of Pan et al. [19], the
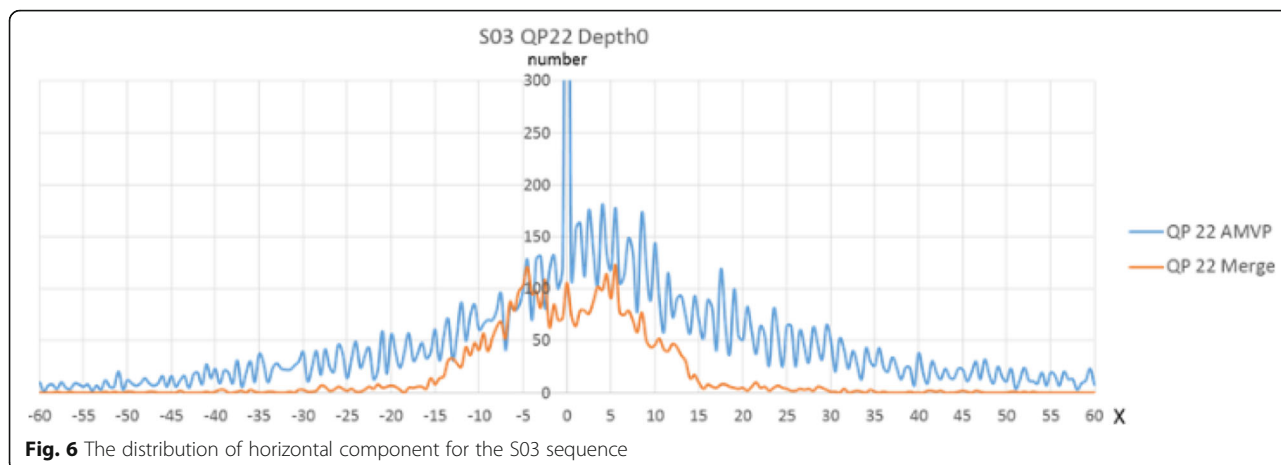

Fig. 4 Reference locations of Merge and AMVP candidates

Fig. 5 The flowchart of the fast CU coding
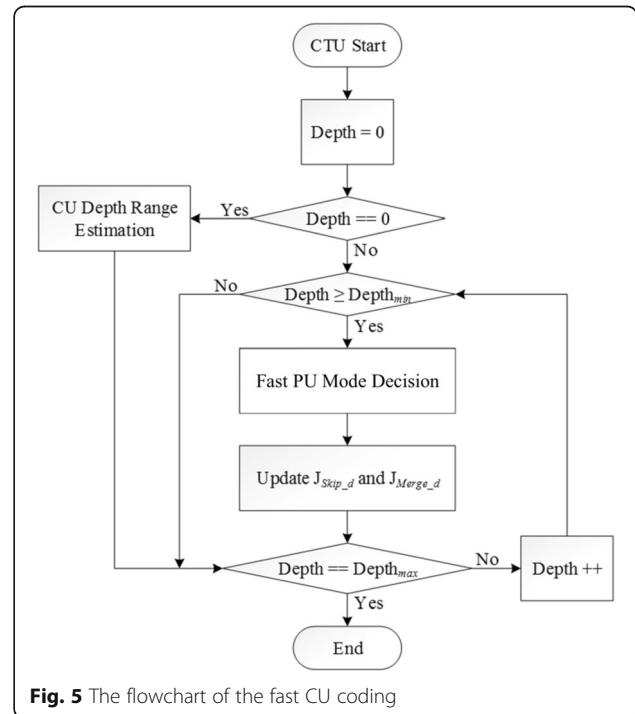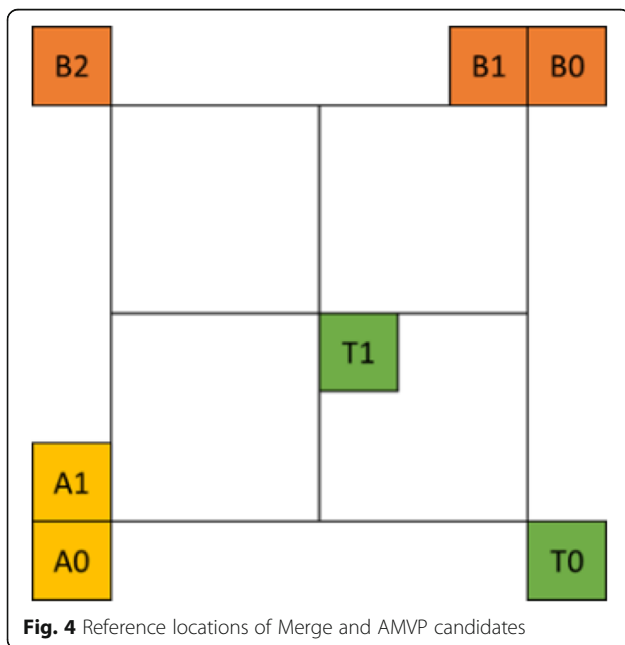
Fig. 6 The distribution of horizontal component for the S03 sequence

CU splitting will be early terminated if the residual and MV length are all 0. Wu et al. [20] propose the joint constraint of the best PU mode, the second-best PU mode and the CBF information for early termination of the coding of the CTU. Ahn et al. [21] use the information of the SAO (sample adaptive offset) parameter to calculate the complexity of the current CU. In addition, they set thresholds according to the RD-costs of the CUs with different complexities to early terminate the coding process.

Most of fast algorithms utilize the RD-costs and CBF to early terminate the prediction process under the combination of different situations. In this paper, a novel fast coding criterion is proposed. We analyze the motion vectors from Merge/Skip 2N×2N and AMVP. The MV range will be calculated and further determined whether to disable Merge or AMVP in a PU to accelerate the prediction. Besides, the early termination by RD-costs and search range reduction are also designed to elevate the coding time-saving. The rest of this paper is arranged as follows. Section 3 describes the proposed fast inter-prediction algorithm comprehensively. Section 4

demonstrates the experimental results. Finally, the work is briefly concluded in Section 5.

3 Proposed fast inter-prediction algorithm
The fast inter-prediction algorithm in this paper can be divided into fast CU coding and fast PU mode decision. We incorporate the fast CU decision in [22], which avoids any redundant splitting process, with the proposed fast PU mode decision to speed up the prediction procedure. The detailed explanations are provided as follows.

3.1 Fast CU coding
3.1.1 RD-cost application
Skip mode is a distinctive situation in Merge 2N×2N, which means the transformed residual is 0 and with low distortion and required bitrate. As a result, if the RD-cost of the current PU is smaller than the average RD-cost of previous coded Skip modes, it indicates that the prediction is fairly precise and the succeeding PU modes may be omitted. We average the latest five RD-costs of the Skip modes [13, 22] and Merge modes which are selected as the best mode and denoted as
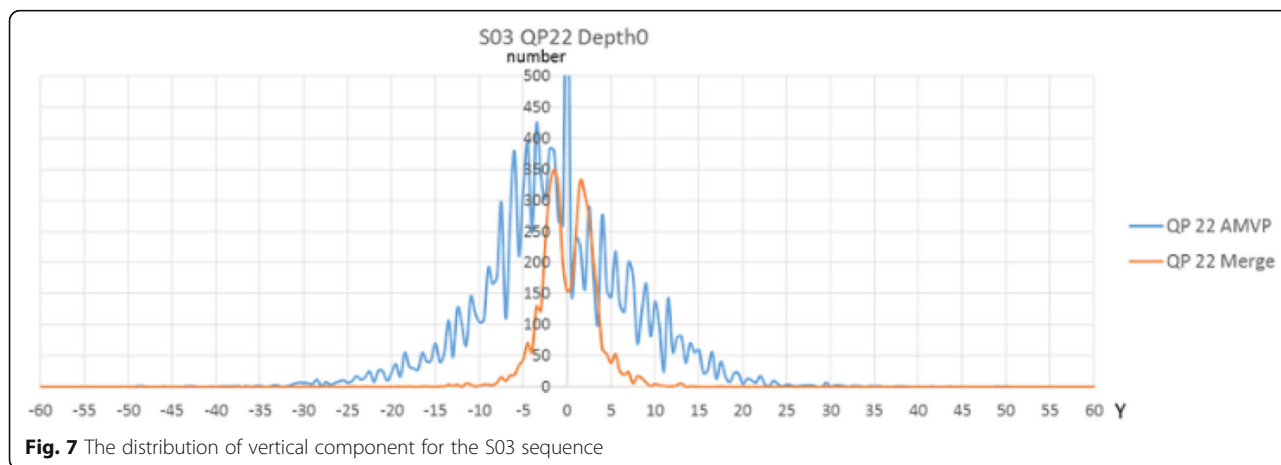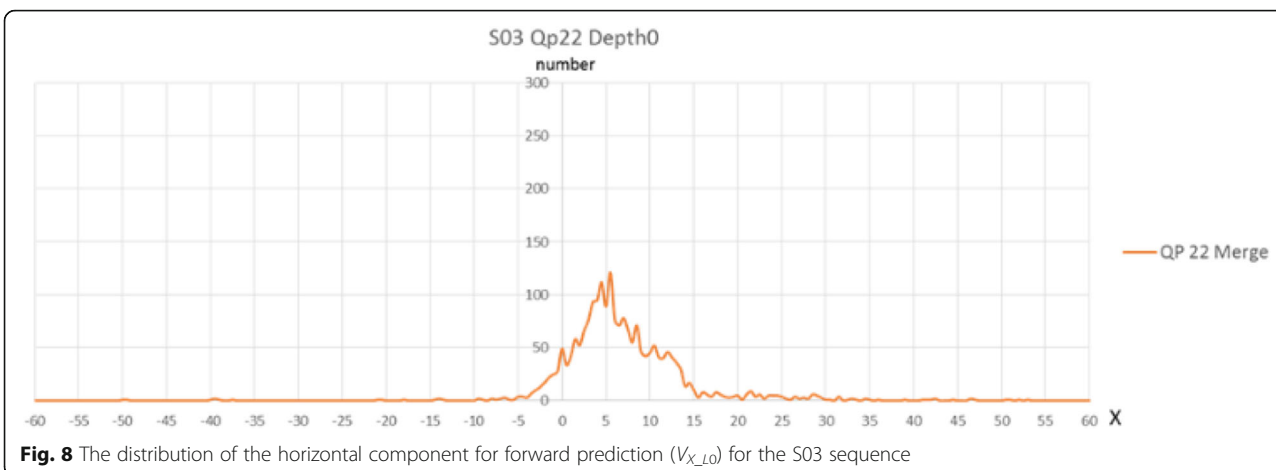

Fig. 7 The distribution of vertical component for the S03 sequence

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 5 of 17

Fig. 8 The distribution of the horizontal component for forward prediction (V_{X_L0}) for the S03 sequence

J_{Skip_d} and J_{Merge_d} by (1)(2). d means the CU depth, n is denoted as the number of Skip modes selected as best modes, and m indicates the number of Merge modes selected as the best mode.

$$J_{Skipd} = \frac{1}{5} \sum_{i=0}^{4} J_{Skipd_{n-i}} \tag{1}$$

$$J_{Merged} = \frac{1}{5} \sum_{i=0}^{4} J_{Merged_{m-i}} \tag{2}$$

3.1.2 CU depth range estimation
For a complex area, CU tends to be split into smaller sizes. Alternatively, larger CU sizes are more suitable for a smooth region. We apply the CU depth estimation and CU depth adjustment in [22] to the proposed method.

3.1.3 Fast CU coding
The overall algorithm of fast CU coding which combines the methods in Section 3.1.1 and Section 3.1.2 is shown

in Fig. 5. We calculate the average RD-costs of Skip mode and Merge mode in Section 3.1.1. Moreover, the CU depth range is estimated in Section 3.1.2. During the fast CU coding, we avoid the redundant CU coding if the CU depths are not within the interval of estimated CU depth range. In addition, the fast PU mode decision in Fig. 5 will be introduced in Section 3.2.

3.2 Fast PU mode decision
Most of the existing criteria usually early terminate the PU mode decision by RD-cost. In the proposed method, we take MV candidates of Merge 2N×2N into consideration to avoid unnecessary computation of Merge or AMVP. The complete explanations of fast PU mode decision are described below.

3.2.1 Motion vector analysis
Both the Merge mode and AMVP acquire the neighboring MVs to perform motion vector prediction. In addition, the reference neighboring PUs of MV candidates are the same, and the only difference is the number and priority of the reference candidates. We
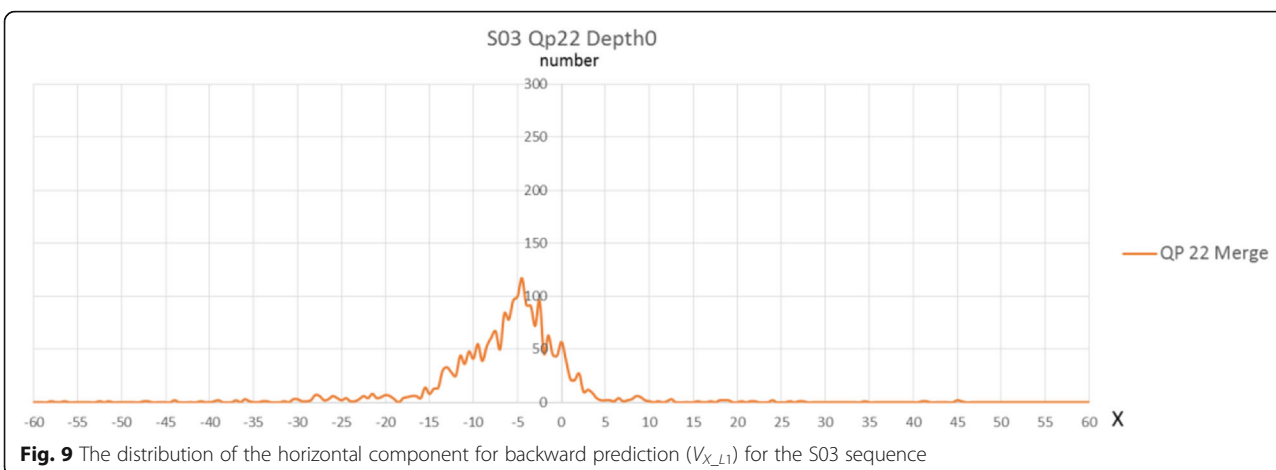
Fig. 9 The distribution of the horizontal component for backward prediction (V_{X_L1}) for the S03 sequence
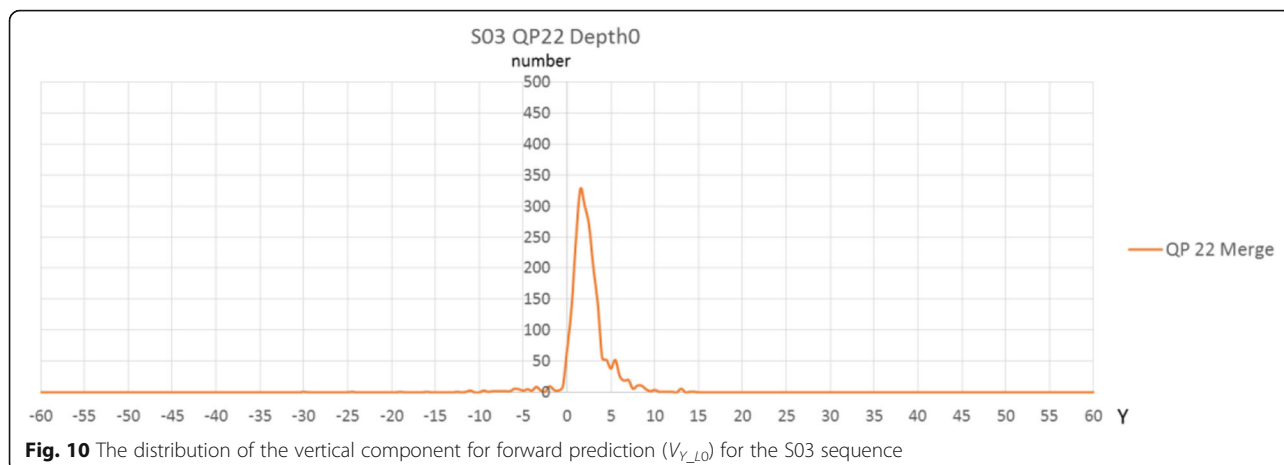
Fig. 10 The distribution of the vertical component for forward prediction (V_{Y_L0}) for the S03 sequence

consider that the selected MVs are probably related to the best prediction mode. As a result, we analyze the motion vector properties of Merge 2N×2N and Inter 2N×2N when these two modes are individually selected as the best mode. For Merge 2N×2N, we acquire forward MV and backward MV from the location of the best reference PU. For Inter 2N×2N, we acquire the forward MVs and backward MVs from the location of the two reference PU candidates of AMVP. Consequently, the number of Merge MVs is two (forward and backward MVs) while the number of MVs of the AMVP candidates of Inter 2N×2N is four (two candidates with forward and backward MVs).

Then, we investigate the range of Merge MVs and the AMVP candidates. Figures 6 and 7 show the distributions of Merge MVs and AMVP candidates of the S03 sequence at depth 0 with QP22 for horizontal and vertical components, respectively. We can find that the MV lengths of the horizontal component and the vertical component for AMVP candidates are usually larger than those of Merge 2N×2N.

We further separate the Merge MVs into forward prediction ($Merge\_MV_{L0}$) and backward prediction ($Merge\_MV_{L1}$) with the horizontal component and the vertical component as in (3). V_{X_L0} and V_{Y_L0} indicate the horizontal and the vertical components for forward prediction, respectively. V_{X_L1} and V_{Y_L1} denote the horizontal and the vertical components for backward prediction, respectively. Figures 8, 9, 10, and 11 show the distributions of Merge MV components with different prediction directions. It can be seen that most of the MVs of forward prediction (V_{X_L0} and V_{Y_L0}) are positive, whereas most of Merge MVs of backward prediction (V_{X_L1} and V_{Y_L1}) are negative for the S03 sequence.

$$MergeMV_{L0} = (V_{XL0}, V_{YL0})$$
$$MergeMV_{L1} = (V_{XL1}, V_{YL1}) \tag{3}$$

Excluding the zero vector, we also individually average the horizontal and vertical components of forward/
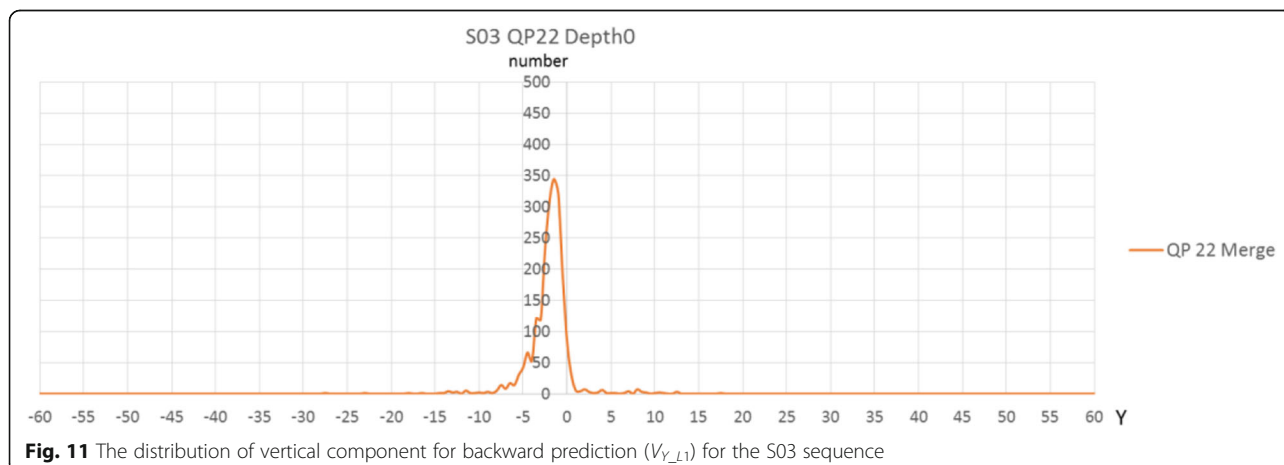
Fig. 11 The distribution of vertical component for backward prediction (V_{Y_L1}) for the S03 sequence

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 7 of 17
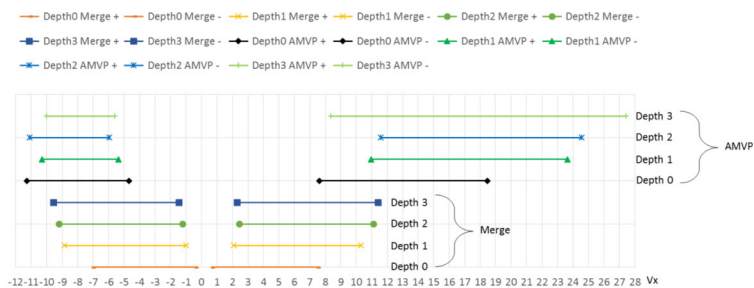
(a) MV range distribution of the horizontal component (V_X)
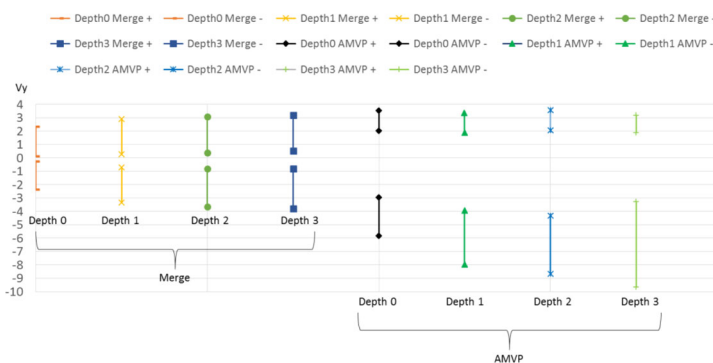
(b) MV range distribution of the vertical component (V_Y)

Fig. 12 MV range distribution of the (**a**) horizontal component (V_X) and (**b**) vertical component V_Y of the S03 sequence at QP22
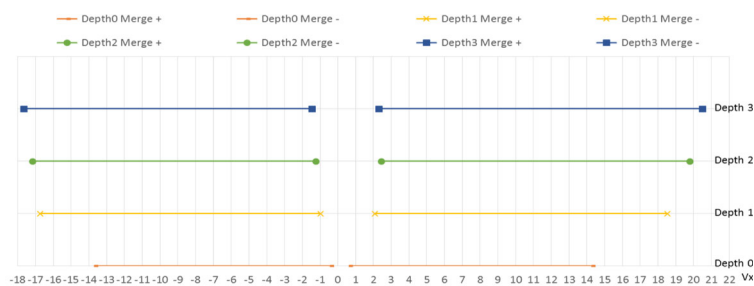
(a) MV range distribution of the horizontal component (V_X) after extension
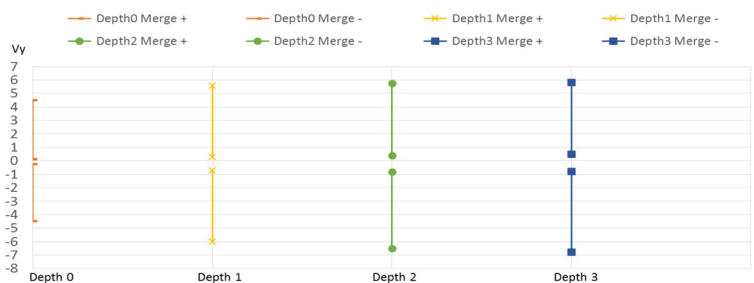
(b) MV range distribution of the vertical component (V_Y) after extension

Fig. 13 MV range distribution of the (**a**) horizontal component (V_X) and (**b**) vertical component V_Y of the S03 sequence at QP22 after extension
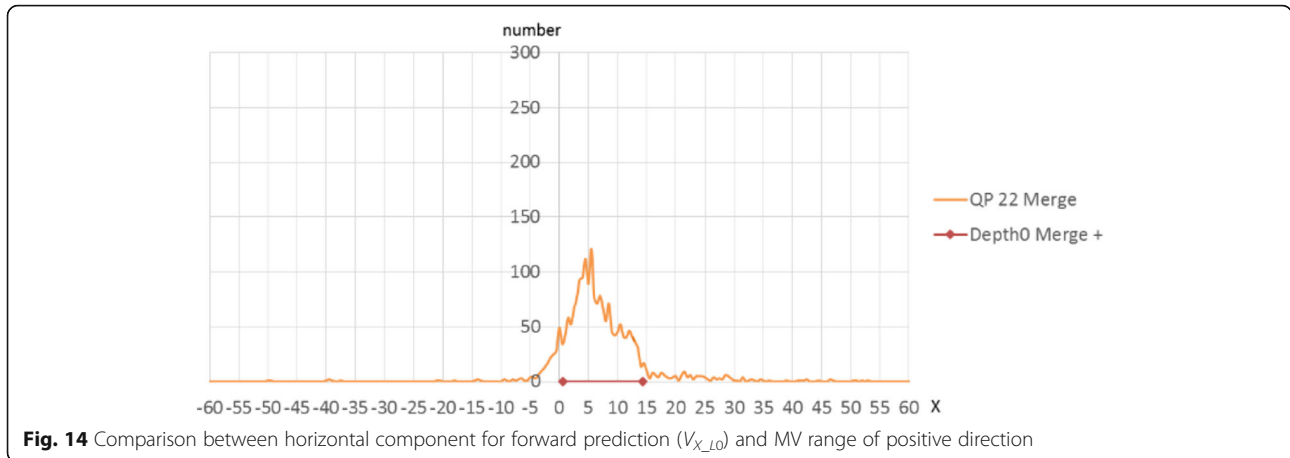
Fig. 14 Comparison between horizontal component for forward prediction (V_{X_L0}) and MV range of positive direction

backward predictions for both Merge MVs as well as the AMVP candidates and classify them into positive or negative directions. The zero vector is not positive or negative and is usually inserted due to the absence of candidates, so the zero vector is not taken into consideration. Figure 12 shows the average MV distributions of horizontal and vertical components for both Merge MVs and AMVP candidates. The average MV components form several ranges in different CU depths for both Merge MVs and AMVP candidates.

3.3 MV range calculation

From the analysis in Section 3.2.1, we know that the average MV ranges of Merge 2N×2N of forward and backward predictions are different from the average MV ranges of AMVP candidates. As a result, we are able to determine either Merge mode or AMVP is suitable for the prediction process of the current PU according to the average MV ranges. Therefore, we propose to average the MVs of Merge 2N×2N which are selected as the best prediction modes from training frames to estimate the MV ranges.

After executing the prediction of Merge/Skip 2N×2N and Inter 2N×2N, we classify the MVs into forward and backward predictions by (3) if Merge mode is selected as the best mode. As $posV_{D\_Li}$ and $negV_{D\_Li}$ shown in (4), the MVs are categorized into

positive or negative directions (*pos* or *neg*), horizontal or vertical component, and backward or forward predictions (*L*0 or *L*1). D indicates a horizontal component (X) or vertical component (Y). Li denotes forward or backward prediction, in which $i = 0$ means forward prediction and $i = 1$ means backward prediction.

$$posV_{DLi} = V_{DLi}, \quad if \;\; V_{DLi} > 0 \quad\quad i \in \{0, 1\}, \quad D \in \{X, Y\}$$
$$negV_{DLi} = -V_{DLi}, \quad if \;\; V_{DLi} < 0 \quad\quad i \in \{0, 1\}, \quad D \in \{X, Y\}$$
$$(4)$$

$$V_{MergeposDLi} = \frac{1}{N} \sum_{n=0}^{N} posV_{DLi_n} \quad\quad i \in \{0, 1\}, \quad D \in \{X, Y\}$$

$$V_{MergenegDLi} = \frac{1}{N} \sum_{n=0}^{N} negV_{DLi_n} \quad\quad i \in \{0, 1\}, \quad D \in \{X, Y\}$$
$$(5)$$

By (5), we can calculate the average positive or negative component length ($V_{Merge\_posD\_Li}$ or $V_{Merge\_negD\_Li}$), where N is the number of the Merge MVs used in statical stage. By (6), we can obtain the MV range of the horizontal component via determining the minimum and maximum lengths of the positive horizontal component ($V_{Merge\_posX\_min}$ and $V_{Merge\_posX\_max}$) and the negative horizontal component ($V_{Merge\_negX\_min}$ and $V_{Merge\_negX\_max}$). The same method in (6) is used to determine the information of the vertical component.

Table 1 Probability distribution for different search ranges [23]

Training sequence	Probability (%)		
	SR = 1	SR = 2	SR = 4
Class A	92.60	98.14	98.28
Class B	90.37	96.58	96.61
Class C	93.41	98.24	98.47
Class D	91.97	98.57	98.59
Class E	96.89	98.89	98.90
Average	93.05	98.08	98.17

Table 2 The configuration settings of experimental environment

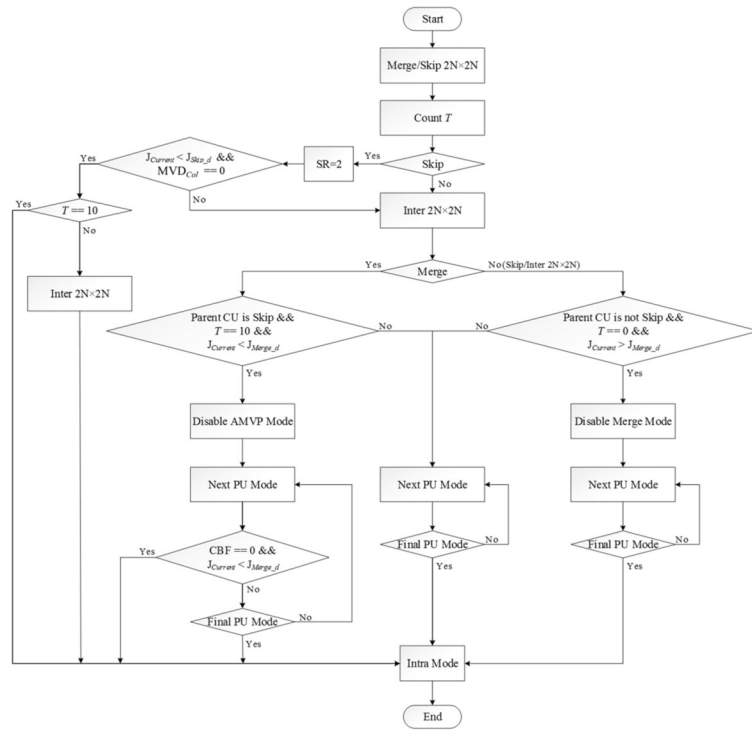| Configurations	Settings
HM version	HM-16.3 [24] and HM-16.4 [25]
Configurations	Random-access
GOPSize	8
IntraPeriod	32
Search method	TZSearch

Fig. 15 The proposed fast PU mode decision

Table 3 The information of the testing sequences

Class	Name		Resolution	Frames	FPS
A	S01	Traffic	2560 × 1600	150	30
	S02	PeopleOnStreet	2560 × 1600	150	30
B	S03	Kimono	1920 × 1080	240	24
	S04	ParkScene	1920 × 1080	240	24
	S05	Cactus	1920 × 1080	500	50
	S06	BasketballDrive	1920 × 1080	500	50
	S07	BQTerrace	1920 × 1080	600	60
C	S08	BasketballDrill	832 × 480	500	50
	S09	BQMall	832 × 480	600	60
	S10	PartyScene	832 × 480	500	50
	S11	RaceHorsesC	832 × 480	300	30
D	S12	BasketballPass	416 × 240	500	50
	S13	BQSquare	416 × 240	600	60
	S14	BlowingBubbles	416 × 240	500	50
	S15	RaceHorses	416 × 240	300	30
E	S16	Vidyo1	1280 × 720	600	60
	S17	Vidyo3	1280 × 720	600	60
	S18	Vidyo4	1280 × 720	600	60
	–	FourPeople	1280 × 720	600	60
	–	Johnny	1280 × 720	600	60
	–	KristenAndSara	1280 × 720	600	60

$$V_{MergeposDmin} = min\{V_{MergeposDLi}\}$$
$$V_{MergeposDmax} = max\{V_{MergeposDLi}\}$$
$$V_{MergenegDmin} = min\{V_{MergenegDLi}\} \quad (6)$$
$$V_{MergenegDmax} = max\{V_{MergenegDLi}\}$$
$$i \in \{0, 1\}, \quad D \in \{X, Y\}$$

$$EV_{posD} = V_{MergeposDmax} - V_{MergeposDmin}$$
$$EV_{negD} = V_{MergenegDmax} - V_{MergenegDmin} \quad (7)$$
$$D \in \{X, Y\}$$

With EV in (7), we hope to extend the MV range obtained by (8), namely by adjusting the value obtained in (6) to make it similar to the MV distribution of the Merge 2N×2N in Section 3.2.1. We also define the length of the component in positive ($posV_D$) or negative ($negV_D$) direction, respectively, as shown in (8) without considering forward and backward predictions. V_D means the value of the horizontal or vertical component. According to (6), (7), and (8), we can define the adjusted MV ranges as shown in (9).

$$posV_D = V_D, \quad if V_D > 0 \qquad D \in \{X, Y\}$$
$$negV_D = -V_D, \quad if V_D < 0 \qquad D \in \{X, Y\} \quad (8)$$

$$V_{MergeposDmin} < posV_D < (V_{MergeposDmax} + EV_{posD})$$
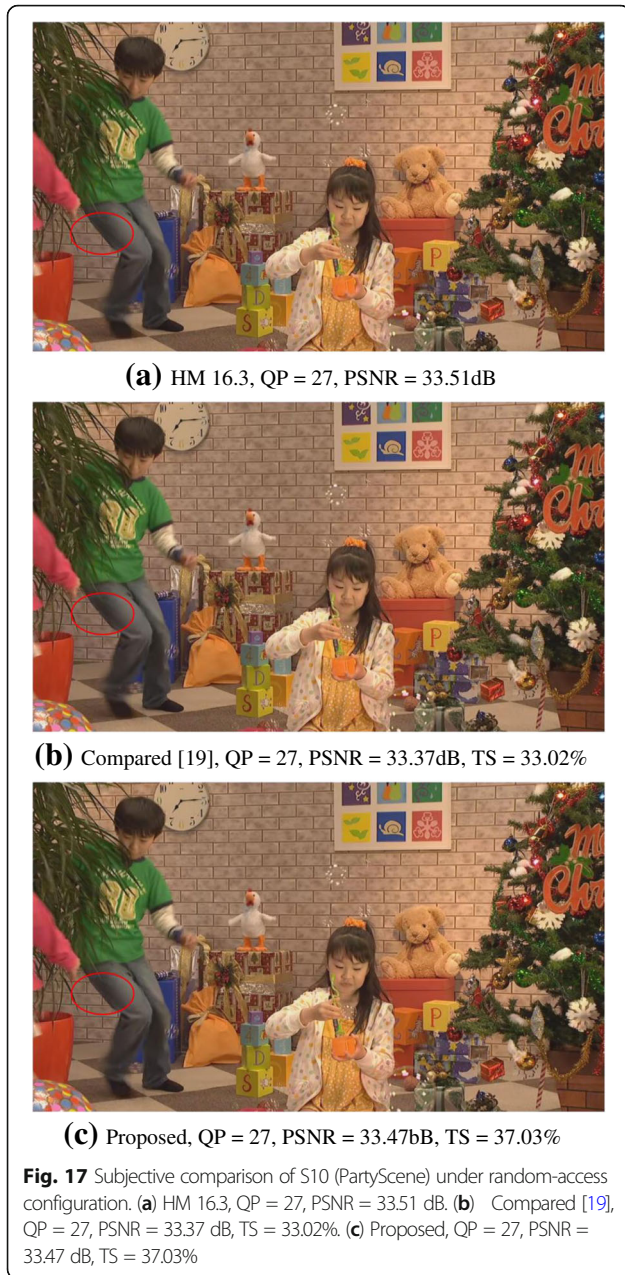$$V_{MergenegDmin} < negV_D < (V_{MergenegDmax} + EV_{negD}) \quad (9)$$
$$D \in \{X, Y\}$$

Figure 13 shows the MV ranges after the extension by (9). Figure 14 shows the comparison between the MV

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 10 of 17

Table 4 Performance comparison between the proposed method and [19] under the random-access configuration

HM 16.3			[19]			Proposed		
Class	Sequence		BDPSNR (dB)	BDBR (%)	TS (%)	BDPSNR (dB)	BDBR (%)	TS (%)
A	S01	Traffic	− 0.02	0.72	56.98	− 0.04	1.24	54.15
	S02	PeopleOnStreet	− 0.03	0.67	27.09	− 0.03	0.75	39.18
		Average	− 0.03	0.70	42.04	− 0.04	1.00	46.67
B	S03	Kimono	− 0.01	0.43	46.86	− 0.04	1.23	59.17
	S04	ParkScene	− 0.02	0.50	53.77	− 0.03	1.03	50.02
	S05	Cactus	− 0.01	0.60	48.18	− 0.02	0.91	50.26
	S06	BasketballDrive	− 0.02	0.68	44.15	− 0.02	1.02	49.31
	S07	BQTerrace	− 0.03	1.68	55.27	− 0.03	1.62	52.67
		Average	− 0.02	0.78	49.65	− 0.03	1.16	52.29
C	S08	BasketballDrill	− 0.01	0.13	35.45	− 0.02	0.50	37.76
	S09	BQMall	− 0.02	0.58	45.97	− 0.02	0.51	43.15
	S10	PartyScene	− 0.01	0.31	38.69	− 0.01	0.20	39.50
	S11	RaceHorsesC	− 0.04	0.99	36.92	− 0.05	1.23	36.36
		Average	− 0.02	0.50	39.26	− 0.02	0.61	39.19
D	S12	BasketballPass	− 0.02	0.46	29.31	− 0.02	0.36	32.87
	S13	BQSquare	− 0.01	0.16	48.65	0.01	− 0.14	42.71
	S14	BlowingBubbles	− 0.02	0.39	39.93	− 0.01	0.20	38.11
	S15	RaceHorses	− 0.04	0.86	22.68	− 0.03	0.58	32.59
		Average	− 0.02	0.47	35.14	− 0.01	0.25	36.57
Total average			− 0.02	0.61	41.99	− 0.02	0.75	43.85
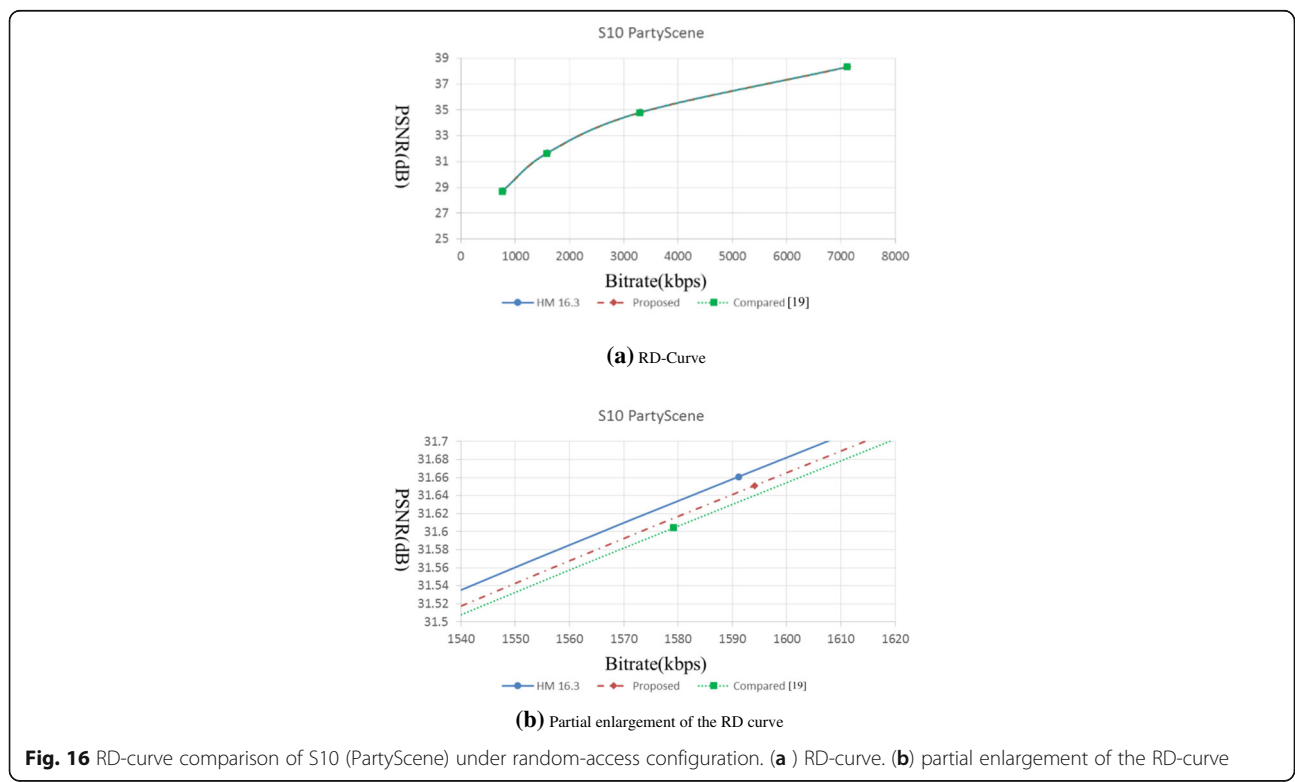
(a) RD-Curve

(b) Partial enlargement of the RD curve

Fig. 16 RD-curve comparison of S10 (PartyScene) under random-access configuration. (**a**) RD-curve. (**b**) partial enlargement of the RD-curve

(a) HM 16.3, QP = 27, PSNR = 33.51dB

(b) Compared [19], QP = 27, PSNR = 33.37dB, TS = 33.02%

(c) Proposed, QP = 27, PSNR = 33.47bB, TS = 37.03%

Fig. 17 Subjective comparison of S10 (PartyScene) under random-access configuration. (**a**) HM 16.3, QP = 27, PSNR = 33.51 dB. (**b**) Compared [19], QP = 27, PSNR = 33.37 dB, TS = 33.02%. (**c**) Proposed, QP = 27, PSNR = 33.47 dB, TS = 37.03%

distribution in Fig. 8 and Fig. 13a for depth 0. It is obvious that the extended MV range is more similar to the statistical result. The proposed MV range is used to verify whether the current MV of the Merge mode is within the range. There are five candidates of the reference location during Merge 2N×2N, including 10 MVs (forward and backward predictions) for Merge 2N×2N candidates. Moreover, the number of MVs within the range is counted to switch between the Merge mode and AMVP in the following prediction. T is defined as the number of the MVs within the MV range and the initialized value is 0. We substitute MV candidates into (8), and then substitute $posV_D$ and $negV_D$ into (9) to make the

judgment of the MV range. If the condition is satisfied, T has 1 added to it except the case that either one of the MV components is 0. For example, there is a Merge candidate MV with values (– 3,2). From (8), the length for the horizontal component in a negative direction ($negV_X$) is 3 and the length for the vertical component in a positive direction ($posV_Y$) is 2. By (10), if the lengths of both components are within the range, T has 1 added to it. In the process, the forward and backward predictions are taken into consideration together. Rather than only calculating the MV length by conventional method, the MV directions and ranges are considered in our approach. For instance, MV (–3,2) and MV (3,2) are with the same length, whereas the MV directions are totally different. The judgment for T is either 0 or 10, in which 0 means none of the acquired MVs is within the MV range and 10 means all of the acquired MVs are within the MV range.

$$
T = T + 1,
$$
$$
if \left\{ \begin{array}{l} V_{MergeposDmin} < posV_D < (V_{MergeposDmax} + EV_{posD}) \\ V_{MergenegDmin} < negV_D < (V_{MergenegDmax} + EV_{negD}) \end{array} \right\}
$$
$$
D \in \{X, Y\}
$$
(10)

3.4 Search range reduction

In Table 1, Tai et al. [23] show the probabilities for different search ranges (SR) which the best motion vector is within the search range if Skip mode is selected as the best mode after executing Merge/Skip 2N×2N. Both of the probabilities of SR = 2 and SR = 4 are higher than 96%, which means the SR can provide accurate prediction results. As a consequence, we set the SR as 2 if Skip mode is selected as the best mode after Merge/Skip 2N×2N.

3.5 Proposed fast PU mode decision algorithm

The flowchart of the proposed fast PU mode decision is depicted in Fig. 15, which is also the execution block in Fig. 5.

The Merge MV range in (8) is updated by the first three frames of each GOP. The MV range of the first three frames of each GOP is inherited from the previous GOP. J_{Skip_d} is the average RD-cost of the Skip mode from (1). J_{Merge_d} is the average RD-cost of the Merge mode from (2). $J_{Current}$ is the RD-cost of the current mode. T is the number of the MVs within the MV range in (9). MVD_{Col} is the length of the collocated MVD in the reference frame. The following descriptions explain the procedure of the proposed fast mode decision in detail. In addition, the switch between AMVP and Merge is only performed at the CTUs excluding left and top boundary of the coding frame.

(a) HM 16.3

(b) Compared [19]   **(c)** Proposed
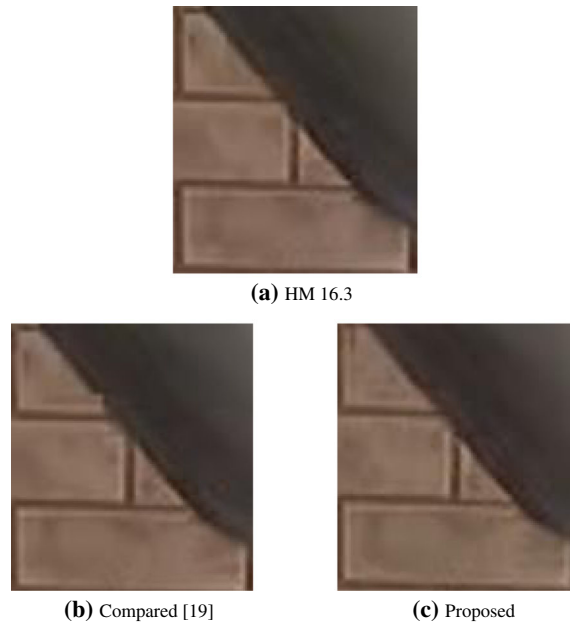
Fig. 18 Partial enlargements of the subjective comparison in the red circled region of Fig. 17. (**a**) HM 16.3. (**b**) Compared [19]. (**c**) Proposed
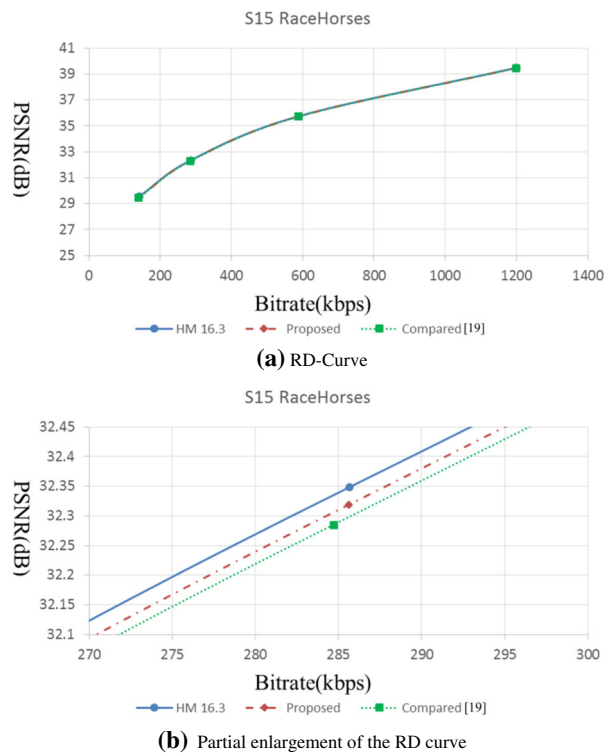
(a) RD-Curve

(b) Partial enlargement of the RD curve

Fig. 19 RD-curve comparison of S15 (RaceHores) under random-access configuration. (**a**) RD-curve. (**b**) partial enlargement of the RD-curve

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 13 of 17

(a) HM 16.3, QP = 27, PSNR = 34.42dB

(b) Compared [19], QP = 27, PSNR = 34.30dB, TS = 16.90%

(c) Proposed, QP = 27, PSNR = 34.37dB, TS = 28.77%

Fig. 20 Subjective comparison of S15 (RaceHores) under random-access configuration. (**a**) HM 16.3, QP = 27, PSNR = 34.42 dB. (**b**) Compared [19], QP = 27, PSNR = 34.30 dB, TS = 16.90%. (**c**) Proposed, QP = 27, PSNR = 34.37 dB, TS = 28.77%
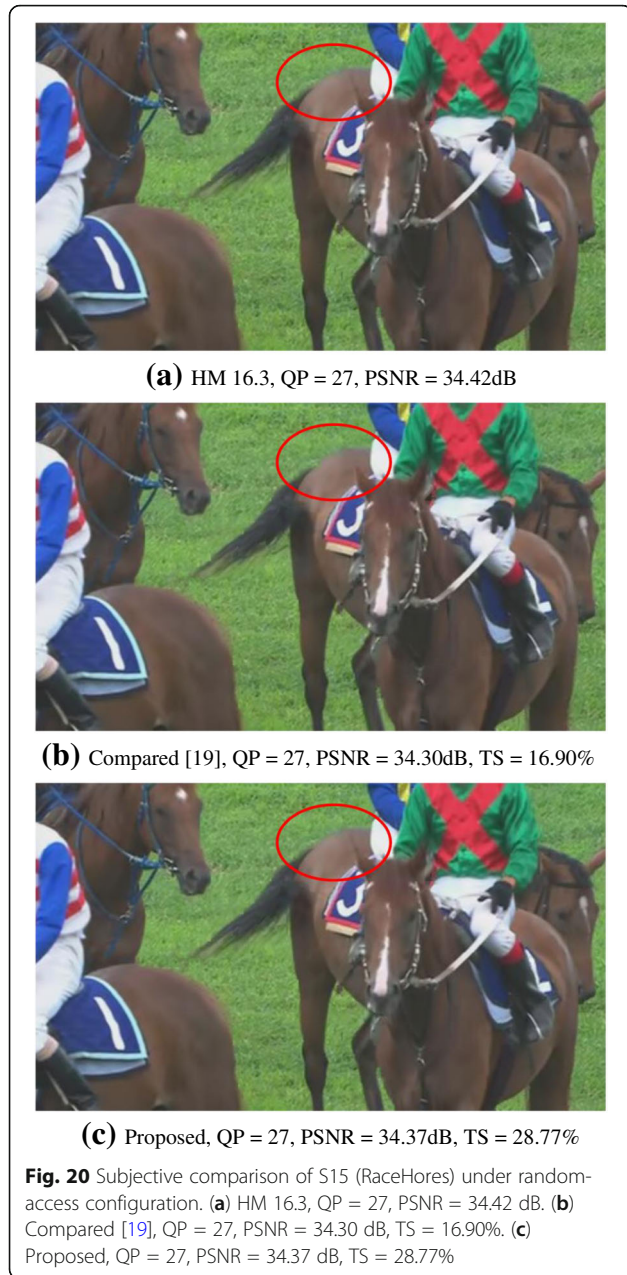
Step 1. Execute Merge/Skip 2N×2N and count T.

Step 2. If Skip mode is selected as the best mode, go to Step 3. Otherwise, go to Step 6.

Step 3. Set SR as 2.

Step 4. If $J_{Current}$ is smaller than J_{Skip_d} and MVD_{Col} is 0, go to Step 5. Otherwise, go to Step 6.

Step 5. If T is 10, go to Step 16. Otherwise, execute Inter 2N×2N and then go to Step 16.

Step 6. Execute Inter 2N×2N.

Step 7. If Merge mode is selected as the best mode, go to Step 8. Otherwise, go to Step 12.

Step 8. If the best mode in the parent CU is Skip mode, T is 10 and $J_{Current}$ is smaller than J_{Merge_d}, go to Step 9. Otherwise, go to Step 14.

Step 9. Disable the procedure of AMVP in the following PU modes.

Step 10. Execute the next PU mode of inter-prediction. If CBF is 0 and $J_{Current}$ is smaller than J_{Merge_d}, go to Step 16.

Step 11. If the current PU mode is the last inter-prediction mode, go to Step 16. Otherwise, go to Step 10.

Step 12. If the best mode in the parent CU is not Skip mode, T is 0 and $J_{Current}$ is larger than J_{Merge_d}, go to Step 13. Otherwise, go to Step 14.

Step 13. Disable the procedure of Merge in the following PU modes.

Step 14. Execute the next PU mode of inter-prediction.

Step 15. If the current PU mode is the last inter-prediction mode, go to Step 16. Otherwise, go to Step 14.

Step 16. Execute intra prediction and select the best mode.
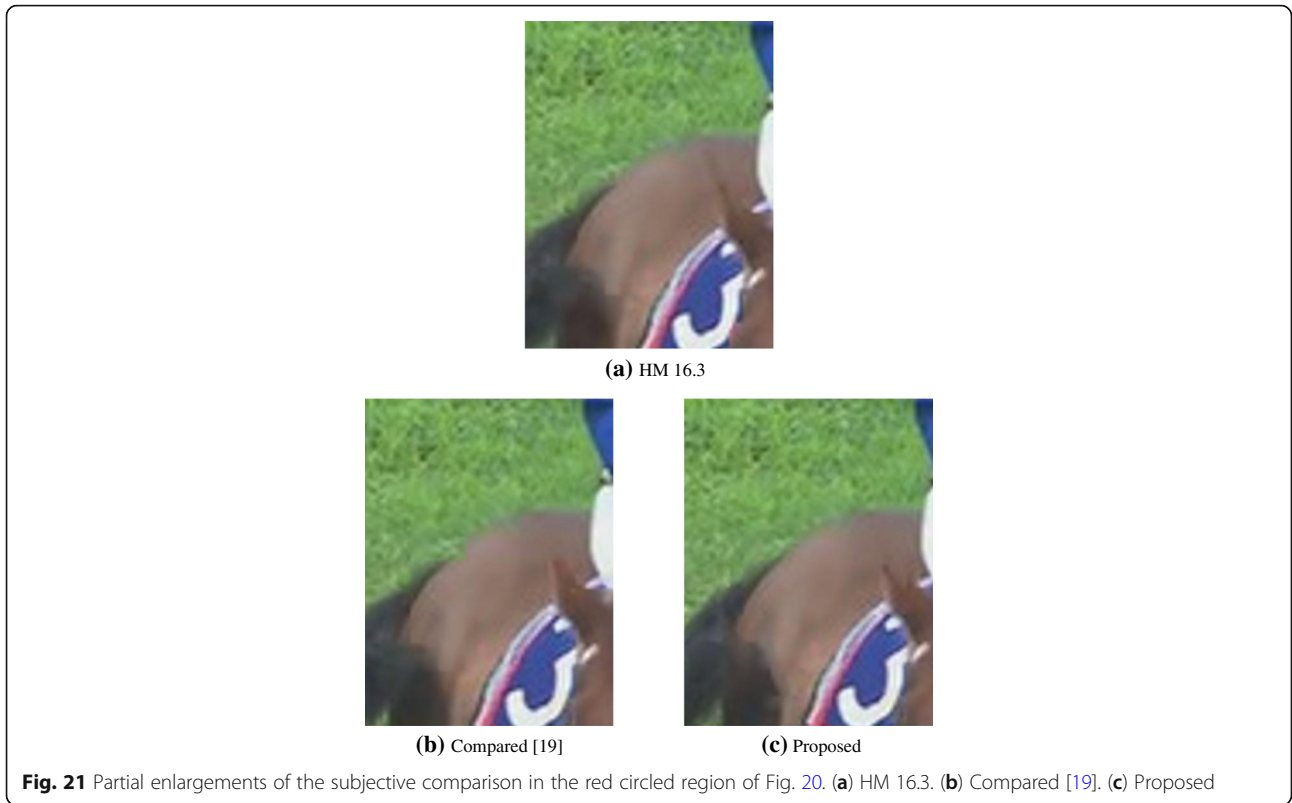
Step 17. Finish the PU mode decision.

4 Experimental results and discussion

We implement the proposed algorithm on HEVC reference software version 16.3 and 16.4 (HM-16.3 [24] and HM-16.4 [25]). The test sequences consist of classes A~E with QP 22, 27, 32, 37 under random-access configuration and default settings. Table 2 tabulates the configuration settings of experimental environment. Table 3 describes the sequence information, including different resolutions, numbers of frames, and frame rates (FPS). We use (11) to compute time-saving and evaluate the coding efficiency by BDBR and BDPSNR [26, 27].

$$\text{TS}(\%) = \frac{\text{Time}_{HM} - \text{Time}_{proposed}}{\text{Time}_{HM}} \times 100(\%) \qquad (11)$$

4.1 Compared to [19] on HM-16.3

First, we implement both of the proposed algorithm and [19] on HM-16.3. From Table 4, the proposed algorithm saves 43.85% of the average coding time, which is better than 41.99% of the average coding time in [19]. The method in [19] is without CU early split, and early terminates the PU prediction only by motion vector and residual, so it will lead to the rising BDBR and less time-saving in the sequences with small resolution and high motion, such as Class D. Figure 16 draws the RD curve comparison of HM.16.3, the proposed method, and [19] for PartyScene (S10) sequence. From the partial enlargements in Fig. (16b), the curve of the proposed

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 14 of 17

Fig. 21 Partial enlargements of the subjective comparison in the red circled region of Fig. 20. (**a**) HM 16.3. (**b**) Compared [19]. (**c**) Proposed

Table 5 Performance comparison with [20, 21] under random-access configuration

HM 16.4		[20]		[21]		Proposed	
Class	Sequence	BDBR (%)	TS (%)	BDBR (%)	TS (%)	BDBR (%)	TS (%)
A	Traffic	1.29	55.19	0.91	59.13	0.94	53.22
	PeopleOnStreet	1.25	24.87	0.93	25.76	0.54	38.25
	Average	1.27	40.03	0.92	42.45	0.74	45.74
B	Kimono	0.84	35.77	1.38	56.44	1.23	58.09
	ParkScene	1.22	52.10	1.32	52.29	0.95	48.60
	Cactus	1.45	45.32	2.73	52.64	0.70	48.44
	BasketballDrive	0.58	38.68	1.94	46.82	0.84	47.12
	BQTerrace	1.04	53.05	1.78	50.71	0.75	51.30
	Average	1.03	44.98	1.83	51.78	0.89	50.71
C	BasketballDrill	0.66	35.63	1.91	41.45	0.63	37.15
	BQMall	1.66	36.79	2.31	43.31	0.52	42.15
	PartyScene	1.11	33.41	0.93	37.17	0.61	37.96
	RaceHorsesC	0.98	25.13	2.22	31.70	0.93	34.06
	Average	1.10	32.74	1.84	38.41	0.67	37.83
D	BasketballPass	1.40	38.23	1.55	33.66	0.40	31.31
	BQSquare	0.91	46.06	0.70	44.41	0.51	41.00
	BlowingBubbles	1.20	32.89	0.76	36.28	0.36	35.90
	RaceHorses	1.35	22.75	1.03	26.56	0.38	30.41
	Average	1.22	34.98	1.01	35.23	0.41	34.65
Total average		1.13	38.39	1.49	42.56	0.69	42.33

Table 6 Performance comparison with CFM [5], ECU [6] and ESD [7] under random-access configuration

HM 16.4			CFM [5]			ECU [6]			ESD [7]			Proposed		
Class	Sequence		BDPSNR (dB)	BDBR (%)	TS (%)	BDPSNR (dB)	BDBR (%)	TS (%)	BDPSNR (dB)	BDBR (%)	TS (%)	BDPSNR (dB)	BDBR (%)	TS (%)
A	S01	Traffic	− 0.022	0.67	49.20	− 0.020	0.63	53.83	− 0.005	0.16	44.00	− 0.031	0.94	53.22
	S02	PeopleOnStreet	− 0.063	1.44	30.54	− 0.025	0.55	21.93	− 0.018	0.40	22.50	− 0.024	0.54	38.25
B	S03	Kimono	− 0.018	0.61	41.45	− 0.012	0.41	44.01	− 0.008	0.26	36.33	− 0.037	1.23	58.09
	S04	ParkScene	− 0.022	0.68	46.80	− 0.014	0.45	50.14	− 0.006	0.19	41.61	− 0.030	0.95	48.60
	S05	Cactus	− 0.017	0.78	40.29	− 0.015	0.72	43.47	− 0.006	0.28	36.15	− 0.015	0.70	48.44
	S06	BasketballDrive	− 0.021	0.95	38.85	− 0.008	0.36	40.31	− 0.008	0.38	33.70	− 0.018	0.84	47.12
	S07	BQTerrace	− 0.011	0.70	45.57	− 0.012	0.75	50.60	− 0.004	0.25	41.42	− 0.012	0.75	51.30
C	S08	BasketballDrill	− 0.029	0.70	33.65	− 0.010	0.25	31.87	− 0.009	0.22	29.12	− 0.026	0.63	37.15
	S09	BQMall	− 0.050	1.31	41.98	− 0.024	0.62	41.04	− 0.013	0.34	35.67	− 0.020	0.52	42.15
	S10	PartyScene	− 0.032	0.75	36.08	− 0.027	0.62	33.03	− 0.011	0.25	30.14	− 0.026	0.61	37.96
	S11	RaceHorsesC	− 0.060	1.64	30.34	− 0.012	0.33	19.59	− 0.017	0.46	20.74	− 0.035	0.93	34.06
D	S12	BasketballPass	− 0.059	1.23	32.08	− 0.025	0.52	24.65	−0.018	0.38	24.98	− 0.019	0.40	31.31
	S13	BQSquare	− 0.028	0.72	44.89	− 0.012	0.32	42.87	− 0.009	0.24	38.93	− 0.020	0.51	41.00
	S14	BlowingBubbles	− 0.033	0.78	37.76	− 0.026	0.62	34.86	− 0.010	0.23	31.55	− 0.015	0.36	35.90
	S15	RaceHorses	− 0.089	1.94	30.00	− 0.026	0.56	17.16	− 0.017	0.38	19.54	− 0.017	0.38	30.41
E	S16	Vidyo1	− 0.017	0.51	55.32	− 0.005	0.16	66.05	− 0.004	0.12	51.69	− 0.012	0.38	63.19
	S17	Vidyo3	− 0.026	0.85	54.20	− 0.006	0.21	64.65	− 0.007	0.23	50.64	− 0.010	0.35	62.97
	S18	Vidyo4	− 0.015	0.47	54.44	− 0.002	0.06	65.78	− 0.005	0.16	51.26	− 0.018	0.60	64.22
	–	FourPeople	− 0.013	0.36	53.33	− 0.005	0.13	65.41	− 0.002	0.06	50.56	− 0.013	0.35	59.60
	–	Johnny	− 0.014	0.55	56.90	− 0.001	0.03	70.56	− 0.004	0.16	53.65	− 0.010	0.39	69.19
	–	KristenAndSara	− 0.015	0.47	54.83	− 0.003	0.09	66.82	− 0.003	0.09	51.39	− 0.013	0.43	65.27
Total Average			− 0.031	0.86	43.26	− 0.014	0.40	45.17	− 0.009	0.25	37.88	− 0.020	0.61	48.54
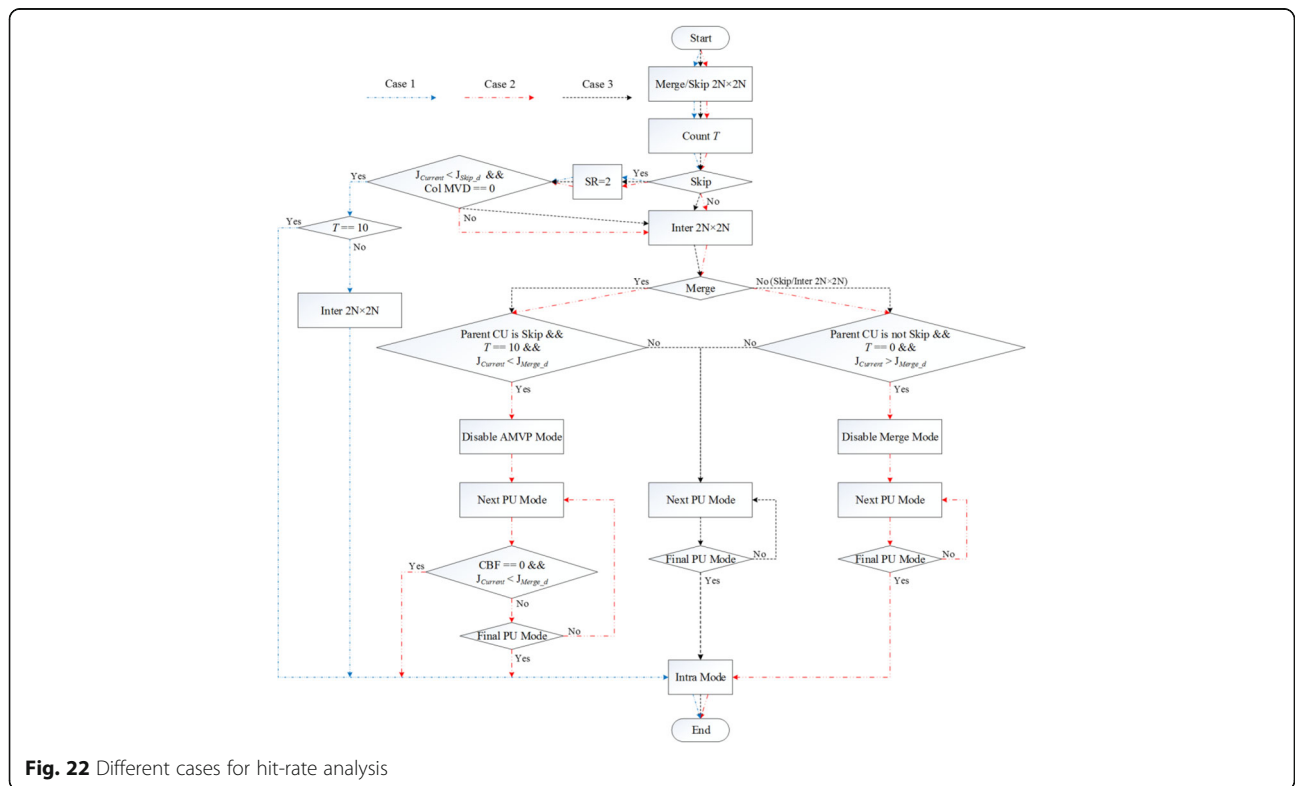
Fig. 22 Different cases for hit-rate analysis

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 16 of 17

Table 7 Hit-rate analysis of the proposed method

Sequence	Case 1 hit (%)	Case 2 hit (%)
S01	99.16	80.92
S02	97.73	83.76
S03	99.55	84.34
S04	98.88	80.07
S05	99.01	84.06
S06	99.38	81.33
S07	98.74	81.34
S08	98.95	80.86
S09	98.51	78.59
S10	97.84	80.70
S11	97.54	79.76
S12	98.16	82.96
S13	98.16	82.96
S14	97.02	79.65
S15	96.49	78.47
Average	98.34	81.32

method is closer to HM-16.3 than [19], which indicates the better coding efficiency. The subjective comparison of the sequence PartyScene (S10) is illustrated in Fig. 17, while Fig. 18 is the partial enlargement of the red circled region in Fig. 17. There are saw edges in the result of [19], whereas the result of the proposed method reconstructs smoother edges with closer image quality to HM-16.3. The RD-curve and subjective comparison for the sequence RaceHores(S15) are shown in Fig. 19, Fig. 20, and Fig. 21, which conclude the same observation.

4.2 Compared to [20, 21] on HM-16.4

We also implement the proposed algorithm on HM-16.4 and compare the experimental results to [20, 21]. As shown in Table 5, the BDBR and time-saving of our method are 0.69% and 42.33%, respectively, while corresponding results for [20] are 1.13% and 38.39%, respectively. The BDBR and time-saving of the proposed algorithm outperforms [20] because the early termination algorithms of CU and PU in [20] use less information to make decisions. The method in [21] focuses on early terminations of CU and PU. However, our method proposes a novel PU decision scheme according to MV range and combines an efficient CU decision algorithm in [22]. As a result, the BDBR in [21] is 1.49%, while that of the proposed method is 0.69%, which is almost half of [21] with the time-saving similar to [21]. In the higher-resolution sequence S02 (PeopleOnStreet), the BDBR and time-saving of our method are 0.54% and 38.25%, respectively, whereas [21] has inferior BDBR and time-saving results of 0.93% and 25.76%, respectively.

4.3 Compared to the fast coding configuration of HM-16.4

The fast coding configuration is optional in HEVC encoder and also provides significant coding time reduction. We also make a comparison to the existing and available tools in HEVC encoder as demonstrated in Table 6, including CFM [5], ECU [6], and ESD [7]. From Table 6, the proposed strategy accelerates the average coding time by 48.54% with BDPSNR and BDBR degradation by only − 0.02 dB and 0.61%, respectively. The time-saving of the proposed algorithm outperforms both ECU and ESD. Moreover, the proposed algorithm is superior to CFM for these three target-evaluating indexes. Owing to that the proposed algorithm is designed for sequences with high resolution, it can be noticed that the high-resolution sequences, such as Class A, Class B, and Class E, provide greater time-saving and remain the BDBR at the same time. In spite of the less time-saving in the low-resolution sequences of Class D, we receive lower BDBR degradation as well.

4.4 Availability test for the proposed method

To further verify the availability, we calculate the hit rates of the proposed method under three cases, which are the probabilities that the prediction results of the proposed method are the same with the ground truth from the original encoder. Case 1 is the application of Skip mode and Case 2 indicates the switch between Merge and AMVP. Case 3 is without neglecting any PU modes, Merge, or AMVP, so it is not necessary to calculate the hit rate under Case 3. The detailed notes are illustrated in Fig. 22 and the hit rates are shown in Table 7. We can perceive that the average hit rate under Case 1 is up to 98.34% and the average hit rate under Case 2 is up 81.32%, which specify the high validity of the proposed method.

5 Conclusions

In relation to the demand for high-resolution and quality videos with great compression efficiency, HEVC/H.265 is standardized as the newest video coding standard. It provides advanced coding tools; however, it increases the coding complexity at the same time. The coding complexity of HEVC mainly results from the executing processes of the CU and PU. We find the appropriate Merge MV range to distinguish the Merge mode and AMVP mode to accelerate the PU mode decision. The method in this paper verifies whether the MVs of Merge 2N×2N candidates are within the MV range, early terminates the prediction process and disables the Merge or AMVP modes of following prediction modes. The experimental results show that the proposed algorithm can reduce the average coding time by 48.54% while

Lin *et al. EURASIP Journal on Image and Video Processing* (2018) 2018:99

Page 17 of 17

increasing the average BDBR by only 0.61% on HM-16.4. The proposed method speeds up the coding process and maintains the video quality simultaneously.

Availability of data and materials
Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Authors' contributions
KML and MJC conceived and designed the study. KML performed the experiments. KML, JRL, MJC, CHY, and CAL wrote the paper and edited the manuscript. All authors read and approved the manuscript.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details
[1]Department of Electrical Engineering, National Dong Hwa University, Hualien, Taiwan. [2]Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan. [3]Department of Electrical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan.

References
1. G.J. Sullivan, J.R. Ohm, W.J. Han, T. Wiegand, Overview of the high efficiency video coding (HEVC) standard. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1649–1668 (2012)
2. T. Wiegand, G.J. Sullivan, G. Bjøntegaard, A. Luthra, Overview of the H.264/AVC video coding standard. IEEE Trans. Circuits Syst. Video Technol. **13**(7), 560–576 (2003)
3. J.L. Lin, Y.W. Chen, Y.W. Huang, S.M. Lei, Motion vector coding in the HEVC standard. IEEE J. Sel. Topics Signal Process. **7**(6), 957–968 (2013)
4. P. Helle, S. Oudin, B. Bross, D. Marpe, M.O. Bici, K. Ugur, J. Jung, G. Clare, T. Wiegand, Block merging for quadtree-based partitioning in HEVC. IEEE Trans. Circuits Syst. Video Technol. **22**(12), 1720–1731 (2012)
5. R.H. Gweon, Y.L. Lee, J. Lim, in *Document: JCTVC-F045*. Early termination of CU encoding to reduce HEVC complexity (2011)
6. K. Choi, E.S. Jang, in *Document: JCTVC-F092*. Coding tree pruning based CU early termination (2011)
7. J. Yang, J. Kim, K. Won, H. Lee, B. Jeon, in *Document: JCTVC-G543*. Early skip detection for HEVC (2011)
8. A. BenHajyoussef, T. Ezzedine, A. Bouallègue, Gradient-based pre-processing for intra prediction in high efficiency video coding. EURASIP J. Image Video Process. **2017**(9), 1–13 (2017)
9. M. Zhang, X. Zhai, Z. Liu, Fast and adaptive mode decision and CU partition early termination algorithm for intra-prediction in HEVC. EURASIP J. Image Video Process. **2017**(86), 1–11 (2017)
10. X. Shen, L. Yu, CU splitting early termination based on weighted SVM. EURASIP J. Image Video Process. **2013**(4), 1–11 (2013)
11. X. Huang, P. An, Q. Zhang, Efficient AMP decision and search range adjustment algorithm for HEVC. EURASIP J. Image Video Process. **2017**(75), 1–15 (2017)
12. X.Q. Huang, C.H. Kuo, C.P. Mao, Y.S. Ciou, in *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference(APSIPA)*. Adaptive depth search range for HEVC coding unit size selection (2014)
13. H.M. Yoo, J.W. Suh, Fast coding unit decision based on skipping of inter and intra prediction units. Electron. Lett. **50**(10), 750–752 (2014)
14. S. Yang, H. Lee, H.J. Shim, B. Jeon, in *Proceedings of the 2013 11th IEEE IVMSP Workshop*. Fast inter mode decision process for HEVC encoder (2013)
15. X. Jiang, T. Song, W. Shi, L. Wang, T. Shimamoto, in *Proceedings of the IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW)*. Merge prediction algorithm for adaptive parallel (2015), pp. 310–311
16. S.H. Yang, K.S. Huang, HEVC fast reference picture selection. Electron. Lett. **51**(25), 2109–2111 (2015)
17. L. Shen, Z. Zhang, Z. Liu, Adaptive inter-mode decision for HEVC jointly utilizing inter-level and spatiotemporal correlations. IEEE Trans. Circuits Syst. Video Technol. **24**(10), 1709–1722 (2014)
18. H.L. Tan, C.C. Ko, S. Rahardja, Fast coding quad-tree decisions using prediction residuals statistics for high efficiency video coding (HEVC). IEEE Trans. Broadcast. **62**(1), 128–133 (2016)
19. Z. Pan, S. Kwong, M.T. Sun, J. Lei, Early merge mode decision based on motion estimation and hierarchical depth correlation for HEVC. IEEE Trans. Broadcast. **60**(2), 405–412 (2014)
20. J. Wu, B. Guo, J. Hou, Y. Yan, J. Jiang, in *Proceedings of the IEEE International Conference on Imaging Systems and Techniques (IST)*. A fast CU encoding scheme based on the joint constraint of best and second-best PU modes for HEVC inter coding (2015)
21. S. Ahn, B. Lee, M. Kim, A novel fast CU encoding scheme based on spatiotemporal encoding parameters for HEVC inter coding. IEEE Trans. Circuits Syst. Video Technol. **25**(3), 422–435 (2015)
22. Y.D.W. MJ Chen, C.H. Yeh, K.M. Lin, S.D. Lin, Efficient CU and PU decision based on motion information for inter-prediction of HEVC. IEEE Trans. Ind. Informat. Early Access (2018)
23. K.H. Tai, M.Y. Hsieh, M.J. Chen, C.Y. Chen, C.H. Yeh, A fast HEVC encoding method using depth information of collocated CUs and RD cost characteristics of PU modes. IEEE Trans. Broadcast. **63**(4), 680–692 (2017)
24. JCT-VC HEVC Reference Software Version HM 16.3, available at https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.3/
25. JCT-VC HEVC Reference Software Version HM 16.4, available at https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.4/
26. G. Bjøntegaard, *Calculation of average PSNR differences between RD curves* (ITU-T SG16/Q6 Document, VCEG-M33, Austin, 2001)
27. G. Bjøntegaard, *Improvements of the BD-PSNR model, ITU-T SG16/Q6, Document* (VCEG-AI11, Berlin, 2008)