Fast CU Encoding Schemes Based on Merge Mode and Motion Estimation for HEVC Inter Prediction

Jinfu Wu[1], Baolong Guo[1], Jie Hou[1], Yunyi Yan[1] and Jie Jiang[2]
[1]School of Aerospace Science and Technology, Xidian University
Xi'an, Shaanxi 710071, China
[2]Ocean Electronic Lab, Hangzhou Dianzi University, Hangzhou 310018, China
[e-mail: wjf505@gmail.com, {blguo, yyyan}@xidian.edu.cn,
jie.hou.xdu@gmail.com, jjgirl2008@126.com]
*Corresponding author: Jinfu Wu

Abstract

The emerging video coding standard High Efficiency Video Coding (HEVC) has shown almost 40% bit-rate reduction over the state-of-the-art Advanced Video Coding (AVC) standard but at about 40% computational complexity overhead. The main reason for HEVC computational complexity is the inter prediction that accounts for 60%-70% of the whole encoding time. In this paper, we propose several fast coding unit (CU) encoding schemes based on the Merge mode and motion estimation information to reduce the computational complexity caused by the HEVC inter prediction. Firstly, an early Merge mode decision method based on motion estimation (EMD) is proposed for each CU size. Then, a Merge mode based early termination method (MET) is developed to determine the CU size at an early stage. To provide a better balance between computational complexity and coding efficiency, several fast CU encoding schemes are surveyed according to the rate-distortion-complexity characteristics of EMD and MET methods as a function of CU sizes. These fast CU encoding schemes can be seamlessly incorporated in the existing control structures of the HEVC encoder without limiting its potential parallelization and hardware acceleration. Experimental results demonstrate that the proposed schemes achieve 19%-46% computational complexity reduction over the HEVC test model reference software, HM 16.4, at a cost of 0.2%-2.4% bit-rate increases under the random access coding configuration. The respective values under the low-delay B coding configuration are 17%-43% and 0.1%-1.2%.

**Keywords:** High Efficiency Video Coding (HEVC), inter prediction, mode decision, early termination

1. Introduction

The latest international video coding standard, High Efficiency Video Coding (HEVC), has been established by the Joint Collaborative Team on Video Coding (JCT-VC) under the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG) standardization organizations [1, 2]. The main goal of HEVC is to achieve 50% or more bit-rate over H.264/Advanced Video Coding (AVC) for equal perceptual video quality [2-4]. The final draft of the HEVC standard has been approved in January 2013.

HEVC adopts the conventional hybrid video coding scheme (intra/inter prediction, 2-D transform coding, and entropy coding) used in the prior video coding standards since H.261. To improve the coding efficiency significantly, many advanced tools are introduced in HEVC. Among these, HEVC adopts a hierarchical coding structure consisting of coding unit (CU), prediction unit (PU), and transform unit (TU) [2]. **Fig. 1** shows an example of the hierarchical coding structure of HEVC.
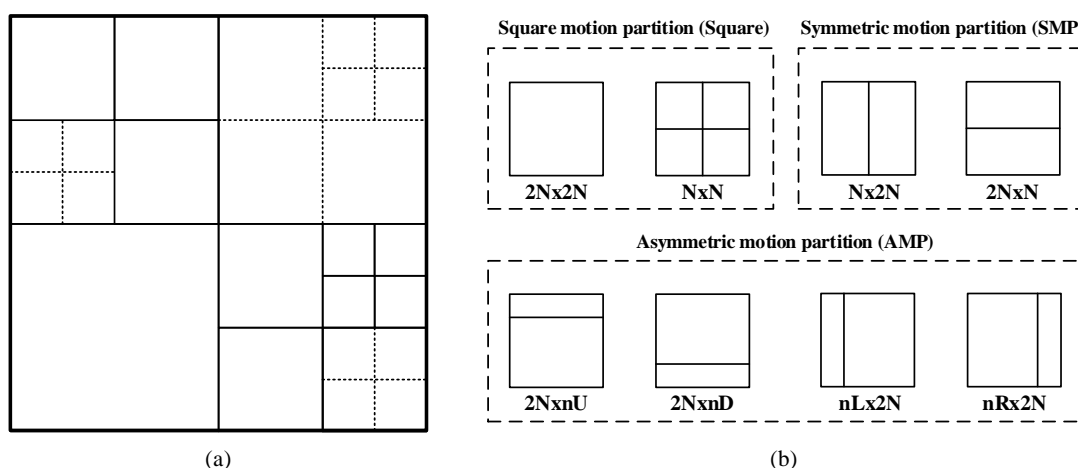
Fig. 1. Hierarchical coding structure of HEVC. (a) Subdivision of a coding tree unit into CUs and TUs. Solid lines indicate CU boundaries and dotted lines indicate TU boundaries. (b) All PU modes for a CU.

As shown in **Fig. 1**, every picture in a video sequence is partitioned into a group of coding tree units (CTUs), each which consists of one luma coding tree block (CTB), two corresponding chroma CTBs, and associated syntax elements. According to the quadtree syntax, a CTU allows to be split into smaller coding units (CUs) based on the signal characteristics of the region that is covered by the CTU. The CU is the basic unit for intra/inter prediction, whose size can be defined as $2N \times 2N$, $N \in \{4, 8, 16, 32\}$. For a $2N \times 2N$ CU three PU mode sets are enabled: square motion partition (Square), symmetric motion partition (SMP), and asymmetric motion partition (AMP) [16]. Specifically, Square set contains two PU modes, $2N \times 2N$ and $N \times N$, the latter one is only allowed when the CU size is equal to the minimum allowed CU size; SMP set has two PU modes $N \times 2N$ and $2N \times N$; AMP set includes four PU modes, $2N \times nU$, $2N \times nD$, $nL \times 2N$, and $nR \times 2N$, where $2N \times nU$ ($2N \times nD$) and $nL \times 2N$ ($nR \times 2N$) indicate two PUs with 1:3 (3:1) partition in vertical and horizontal directions, respectively. An additional PU mode known as Merge mode

is introduced for inter prediction to derive the motion information from spatially or temporally neighboring PUs [5]. TUs are used for the transform coding in a behavior referred as residual quadtree (RQT), which is a nested quadtree structure rooted to CUs. The allowed TU sizes in HEVC are from 32×32 down to 4×4 [2, 6-8].

The flexible coding structure of HEVC is the primary factor for the coding gain as it can be efficiently adjusted to different texture of the picture. However, it causes the majority of the computational complexity, and the inter prediction in HEVC even accounts for 60%-70% of the whole encoding time [6, 7]. In the HEVC inter prediction, the encoder needs to determine the best combination of CU sizes and PU modes from all the possible combinations for each CTU according to the minimization of the Lagrangian cost function [4, 9],

$$c^* = \arg \min_{c \in C} D(c) + \lambda \cdot R(c) \tag{1}$$

where C is the set of all the possible combinations of CU sizes and PU modes for each CTU; $D(c)$ represents the sum of squared differences (SSD) between the original CTU o and its reconstructed CTU o' that is obtained by coding the CTU o with the combination c; $R(c)$ denotes the number of bits needed for encoding the CTU o with the combination c; λ is the Lagrangian multiplier. Therefore, this brute-force method costs high computational complexity, and fast algorithms for HEVC inter prediction are very desirable for real-time implementation of HEVC encoders.

In this paper, we analyze the relationships between the HEVC inter prediction and the Merge mode as well as motion estimation information, and propose an early Merge mode decision method based on motion estimation (EMD) and a Merge mode based early termination method (MET). To provide a better balance between computational complexity and coding efficiency, several fast CU encoding schemes are surveyed according to the rate-distortion-complexity (RDC) characteristics of EMD and MET methods as a function of CU sizes.

The remainder of this paper is organized as follows. Section 2 presents a brief literature review of fast algorithms for HEVC inter prediction. Section 3 describes the motivations and details of the proposed EMD and MET methods. Experiments are carried out and analyzed in Section 4 to demonstrate the effectiveness of the proposed methods. A survey of several fast CU encoding schemes based on the proposed EMD and MET methods is given in Section 5. Section 6 concludes the paper.

2. Background and Related Works

Recently, many fast algorithms have been proposed to reduce the computational complexity caused by the HEVC inter prediction. During the HEVC standardization, the HEVC test model reference software (HM) had adopted three optional fast algorithms to speed up its inter prediction: early CU termination (ECU) [10], early skip detection (ESD) [11], and coded block flag (CBF) fast mode (CFM) [12]. The CBF information and Skip mode are newly introduced in HEVC. Specifically, the CBF specifies whether there exists non-zero transform efficient inside a CU, while the Skip mode is a special case of the Merge mode when all CBFs of a CU are equal to zero, in this case no prediction residual is coded. In ECU, if the best PU mode of current CU was Skip mode, the PU mode decisions for smaller CUs were skipped; When ESD was enabled, the evaluation of $2N \times 2N$ PU mode was conducted before the Merge mode to discover the motion information and CBFs for current CU. If the $2N \times 2N$ PU contained no supplementary motion information and its CBF was equal to zero, the Skip mode

was early selected as the best PU mode for current CU, followed by the PU mode decisions for smaller CUs; In CFM, all the PU modes (Square, SMP, and AMP) were evaluated in order for current CU. Once the CBF of an evaluated PU mode was equal to zero, then this PU mode was selected as the best PU mode for current CU, followed directly by the PU mode decisions for smaller CUs.

Shen *et al*. proposed a fast CU size decision method and a fast PU mode decision method for inter prediction in [13] and [14], respectively. In [13], an adaptive CU depth range algorithm was exploited to skip the evaluations of those CU sizes rarely used in the previously coded neighboring CUs. In addition, three early termination methods based on motion homogeneity, rate-distortion (RD) cost, and Skip mode, respectively, were proposed to skip the motion estimation on unnecessary CU sizes. These three supplementary information were also utilized to skip the evaluations of unnecessary PU modes for CUs in [14]. However, all these proposed fast algorithms were inefficient for encoding the image regions with inhomogeneous motion activities. Moreover, they had a limitation for parallel processing due to their spatial and temporal dependability of neighboring CUs.

Xiong *et al*. [15] proposed a fast CU size selection method based on pyramid motion divergence (PMD). The PMD was represented as the variances of the pixel motion vectors (MVs) of current CU and its corresponding four sub-CUs. The best CU size of current CU was obtained by determining whether its k most similar CUs (in terms of PMD) were with a same CU size. However, this method required additional hardware implementation to calculate pixel-wise optical flows for HEVC encoders.

Vanne *et al*. [16] proposed a series of efficient PU mode decision schemes based on the rate-distortion-complexity (RDC) characteristic analysis of SMP and AMP mode decisions for different CU sizes. For each CU size, the evaluations of SMP and AMP modes were conditionally conducted according to two different preconditions: neither SMP nor AMP modes was evaluated when Merge or Skip mode yielded a smaller RD cost than $2N \times 2N$ PU mode under the joint precondition; For the distributed precondition, the candidate SMP and AMP modes were selectively tried by comparing the RD costs of Skip mode, $2N \times 2N$ PU mode, and Merge mode.

Ahn *et al*. [17] studied the relationships between the picture texture and a spatial encoding parameter, sample adaptive offset (SAO), and proposed a SAO based fast CU size decision method. The SAO is a newly-adopted approach in HEVC to reduce pixel distortion by adding an offset value to each pixel. Additional temporal encoding parameters, such as motion vector and CBFs were also utilized to categorize current CU into a simple or complex motion region. Then, the CU size decision for current CU was performed differently according to its category. However, the proposed SAO based fast CU size decision had a limitation for parallelization in hardware implementation due to the fact that the SAO encoding parameter for current CU was obtained from its temporally collocated CUs.

Pan *et al*. [18] proposed an early Merge mode decision method based on the motion and CBF information. For a 64×64 CU, the Merge mode was early selected as its best PU mode when its motion vector as well as CBF were equal to zero; For the smaller CUs covered by 64×64 CUs, according to the spatial correlations, an additional condition was provided to early determine the Merge mode as their best PU modes: the best PU mode of the 64×64 CU was the Merge mode.

Lee *et al*. [19] proposed a fast CU size decision algorithm based on statistical analysis of the distributions of the optimal CU sizes as well as the PU modes. First, an early Skip mode decision was performed if the RD cost of the Skip mode was less than an adaptive threshold. Second, different thresholds were assigned for a CU skip estimation method and an early CU

termination method to determine the CU size range at an early stage. This algorithm, however, required a periodic update of thresholds depending on various characteristics of video sequences. Therefore, there might be some limitation on computational complexity reduction.

As mentioned above, the Merge mode and picture texture parameters, e.g., CBF and MV, have been validated to have a significant influence on fast PU mode decision. However, the relationships between the Merge mode and CU size decision are not fully exploited. In this paper, we analyze the relationships between the HEVC inter prediction and the Merge mode as well as motion estimation information, and propose an early Merge mode decision method based on motion estimation (EMD) and a Merge mode based early termination method (MET). To provide a better balance between computational complexity and coding efficiency, several fast CU encoding schemes are surveyed according to the RDC characteristics of EMD and MET methods as a function of CU sizes.

3. Proposed Methods Based on Merge Mode and Motion Estimation

3.1 Motivation

It is reported that the inter prediction in HEVC improves the coding efficiency significantly with a large computational complexity overhead, as the best combination of CU sizes and PU modes is determined by exploring all the candidates according to (1). **Fig. 2** shows the mode decision process of the HEVC inter prediction.
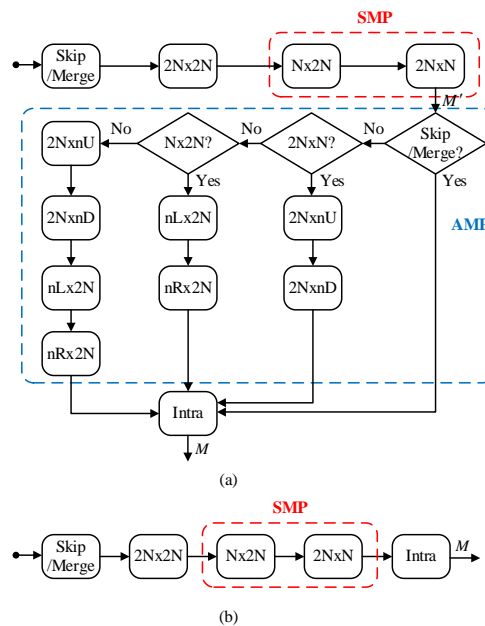
Fig. 2. Mode decision of the HEVC inter prediction. (a) $N \in \{8, 16\}$. (b) $N \in \{4, 32\}$.

It can be observed from **Fig. 2**, for a $2N \times 2N$ CU, the Skip, Merge, Square, and SMP modes are evaluated individually to select a best interim mode M' among them. If $N \in \{8, 16\}$, M' is used to assign a mode set for the subsequent AMP mode decision [**Fig. 2(a)**]. The mode set contains none, half, or all of the AMP modes when M' belongs to {Skip, Merge}, or to SMP modes, or is selected as $2N \times 2N$ PU mode, respectively. After the AMP

mode evaluations, the HEVC encoder evaluates the Intra mode and resolves the best mode M. The AMP modes evaluations are disabled to avoid unavailable block dimensions and achieve a low complexity when $N \in \{4, 32\}$, so the flowchart of mode decision for $N \in \{4, 32\}$ is simplified and depicted in **Fig. 2(b)** [16].

Among the numerous PU modes, the Merge mode achieves good coding efficiency and costs little computational complexity, such that it has been exploited to speed up the mode decision of HEVC inter prediction [10, 16, 18, 19]. The Merge mode is especially suitable for encoding HD and Ultra-HD videos, where exist substantial amounts of background and static regions.

In addition, two picture texture parameters including CBF and MV are useful for improving the inter prediction accuracy. The CBF specifies whether there exists non-zero transform coefficient inside a CU. In video coding, the CUs in background or static regions tend to be encoded in a larger PU size such that their predicted residuals have high probabilities to be transformed and quantized to zeros [11, 12]. Therefore, the CBF information is highly related to the picture texture, and can be used for fast inter prediction.

The motion vector (MV) of a CU is defined as,

$$MV = \frac{1}{k} \sum_{(i,j),(m,n) \in \theta} |x_i - x_m| + |y_j - y_n| \tag{2}$$

where (i, j) and (m, n) represent the initial search point and the final best search point in the motion estimation (ME) process, respectively; k is the number of the reference frame list for encoding the CU with $2N \times 2N$ PU mode; θ denotes the best reference frame in each reference frame list. When $MV = 0$, the CU to be encoded is likely to be inside a region with a slow motion or motionless content [20, 21]. This feature of MV can also be used to speed up the inter prediction.

Table 1. Prediction accuracy distribution of using the Merge mode and two picture texture parameters for fast inter prediction (%)

Sequence	Resolution	M = Merge				M = Merge, CBF = 0, and MV = 0			
		CU0	CU1	CU2	CU3	CU0	CU1	CU2	CU3
Traffic	2560x1600	94.21	96.99	98.67	99.20	98.28	99.35	99.83	99.96
ParkScene	1920x1080	93.22	95.75	97.69	98.40	98.09	99.28	99.79	99.93
BasketballDrill	832x480	89.24	93.53	95.86	96.21	98.00	98.99	99.75	99.95
BQSquare	416x240	91.22	95.79	98.22	99.35	95.63	98.54	99.60	99.88
FourPeople	1280x720	95.87	97.90	99.29	99.60	98.82	99.52	99.90	99.98
Average		92.85	95.99	97.95	98.55	97.76	99.14	99.77	99.94

To validate the effectiveness of using the Merge mode and these two picture texture parameters (CBF and MV) for fast inter prediction, extensive experiments are conducted to analyze the individual prediction accuracies of choosing the Merge mode as the best PU mode with or without considering the conditions: CBF = 0 and MV = 0. Five video sequences with different resolutions and picture texture are tested under the HEVC test model reference software, HM 16.4 [22]: "Traffic" (2560 × 1600), "ParkScene" (1920 × 1080), "BasketballDrill" (832 × 480), "BQSquare" (416 × 240), and "FourPeople" (1280 × 720). Among these video sequences, "Traffic" has complex backgrounds and medium motions.

"ParkScene" and "BQSquare" are with moderate motion activities. "BasketballDrill" has fast moving objects. "FourPeople" has a simple background, and the objects move slowly. The experiments are performed under the JCT-VC common test conditions [23]: each video sequence is encoded under both random access (RA) and low-delay B (LB) coding configurations; four quantization parameters (QPs) of 22, 27, 32, and 37 are tested; rate distortion optimization quantization (RDOQ) is enabled; search range of ME is set to 64; the minimum and maximum CU sizes are specified as 8 and 64, respectively. The prediction accuracies are surveyed as a function of CU sizes in **Table 1** (CU0~CU3 represent $64 \times 64 \sim 8 \times 8$ CU sizes, respectively), because the picture texture influences the inter prediction differently for various CU sizes.

From **Table 1** it can be observed that, no matter whether the picture texture parameters are considered or not, the prediction accuracies of selecting the Merge mode as the best PU mode increase as a function of CU sizes. The prediction accuracies without considering the picture texture parameters are on average 92.85%, 95.99%, 97.95%, and 98.55% for CU0, CU1, CU2, and CU3, respectively. The respective values for the case with CBF = 0 and MV = 0 are 97.76%, 99.14%, 99.77%, and 99.94%. These values demonstrate that it is reasonable to early determine the Merge mode as the best PU mode for all kinds of CU sizes. In addition, the picture texture parameters are effective to improve the fast inter prediction accuracy when CBF = 0 and MV = 0. Even for the video sequence with fast motions, e.g., "BasketballDrill", more than 98% (up to 99.95%) CUs are accurately encoded in the Merge mode when considering the picture texture parameters. Based on these analyses, an early Merge mode decision method based on motion estimation (EMD) and a Merge mode based early termination method (MET) are proposed in the following sections.

3.2 Early Merge Mode Decision Method Based on Motion Estimation (EMD)

It is reported that disabling the SMP and AMP modes entirely in the mode decision achieves around 60% computational complexity reduction with an unacceptable high bit-rate penalty (around 4%) [16]. But if the SMP and AMP modes can be disabled appropriately, a desirable balance between computational complexity and coding efficiency will be achieved.

As we mentioned in **Section 3.1**, the Merge mode has a high likelihood to be chosen as the best PU mode for all kinds of CU size, and the picture texture parameters (CBF and MV) are effective to improve the prediction accuracy. Therefore, if the Merge mode is selected accurately as the best PU mode prior to the evaluations of the SMP and AMP modes in the mode decision, a large number of encoding time can be saved with negligible coding efficiency loss. **Fig. 3** depicts the flowchart of the proposed early Merge mode decision method based on motion estimation (EMD).
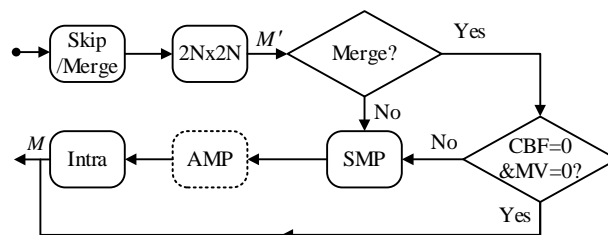

Fig. 3. Mode decision of the proposed EMD method (uniform for each N except for the dotted AMP).

As shown in **Fig. 3**, for a $2N \times 2N$ CU, its best interim mode M' is selected among the Skip, Merge, and Square modes. If $M' = Merge$, an additional decision is made whether the picture texture parameters of current CU satisfy the conditions of CBF = 0 and MV = 0. If yes, the Merge mode is selected as the best PU mode for current CU without the evaluations of SMP and AMP modes; otherwise, the default mode decision is performed for current CU. Note that the AMP mode evaluations are only allowed for $N \in \{4, 32\}$.

In order to evaluate the efficiency of the proposed methods, determination rate (DR) and hit rate (HR) are adopted, which correspond to the computational complexity reduction and prediction accuracy, respectively,

$$\begin{cases} S_{DR}(\mathbf{V}|\mathbf{U}) = N(\mathbf{V}|\mathbf{U})/N(\mathbf{U}) \times 100\% \\ T_{HR}(\mathbf{U}|\mathbf{V}) = N(\mathbf{U}|\mathbf{V})/N(\mathbf{V}) \times 100\% \end{cases} \tag{3}$$

where $S_{DR}(\mathbf{V}|\mathbf{U})$ and $T_{HR}(\mathbf{U}|\mathbf{V})$ denote the DR and HR, respectively; $\mathbf{V}|\mathbf{U}$ and $\mathbf{U}|\mathbf{V}$ represent two conditional events, where $\mathbf{U}$ and $\mathbf{V}$ are two individual events; $N(\cdot)$ represents the number of total CUs of the corresponding event. The computational complexity reduction grows as a function of DR. If HR is so large that close to 100%, it means that almost the best PU modes are correctly predicted with negligible coding efficiency loss.

To evaluate the efficiency of the proposed EMD method, the DR and HR in (3) are used. The event $\mathbf{U}$ and $\mathbf{V}$ represent $M = Merge$ and $M' = Merge \& CBF = 0 \& MV = 0$, respectively, where & is the logical AND operation. The experiments are conducted with the same video sequences and test conditions as in **Section 3.1**. The detailed results of DR and HR of the EMD method are listed in the columns 3~4 of **Table 2**.

Table 2. Respective DR and HR of the EMD and MET methods (%)

Sequence	Resolution	EMD		MET	
		$S_{DR}(\mathbf{V}|\mathbf{U})$	$T_{HR}(\mathbf{U}|\mathbf{V})$	$S_{DR}(\mathbf{V}|\mathbf{U})$	$T_{HR}(\mathbf{U}|\mathbf{V})$
Traffic	2560x1600	93.53	99.90	88.91	97.56
ParkScene	1920x1080	92.62	99.87	87.84	97.39
BasketballDrill	832x480	85.44	99.87	75.95	97.57
BQSquare	416x240	90.45	99.77	87.28	95.15
FourPeople	1280x720	94.92	99.94	91.45	98.64
Average		91.39	99.87	86.29	97.26

From **Table 2**, it can be observed that the DR of the proposed EMD method is from 85.44% to 94.92%, 91.39% on average, which means that 91.39% CUs satisfy the early Merge mode decision conditions when they are encoded in the Merge mode. The HR of the proposed algorithm is almost as high as 100% for all the video sequences with different picture texture. In other words, almost all the CUs are correctly encoded in the Merge mode under the early Merge mode decision conditions. These values demonstrate that the proposed EMD method works efficiently.

3.3 Merge Mode Based Early Termination Method (MET)

In HEVC inter prediction, choosing a small CU size usually results in a lower energy residual after motion compensation but requires a larger number of bits to signal prediction information. For the regions with a smooth or slow motion, they can be predicted more effectively using larger CU size as less bits are required for encoding.

It is reported that CUs encoded in the Merge mode have a high probability to be located in a region with homogeneous motions or even without motion [18]. Therefore, the Merge mode reflecting the motion inside the CUs can be used to skip the mode decisions for smaller CUs. For improving the accuracy, the picture texture parameters (CBF and MV) are also taken into account. Based on these analyses, we propose a Merge mode based early termination method (MET) for inter prediction, which is conducted according to a syntax element *splitflag*,

$$splitflag = \begin{cases} unsplit & M' = Merge \,\&\, CBF = 0 \,\&\, MV = 0 \\ default & otherwise \end{cases} \tag{4}$$

In the proposed MET method, for a $2N \times 2N$ CU, if the Merge mode is determined as its best interim mode M' with the picture texture parameters satisfying CBF = 0 and MV = 0, the syntax element *splitflag* is set as *unsplit* to skip the remaining mode decisions for smaller CUs. Otherwise, all the mode decisions for current CU and its smaller CUs will be performed just as the *default* inter prediction does. Note that the proposed MET method has no effect on the mode decisions of the indicated minimum CU size 8×8.

The DR and HR in (3) are also used to evaluate the efficiency of the proposed MET method. In this case, the event **U** and **V** denote *splitflag = unsplit* and $M' = Merge \,\&\, CBF = 0 \,\&\, MV = 0$, respectively. The experiments are conducted with the same video sequences and test conditions as in **Section 3.1**, and the statistical results of DR and HR of the MET method are listed in the columns 5~6 of **Table 2**.

It can be seen from **Table 2** that the DR of the proposed MET method is from 75.95% to 91.45%, 86.29% on average. It also means that 86.29% CUs tend to select their best PU modes on current CU sizes when $M' = Merge$, CBF = 0, and MV = 0. The HR of the proposed MET method is from 95.15% to 98.64%, 97.26% on average, which means that 97.26% CUs are encoded in the correct CU size as in the original HEVC encoder. These values illustrate the effectiveness of the proposed MET method.

3.4 Overall Algorithm

Based on the analyses above, the proposed overall algorithm is summarized in **Table 3**. As shown in **Table 3**, for a $2N \times 2N$ CU, if $M' = Merge$, CBF = 0, and MV = 0, both the mode decision evaluations for the remaining PU modes and smaller CUs are skipped to reduce the computational complexity significantly.

Table 3. Pseudo-code of the overall algorithm

Overall algorithm for fast inter prediction

Input: CTU size = 64×64; minimum CU size = 8×8; splitflag = default
Output: The best combinations of CU sizes and PU modes for the CTU
 1: **for** 0≤*Depth*≤3 **do**
 2:     // Encode the current CU with the Merge mode
 3:     // Encode the current CU with 2N×2N PU mode, and acquire the CBF and MV information
 4:     **if** M'=Merge & CBF=0 & MV=0 **then**
 5:         // M=Merge, splitflag=unsplit
 6:     **else**
 7:         // Encode the current CU with the remaining PU modes
 8:     **end if**
 9:     **if** splitflag=unsplit **then**
10:         // Skip the mode decisions for smaller CUs inside current CU
11:     **end if**
12: **end for**
13: // Proceed to the next CTU

4. Experimental Results

In order to validate the effectiveness of the proposed methods, we implemented them on the HEVC test model reference software, HM 16.4 [22]. Five classes of video sequences with different resolutions including class A ($4K \times 2K$), class B (1080p), class C (WVGA), class D (QWVGA) and class E (720p), are tested under the common test conditions [23]. The detailed common test conditions are as follows: Each video sequence is encoded under both RA and LB coding configurations, which corresponds to a broadcast scenario with a maximum group of picture (GOP) size of 8 and to a low-delay scenario with no picture reordering, respectively. Four QPs of 22, 27, 32, and 37 are adopted and the optional fast algorithms (ECU [10], ESD [11], and CFM [12]) are disabled. To save the profiling time, only one fifth of frames in each video sequence are encoded. The hardware platform is Intel Core i5-3470 CPU @ 3.20GHz, 8GB RAM with the Ubuntu 14.04 32-bit operating system. The performance of the proposed methods are measured in terms of encoding time and BD-rate [24], where positive and negative values represent increments and decrements, respectively. The encoding time saving is calculated as

$$\Delta T = \frac{T_{proposed} - T_{anchor}}{T_{anchor}} \times 100\% \tag{5}$$

where $T_{proposed}$ denotes the total encoding time consumed by the proposed methods, and T_{anchor} represents the total encoding time of the original HM 16.4 encoder.

Table 4 and Table 5 show individual results of the EMD method, the MET method, and the overall algorithm under RA and LB configuration, respectively. As shown in Table 4 and Table 5, the EMD method can achieve 31.21% and 29.26% encoding time reduction for all tested sequences under RA and LB configuration, respectively. And the coding efficiency loss introduced by the EMD method is very negligible, i.e., 0.013dB PSNR drops and 0.34%-0.39% BD-rate increases. As far as the MET method is concerned, 36.57% and 33.14% encoding time are saved under RA and LB configuration, respectively, with a maximum of 62.33% in "Johnny" (1280×720, LB) and a minimum of 10.09% in "RaceHorses" (416×240,

LB). In addition, the averaged PSNR drops are 0.033dB and 0.015dB, and the averaged BD-rate increases are 0.89% and 0.42% for all tested sequences under RA and LB configurations, respectively.

By incorporating the EMD and MET methods, the overall algorithm can reduce on average 46.3% and 43.0% encoding time under RA and LB configurations, respectively. Meanwhile, the averaged PSNR drops are 0.041-0.087 dB and the averaged BD-rate increases are 1.21-2.36% for all tested sequences. These values demonstrate that the overall algorithm is an attractive solution for fast inter prediction. However, the BD-rate penalty introduced by the overall algorithm is still too high for the cases where an acceptable RD performance (i.e., coding efficiency) is much more important than the computational complexity reduction. To provide a better balance between computational complexity and coding efficiency, several fast CU encoding schemes are surveyed in the following section according to the RDC characteristic of EMD and MET methods as a function of CU sizes.

Table 4. Individual evaluation results of EMD, MET and overall algorithm under RA configuration

Sequences	Picture Size	EMD			MET			Overall algorithm		
		BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)
PeopleOnStreet	2560x1600	0.34	-0.015	-21.84	0.82	-0.035	-23.07	2.47	-0.112	-31.86
Traffic	2560x1600	0.19	-0.006	-43.17	1.00	-0.033	-54.36	2.51	-0.082	-65.25
BasketballDrive	1920x1080	0.20	-0.005	-31.37	0.37	-0.008	-37.59	1.29	-0.028	-46.91
BQTerrace	1920x1080	0.24	-0.004	-41.53	1.10	-0.020	-52.70	1.93	-0.036	-63.05
Cactus	1920x1080	0.28	-0.006	-36.12	1.18	-0.025	-44.38	3.21	-0.068	-54.20
Kimonol	1920x1080	0.33	-0.011	-29.04	0.52	-0.017	-34.27	1.75	-0.055	-43.51
ParkScene	1920x1080	0.31	-0.009	-41.46	0.96	-0.029	-51.16	2.58	-0.078	-62.13
RaceHorses	832x480	0.56	-0.022	-18.68	0.63	-0.026	-18.57	2.45	-0.096	-27.00
BasketballDrill	832x480	0.19	-0.008	-29.63	0.51	-0.021	-34.54	1.44	-0.059	-43.91
BQMall	832x480	0.49	-0.020	-30.47	1.06	-0.042	-35.00	2.93	-0.115	-44.94
PartyScene	832x480	0.27	-0.012	-27.64	1.01	-0.046	-32.66	2.05	-0.095	-41.14
RaceHorses	416x240	0.76	-0.034	-17.52	0.93	-0.044	-15.48	3.22	-0.152	-24.20
BasketballPass	416x240	0.41	-0.019	-32.44	1.10	-0.052	-36.82	3.00	-0.143	-47.65
BlowingBubbles	416x240	0.33	-0.013	-28.92	1.18	-0.047	-32.13	2.62	-0.103	-42.16
BQSquare	416x240	0.19	-0.009	-38.26	0.96	-0.043	-45.79	1.89	-0.084	-56.73
Average		0.34	-0.013	-31.21	0.89	-0.033	-36.57	2.36	-0.087	-46.31

Table 5. Individual evaluation results of EMD, MET and overall algorithm under LB configuration

Sequences	Picture Size	EMD			MET			Overall algorithm		
		BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)
BasketballDrive	1920x1080	0.18	-0.004	-25.97	0.19	-0.004	-28.93	0.76	-0.016	-38.45
BQTerrace	1920x1080	0.39	-0.006	-35.94	0.77	-0.013	-43.61	1.57	-0.027	-54.08
Cactus	1920x1080	0.40	-0.010	-30.12	0.61	-0.014	-35.11	2.10	-0.049	-44.80
Kimonol	1920x1080	0.24	-0.008	-21.95	0.21	-0.007	-23.57	0.85	-0.029	-32.63
ParkScene	1920x1080	0.40	-0.013	-34.90	0.79	-0.025	-40.59	1.97	-0.061	-51.24
FourPeople	1280x720	0.34	-0.014	-47.29	0.48	-0.017	-59.21	0.80	-0.027	-71.17
Johnny	1280x720	0.69	-0.010	-49.83	0.11	0.000	-62.33	1.01	-0.015	-74.82
KristenAndSara	1280x720	0.63	-0.020	-48.72	0.01	-0.002	-60.69	0.87	-0.027	-73.01
RaceHorses	832x480	0.27	-0.011	-14.74	0.26	-0.011	-13.16	0.85	-0.035	-20.88
BasketballDrill	832x480	0.18	-0.007	-24.79	0.21	-0.009	-26.72	0.66	-0.027	-36.14
BQMall	832x480	0.45	-0.018	-25.36	0.57	-0.023	-27.29	1.41	-0.056	-37.05
PartyScene	832x480	0.28	-0.012	-19.65	0.47	-0.021	-20.18	1.01	-0.045	-28.18

RaceHorses	416x240	0.39	-0.020	-13.15	0.48	-0.024	-10.09	1.26	-0.062	-17.80
BasketballPass	416x240	0.40	-0.019	-27.57	0.45	-0.021	-29.66	1.32	-0.061	-39.97
BlowingBubbles	416x240	0.57	-0.022	-22.09	0.63	-0.024	-21.77	1.71	-0.065	-30.86
BQSquare	416x240	0.49	-0.020	-26.03	0.55	-0.022	-27.30	1.22	-0.050	-37.64
Average		0.39	-0.013	-29.26	0.42	-0.015	-33.14	1.21	-0.041	-43.04

5. Discussion

It is well known that when designing fast inter prediction algorithms, the computational complexity reduction for the encoder is proportional to the RD performance degradation. Hence, a good decision condition should be designed to yield the best trade-off between the computational complexity and coding efficiency. From **Table 1** we can see that, the prediction accuracies of selecting the Merge mode as the best PU mode vary among different CU sizes. That is, enabling various CU sizes for EMD and MET methods in the proposed overall algorithm may yield different trade-offs between the computational complexity and coding efficiency.

Based on this observation, extensive experiments are conducted in terms of the rate-distortion-complexity (RDC) characteristic. First, the individual RDC characteristics of the EMD and MET methods are investigated. Then, the most feasible encoding schemes are surveyed according to the RDC characteristics of EMD and MET methods as a function of CU sizes. Finally, all the encoding schemes are compared with the existing fast inter prediction algorithms. All the experiments in this section are performed with the same video sequences and test conditions as in **Section 4**. Each video sequence is encoded under both RA and LB coding configurations. Four QPs of 22, 27, 32, and 37 are selected to report the QP-specific delta bit-rate. The encoding time in (3) and BD-rate [24] are used to measure the performance of the encoding schemes. Here, each encoding scheme is identified as S_x, where x represents a consecutive numbering of these schemes. The allowed CU sizes for EMD and MET methods are $N_{EMD} \in \{4, 8, 16, 32\}$ and $N_{MET} \in \{8, 16, 32\}$, respectively.

5.1 EMD/MET Only Schemes

The individual performance of the EMD and MET methods presented in **Table 4** and **Table 5** have shown that the EMD and MET methods are effective to speed up the inter prediction with negligible coding efficiency loss. **Table 6** presents the individual performance of the EMD and MET methods in terms of the RDC characteristics, where the encoding schemes disabling either EMD or MET methods are denoted as S_0 and S_1, respectively.

As shown in **Table 6**, enabling the EMD method entirely achieves 31% encoding time reduction with 0.3% BD-rate increments under the RA configuration. The respective values for the MET method is 37% and 0.9%. That is, the MET method reduces 6% more encoding time with triple BD-rate penalty by the EMD method. For the LB case, the EMD and MET methods achieve 29% and 33% encoding time reduction, respectively, with a same BD-rate increments (4%). Note that a similar conclusion can be drawn for different QPs. Hence, compared to the EMD method, the MET method provides more room for obtaining a better balance between the computational complexity and coding efficiency.

Table 6. Impact of the EMD and MET methods in terms of RDC characteristics (%)

Cfg.	ID	N_{EMD}	N_{MET}	QP=22		QP=27		QP=32		QP=37		BDBR	ΔT
				Δbitrate	ΔT	Δbitrate	ΔT	Δbitrate	ΔT	Δbitrate	ΔT		
RA	S_0	{4,8,16,32}	ϕ	0.2	-18	0.2	-28	0.3	-36	0.4	-43	0.3	-31
	S_1	ϕ	{8,16,32}	0.5	-20	0.7	-32	0.9	-43	1.2	-52	0.9	-37
LB	S_0	{4,8,16,32}	ϕ	0.2	-16	0.3	-26	0.4	-34	0.5	-41	0.4	-29
	S_1	ϕ	{8,16,32}	0.2	-17	0.3	-28	0.4	-39	0.6	-48	0.4	-33

5.2 Encoding Schemes with Various CU Size Ranges

Since enabling various CU sizes for EMD and MET methods have a different influence on the performance of the overall algorithm, it is necessary to analyze the relationships between various CU size ranges and the performance of the overall algorithm. Here, four CU size ranges are enabled for the EMD method: $N_{EMD} \in \{4\}$, $N_{EMD} \in \{4,8\}$, $N_{EMD} \in \{4,8,16\}$, and $N_{EMD} \in \{4,8,16,32\}$. The respective CU size ranges allowed for the MET method are $N_{MET} \in \{8\}$, $N_{MET} \in \{8,16\}$, and $N_{MET} \in \{8,16,32\}$. These 12 candidate encoding schemes include the most feasible compositions of the EMD and MET methods with various CU size ranges, and their RDC characteristics are summarized in **Table 7.**

As shown in **Table 7**, the schemes S_2 - S_5 ($N_{MET} = 8$) reduce the computational complexity by 19%-39% at a cost of 0.2%-0.8% BD-rate increments under the RA configuration. Expanding $N_{MET} = 8$ to $N_{MET} \in \{8,16\}$ in S_6 - S_9 doubles the BD-rate penalty but with only 5% more computational complexity reduction except for the case $N_{EMD} = 4$, which reduces 12% more encoding time with an extra 0.3% BD-rate overhead. For the the schemes S_{10} - S_{13} with $N_{MET} \in \{8,16,32\}$, 38%-46% encoding time can be saved with at least 1% BD-rate penalty. These dramatic RD performance degradations would limit their applications in high-quality video coding.

In the LB case, The saving encoding time and BD-rate penalties by the schemes S_2 - S_5 are 17%-36% and 0.1%-0.6%, respectively. The schemes S_6 - S_9 lower 29%-40% computational complexity at a cost of 0.3%-0.9% BD-rate overhead. Enabling $N_{MET} \in \{8,16\}$ to $N_{MET} \in \{8,16,32\}$ in S_{10} - S_{13} reduces 3%-6% more computational complexity with around 0.2% additional BD-rate overhead. An interesting fact is that the schemes S_9 and S_{12} have a similar RDC characteristics, which means allowing the CU size $N = 32$ for either EMD or MET method beyond S_8 would make an equal contribution to the performance of the overall algorithm.

Table 7. Impact of proposed schemes with various CU size ranges in terms of RDC characteristics (%)

| Cfg.	ID	N_{EMD}	N_{MET}	QP=22		QP=27		QP=32		QP=37		BDBR	ΔT
				Δbitrate	ΔT	Δbitrate	ΔT	Δbitrate	ΔT	Δbitrate	ΔT		
RA	S_2	{4}	{8}	0.2	-12	0.3	-17	0.3	-21	0.4	-24	0.2	-19
	S_3	{4,8}	{8}	0.5	-18	0.7	-26	0.9	-33	1.0	-38	0.6	-29
	S_4	{4,8,16}	{8}	0.5	-21	0.8	-32	1.0	-40	1.1	-47	0.7	-35
	S_5	{4,8,16,32}	{8}	0.5	-22	0.7	-34	0.9	-45	1.1	-53	0.8	-39
	S_6	{4}	{8,16}	0.4	-18	0.5	-28	0.7	-36	0.9	-43	0.5	-31
	S_7	{4,8}	{8,16}	0.6	-20	0.8	-30	1.0	-39	1.2	-45	0.8	-34
	S_8	{4,8,16}	{8,16}	0.7	-24	1.1	-36	1.6	-46	2.1	-54	1.5	-40
	S_9	{4,8,16,32}	{8,16}	0.7	-25	1.2	-39	1.6	-50	2.1	-60	1.6	-43
	S_{10}	{4}	{8,16,32}	0.5	-21	0.7	-33	1.0	-44	1.3	-53	0.9	-38
	S_{11}	{4,8}	{8,16,32}	0.7	-23	1.0	-36	1.3	-47	1.7	-55	1.3	-40
	S_{12}	{4,8,16}	{8,16,32}	0.8	-25	1.3	-38	1.7	-49	2.3	-58	1.8	-43
	S_{13}	{4,8,16,32}	{8,16,32}	0.9	-27	1.4	-41	2.0	-54	2.8	-64	2.4	-46
LB	S_2	{4}	{8}	0.1	-11	0.2	-16	0.3	-20	0.3	-23	0.1	-17
	S_3	{4,8}	{8}	0.3	-16	0.5	-25	0.6	-32	0.7	-37	0.4	-28
	S_4	{4,8,16}	{8}	0.3	-19	0.4	-29	0.6	-38	0.8	-46	0.6	-33
	S_5	{4,8,16,32}	{8}	0.4	-20	0.4	-32	0.6	-42	0.8	-51	0.6	-36
	S_6	{4}	{8,16}	0.3	-16	0.5	-25	0.6	-33	0.8	-40	0.3	-29
	S_7	{4,8}	{8,16}	0.5	-18	0.7	-28	0.9	-36	0.9	-43	0.5	-32
	S_8	{4,8,16}	{8,16}	0.6	-21	0.9	-33	1.2	-43	1.4	-52	0.8	-37
	S_9	{4,8,16,32}	{8,16}	0.6	-22	0.9	-36	1.2	-47	1.4	-57	0.9	-40
	S_{10}	{4}	{8,16,32}	0.4	-19	0.7	-30	0.8	-40	1.0	-50	0.5	-35
	S_{11}	{4,8}	{8,16,32}	0.6	-21	1.0	-33	1.1	-43	1.2	-53	0.7	-38
	S_{12}	{4,8,16}	{8,16,32}	0.7	-22	1.1	-35	1.4	-46	1.6	-55	1.0	-40
	S_{13}	{4,8,16,32}	{8,16,32}	0.8	-24	1.2	-38	1.6	-50	1.8	-61	1.2	-43

5.3 Comparison with Existing Techniques

To further evaluate the effectiveness of the proposed encoding schemes, three built-in fast inter prediction algorithms (ECU [10], ESD [11], and CFM [12]) and two state-of-the-art fast inter prediction algorithms (CSD [13], PMD [15]) are implemented on HM 16.4. All the existing techniques are conducted with the same video sequences and test conditions as the proposed encoding schemes so that a fair comparison can be obtained. **Fig. 4** depicts the RDC characteristics of the five existing techniques against the proposed encoding schemes, where the existing techniques and proposed encoding schemes are marked by cross and dot, respectively.

In **Fig. 4**, the proposed schemes belonging to S_2 - S_5, S_6 - S_9, and S_{10} - S_{13} (with same CU size ranges for the MET method) are connected with dashed lines to improve their visual comparison. Although the proposed schemes provide a wide range of RDC characteristics, the crossing lines among the schemes indicate that none of them obtains the best gain alone. This is due to the fact that the contributions of the EMD and MET methods to the proposed overall algorithm are overlapped, which is consistent with the results in **Table 4** and **Table 5**. The common trend between RA and LB cases is that the proposed schemes are superior or comparable with ESD, CFM, CSD and PMD. For example, S_0, S_1, S_8, and S_{12} achieve higher or equal computational complexity reduction with a smaller BD-rate overhead than ESD, CFM, CSD, and PMD in the RA case, respectively. In the LB case the respective schemes are S_6, S_8, S_{12}, and S_{13}.

In addition, an advantage of these proposed encoding schemes over CSD and PMD is that they can be seamlessly incorporated in the existing control structure of the HEVC encoder without limiting its potential parallelization and hardware acceleration.
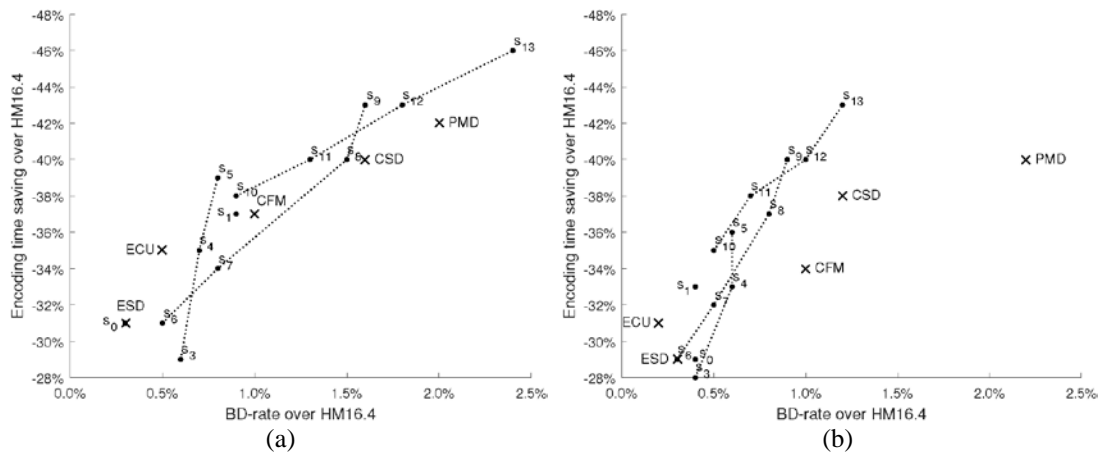
Fig. 4. Comparison of the proposed and contemporary fast inter predicion schemes in terms of RDC characteristics (HM 16.4). (a) RA case. (b) LB case.

6. Conclusion

In this paper, we propose several fast CU encoding schemes based on the Merge mode and motion estimation information to reduce the computational complexity of the HEVC encoder. Firstly, an early Merge mode decision method based on motion estimation (EMD) is proposed for each CU size. Then, a Merge mode based early termination method (MET) is developed to determine the CU size at an early stage. To provide a better balance between computational complexity and coding efficiency, several fast CU encoding schemes are surveyed according to the RDC characteristics of EMD and MET methods as a function of CU sizes. Experimental results demonstrate the effectiveness of the proposed schemes.

For future work, we plan to apply the proposed schemes to a practical video encoder on many-core processors. A practical video encoder needs low computational complexity, acceptable visual quality, and a parallel framework. Note that there have been some recent developments for video coding on many-core processors [25, 26], which can be useful for designing a practical video coder.

References

[1]   "High efficiency video coding," ITU-T and R. H. ISO/IEC 23008-2 (HEVC), Oct. 2014.
[2]   G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649-1668, Dec. 2012. Article (CrossRef Link)
[3]   "Advanced video coding for generic audiovisual services," ITU-T and R. H. ISO/IEC 14496-10 (AVC), Mar. 2009.
[4]   J. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards - including high efficiency video coding (HEVC)," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1669-1684, Dec. 2012. Article (CrossRef Link)
[5]   P. Helle *et al.*, "Block merging for quadtree-based partitioning in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1720-1731, Dec. 2012. Article (CrossRef Link)

[6]   F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1685-1696, Dec. 2012. Article (CrossRef Link)

[7]   I. K. Kim, J. Min, T. Lee, W. J. Han, and J. Park, "Block partitioning structure in the HEVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1697-1706, Dec. 2012. Article (CrossRef Link)

[8]   Y. Yuan *et al.*, "Quadtree based non-square block structure for inter frame coding in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1707-1719, Dec. 2012. Article (CrossRef Link)

[9]   K. McCann, C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. J. Sullivan, "High efficiency video coding (HEVC) test model 16 (HM 16) encoder description," *ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC)*, Document JCTVC-R1002, Jul. 2014.

[10] K. Choi, S.-H. Park, and E. S. Jang, "Coding tree pruning based CU early termination," *Document JCTVC-F092*, Torino, Italy, Jul. 2011.

[11] J. Yang, J. Kim, K. Won, H. Lee, and B. Jeon, "Early skip detection for HEVC," *Document JCTVC-G543*, Geneva, Switzerland, Nov. 2011.

[12] R. H. Gweon and Y. L. Lee, "Early termination of CU encoding to reduce HEVC complexity," *Document JCTVC-F045*, Torino, Italy, Jul. 2011.

[13] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 465-470, Feb. 2013. Article (CrossRef Link)

[14] L. Shen, Z. Zhang, and Z. Liu, "Adaptive inter-mode decision for HEVC jointly utilizing inter-level and spatiotemporal correlations," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 10, pp. 1709-1722, Oct. 2014. Article (CrossRef Link)

[15] J. Xiong, H. Li, Q. Wu, and F. Meng, "A fast HEVC inter CU selection method based on pyramid motion divergence," *IEEE Transactions on Multimedia*, vol. 16, no. 2, pp. 559-564, Feb. 2014. Article (CrossRef Link)

[16] J. Vanne, M. Viitanen, and T. Hamalainen, "Efficient mode decision schemes for HEVC inter prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 9, pp. 1579-1593, Sep. 2014. Article (CrossRef Link)

[17] S. Ahn, B. Lee and M. Kim, "A novel fast CU encoding scheme based on spatiotemporal encoding parameters for HEVC inter coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 422-435, Mar. 2015. Article (CrossRef Link)

[18] Z. Pan, S. Kwong, M.-T. Sun, and J. Lei, "Early MERGE mode decision based on motion estimation and hierarchical depth correlation for HEVC," *IEEE Transactions on Broadcasting*, vol. 60, no. 2, pp. 405-412, Jun. 2014. Article (CrossRef Link)

[19] J. Lee, S. Kim, K. Lim, and S. Lee, "A fast CU size decision algorithm for HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 411-421, Mar. 2015. Article (CrossRef Link)

[20] H. Zeng, C. Cai, and K.-K. Ma, "Fast mode decision for H.264/AVC based on macroblock motion activity," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 4, pp. 491-499, Apr. 2009. Article (CrossRef Link)

[21] L. Shen, Z. Liu, T. Yan, Z. Zhang, and P. An, "View-adaptive motion estimation and disparity estimation for low complexity mulatiview video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 6, pp. 925-930, Jun. 2010. Article (CrossRef Link)

[22] JCT-VC (2015) Subversion repository for the HEVC test model reference software, ver. HM 16.4 Article (CrossRef Link)

[23] F. Bossen, "Common test conditions and software reference configurations," *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11*, Document: JCTVC-L1100, 12th Meeting: Geneva, CH, 2013.

[24] G. Bjontegaard, "Calculation of average PSNR difference between RD-curves," VCEG-M33, Austin, TX, USA, Apr. 2001.

[25] C. Yan, Y. Zhang, J. Xu *et al.*, "A highly parallel framework for HEVC coding unit partitioning tree decision on many-core processors," *IEEE Signal Processing Letters*, vol. 21, no. 5, pp. 573-576, May 2014. Article (CrossRef Link)

[26] C. Yan, Y. Zhang, J. Xu *et al.*, "Efficient parallel framework for HEVC motion estimation on many-core processors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 12, pp. 2077-2089, Dec. 2014. Article (CrossRef Link)
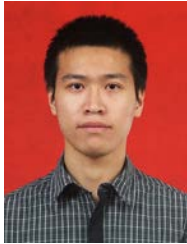
Jinfu Wu received his B.S. degrees in control technology and instrument from Xidian University, Xi'an, P.R. China in 2011. He is currently a Ph.D. student with the School of Aerospace Science and Technology, Xidian University. His research interests include video coding and image processing.

Baolong Guo received his B.S., M.S. and Ph.D. degrees from Xidian University in 1984, 1988 and 1995, respectively, all in communication and electronic system. From 1998 to 1999, he was a visiting scientist at Doshisha University, Japan. He is currently the director of the Institute of Intelligent Control & Image Engineering (ICIE) at Xidian University. His research interests include neural networks, pattern recognition, intelligent information processing, and image communication.

Jie Hou obtained B.S. degree in Automation from Xidian University, P.R.China in 2011. Currently, He's pursuing Ph.D. degree in Measuring Technology and Instruments at Xidian University. His research mainly focuses on video processing and object recognition.

Yunyi Yan received his B.S. degree in automation, M.S. degree in pattern recognizaion andintelligent system, and Ph.D degree in electronic science and technology from Xidian University in 2001, 2004 and 2008 respectively in China. Since 2004, he has been with ICIE Institute of Areospace Science and Technology School of Xidian University, where he is currently an associated professor. Since 2013, he served as the director of Intelligent Detection Department in Xidian University. He worked as a visiting scholar in Montreal Neurological Institute of McGill University,Canada from 2010 to 2011. His research interests include system reliablity analysis, medical image processing, and swarm intelligence.

Jie Jiang received her Ph.D. degree in signal and information processing from Xidian University in 2012. Then she joined Hangzhou Dianzi University. Her research interests include video coding and communication.