Neyman-Pearson-Based Early Mode Decision for HEVC Encoding

Qiang Hu, Xiaoyun Zhang, *Member, IEEE*, Zhiru Shi, and Zhiyong Gao

***Abstract*—The high efficiency video coding (HEVC) standard has highly improved the coding efficiency by adopting hierarchical structures of coding unit (CU), prediction unit (PU), and transform unit (TU). However, enormous computational complexity is introduced due to the recursive rate-distortion optimization (RDO) process on all CUs, PUs and TUs. In this paper, we propose a fast and efficient mode decision algorithm based on the Neyman-Pearson rule, which consists of early SKIP mode decision and fast CU size decision. First, the early mode decision is modeled as a binary classification problem of SKIP/non-SKIP or split/unsplit. The Neyman-Pearson-based rule is employed to balance the rate-distortion (RD) performance loss and the complexity reduction by minimizing the missed detection with a constrained incorrect decision rate. A nonparametric density estimation scheme is also developed to calculate the likelihood function of the statistical parameters. Furthermore, an online training scheme is employed to periodically update the probability density distributions for different quantization parameters (QPs) and CU depth levels. The experimental results show that the proposed overall algorithm can save 65% and 58% computational complexity on average with a 1.29% and 1.08% Bjontegaard Delta bitrate (BDBR) increase for various test sequences under random access and low delay P conditions, respectively. The proposed overall scheme also has the advantage that it can make the trade-off between the RD performance and time saving by setting different values for the incorrect decision rate.**

***Index Terms*—High efficiency video coding (HEVC), mode decision, Neyman-Peason, nonparametric density estimation.**

I. Introduction

RECENTLY, the fast development of video capture and display devices has brought a dramatic demand for high definition (HD) and Ultra high definition (UHD) videos. However, the existing video coding standard H.264/AVC [1] is still not efficient enough to compress the UHD videos, which become the burden of video storage and transmission. To further improve the compression performance, the *High Efficiency Video Coding* (HEVC) standard is developed by the *Joint Collaborative Team on Video Coding* (JCT-VC) [2].

Manuscript received May 18, 2015; revised November 5, 2015; accepted December 14, 2015. Date of publication December 25, 2015; date of current version February 18, 2016. This work was supported in part by the National Natural Science Foundation of China under Grant 61221001, Grant 61133009, and Grant 61301116, in part by the 111 Project under Grant B07022, in part by the Shanghai Key Laboratory of Digital Media Processing and Transmissions under STCSM Grant 12DZ2272600, and in part by the National Key Technology R&D Program of China under Grant 2013BAH53F04. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yap-Peng Tan. *(Corresponding author: Xiaoyun Zhang.)*

The authors are with the Institute of Image Communication and Network Engineering, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: hq2902108007@sjtu.edu.cn; xiaoyun.zhang@sjtu.edu.cn; zhiru.shi@sjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2512799

Compared with the previous standard H.264/AVC, the HEVC provides higher compression performance by 50% bitrate saving on average at a similar perceptual image quality [3]. The main improvement comes from utilizing more flexible partitioning structures, which include *coding unit* (CU), *prediction unit* (PU) and *transform unit* (TU) [4], [5]. The quad-tree based prediction structures can obtain more accurate partitioning and improve the coding performance significantly. However, the flexible partitioning structures bring intensive computational complexity. In the recursive searching for the best partitioning mode, RDO process is performed for each size of CU, PU and TU. Therefore, reducing the computational complexity of HEVC encoding is essential for real-time application.

Recently, many approaches [6]–[25] have been proposed to reduce the mode decision complexity of HEVC. In [6]–[13], fast algorithms for CU mode decision are proposed. Several PU mode decision algorithms are proposed in [14]–[17]. Although these works have performed well on decreasing the complexity, only speeding up a single type of partitioning structures is still not enough. Further efforts are introduced in [18]–[25] to deal with both the CU and the PU mode decisions in order to obtain more computation reduction.

In our study on HEVC encoding of high resolution videos, the *SKIP* mode shows a very high occurrence probability. If the *SKIP* mode can be correctly determined in advance, a large proportion of RDO process and motion estimation will be skipped without affecting RD performance. Furthermore, as shown in the previous researches [6]–[13], [18]–[25] CU size decision process applying full RDO can bring unacceptable computational complexity for practical application. Therefore, efficient early CU size decision is also necessary. Motivated by these ideas, the early *SKIP* mode decision (ESMD) and early CU size decision (ECUSD) based on the Neyman-Pearson rule are proposed in this paper.

Both *SKIP* mode decision and CU size decision are modeled as the binary classification problems of *SKIP/non-SKIP* and *split/unsplit* in our work. In fact, not all the misclassification will cause RD performance loss. For example, the missed detection of *SKIP* mode will only lead to more computation, while the incorrect decision of the *non-SKIP* mode as the *SKIP* mode will affect RD performance. To control the RD performance loss caused by the incorrect decision, the Neyman-Pearson decision rule is applied by using the RD cost as the criterion. The probability of the missed detection is minimized subject to the condition that the incorrect decision is not larger than a given constraint. Therefore, with the constrained RD performance loss, the complexity can be reduced as much as possible.

In order to improve the accuracy of the Neyman-Pearson decision, an adaptive mechanism is employed for the probability

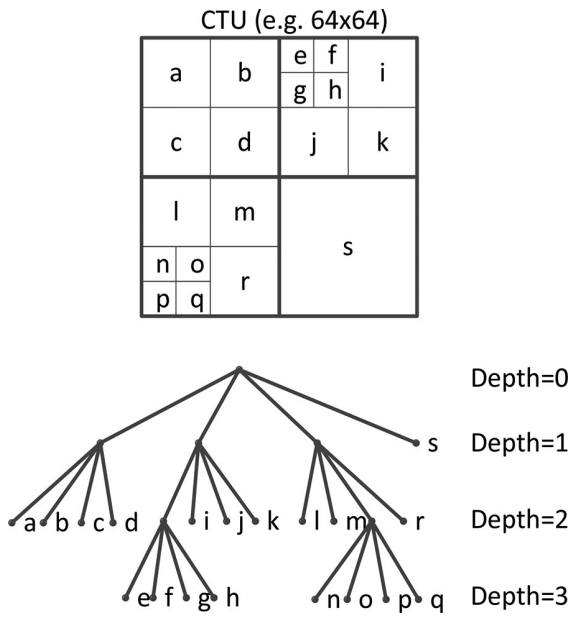

Fig. 1. Quad-tree structure of HEVC.

density estimation. Since the RD cost distributions are different for different QPs and CU depth levels, an online training scheme is developed to update the probability density distributions for various QPs and CU sizes. The final experimental results demonstrate that the proposed overall algorithm can reduce the computational complexity in the range of 50% to 82%, while the bit rate increase is only 1.29% on average.

The rest of the paper is organized as follows. Section II presents the partitioning structures for HEVC and related works on fast mode decision algorithms for reducing the video coding computational complexity. Section III analyzes the *SKIP* mode and CU size distributions. Section IV introduces the proposed fast algorithms in detail. Performance evaluations and analyses are presented in Section V. At last, the conclusion is drawn in Section VI.

II. HEVC Partitioning Structures and Related Works

A. HEVC Partitioning Structures

HEVC inherits the block-based hybrid coding architecture as its predecessor H264/AVC. The video frames are partitioned into sequences of *coding tree units* (CTUs), which can be recursively divided into four equal sub-CUs. In the HEVC reference software, there are four CU depth levels with CU sizes ranging from $8\times$ 8 to 64 $\times 64$. The quad-tree structure of HEVC is shown in Fig. 1.

According to the partition mode, each CU may include one or more PUs, which are the basic unit used in either inter-frame or intra-frame prediction. As shown in Fig. 2, there are up to 8 partition modes for inter-coded CUs and 2 partition modes for intra-coded CUs. And for each PU in one kind of partition, there are various prediction modes such as *SKIP*, *merge*, forward, backward, bi-prediction for inter PUs and 35 direction modes for intra PUs. Similar to H.264/AVC, the mode decision process in HEVC is conducted by checking all possible depth

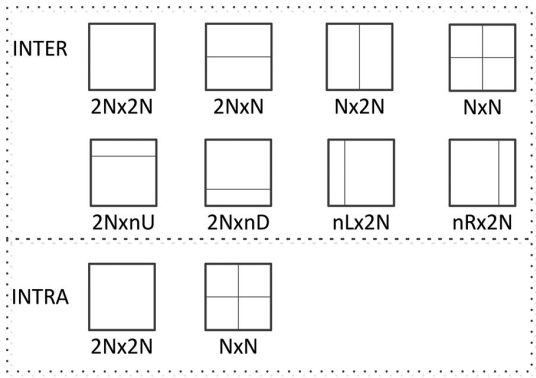

Fig. 2. Partitions for inter-coded CUs and intra-coded CUs in HEVC.

levels, partitions and prediction modes to choose the one with the minimal RD cost as the optimal mode, which is expressed by

$$\min\{J\}\ J = \mathrm{SSE} + \lambda \cdot \mathrm{Bits} \tag{1}$$

where J denotes the RD cost, and *Bits* specifies the bitrate cost dependent on each mode decision case; *SSE* is the sum of squared errors between the current block and the matching block, and λ is the Lagrange multiplier. However, checking all possible depth levels and prediction modes will lead to extremely high computational complexity and make it difficult for real-time implementation. Therefore, efficient algorithms with reduced complexity and negligible RD performance loss are crucially important and highly desired.

B. Related Works

A number of fast algorithms have been proposed to reduce the mode decision complexity of HEVC encoding. These works can be categorized into three classes. The first one is fast algorithms for determining the CU size [6]–[13]. The early CU (ECU) termination algorithm is proposed in [6] to avoid unnecessary CU splitting, when *SKIP* is detected as the best mode of the current CU. In [7], a fast HEVC CU size decision algorithm is proposed by using the neighbor and co-located CUs information to decide the depth range. A pyramid motion divergence (PMD) based fast CU size selection method is proposed in [8], which employs the variance of the down-sampled optical flow and k nearest neighbors like method. In [9], a fast CU depth decision method is proposed by using the depth information of spatio-temporal adjacent CTUs to reduce the depth levels. The RD cost based scheme [10] is proposed to reduce the complexity by top skip and early termination of CUs. Shen [11] proposes a CU splitting early termination algorithm based on support vector machines (SVM), where several features like RD cost, motion activity and prediction error are used. In [12], a Bayesian decision rule is applied for early CU *split* decision using relevant and computational-friendly features to assist the decision. Zhang [13] proposes a machine learning-based fast CU depth decision method, which optimizes the complexity allocation at CU level with given RD cost constraints.

The second category of fast algorithms is to reduce the complexity of PU mode decision [14]–[17]. Since *SKIP* mode is a special PU mode, Kim et al [14] proposes an early *SKIP*

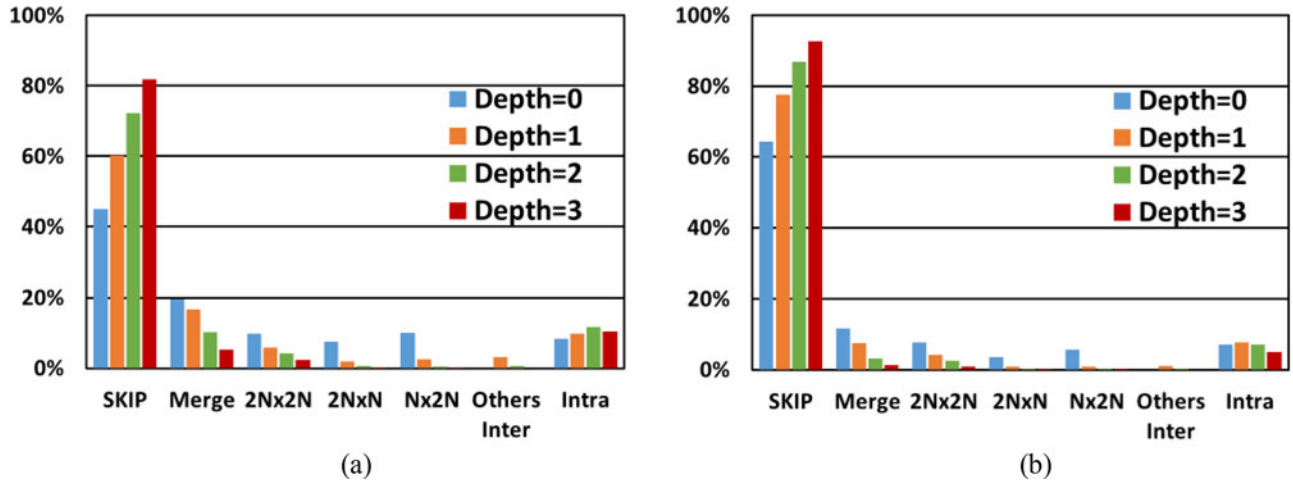

Fig. 3. PU mode distribution for different CU depth levels and QP values for BasketballDrive (100 frames) under RA-Main configuration. (a) BasketballDrive, QP $= 27$. (b) BasketballDrive, QP $= 37$.

detection (ESD) algorithm, in which the different motion vectors and coded block flag (CBF) of inter $2N \times 2N$ mode are utilized. This method has been adopted into the HEVC reference software model HM16.0[1] as a fast option. In [15], the rate distortion complexity characteristics of HEVC inter prediction are analyzed and used by proposing optimized mode decision schemes. Based on the fact that *SKIP* mode is spatially related, a global and local level based fast *SKIP* mode decision algorithm is presented in [16]. The CBF fast method (CFM) is proposed in [17] to skip the remaining PU mode decision process of the current CU, when all CBF values of luminance and chrominance are zero.

Finally, some approaches optimize both the CU and the PU mode decision processes [18]–[25]. In [18], fast HEVC encoding based on decision trees is described. The decision trees are obtained by the data mining technique. Lee [19] combined three approaches together including *SKIP* mode decision (SMD), early CU termination (ECUT), and CU skip estimation (CUSE). Thresholds are calculated based on Bayes' rule with a complexity factor. In [20], an early *SKIP* detection method and a fast CU *split* decision method are introduced based on spatio-temporal encoding parameters. The algorithm utilizes sample-adaptive-offset parameters as the spatial encoding parameters to estimate the texture complexity that affects the CU partitions. The motion vectors, TU size and coded block flag information are used as the temporal encoding parameters to estimate the temporal complexity that affects the CU partitions. In [21], a fast inter mode decision algorithm is proposed by using the prediction information from spatially or temporally adjacent CUs. Three adaptive inter mode schemes are introduced, i.e., early *SKIP* mode decision, prediction size correlation and RD cost correlation based mode decision. In [22], the Markov Random Field (MRF) is employed for fast inter CU decision, and the pyramid variance of the absolute differences (PVAD) is used as the feature to choose the CU size. Hu [23] proposes an efficient prediction scheme for x265, which includes decreasing the number of RDO times, early *SKIP* detection and fast intra mode decision. In [24], a fast inter CU decision is proposed based on the latent sum of absolute differences estimation. In [25], a fast CU size decision method for HEVC intra coding is proposed.

These previous related works can reduce the computational complexity of HEVC encoder, but more efficient methods are still in great need for real-time video services. In this article, early *SKIP* mode and CU size decision algorithms are proposed based on the Neyman-Pearson decision rule, which can further decrease computational complexity while maintain the coding performance.

III. Observations and Statistical Analysis

In HEVC, *SKIP* mode is a very efficient inter prediction tool, since its coefficients are all zeros and motion information is predicted from neighboring blocks directly. A PU can be coded without motion estimation and residual coding process if it is early detected as *SKIP* mode, therefore *SKIP* is the coding mode with the lowest computational complexity. On the other side, checking each CU depth level needs tremendous complexity to compute RD cost. And there are generally many homogenous regions in HD and UHD videos, where large CU depth level checking is not necessary. Therefore, it is desirable and beneficial to early terminate CU depth decision to exclude unnecessary CU depth levels. In this section, statistics of the *SKIP* mode distribution and CU size distribution are fully investigated.

A. SKIP Mode Distribution

In order to investigate the distribution of the optimal PU modes for each CU depth level, the reference encoder HM16.0 is used to encode the HD test sequence BasketballDrive (100 frames) under "encoder_randomaccess_main" (RA-Main) configuration [26]. The PU mode distribution results are listed in Fig. 3. It shows that *SKIP* is the most frequently selected mode and takes up the major proportion ranging from 45% to 95%.

[1]"JCT-VC Subversion Respository for the HEVC test Model Version HM16.0, " [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.0

Fig. 4. CU partition of BasketballDrive encoded by HM, QP = 27 under RA-Main configuration.

TABLE I
CU DEPTH LEVEL DISTRIBUTION FOR DIFFERENT SEQUENCES UNDER RA-MAIN CONFIGURATION

Sequences	Depth = 0	Depth = 1	Depth = 2	Depth = 3
PeopleOnStreet	21%	31%	33%	15%
Traffic	62%	22%	12%	4%
BQTerrace	59%	20%	14%	7%
Cactus	54%	25%	15%	6%
RaceHorsesC	14%	43%	30%	13%
RaceHorses	5%	34%	43%	18%
Average	**36%**	**29%**	**24%**	**11%**

Motivated by these observations, the appropriate *SKIP* mode early detection can decrease vast coding computation, while RD performance will not be affected.

According to Fig. 3, we can also see that the percentage of *SKIP* modes is various for different QP values and CU depth levels. For the low QP values (QP 27), less residual coefficients are quantized to zeros, so that less *SKIP* modes are encoded. But the *SKIP* modes still take up more than 60%. While for the high QP situation, residual coefficients are more likely to be quantized into zeros, which leads to more *SKIP* selected, about 80% of CUs are usually coded as *SKIP* modes. When considering CU depth levels, the small blocks have strong correlation and similar motion vectors can be predicted from neighboring blocks directly. In this case, *SKIP* modes take up more than 90%. To achieve the best complexity reduction performance, the *SKIP* mode detection scheme should be adaptive for different QPs and CU depth levels according to the statistical analysis.

B. CU Size Distribution

HEVC encoding performs full RD cost computation on all possible CU depth levels and PU modes to find the optimal one. Fig. 4 is an example of the final CU size decision of HEVC encoding under RA-Main configuration. It is appropriate to choose a large CU size as the best mode for the homogeneous regions, such as the background. While for the regions with active motion or rich texture like moving peoples, it usually selects small CU size. The CU depth level distribution of test sequences with different motion activities and texture is presented in Table I under RA-Main configuration. The result shows that the total probability of CU depth level 0 and 1 is almost 80% for sequences with smooth motion and static background (such as Traffic, BQTerrace and Cactus). For sequences with complex motion or rich texture (such as PeopleOnStreet, RaceHorses and RaceHorsesC), the possibility of selecting CU depth level 2 and 3 is rising. But the selection of small CU depth still has high probability. From the overall test results, about 36% and 29% of CUs choose the depth level 0 and level 1 respectively as the optimal levels. Only 11% of CUs choose the depth level 3. The conclusion obtained from this observation is that most of tree blocks select the first two depth levels as the optimal depth levels, especially for homogenous sequences.

Thus, efficient algorithm of early CU depth termination could reduce dramatic RDO computation complexity by excluding small size CUs. According to these observations, we propose the ECUSD algorithm based on the Neyman-Pearson decision rule to maximize the complexity reduction under the condition of constraining the incorrect decision rate of *unsplit*.

IV. PROPOSED EARLY MODE DECISION ALGORITHM

A. Early SKIP Mode Decision (ESMD)

In PU mode selection, *SKIP* mode decision can be considered as a two-category classification problem, in which there are two categories $\{\omega_S^d, \omega_{nS}^d\}$. Category ω_S^d represents that *SKIP* mode is considered as the optimal mode in the given CU depth d, and ω_{nS}^d denotes that the optimal mode is *non-SKIP* where other modes should be searched. In our algorithm, the RD cost (J_S^d) of *SKIP* mode at the given CU depth d is calculated and used as the criterion for the classification problem. Computing the RD cost of *SKIP* mode has the lowest complexity, because the calculation for the coefficients bits and decoder loop are not executed. Furthermore, the cost J_S^d is firstly obtained before computing the RD costs of other PU modes. Therefore, the J_S^d can be conveniently used as the criterion for ESMD without additional computation.

The histogram distributions of *SKIP* RD cost J_S^d in terms of $\omega_S^d$and ω_{nS}^d are investigated. Taking the sequence BQTerrace with CU depth level 2 as an example, the histogram of *SKIP* RD cost J_S^2 is illustrated in Fig. 5. *SKIP* RD cost J_S^2 of category ω_S^2 shows a concentrated distribution centered in a narrow RD cost range. Meanwhile, J_S^2 of *non-SKIP* class ω_{nS}^2 distributes in a relatively wide and flat range. According to the histogram, the possibility of *non-SKIP* mode is correspondingly low at the region with small J_S^2. As a consequence, PUs with small J_S^2 are more likely to be determined as *SKIP* modes. But for the region with large J_S^2, the *SKIP* and *non-SKIP* J_S^2 distributions are overlapped. The incorrect early decision may result in RD performance loss. In this case, the Neyman-Pearson based decision is adopted to reduce computation as much as possible under the condition of a given error rate. Thus, we define the following two cases in the mode decision:

Missed detection: A missed detection is the case that the optimal mode should be *SKIP*, but it is not early detected and *non-SKIP* mode decision process is redundantly implemented. For the missed detection, there is no RD performance loss but

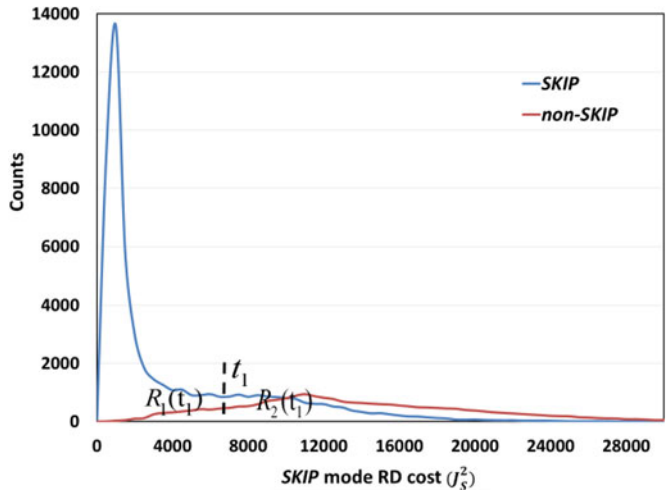

Fig. 5. Distributions of J_S^2 for *SKIP* mode and *non-SKIP* mode for BQTerrace with $\mathrm{QP} = 27$ and depth level $= 2$.

with more computation. We denote the $p_1(e)$ as the probability of the missed detection.

Incorrect decision: An incorrect decision is the case that we make decision as the *SKIP* mode while the ground truth should be *non-SKIP*. This kind of incorrect decision will lead to RD performance loss. We denote the $p_2(e)$ as the probability of the incorrect decision.

According to the Neyman-Pearson decision theory, our decision rule α divides the distribution space into two regions: $R_1(t_1)$ and $R_2(t_1)$

$$\begin{aligned} R_1(t_1) &= \left\{ J_S^d : J_S^d \le t_1 \,\&\, \alpha(J_S^d) = \omega_S^d \right\} \\ R_2(t_1) &= \left\{ J_S^d : J_S^d > t_1 \,\&\, \alpha(J_S^d) = \omega_{nS}^d \right\}. \end{aligned} \tag{2}$$

The decision boundary t_1 is illustrated as dotted boundary line between the two decision regions in Fig. 5. The probability of missed detection can be expressed as

$$p_1(e) = \int_{R_2(t_1)} p(J_S^d|\omega_S^d) dJ_S^d. \tag{3}$$

The probability of incorrect decision can be expressed as

$$p_2(e) = \int_{R_1(t_1)} p\left(J_S^d|\omega_{nS}^d\right) dJ_S^d \tag{4}$$

where $p(J_S^d|\omega_i^d), i \in \{S, nS\}$ denotes the class-conditional probability density function (PDF) of J_S^d given ω_i^d and $p(\omega_i^d)$ is the priori probability of class ω_i^d. Then we define the total error rate as

$$P(e) = p_2(e)p\left(\omega_{nS}^d\right) + p_1(e)p(\omega_S^d). \tag{5}$$

The priori probability $p(\omega_i^d)$ can be estimated as follows:

$$p\left(\omega_i^d\right) = \frac{N_i^d}{\sum_i N_i^d}, i \in \{S, nS\} \tag{6}$$

where N_S^d and N_{nS}^d represent the number of *SKIP* and *non-SKIP* CUs for the current depth d, respectively.

It should be noticed that there is no RD performance loss for the missed detection, but only with complexity increase. However, the incorrect decision of making *SKIP* mode as the optimal one will lead to RD performance loss. According to the Neyman-Pearson rule, we make the missed detection rate of *SKIP* mode $p_1(e)$ as small as possible in order to reduce the computation, while the incorrect decision rate of *SKIP* mode $p_2(e)$ is subject to a constraint to maintain the RD performance. We obtain the Neyman-Pearson rule as

$$\begin{cases} \min p_1(e) \\ \text{s.t. } p_2(e) = \varepsilon_1 \end{cases} \tag{7}$$

where ε_1 is the maximum acceptable incorrect decision rate of *SKIP* mode, which is used as the trade-off parameter between complexity reduction and RD performance. The Lagrange multiplier method is introduced to solve this problem. The Lagrange function is defined as follows:

$$L = p_1(e) + \lambda_1 \cdot (p_2(e) - \varepsilon_1) \tag{8}$$

where λ_1 is the Lagrange multiplier. Then, substituting (3) and (4) into (8), we obtain

$$\begin{aligned} L = &\int_{R_2(t_1)} p\left(J_S^d \middle| \omega_S^d\right) dJ_S^d + \lambda_1 \\ &\cdot \left(\int_{R_1(t_1)} p\left(J_S^d \middle| \omega_{nS}^d\right) dJ_S^d - \varepsilon_1 \right). \end{aligned} \tag{9}$$

It should be noticed that

$$\int_{R_1(t_1)} p\left(J_S^d \middle| \omega_S^d\right) dJ_S^d + \int_{R_2(t_1)} p\left(J_S^d \middle| \omega_S^d\right) dJ_S^d = 1. \tag{10}$$

Then, (9) can be expressed as

$$L = 1 - \lambda_1 \cdot \varepsilon_1 + \int_{R_1(t_1)} \left(\lambda_1 \cdot p\left(J_S^d \middle| \omega_{nS}^d\right) - p\left(J_S^d \middle| \omega_S^d\right)\right) dJ_S^d. \tag{11}$$

In order to get the extremum of (11), we should calculate the derivation of t_1 and λ_1, respectively. And set $\frac{\partial L}{\partial \lambda_1} = 0$, $\frac{\partial L}{\partial t_1} = 0$. Then, we can get the solution as

$$\int_{R_1(t_1)} p\left(J_S^d \middle| \omega_{nS}^d\right) dJ_S^d = \varepsilon_1 \tag{12}$$

$$\lambda_1 = \frac{p\left(t_1 \middle| \omega_S^d\right)}{p\left(t_1 \middle| \omega_{nS}^d\right)}. \tag{13}$$

So the Neyman-Pearson decision rule can be rewritten as

$$\begin{cases} p\left(J_S^d \middle| \omega_S^d\right) > \lambda_1 \cdot p\left(J_S^d \middle| \omega_{nS}^d\right), & \text{choosing } \omega_S^d \\ \text{else}, & \text{choosing } \omega_{nS}^d \end{cases}. \tag{14}$$

The class-conditional probability density function $p(J_S^d|\omega_i^d)$, $i \in \{S, nS\}$ and decision boundary t_1 are estimated by using nonparametric density estimation method which is described in Part C of this section.

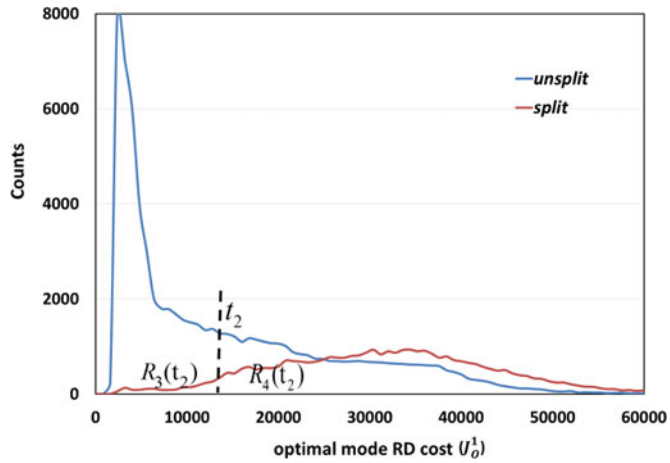

Fig. 6. Distributions of J_O^1 for *split* and *unsplit* CUs for BQTerrance with $\mathrm{QP} = 27$ and depth level $= 1$.

B. Early CU Size (Depth) Decision (ECUSD)

As we analyzed in Sub-Section III-B, the majority of CUs select depth level 0 and 1 as the optimal depth levels. Therefore, it is desirable to introduce an early CU size decision method to reduce the redundant complexity of small CU size checking.

The CU size decision also can be treated as a binary classification problem with two categories $W = \{\omega_{\mathrm{split}}^d, \omega_{\mathrm{unsplit}}^d\}$ for each $d \in \{0, 1, 2\}$. The category $\omega_{\mathrm{split}}^d$ represents the current CU needs to be split into four sub-CUs, while the category $\omega_{\mathrm{unsplit}}^d$ denotes that the current CU does not need to be further split. In our proposed ECUSD algorithm, the decision is conducted after the RDO checking of the current CU depth. The RD cost (J_O^d) of the optimal mode in the given CU depth d can be obtained and used as the criterion for ECUSD, which avoids additional computation.

To reveal the distribution of RD cost J_O^d, the HD sequence BQTerrace is encoded by HM16.0 with $\mathrm{QP} = 27$. Taking CU depth 1 as an example, the histogram distributions of the optimal RD cost J_O^1 in terms of categories $\omega_{\mathrm{split}}^1$ and $\omega_{\mathrm{unsplit}}^1$ are illustrated in Fig. 6. According to the figure, the histogram curves are similar to that of the *SKIP* mode decision. The optimal RD cost J_O^1 of *unsplit* class $\omega_{\mathrm{unsplit}}^1$ shows a concentrated distribution centered in a narrow RD cost range. Meanwhile, J_O^1 of *split* class $\omega_{\mathrm{split}}^1$ distributes in a relatively wide and flat RD cost range. Since the possibility of *unsplit* decision is correspondingly high at small J_O^1 range, the CUs with small J_O^1 are more likely to be *unsplit*. The decision regions of categories $\omega_{\mathrm{unsplit}}^1$ and $\omega_{\mathrm{split}}^1$ are denoted as $R_3(t_2)$ and $R_4(t_2)$ respectively. The dotted line denoted as t_2 is the boundary of the two decision regions.

It can be seen in Fig. 6 that the J_O^1 distributions of *split* and *unsplit* are overlapped in the range of large J_O^1. Making incorrect decision of *unsplit* in the region $R_3(t_2)$ will cause RD performance loss. While the missed detection of *unsplit* in the region $R_4(t_2)$ will only result in complexity rising and not affect RD performance. It is better to make the missed detection rate of *unsplit* as small as possible under the condition of limiting the incorrect decision rate of *unsplit*. In this case, a Neyman-Pearson based decision is adopted to make an early CU size decision. The boundary point t_2 and optimal Lagrange multiplier λ_2 are calculated as follows:

$$\int_{R_3(t_2)} p\left(J_O^d \middle| \omega_{\mathrm{split}}^d\right) dJ_O^d = \varepsilon_2 \tag{15}$$

$$\lambda_2 = \frac{p\left(t_2 \middle| \omega_{\mathrm{unsplit}}^d\right)}{p\left(t_2 \middle| \omega_{\mathrm{split}}^d\right)} \tag{16}$$

where ε_2 is the maximum acceptable incorrect decision rate of *unsplit* $p(J_O^d|\omega_j^d) j \in \{\mathrm{split}, \mathrm{unsplit}\}$ denotes the class-conditional probability density function of J_O^d given ω_j^d and the priori probability of class ω_j^d is estimated as follows:

$$p\left(\omega_j^d\right) = \frac{N_j^d}{\sum_j N_j^d} \tag{17}$$

where N_{split}^d and N_{unsplit}^d represent the number of *split* and *unsplit* CUs for the given depth d, respectively.

Thus, we obtain the Neyman-Pearson decision rule for CU size determination as

$$\begin{cases} p\left(J_O^d \middle| \omega_{\mathrm{unsplit}}^d\right) > \lambda_2 \cdot p\left(J_O^d \middle| \omega_{\mathrm{split}}^d\right), & \text{choosing } \omega_{\mathrm{unsplit}}^d \\ \text{else}, & \text{choosing } \omega_{\mathrm{split}}^d \end{cases} \tag{18}$$

where the class-conditional probability density function $p(J_O^d|\omega_j^d)$ and decision boundary t_2 are estimated using a non-parametric density estimation method.

C. Nonparametric Likelihood Estimation

In HEVC, the hierarchical group of picture (GOP) structure provides great coding efficiency. Our schemes are tested by using the two hierarchical GOP structures recommended by HEVC common test condition, i.e., "encoder_lowdelay _P_main" (LP-Main) and "encoder_randomaccess_main" (RA-Main) [26]. For the LP-Main configuration, all pictures are coded in display order as shown in Fig. 7(a). The QP of the lowest hierarchy level P pictures is increased by one relative to that of I pictures, and for each hierarchy level, the QP value increases one from that of the lower hierarchy level. For the hierarchical prediction structure of RA-Main as shown in Fig. 7(b), all pictures are coded as B pictures except at random access refresh points (where I pictures are used). Similarly, QPs of B pictures are derived by adding different offsets to the QPs of I pictures according to the hierarchy levels.

As mentioned in Section IV, the RD cost distributions are quite different for different QPs and CU depth levels. Therefore, an adaptive mechanism should be considered for the probability density function estimation. In our proposed algorithm, a training scheme is employed to record and estimate the statistical parameters of different QP offsets and CU depth levels. Within every m frames, the first inter frames encoded by different QP offsets (hierarchy levels) are applied for the training process individually. So that the rest frames within the m-frames period can use the statistical parameters corresponding to

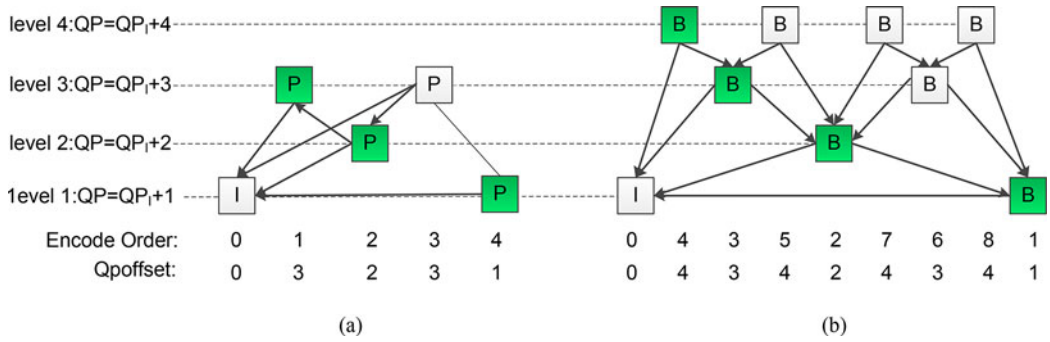

Fig. 7. Example of hierarchical temporal prediction structures and the selection of training frames (in green color). (a) LP-Main, (b) RA-Main.

different QP offsets and CU depth levels. For example, the first inter frames in each hierarchy level (pictures in green color) are used for the training process as shown in Fig. 7. During this procedure, statistical parameters $p(J_S^d|\omega_i^d)$, N_i^d, $i \in \{S, nS\}$ for $d \in \{0,1,2,3\}$ and $p(J_O^d|\omega_j^d), N_j^d, j \in \{\text{split}, \text{unsplit}\}$ for $d \in \{0,1,2\}$ are calculated for each QP offset and CU depth. Based on these statistical parameters, the Neyman-Pearson decisions can then be conducted on the rest frames. According to the video sequence temporal correlation, the statistics of the following frames have similar characteristics with the same hierarchically positioned trained frame in a short time. Meanwhile, the training period should not be too short, since the training process will employ full CU mode searching which will lead to complexity increase. In practice, the training period $m = 50$ is used to balance the complexity of the training process and the encoding performance.

According to the RD cost histogram distributions shown in Figs. 5 and 6, it's difficult to represent the distributions of different QPs and CU depth levels by using a particular probability function. Therefore, the nonparametric likelihood estimation is employed in our proposed algorithm, which does not rely on any particular distribution and can provide more accurate decision model.

Based on the data from the training process, the likelihood ratio function of the PDF $p(J_S^d|\omega_i^d), i \in \{S, nS\}$ and $p(J_O^d|\omega_j^d), j \in \{\text{split}, \text{unsplit}\}$ are estimated using nonparametric estimation methods [27]–[30]

$$p\left(J_S^d \middle| \omega_i^d\right) \cong \frac{H_i^d(k)}{N_i^d}, \quad i \in \{S, nS\} \tag{19}$$

$$p\left(J_O^d \middle| \omega_j^d\right) \cong \frac{H_j^d(l)}{N_j^d}, \quad j \in \{\text{split, unsplit}\} \tag{20}$$

where $H_i^d(k)$ is the number of PUs that are decided as the class ω_i^d for the kth interval of the normalized histogram. $H_j^d(l)$ is the number of CUs that are determined as the class ω_j^d for the lth interval of the normalized histogram.

For the ease of implementation and analysis, the RD cost values are right shifted into 800 bins to approximate the histogram of the RD cost distribution. Since the current CU is four times size of the next depth level CU and the RD cost values differ widely for different CU depth levels, the right shift factor is set to be decreased with steps of two from the large CUs to the small CUs. For each CU depth d, the shift factor is defined as

$$\text{shift}[d] = \{9, 7, 5, 3\}. \tag{21}$$

According to the previous analysis, our concerned RD cost is distributed in the range of relatively small values. By using the shift factor defined in (21), most portions of concerned RD cost are mapped in the range of 0 to 799 according to the practical statistical test. With this shift factor, the RD cost distribution in each bin does not show great statistical fluctuation and a good histogram can be estimated.

Moreover, the decision boundaries t_1 and t_2 play important roles in (12) and (15). In this algorithm, area percent boundary is adopted to calculate the decision boundaries [31]. The calculation of decision boundary t_1 of ESMD is expressed as

$$\begin{aligned} &\text{for } n = 0 : N-1 \\ &\qquad \text{if } \sum_{\substack{k=0 \\ \text{break}}}^{n} H_{nS}^d(k) > \varepsilon_1 \cdot \sum_{k=0}^{N-1} H_{nS}^d(k) \\ &\qquad\quad t_1 = n \end{aligned} \tag{22}$$

where N is the number of total intervals (800 here). $H_{nS}^d(k)$ is the number of PUs that are determined as the class ω_{nS}^d for the kth RD cost interval of the normalized histogram. k and n stand for the interval number ranging from 0 to 799, respectively. Since the RD cost values are right shifted to 0 to 799, k and n are also the shifted RD cost values. The boundary t_1 is determined as the shifted RD cost value n when the area percent is just greater than ε_1. While for ECUSD algorithm, the $H_{\text{split}}^d(l)$ denotes the number of CUs, which are going to be divided further for the lth interval of the RD range. The calculation of t_2 can be written as

$$\begin{aligned} &\text{for } n = 0 : N-1 \\ &\qquad \text{if } \sum_{\substack{l=0 \\ \text{break}}}^{n} H_{\text{split}}^d(l) > \varepsilon_2 \cdot \sum\nolimits_{l=0}^{N-1} H_{\text{split}}^d(l) \\ &\qquad\quad t_2 = n. \end{aligned} \tag{23}$$

D. Overall Algorithm

Based on the preceding analysis, the proposed scheme consists of two stages, which are the training stage and the fast mode

decision stage. During the training stage, the RDO process is performed for all sizes of CU and PU in a recursive manner to decide the optimal mode and to estimate the statistical parameters. Then the fast mode decision stage by employing ESMD and ECUSD is conducted. The flowchart of the overall algorithm is shown in Fig. 8. The detailed steps are summarized as follows.

Step 1: Start encoding a frame.
Step 2: Check whether the current frame is the trained frame. If it is, go to Step 3. Otherwise, go to Step 4.
Step 3: Training stage: Encode all sizes of the CU in a recursive manner to decide the optimal mode. The statistical parameters are collected based on various QPs and CU depth levels. Then, the normalized histograms are established for computing t_1, λ_1, t_2 and λ_2 according to (12), (13), (15) and (16).
Step 4: Check whether the current frame is inter prediction frame. If it is, start fast mode decision.
Step 5: Start to find the optimal CTU partition in the frame.
Step 6: Test the *SKIP* mode and obtain the RD cost J_S^d of the *SKIP* mode for the current CU depth d.
Step 7: If the current CU satisfies the early *SKIP* mode decision condition according to (14), go to Step 9.
Step 8: Check the remaining PU modes for the current depth.
Step 9: Choose the mode with the minimum RD cost (J_O^d) as the optimal mode for the current depth.
Step 10: If the CU splitting termination condition is satisfied according to (18), go to Step 11. Otherwise, go to Step 5 for the next CU depth checking.
Step 11: Determine the optimal CTU partition.

V. Experimental Results and Discussion

A. Test Conditions

In this section, the performances of proposed fast mode decision schemes are evaluated in terms of the encoding time reduction and Bjontegaard Delta bitrate (BDBR) [32]. The proposed algorithm is implemented on the HEVC test model HM16.0. The experimental conditions are based on the "encoder_randomaccess_main" (RA-Main) and "encoder_lowdelay_P_main" (LP-Main) settings and aligned with the common test conditions recommended by the *JCT-VC* [26]. The test sequences recommended by the *JCT-VC* with 5 resolutions (416×240, 832×480, 1280×720, 1920×1080, and 2560×1600) are adopted. Intel(R) Core(TM) i7-3770 CPU@3.4 GHz with windows 7 operating system is used as the test platform. The time savings are calculated as

$$\Delta T = \frac{T_{\text{reference}} - T_{\text{proposed}}}{T_{\text{reference}}} \tag{24}$$

where $T_{\text{reference}}$ denotes the total encoding time of the original reference software HM16.0, and T_{proposed} is the overall encoding time of the HM16.0 implemented with the proposed algorithm.

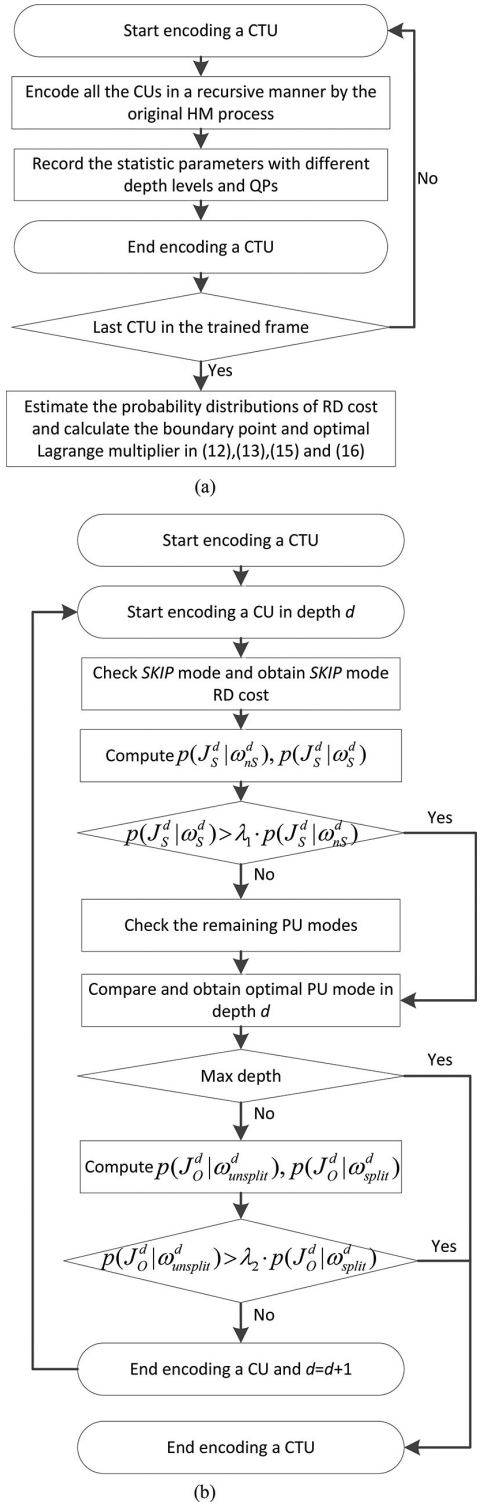

Fig. 8. Flowchart of the proposed overall algorithm. (a) Training process for statistical parameters. (b) Fast mode decision process.

B. Results of the Proposed Individual Algorithm

The individual performance of the proposed ESMD and ECUSD compared with the state-of-the-art fast algorithms under the RA-Main configuration are shown in Tables II and III, respectively. It can be seen from Tables II and III that the two proposed algorithms (ESMD and ECUSD) can greatly reduce

TABLE II
RESULTS OF PROPOSED ESMD COMPARED WITH THE STATE-OF-THE-ART FAST ALGORITHMS UNDER THE RA-MAIN CONFIGURATION

		ESD [14]		Kim [16]		Proposed ESMD with $\varepsilon_1 = 0.05$		Proposed ESMD with $\varepsilon_1 = 0.1$		Proposed ESMD with $\varepsilon_1 = 0.15$	
	Sequences	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)
Class A	PeopleOnStreet	0.32	26.13	0.43	21.69	0.30	28.12	0.35	37.10	0.59	39.80
(2560×1600)	Traffic	0.11	44.96	0.37	38.45	0.15	44.56	0.20	54.21	0.51	56.26
	BasketballDrive	0.34	31.80	0.30	27.03	0.23	31.65	0.28	39.15	0.66	41.89
	BQTerrace	0.18	38.98	0.60	36.94	0.13	36.86	0.16	42.22	0.41	45.09
Class B	Cactus	0.18	32.34	0.50	31.04	0.24	33.56	0.29	40.85	0.59	43.81
(1920×1080)	Kimonol	0.15	26.53	0.42	30.39	0.22	27.04	0.26	31.18	0.43	34.82
	ParkScene	0.20	36.76	0.31	31.36	0.23	36.86	0.27	41.67	0.61	43.91
	BasketballDrill	0.17	31.81	0.19	26.62	0.25	32.23	0.36	39.23	0.88	41.45
Class C	BQMall	0.23	31.64	0.43	26.92	0.23	32.05	0.28	38.88	0.62	40.69
(WVGA)	PartyScene	0.14	28.04	0.55	22.53	0.18	25.98	0.62	37.91	1.02	40.12
	RaceHorsesC	0.49	23.20	0.68	22.81	0.26	20.32	0.39	24.88	0.96	27.21
	BasketballPass	0.35	41.75	0.19	29.36	0.27	39.45	0.44	52.57	0.90	53.98
Class D	BQSquare	0.11	40.96	0.25	29.22	0.13	39.02	0.27	48.71	0.77	50.36
(WQVGA)	BlowingBubbles	0.23	26.93	0.46	21.48	0.23	25.31	0.36	30.84	0.78	32.78
	RaceHorses	0.37	17.85	0.68	16.75	0.38	17.21	0.46	19.49	0.91	21.44
	FourPeople	0.01	53.17	0.07	42.75	0.12	49.28	0.24	61.01	0.63	63.54
Class E	Johnny	0.07	57.73	0.08	41.57	0.15	58.55	0.59	67.04	0.93	69.82
(720P)	KristenAndSara	0.13	53.53	0.35	44.42	0.17	52.21	0.24	62.18	0.76	64.47
	Vidyo1	0.25	52.98	0.20	42.45	0.22	48.12	0.30	59.92	0.81	62.16
Average		**0.21**	**36.69**	**0.37**	**30.72**	**0.22**	**35.70**	**0.33**	**43.63**	**0.72**	**45.98**

TABLE III
RESULTS OF PROPOSED ECUSD COMPARED WITH THE STATE-OF-THE-ART FAST ALGORITHMS UNDER THE RA-MAIN CONFIGURATION

		PMD [8]		SVM [11]		Proposed ECUSD with $\varepsilon_2 = 0.05$		Proposed ECUSD with $\varepsilon_2 = 0.1$		Proposed ECUSD with $\varepsilon_2 = 0.15$	
	Sequences	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)
Class A	PeopleOnStreet	1.75	36.38	1.01	28.76	0.63	33.64	1.06	40.55	1.71	44.01
(2560x1600)	Traffic	2.03	45.39	1.14	56.79	0.55	63.60	0.81	65.31	1.44	68.24
	BasketballDrive	1.50	48.96	0.82	44.43	0.35	42.89	0.45	46.84	1.33	50.45
	BQTerrace	1.45	50.07	1.36	55.28	0.44	53.09	0.50	55.08	1.21	58.32
Class B	Cactus	2.10	44.12	1.40	47.21	0.58	44.81	0.83	47.37	1.44	50.66
(1920×1080)	Kimonol	1.86	42.86	1.55	43.89	0.46	36.82	0.56	38.47	1.41	44.17
	ParkScene	1.72	44.89	1.07	47.27	0.46	47.91	0.62	49.02	1.16	54.24
	BasketballDrill	2.12	44.36	0.66	37.96	0.33	41.80	0.66	43.71	0.95	44.58
Class C	BQMall	2.65	42.17	1.61	37.17	0.55	40.51	0.91	42.96	1.17	43.74
(WVGA)	PartyScene	2.35	37.47	1.46	31.07	0.50	35.40	0.69	36.94	1.05	38.51
	RaceHorsesC	3.80	32.46	2.76	28.78	0.54	28.71	0.79	30.90	1.12	34.42
	BasketballPass	1.35	45.90	0.15	41.24	0.61	50.83	0.74	51.68	1.02	53.96
Class D	BQSquare	1.06	44.28	0.75	39.00	0.38	49.70	0.53	51.41	0.94	53.88
(WQVGA)	BlowingBubbles	1.90	31.16	1.26	26.38	0.54	35.18	0.80	38.34	1.27	40.11
	RaceHorses	2.20	22.15	1.37	21.43	0.68	23.95	1.08	25.89	1.64	28.36
	FourPeople	0.70	65.40	0.62	64.52	0.16	58.93	0.62	61.67	0.90	63.22
Class E	Johnny	0.42	64.19	0.42	65.34	0.16	62.85	0.66	67.31	0.86	69.53
(720P)	KristenAndSara	1.11	66.59	0.91	67.09	0.12	63.76	0.63	68.27	0.75	70.35
	Vidyo1	0.63	64.10	0.46	65.27	0.44	59.81	1.21	63.91	1.44	65.86
Average		**1.72**	**45.94**	**1.09**	**44.68**	**0.45**	**46.01**	**0.74**	**48.72**	**1.20**	**51.40**

the computational complexity with little RD performance loss. For the ESMD method, the coding time has been reduced by 35.7%, 43.6%, and 45.9% on average, with the maximum acceptable incorrect decision rate $\varepsilon_1 = 0.05, 0.1$ and 0.15, respectively. Meanwhile, the RD performance almost has no loss in terms of BDBR (0.22%, 0.33% and 0.72% increase, respectively). From Table II, it is confirmed that the proposed ESMD method with $\varepsilon_1 = 0.1$ outperforms the ESD [14] and the Kim's algorithm [16] in terms of time saving with similar RD performance. As shown in Table III, the proposed ECUSD approach can save 46.0%, 48.7% and 51.4% encoding time with the maximum acceptable incorrect decision rate $\varepsilon_2 = 0.05, 0.1$ and 0.15, respectively. The loss of coding efficiency is negligible (with 0.45%, 0.74% and 1.20% BDBR increase, respectively). With $\varepsilon_2 = 0.05$, the proposed ECUSD method (0.45% BDBR increase) outperforms the PMD [8] (1.72% BDBR increase) and the SVM [11] (1.09% BDBR increase) in terms of BDBR with similar encoding time reduction.

It can be noticed that the proposed ESMD and ECUSD have more time saving for homogenous sequences (such as Traffic

TABLE IV
RESULTS OF PROPOSED OVERALL ALGORITHM WITH DIFFERENT INCORRECT DECISION RATES UNDER RA-MAIN AND LP-MAIN CONFIGURATIONS

		RA-Main						LP-Main					
		$\varepsilon_1 = 0.03$, $\varepsilon_2 = 0.03$		$\varepsilon_1 = 0.05$, $\varepsilon_2 = 0.05$		$\varepsilon_1 = 0.1$, $\varepsilon_2 = 0.05$		$\varepsilon_1 = 0.03$, $\varepsilon_2 = 0.03$		$\varepsilon_1 = 0.05$, $\varepsilon_2 = 0.05$		$\varepsilon_1 = 0.1$, $\varepsilon_2 = 0.05$	
	Sequences	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR (%)	ΔT (%)	BDBR	ΔT (%)
Class A (2560×1600)	PeopleOnStreet	0.46	41.10	0.96	52.43	1.35	55.68	0.42	39.41	0.86	43.25	1.33	50.12
	Traffic	0.68	58.42	0.99	67.51	1.37	70.21	0.65	45.44	0.76	56.71	1.15	61.88
	BasketballDrive	1.43	45.86	1.51	54.23	1.77	60.10	0.99	38.46	1.20	49.24	1.46	54.25
	BQTerrace	0.40	54.85	0.81	63.90	1.08	66.80	0.25	48.41	0.41	56.10	0.55	60.10
Class B (1920×1080)	Cactus	0.75	48.73	1.56	58.14	1.67	62.21	0.66	41.39	1.30	50.12	1.56	53.76
	Kimonol	0.52	38.17	1.25	43.46	1.59	50.62	0.26	30.66	0.65	35.53	1.02	42.82
	ParkScene	0.46	51.18	1.16	61.88	1.33	66.04	0.40	45.89	0.89	54.30	1.08	57.59
	BasketballDrill	0.92	53.59	1.18	60.26	1.25	62.97	0.88	48.87	0.94	54.03	1.09	55.58
Class C (WVGA)	BQMall	1.19	55.96	1.71	59.34	1.77	61.82	1.23	50.08	1.47	52.74	1.56	55.72
	PartyScene	0.63	47.81	0.88	54.22	0.95	57.46	0.82	42.70	0.95	45.90	1.03	48.97
	RaceHorsesC	1.37	44.53	1.97	52.73	2.04	56.13	0.93	42.60	1.17	46.45	1.19	50.04
	BasketballPass	0.77	58.76	1.01	64.00	1.19	67.87	0.65	55.42	0.82	57.9	0.92	59.18
Class D (WQVGA)	BQSquare	0.37	58.83	0.39	61.63	0.49	66.57	0.31	49.93	0.37	53.21	0.50	54.04
	BlowingBubbles	0.95	50.88	1.56	53.62	1.66	57.48	0.84	41.61	1.39	45.25	1.48	48.12
	RaceHorses	1.03	41.03	1.89	47.53	1.92	52.30	0.69	40.34	1.02	42.70	1.12	45.64
	FourPeople	0.22	75.39	0.48	79.87	0.53	81.61	0.21	66.73	0.29	73.96	0.36	76.55
Class E (720P)	Johnny	0.51	71.02	0.77	80.00	0.83	82.08	0.45	69.04	1.14	74.99	1.32	76.16
	KristenAndSara	0.58	70.93	0.70	78.03	0.76	80.04	0.35	66.36	0.63	73.60	0.81	75.41
	Vidyo1	0.46	72.93	0.83	77.02	1.05	79.50	0.28	67.07	0.50	75.51	0.96	77.21
Average		**0.72**	**54.74**	**1.14**	**61.57**	**1.29**	**65.13**	**0.60**	**48.97**	**0.87**	**54.82**	**1.08**	**58.06**

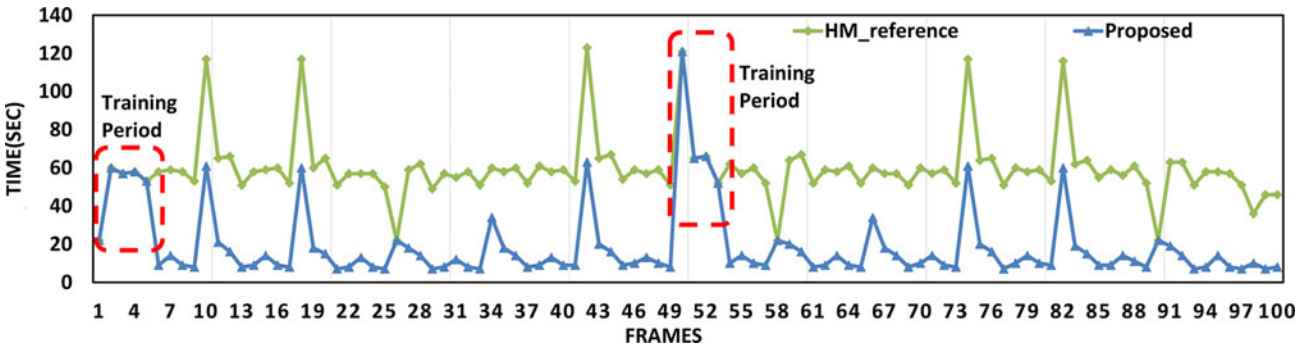

Fig. 9. Encoding time comparison of each frame for sequence "Traffic" at QP 32 under the RA-Main configuration.

and BQTerrace). For these homogenous sequences, *SKIP* mode takes higher percentage and more large CUs can be early terminated by the proposed schemes.

C. Results of the Proposed Overall Algorithm

In the following section, the early decision schemes ESMD and ECUSD are jointly implemented in HM16.0 to evaluate the overall performance when further coding time reductions are required. To show the ability of balancing between time saving and RD performance loss, three sets of different ε_1 and ε_2 values are tested and the results are listed in Table IV. It can be noticed that the time saving is going up with the increasing of the incorrect decision rates ε_1 and ε_2, and the BDBR increment is less with smaller ε_1 and ε_2.

With the set of $\varepsilon_1 = 0.1, \varepsilon_2 = 0.05$, the proposed algorithm can reduce the coding time by 65.13% and 58.06% with only 1.29% and 1.08% BDBR increment on average under the RA-Main and LP-Main configurations, respectively. In particular, for slow motion sequences (such as Traffic, BQTerrace and Johnny), the time saving is more than 66% with little RD performance loss. For high activity sequences (such as PeopleOnStreet, BasketballPass and RaceHorses), the proposed algorithm also can save more than 52% encoding time. The great reduction of the computational complexity illustrates that the proposed algorithm is efficient for both slow and fast motion sequences.

To further investigate the complexity reduction performance, the frame by frame encoding time comparison of HM and the proposed algorithm with $\varepsilon_1 = 0.1, \varepsilon_2 = 0.05$ is demonstrated in Fig. 9. The frames highlighted in the red rectangle are the training stages of the proposed algorithm, where the full mode decisions are conducted and the encoding time is almost the same as HM. After these periods, the time consumption of proposed one is decreased dramatically.

The results are compared with the state-of-the-art fast algorithms, i.e., the Ahn's algorithm [20], the Shen's algorithm [21],

TABLE V
RESULTS OF PROPOSED ALGORITHM COMPARED WITH THE STATE-OF-THE-ART FAST ALGORITHMS UNDER THE RA-MAIN CONFIGURATION

		Ahn [20]		Shen [21]		MRF [22]		fast HEVC		Proposed Scheme with $\varepsilon_1 = 0.1, \varepsilon_2 = 0.05$	
	Sequences	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)
Class A	PeopleOnStreet	1.35	31.56	1.14	37.91	1.86	52.38	1.96	34.67	1.35	55.68
(2560×1600)	Traffic	1.67	59.27	0.93	59.75	3.40	68.78	1.78	65.19	1.37	70.21
	BasketballDrive	0.93	48.88	0.92	49.46	3.71	63.34	1.14	47.02	1.77	60.10
	BQTerrace	1.72	59.26	1.11	58.90	2.47	67.60	1.09	62.67	1.08	66.80
Class B	Cactus	1.87	52.33	0.76	53.30	3.51	63.11	1.99	52.01	1.67	62.21
(1920×1080)	Kimonol	1.97	49.30	0.83	51.50	3.79	66.73	1.13	41.44	1.59	50.62
	ParkScene	1.47	51.67	0.81	56.84	2.98	66.52	1.69	58.16	1.33	66.04
	BasketballDrill	0.96	42.67	0.62	36.18	3.43	57.60	1.19	43.48	1.25	62.97
Class C	BQMall	2.08	40.98	0.97	47.95	2.68	52.24	2.10	47.59	1.77	61.82
(WVGA)	PartyScene	1.86	35.48	0.82	43.38	1.58	50.86	1.30	39.15	0.95	57.46
	RaceHorsesC	2.58	32.23	1.13	34.35	3.07	49.87	2.04	29.08	2.04	56.13
	BasketballPass	0.20	44.57	0.80	33.56	1.81	64.90	2.39	58.09	1.19	67.87
Class D	BQSquare	0.78	42.61	0.68	48.54	1.08	60.74	1.03	57.85	0.49	66.57
(WQVGA)	BlowingBubbles	1.47	30.10	0.79	41.02	2.63	49.76	1.73	35.19	1.66	57.48
	RaceHorses	1.57	25.50	1.15	29.75	2.12	46.03	2.40	26.98	1.92	52.30
	FourPeople	0.76	68.53	0.43	69.21	1.76	78.19	0.21	78.49	0.53	81.61
Class E	Johnny	0.55	68.59	0.54	73.56	2.87	79.28	0.71	78.44	0.83	82.08
(720P)	KristenAndSara	1.19	70.42	0.66	68.23	1.98	77.22	0.72	78.15	0.76	80.04
	Vidyo1	0.62	69.58	0.48	72.92	2.81	77.40	0.67	78.10	1.05	79.50
Average		**1.35**	**48.61**	**0.82**	**50.86**	**2.61**	**62.77**	**1.44**	**53.25**	**1.29**	**65.13**

TABLE VI
RESULTS OF PROPOSED ALGORITHM COMPARED WITH THE STATE-OF-THE-ART FAST ALGORITHMS UNDER THE LP-MAIN CONFIGURATION

		Ahn [20]		Shen [21]		MRF [22]		fast HEVC		Proposed Scheme with $\varepsilon_1 = 0.1, \varepsilon_2 = 0.05$	
	Sequences	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)	BDBR(%)	ΔT(%)
Class A	PeopleOnStreet	1.67	28.87	1.14	36.11	1.66	53.27	1.04	25.44	1.33	50.12
(2560×1600)	Traffic	2.03	53.03	1.36	52.23	2.56	58.23	1.01	54.21	1.15	61.88
	BasketballDrive	1.71	44.89	0.75	42.55	3.43	60.90	0.79	37.03	1.46	54.25
	BQTerrace	2.23	50.76	1.67	49.92	2.10	60.05	1.33	52.34	0.55	60.10
Class B	Cactus	2.46	47.28	1.09	46.00	3.10	58.42	1.57	40.98	1.56	53.76
(1920×1080)	Kimonol	1.96	45.53	0.58	45.84	2.87	63.43	0.38	31.45	1.02	42.82
	ParkScene	1.96	46.33	1.11	48.57	2.58	61.16	1.40	48.36	1.08	57.59
	BasketballDrill	1.07	39.07	0.84	37.21	2.43	54.78	0.56	33.68	1.09	55.58
Class C	BQMall	2.39	38.05	1.26	42.65	1.97	50.07	1.16	37.20	1.56	55.72
(WVGA)	PartyScene	2.14	29.23	1.15	31.08	1.17	46.03	0.97	26.48	1.03	48.97
	RaceHorsesC	2.29	33.01	0.85	32.41	1.91	46.50	0.85	21.84	1.19	50.04
	BasketballPass	0.73	39.13	0.95	31.73	1.26	61.75	1.21	48.50	0.92	59.18
Class D	BQSquare	1.00	30.90	1.74	38.71	1.29	54.35	0.87	41.93	0.50	54.04
(WQVGA)	BlowingBubbles	1.41	27.49	1.33	34.18	2.14	47.41	1.31	25.78	1.48	48.12
	RaceHorses	2.13	24.80	1.13	31.15	1.58	44.13	1.15	19.70	1.12	45.64
	FourPeople	1.66	65.34	1.41	61.90	1.32	73.45	0.09	71.25	0.36	76.55
Class E	Johnny	0.69	65.47	1.63	67.29	2.28	74.87	0.55	71.03	1.32	76.16
(720P)	KristenAndSara	1.62	67.90	1.54	62.40	1.69	73.41	0.46	70.50	0.81	75.41
	Vidyo1	0.85	65.95	1.34	60.66	2.55	74.02	0.18	71.41	0.96	77.21
Average		**1.68**	**44.37**	**1.20**	**44.87**	**2.10**	**58.75**	**0.89**	**43.64**	**1.08**	**58.06**

the MRF based algorithm [22] and the fast HEVC algorithm in HM which enables the method of ECU [6], ESD [14] and CFM [17]. The Ahn's algorithm [20] is based on spatio-temporal encoding parameters. The Shen's algorithm [21] is designed using the coding information from the adjacent CUs including parent CUs in the upper depth level, spatially CUs and temporally adjacent CUs. The MRF based algorithm [22] uses the PVAD as a feature to select CU mode. The results of the Ahn's algorithm [20], the Shen's algorithm [21], the MRF [22] and the fast HEVC are shown in the first four columns in Tables V and VI. Among these four previous works under the RA-Main configuration, the MRF achieves the best coding time saving and the Shen's algorithm [21] has the best RD performance.

With the set of $\varepsilon_1 = 0.1$ and $\varepsilon_2 = 0.05$, our proposed method can achieve more complexity reduction when compared to the MRF method under the RA-Main configuration, which is 65.13% versus 62.77% in time saving. Meanwhile, the RD performance is much better than that of the MRF, which is 1.29% compared to 2.61% in BDBR increase. In comparison with the fast HEVC and the Ahn's algorithm, our proposed algorithm

TABLE VII
RESULTS OF PROPOSED ALGORITHM WITH $\varepsilon_1 = 0.1, \varepsilon_2 = 0.05$ FOR SCENE-CUT SEQUENCES UNDER THE RA-MAIN AND LP-MAIN CONFIGURATIONS

	RA-Main		LP-Main	
Sequences	BDBR(%)	ΔT (%)	BDBR (%)	ΔT (%)
BasketballPass_BlowingBubbles	1.35	57.98	1.11	61.32
PartyScene_BasketballDrill	1.22	53.89	1.11	57.69
Cactus_BasketballDrive	2.75	64.41	2.99	67.01
Kimono1_ParkScene	1.71	52.76	1.55	54.38
PeopleOnStreet_Traffic	1.44	56.09	1.84	59.73
Average	**1.69**	**57.03**	**1.72**	**60.03**

can achieve about 12%–16% more coding time saving with better RD performance. Furthermore, the proposed algorithm can reduce 54.74% encoding time with only 0.72% bitrate increase with $\varepsilon_1 = 0.03$ and $\varepsilon_2 = 0.03$ as shown in the first column in Table IV, which is better than Shen's algorithm both in time saving and bitrate increase. For homogenous sequences, the proposed overall algorithm can save up to 80% coding time for the sequences in Class E as shown in Table V.

For LP-Main configuration, the proposed method presents much better performance as shown in Table VI. With the set of $\varepsilon_1 = 0.1$ and $\varepsilon_2 = 0.05$, our proposed algorithm achieves 58.06% total encoding time reduction with only 1.08% BDBR increase which are both better than the Ahn's algorithm [20] the Shen's algorithm [21]. The MRF based algorithm [22] reduces the coding time by 58.75%, but the BDBR increase is 2.1% which is more than the proposed. Compared to the fast HEVC, the proposed algorithm can achieve about 14% more encoding time saving with only 0.2% BDBR increase.

The proposed method also has the advantage of making the trade-off between the RD performance and time saving. As shown in Table IV, the time saving and the BDBR increment are going up with the increasing of the incorrect decision rates ε_1 and ε_2. The bigger the incorrect decision rates are, the more encoding time can be saved, but with a little more RD performance loss. Therefore, if we want to maintain the RD performance as much, smaller values for the incorrect decision rates are preferred, such as $\varepsilon_1 = 0.03$ and $\varepsilon_2 = 0.03$. However, if encoding complexity is the key issue, then relatively bigger values for ε_1 and ε_2 can be chosen for more time saving.

Finally, the performance of the proposed algorithm is also tested for scene-cut sequences with different resolutions and frame rates. As shown in Table VII, the five scene-cut sequences are generated by cascading two different sequences for scene change, and every 10 frames has a scene cut frame [21]. It can be seen that the proposed algorithm can reduce 57.03% and 60.03% coding time on average under the RA-Main and LP-Main configurations, respectively. The average increase of BDBR are 1.69% and 1.72%.

VI. CONCLUSION

In this paper, the Neyman-Pearson based fast mode decision algorithms, ESMD and ECUSD, are proposed for HEVC inter prediction. Both *SKIP* mode decision and CU size decision are modeled as the binary classification problems, and the RD cost recorded in the encoding process is used as the decision feature. The two cases of misclassification, i.e., missed detection and incorrect decision, have influence on the encoding complexity and RD performance respectively. Thus, the Neyman-Pearson rule is employed to balance the RD performance loss and complexity reduction by making the missed detection rate as small as possible under the condition of limiting the incorrect decision rate. Also, online training process and nonparametric likelihood estimation are employed to update the RD cost probability density distribution for each QP and CU depth. Experimental results have demonstrated that the proposed overall algorithm can significantly reduce the computational complexity by 58%–65% on average with negligible RD performance loss (1.08%–1.29% BDBR increase) and has the advantage of balancing RD performance and encoding time.

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[2] *High Efficiency Video Coding Text Specification Draft 10*, JCTVC-L1003, ISO/IEC-JCT1/SC29/WG11, 2013.

[3] J. Ohm, G. J. Sullivan, H. Schwarz, T. T. Keng, and T. Wiegand, "Comparison of the coding efficiency of video coding standards—including high efficiency video coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[4] G. J. Sullivan, J. Ohm, H. Woo-Jin, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[5] *HM7: High Efficiency Video Coding (HEVC) Test Model 7 Encoder Description*, JCTVC-I1002, Apr. 2012.

[6] *Coding Tree Pruning Based CU Early Termination*, JCTVC-F092, Jul. 2011.

[7] L. Shen, Z. Liu, X. Zhang, W. Zhao, and Z. Zhang, "An effective CU size decision method for HEVC encoders," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 465–470, Feb. 2013.

[8] J. Xiong, H. Li, Q. Wu, and F. Meng, "A fast HEVC inter CU selection method based on pyramid motion divergence," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 559–564, Feb. 2014.

[9] Y. Zhang, H. Wang, and Z. Li, "Fast coding unit depth decision algorithm for interframe coding in HEVC," in *Proc. Data Compression Conf.*, 2013, pp. 53–62.

[10] M. B. Cassa, M. Naccari, and F. Pereira, "Fast rate distortion optimization for the emerging HEVC standard," in *Proc. Picture Coding Symp.*, 2012, pp. 493–496.

[11] X. Shen and L. Yu, "CU splitting early termination based on weighted SVM," *EURASIP J. Image Video Process.*, vol. 2013, no. 4, pp. 1–11, 2013.

[12] X. Shen, L. Yu, and J. Chen, "Fast coding unit size selection for HEVC based on Bayesian decision rule," in *Proc. Picture Coding Symp.*, 2012, pp. 453–456.

[13] Y. Zhang *et al.*, "Machine learning-based coding unit depth decisions for flexible complexity allocation in high efficiency video coding," *IEEE Trans. Image Process.*, vol. 24, no. 7, pp. 2225–2238, Jul. 2015.

[14] J. Kim, J. Yang, K. Won, and B. Jeon, "Early determination of mode decision for HEVC," in *Proc. Picture Coding Symp.*, 2012, pp. 449–452.

[15] J. Vanne, M. Viitanen, and T. D. Hamalainen, "Efficient mode decision schemes for HEVC inter prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1579–1593, Sep. 2014.

[16] M. Kim, N. Ling, L. Song, and Z. Gu, "Fast skip mode decision with rate-distortion optimization for high efficiency video coding," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops*, Jul. 2014, pp. 1–6.

[17] *Early Termination of CU Encoding to Reduce HEVC Complexity*, JCTVC-F045, Jul. 2011.

[18] G. Correa, P. Assuncao, L. Agostini, and L. da Silva Cruz, "Fast HEVC encoding decisions using data mining," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 4, pp. 660–673, Apr. 2015.

[19] J. Lee, S. Kim, K. Lim, and S. Lee, "A fast CU size decision algorithm for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 411–421, Mar. 2015.

[20] S. Ahn, B. Lee, and M. Kim, "A novel fast CU encoding scheme based on spatiotemporal encoding parameters for HEVC inter coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 422–435, Mar. 2015.
[21] L. Shen, Z. Zhang, and Z. Liu, "Adaptive inter-mode decision for HEVC jointly utilizing inter-level and spatio-temporal correlations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 10, pp. 1709–1722, Oct. 2014.
[22] J. Xiong *et al.*, "MRF-based fast HEVC inter CU decision with the variance of absolute differences," *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2141–2153, Dec. 2014.
[23] Q. Hu, X. Zhang, Z. Gao, and J. Sun, "Analysis and optimization of x265 encoder," in *Proc. Visual Commun. Image Process. Conf.*, 2014, pp. 502–505.
[24] J. Xiong, H. Li, F. Meng, Q. Wu, and K. N. Ngan, "Fast HEVC inter CU decision based on latent SAD estimation," *IEEE Trans. Multimedia*, vol. 17, no. 12, pp. 2147–2159, Dec. 2015.
[25] L. Shen, Z. Zhang, and Z. Liu, "Effective CU size decision for HEVC intracoding," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4232–4241, Oct. 2014.
[26] *Common HM Test Conditions and Software Reference Configurations*, JCTVC-K1100, Oct. 2012.
[27] D. Chai, S. L. Phung, and A. Bouzerdoum, "A Bayesian skin/non-skin color classifier using non-parametric density estimation," in *Proc. Int. Symp. Circuits Syst.*, 2003, vol. 2, pp. II-464–II-467.
[28] R. M. Gray and R. A. Olshen, "Vector quantization and density estimation," in *Proc. Compression Complexity Sequences*, 1997, pp. 172–193.
[29] Q. Xie, C. A. Laszlo, and R. K. Ward, "Vector quantization technique for nonparametric classifier design," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 12, pp. 1326–1330, Dec. 1993.
[30] G. Corder, and D. Foreman, *Nonparametric Statistics for Non-Statisticians: A Step-by-Step Approach*, New York, NY, USA: Wiley, 2009.
[31] X. Hou, and Y. Xue, "Fast coding unit partitioning algorithm for HEVC," in *Proc. IEEE Int. Conf. Consum. Electron.*, Jan. 2014, pp. 7–10.
[32] *Calculation of Average PSNR Differences Between RD Curves*, ITU-T SC16/Q6, VCEG-M33, Apr. 2001.

Qiang Hu received the B.S. degree in electrical and information engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2013, and is currently working toward the Ph.D. degree in electrical engineering from the Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University, Shanghai, China.

His current research interests include image and video coding and perceptual video coding and processing.

Xiaoyun Zhang (M'13) received the B.S. and M.S. degrees in applied mathematics from Xi'an Jiaotong University, Xi'an, China, in 1998 and 2001, respectively, and the Ph.D. degree in pattern recognition from Shanghai Jiaotong University, Shanghai, China, in 2004.

She is currently an Associate Professor with the Department of Electronic and Engineering, Shanghai Jiao Tong University, Shanghai, China. Her research interests include video compression, image and video processing, and the implementation in many core/GPU platforms.

Zhiru Shi received the B.S. degree in information and computing science from Qufu Normal University, Qufu, China, in 2008, and the Ph.D. degree from the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, U.U.K., in 2014.

He is currently an Assistant Researcher with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China. Prior to that, he was a Postdoctoral Researcher with the Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University, Shanghai, China. His current research interests include 3D video coding, video quality assessment, and high performance video transcoding system.

Zhiyong Gao received the B.S. and M.S. degrees in electrical engineering from the Changsha Institute of Technology (CIT), Changsha, China, in 1981 and 1984, respectively, and the Ph.D. degree from Tsinghua University, Beijing, China, in 1989.

From 1994 to 2010, he took several senior technical positions in England, including a Principal Engineer with Snell & Wilcox, Petersfield, U.K., from 1995 to 2000, a Video Architect with 3DLabs, Egham, U.K., from 2000 to 2001, a Consultant Engineer with Sony European Semiconductor Design Center, Basingstoke, U.K., from 2001 to 2004, and a Digital Video Architect with Imagination Technologies, Kings Langley, U.K., from 2004 to 2010. Since 2010, he has been a Professor with Shanghai Jiao Tong University. His research interests include video processing and its implementation, video coding, digital TV, and broadcasting.