Hybrid Intraprediction Based on Local and Nonlocal Correlations

Tao Zhang, Xiaopeng Fan, *Senior Member, IEEE*, Debin Zhao, *Member, IEEE*, Ruiqin Xiong, *Member, IEEE*, and Wen Gao, *Fellow, IEEE*

***Abstract*—In the latest video coding standard, namely, high-efficiency video coding (HEVC), intra coding efficiency is significantly improved by a quadtree partition structure and more intra prediction modes. For intra coding, 35 intra modes are employed including 33 angular intra prediction (AIP) modes that are effective for blocks with strong directions, and a planar mode and a dc mode that are used to predict smooth regions. However, intra prediction in HEVC still cannot handle complicated blocks well. To deal with this problem, this paper proposes a hybrid intra prediction method to improve the intra prediction efficiency. The proposed hybrid intra prediction method consists of three parts: adaptive template matching prediction (ATMP) by exploring nonlocal correlation, combined local and nonlocal prediction by exploring both local correlation for AIP and nonlocal correlation for ATMP, and combined neighboring modes prediction that can generate smooth prediction by exploring more local correlations. Experimental results suggest that the proposed hybrid intra prediction method achieves 2.8% BD-rate reduction on average for luma compared to HEVC reference software HM-14.0 under all intra main configurations. The gain can be up to 6.9%. When integrated into joint exploration model −1.0, the proposed method still can achieve 1.2% BD-rate reduction for luma.**

***Index Terms*—High efficient video coding (HEVC), intra coding, intra prediction, template matching prediction.**

I. Introduction

THE latest High Efficiency Video Coding (HEVC) Standard [1] which was developed by the Joint Collaborative Team on Video Coding (JCT-VC) improves the video coding performance significantly. Compared to previous video coding standard H.264/AVC [2], HEVC achieves about 50% bit rate reduction under the same perceptual video quality. Intra coding plays an important role in recent video coding standards. In HEVC intra coding, a frame is divided into large non-overlapping blocks called coding tree units (CTUs). The CTUs are further partitioned into sub-blocks called coding units (CUs) according to a quadtree-based structure. Each CU can be recursively divided into four equal-sized sub-CUs. The size of the coding unit (CU) can be from 8×8 to 64×64. A CU is further divided into one or four prediction units (PU) and each PU can have different prediction modes. In intra coding, the size of the PU can be the same as the CU which is indicated by 2N × 2N when the CU is not a smallest coding unit (SCU). Otherwise, the size of the PU can be 2N × 2N or N × N. Transform unit (TU) is the basic unit used in the transform which is generated by the quadtree partition of the CU, and the size of TU can be from 4×4 to 32×32. The quadtree partitions make the block-size more flexible and adaptive to the characteristics of the content, which improves the coding efficiency significantly.

In the modern video coding standards such as HEVC and H.264/AVC, intra prediction is generated by copying the reference pixels in previously coded boundaries of local blocks along an angular direction. In HEVC intra prediction [3], there are 35 modes in total, as shown in Fig. 1. Among these intra prediction modes, 33 angular intra prediction (AIP) modes are effective for blocks with directional patterns, and planar mode and DC mode are used to predict smooth regions. Compared to only 9 modes in H.264/AVC, the intra prediction efficiency in HEVC is significantly improved since it can handle blocks with more directions well. However, for complicated blocks, e.g., blocks with complex textures, HEVC intra prediction still cannot work well.

Several methods [4]–[14] have been proposed to further improve the intra coding efficiency. Most of these methods can be considered as filtering-based intra prediction, in which local correlation between pixels is explored. The filtering-based intra prediction methods predict the pixels by weighted average of boundary pixels of the coding block or neighboring pixels within the coding block. Recursive extrapolation approach was proposed in [4], [5]. In this method, image signal is considered as a 2-D non-separable Markov model, and a three-tap filter [4] or a four-tap filter [5] is applied to the nearest neighbors of each pixel to get the prediction for the pixels in the block. The coefficients of the filters are trained offline. Based on the statistical model trained from real images, position-dependent filtering intra prediction method was proposed in [6], in which each position under different block sizes and prediction modes employs

Manuscript received April 26, 2017; revised August 11, 2017; accepted October 31, 2017. Date of publication November 20, 2017; date of current version June 15, 2018. This work was supported in part by the Major State Basic Research Development Program of China (973 Program) under Grant 2015CB351804. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Balakrishnan Prabhakaran. *(Corresponding author: Debin Zhao.)*

T. Zhang was with the Department of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China. He is now with the Tencent, Beijing 100080, China (e-mail: taozhang.hit@hotmail.com).

X. Fan and D. Zhao are with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: fxp@hit.edu.cn; dbzhao@hit.edu.cn).

R. Xiong and W. Gao are with the School of Electronics Engineering and Computer Science, Peking University, Beijing 100080, China (e-mail: rqxiong@pku.edu.cn; wgao@pku.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.2017.2775223

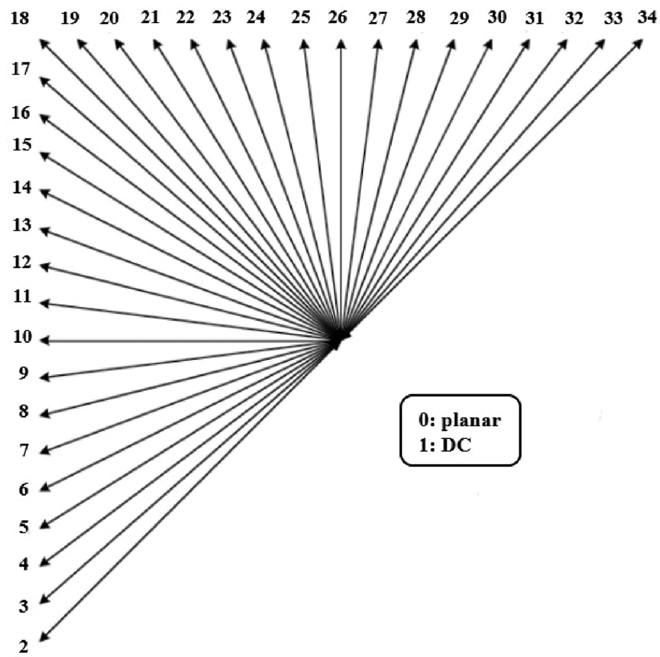


Fig. 1. Intra prediction modes in HEVC.

its own filtering weights. In [7], PDE-based image inpainting method was used in intra prediction with the assumption that image blocks are usually smooth so that the prediction can be generated by solving a PDE problem based on the known boundaries. Based on the analysis that smoothing the copying-based prediction can derive a better prediction, a block-size-dependent iterative filtering method was proposed to smooth the prediction generated by HEVC intra prediction in [8]. A block split based intra coding method was proposed in [9]. This method codes only a half of pixels in each prediction block, while the other half pixels are reconstructed by linear interpolations utilizing the spatial correlation of neighboring pixels. A generalized intra prediction framework based on sparse linear models and geometric transformation was proposed to replace the 33 angular intra prediction modes in HEVC intra coding in [10]. This method unifies the angular HEVC prediction modes and least-squares prediction [11]. A combined intra prediction method was proposed in [12] to exploit the redundancy between neighboring blocks and also between pixels within a coding block. In this method, the predictors generated by HEVC intra prediction and the weighted average of neighboring pixels are combined to get a new predictor. The performance of this method becomes marginal when the residual quad-tree (RQT) structure [1] was adopted in HEVC. RQT can improve the prediction accuracy since the reference pixels become closer to the predicted block. Another method which aims at shortening the distance between the predicted pixels and the references was proposed in [13], in which a non-square block splitting method under a quadtree block structure is employed. The reconstructed non-square block can be used as the reference for the neighboring block. Therefore, the distance between the reference samples and the local samples is reduced, which leads to the improved prediction accuracy. Bi-directional prediction [14] which combines two intra prediction modes was proposed for H.264/AVC intra coding.

In addition to the above mentioned filtering-based methods, some other methods [15]–[18] which exploit non-local correlation for intra prediction were proposed. A block matching based intra prediction was proposed in [15]. In this method, the reconstructed part of the frame is used as the reference, and the prediction for current block is searched in the reference by block matching. The derived block vector which indicates the position of the best matched block needs to be encoded. This prediction method also called intra block copy (IBC) was adopted in the HEVC screen content coding extension [24] because of its high coding performance for screen content coding (SCC) [33]–[35]. Another block matching based method called template matching prediction (TMP) was firstly proposed as an extra prediction mode for H.264/AVC [16], [17]. In TMP, the neighboring boundary pixels of the coding block are taken as the template and the corresponding blocks in the reconstructed region of the same frame with the best matched templates are considered as the prediction candidates for the coding block. Since the template matching process can be also performed at the decoder side, the overhead for signaling the prediction candidate can be neglected.

Our previous work [18] proposed a combined TMP and AIP prediction, which can provide more accurate predictions than TMP and AIP for some blocks with complicated textures. In this paper, we extend the work in [18] by analyzing local and non-local correlations between coding blocks and then propose a hybrid intra prediction method to further improve the prediction efficiency of intra coding. Firstly, in order to predict blocks with complex textures, an adaptive template matching prediction method (ATMP) by exploring non-local correlation is proposed. Secondly, a combined local and non-local prediction (CLNP) method is proposed to improve the prediction efficiency for blocks with both complex textures and strong local features, in which both non-local and local correlations are exploited simultaneously. Furthermore, a combined neighboring modes prediction (CNMP) method is proposed to predict blocks with blurring edges by using more local correlations. Experimental results suggest that compared to HEVC reference software HM-14.0, the proposed hybrid intra prediction method achieves 2.8%, 3.3% and 3.4% BD-rate reduction for Y, U and V under all intra main configuration, respectively. The proposed method can also achieve 1.0%, 1.3% and 1.5% BD-rate reduction for Y, U and V under random access configuration, respectively. When the proposed method is integrated into JEM-1.0 [27], [28] which is a test model for exploring future video coding technology, the coding gains are 1.2%, 2.1% and 2.3% for Y, U and V under all intra main configuration, and 0.4%, 1.1% and 0.6% for Y, U and V under random access configuration.

The rest of the paper is organized as follows. Section II gives motivation for the proposed hybrid intra prediction method. In Section III, the proposed method is described in detail. Experimental results are given in Section IV and Section V concludes the work in this paper.

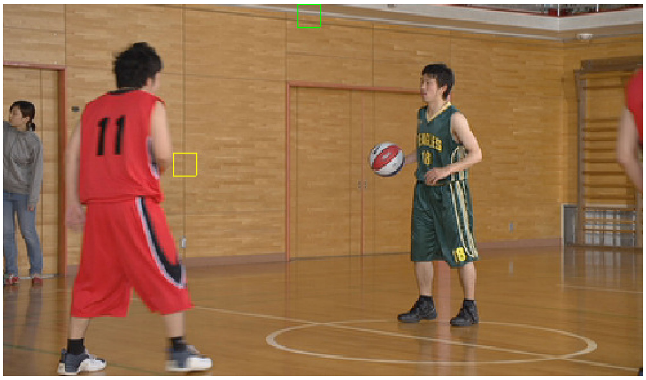


Fig. 2. The first frame of sequence *BasketballPass*.

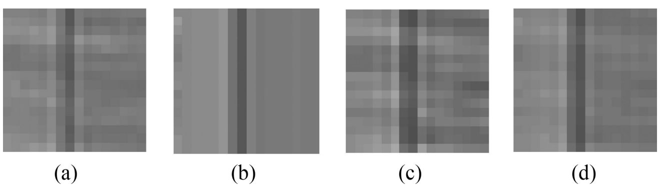


Fig. 3. An example of (a) luma block and its predictions by (b) AIP with mode 26; (c) ATMP; (d) CLNP.

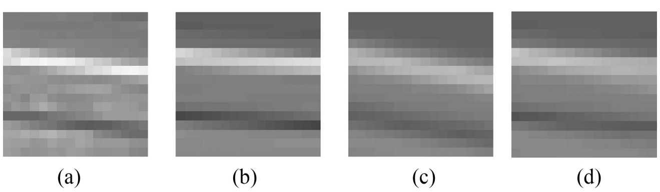


Fig. 4. An example of (a) luma block and its predictions by (b) AIP with mode 11; (c) AIP with mode 12; (d) CNMP with mode 11 and 12.

II. Motivation

As mentioned before, AIP in HEVC can provide good predictions for blocks with strong directional patterns, while its prediction accuracy decreases for blocks with complicated textures and blocks with blurring edges. Fig. 2 shows the first frame of sequence *BasketballPass*. It can be seen that the textures of the wall are relatively complicated, and there exist some regions with blurring edges (the green and white lines on the floor, regions near the celling, and boundaries of the people and objects). To illustrate the different characteristics of different blocks, two examples of 16×16 luma blocks are given in Figs. 3 and 4.

Fig. 3 shows a 16×16 luma block (marked in yellow in Fig. 2) with both complex textures and strong vertical edges. It is obvious the prediction generated by AIP in HEVC [see Fig. 3(b)] can only provide the strong edges with mode 26 while the complicated textures are removed. Fig. 3(c) and (d) give the results generated by the proposed ATMP and CLNP, respectively. It can be observed that both the textures and the edges are kept in Fig. 3(c) by using the non-local prediction in ATMP. However, some unwanted textures are also generated and the strong edges are smoothed. The prediction shown in Fig. 3(d) generated by CLNP which combines local prediction by AIP and non-local prediction by ATMP provides a more accurate prediction, in which the strong edges are remained and the textures are more similar to the original block.

Fig. 4 shows another 16×16 luma block (marked in green in Fig. 2) with relatively blurring edges close to horizontal direction. The predictions generated by AIP with mode 11 and AIP with mode 12 are given in Fig. 4(b) and (c), respectively. We can see that the prediction in Fig. 4(b) has relatively strong edges while the direction of edges in the prediction in Fig. 4(c) deviates from the edges in original block. Fig. 4(d) shows the prediction generated by the proposed CNMP which combines the predictions from AIP with mode 11 and AIP with mode 12. It can be seen that Fig. 4(d) provides more similar edges to the original block. The prediction by CNMP has smoother edges than AIP with mode 11 and provides more accurate direction of edges than AIP with mode 12.

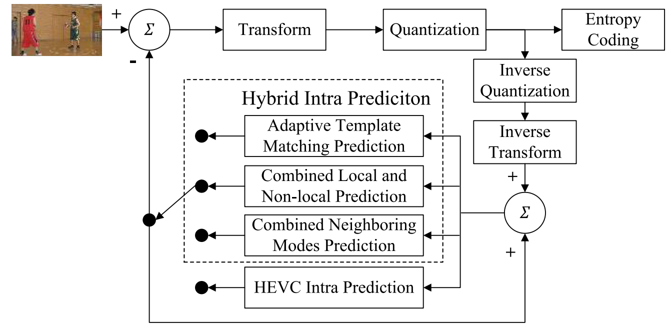


Fig. 5. Proposed hybrid intra prediction for HEVC.

III. Proposed Hybrid Intra Prediction

According to the motivation in Section II, a hybrid intra prediction method is proposed to improve intra prediction. The proposed hybrid intra prediction as shown in Fig. 5 consists of three parts: adaptive template matching prediction (ATMP) which can provide good predictions for blocks with complex textures by exploring non-local correlation, combined local and non-local prediction (CLNP) by exploring both local correlation for AIP and non-local correlation for ATMP, and combined neighboring modes prediction (CNMP) which can generate smoother prediction by exploring more local correlations. In the hybrid intra prediction, these three prediction methods are tested and compared with each other. Then, the best one among these three methods and the best intra prediction mode from HEVC intra prediction are also compared. At last, the one with the smallest rate-distortion (RD) cost is selected as the intra prediction for the coding block. At the encoder side, the best prediction method for each CU is signaled and transmitted. The details for the proposed intra prediction are described as follows.

A. Adaptive Template Matching Prediction

Template matching prediction (TMP) [16] was firstly proposed for H.264/AVC as an additional intra prediction mode.

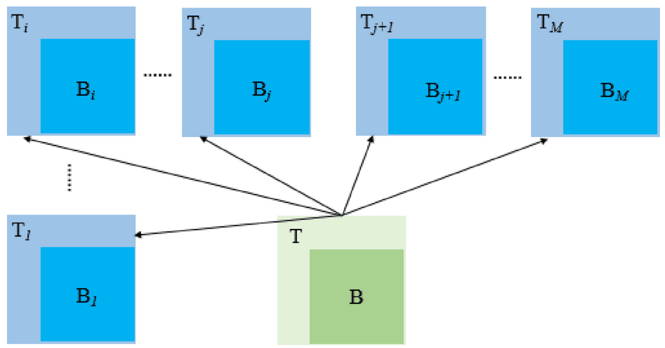


Fig. 6. Template matching process. Inverse-L shaped areas are the templates (in light blue and light green) used in the matching process. The deep blue blocks $\mathbf{B}_1 \sim \mathbf{B}_M$ are the prediction candidates for the coding block $\mathbf{B}$ (in green).

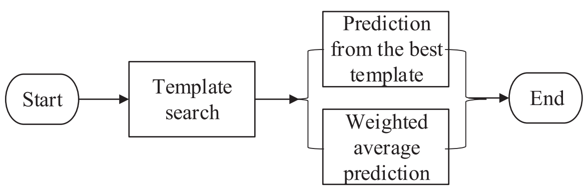


Fig. 7. Flowchart of the proposed ATMP.

TMP can predict blocks with complex textures well when the correlation between the coding block and its template is high and similar patterns can be found for the coding block within the coding frame. The TMP process is shown in Fig. 6. In TMP, template matching is performed for $\mathbf{T}$ in the reconstructed region of the coding frame. Several templates $\mathbf{T}_1 \sim \mathbf{T}_M$ which best match the template $\mathbf{T}$ can be found and the corresponding blocks $\mathbf{B}_1 \sim \mathbf{B}_M$ can be utilized as the predictors for the coding block $\mathbf{B}$. Based on the matched templates, the corresponding block with the smallest sum of squared errors (SSE) for the template [16] or the average block [17] is utilized as the final prediction for $\mathbf{B}$.

However, there have some drawbacks in TMP. Firstly, according to [17], averaging all the candidates can improve the prediction accuracy. Nevertheless, the prediction block with the best matched template can be better if the correlation between the template and its neighboring block is high enough. Therefore, only using one candidate or only averaging all the candidates may be inefficient. Besides, simply averaging all the candidates is also inefficient since the different importance of each candidate is not considered. We propose an adaptive TMP (ATMP) method to further improve the efficiency of TMP, in which two different predictors can be used for prediction and adaptive weighting is designed for the prediction candidates.

The flowchart of the proposed ATMP is given in Fig. 7. In ATMP, the width of the template is set to two pixels, and the vertical search range does not exceed the above border of the above CTU and the horizontal search range does not exceed the left border of the left CTU in order to avoid large memory access and search complexity. An example is shown in Fig. 8 in which the size of the CU (in green) is 32×32 and its search regions contain the left CTU, the above-left CTU, the above CTU and the reconstructed CUs in current CTU.

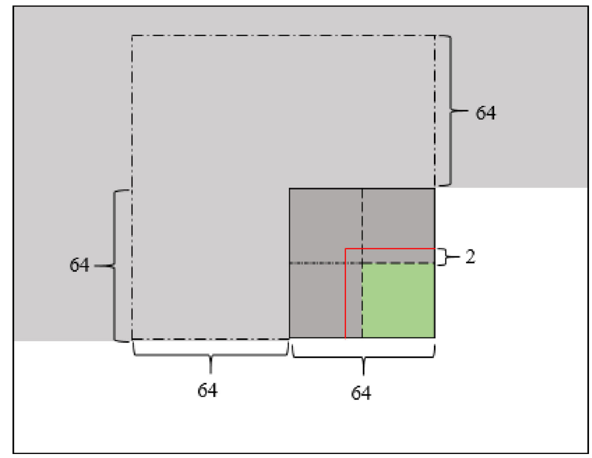


Fig. 8. Search range and template size for the CU in green. The gray regions are already reconstructed.

In ATMP, two predictors based on the matched templates are used, which are the corresponding block with the best matched template and the weighted average block. The first predictor can be represented by

$$\mathbf{P}^{f}_{ATMP}(i,j) = \mathbf{B}_t(i,j), \tag{1}$$

where t is the index of the best matched template, and $\mathbf{B}_t$ is the corresponding block of template $\mathbf{T}_t$.

The weighted average block is calculated from M candidate blocks with the smallest SSEs of the templates. Assume a pixel value p in current template $\mathbf{T}$ is a random variable with mean $\bar{p}$. The corresponding pixel value p_m in the matched template $\mathbf{T}_m$ can be observed as the predictor of p with an error e_m with zero mean and variance σ_m^2. Therefore, E $(p_m) = \bar{p}$. $\bar{p}$ can be estimated by

$$\bar{p} = \sum_{m=1}^{M} p_m / M. \tag{2}$$

The final prediction for each pixel position in the template $\mathbf{T}$ can be calculated by adding the mean value to the weighted average value of the M predictors with mean removal. Thus, we can obtain the estimate for p as the linear combination of p_m as

$$p = \sum_{m=1}^{M} \alpha_m (p_m - \bar{p}) + \bar{p}. \tag{3}$$

The weights can be given by

$$\alpha_m = \frac{1/\sigma_m^2}{\sum_{m=1}^{M} 1/\sigma_m^2}, \tag{4}$$

where σ_m^2 can be estimated by $SSE(\mathbf{T}, \mathbf{T}_m)/N_T$. N_T is the number of pixels in the template $\mathbf{T}$. Based on the high correlations between the templates and the corresponding coding blocks, the second predictor can be represented by

$$\mathbf{P}^{s}_{ATMP}(i,j) = \sum_{m=1}^{M} \alpha_m (\mathbf{B}_m(i,j) - \overline{\mathbf{B}}(i,j)) + \overline{\mathbf{B}}(i,j), \tag{5}$$

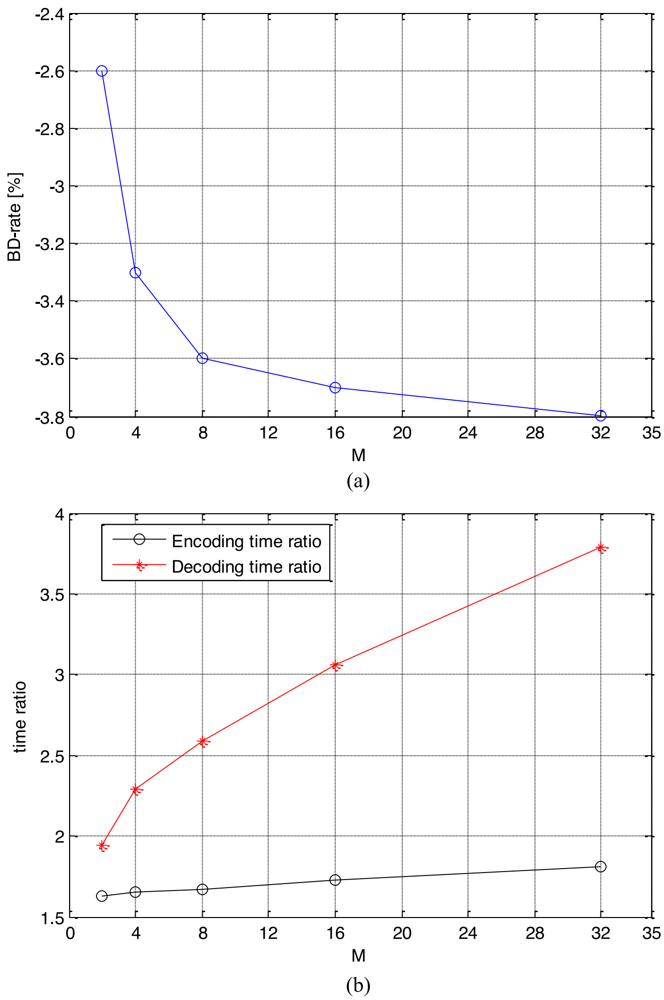


Fig. 9. The relations (a) between BD-rate reduction and M, and (b) between encoding/decoding time and M.

where $\overline{\mathbf{B}}(i,j)$ is the mean value of $\mathbf{B}_m(i,j)$ which is calculated in the same way as (2) and α_m is derived by (4).

The candidate number M is crucial for the prediction performance and encoding/decoding time. In order to obtain an appropriate M, we use four test sequences: *BasketballDrive*, *BQTerrace*, *BasketballDrill* and *Johnny*, to get the relations between average BD-rate reduction (and encoding/decoding time) and M. In the test, M is set to 2^n, where n is from 1 to 5. As shown in Fig. 9, the BD-rate reduction increases with M, and the increase becomes small when M is larger than 8. The encoding and decoding time also increase with M, and the relations are close to linear. Therefore, M is set to 8 in order to get a good balance between coding performance and encoding/decoding complexity.

At the encoder side, the final prediction for current CU is selected between the first predictor and the second predictor by rate-distortion optimization (RDO) process, and the better one is signaled and transmitted. It should be noted that ATMP is applied for both luma and chroma blocks.

The prediction error measured by SSE of the proposed ATMP for Fig. 3(c) is 7224 which is smaller than the SSE (9942) for the prediction by AIP with mode 26 in Fig. 3(b). This indicates that the proposed ATMP is more accurate for predicting this block than AIP.

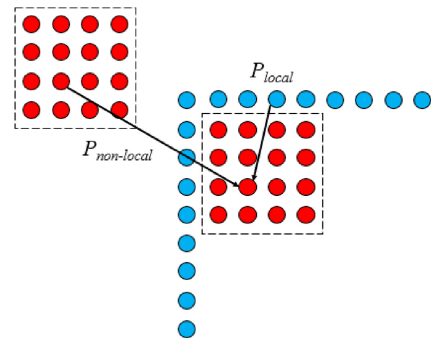


Fig. 10. Illustration of CLNP. It combines AIP which exploits local correlation, and ATMP which exploits non-local correlation.

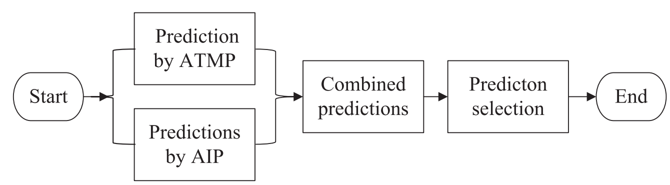


Fig. 11. Flowchart of the proposed CLNP.

B. Combined Local and Non-Local Prediction by AIP and ATMP

For blocks with both complex textures and strong local edges, both AIP and ATMP are inefficient. This is because AIP can only deal well with blocks with local edges while complex textures are difficult to be predicted. ATMP can predict complex textures while the local edges are smoothed or inaccurate because of the averaging of the candidates or the imperfect template matching. In order to handle this problem, in this subsection, a combined local and non-local prediction (CLNP) by exploring both local correlation for AIP and non-local correlation for ATMP is presented. In CLNP, AIP is used to predict the strong edges by using local correlation between the coding block and the boundary pixels and ATMP is used to predict the complex textures by using non-local correlation between the coding block and the matched blocks. Fig. 10 gives the illustration of CLNP.

The flowchart of the proposed CLNP is given in Fig. 11. Different from intra prediction in HEVC, we only use modes of neighboring PUs to generate the local prediction in CLNP. The neighboring PUs used in CLNP are the above PU and the left PU of current CU. It is known that there have high correlations between neighboring PUs, especially when strong directional patterns exist in the local region. Therefore, the modes of neighboring PUs can be utilized to approximate the directional patterns in current PU. It should be noted that neighboring PU may be not encoded by AIP in some cases. In these cases, the intra mode (AIP mode) of the neighboring PU needs to be determined. If the PU is coded by ATMP, the AIP mode is set to DC mode; if the PU is coded by CLNP, the AIP mode is set to the AIP mode in CLNP; if the PU is coded by CNMP, the AIP mode is set to the first mode in CNMP. Another advantage of using modes of neighboring PUs is that the overhead for signaling the intra mode is small since only 1 bit is needed to indicate which PU's mode is used in CLNP when the modes of above PU and left PU are different, and no bit is needed when these two modes are the same.

In CLNP, the weighted prediction by ATMP is combined with the prediction by AIP. The generated prediction is represented by

$$\mathbf{P}_{CLNP}(i,j) = w_{non-local} * \mathbf{P}_{non-local}(i,j) + w_{local} * \mathbf{P}_{local}(i,j), \tag{6}$$

where $\mathbf{P}_{non-local}$ represents the prediction from ATMP and $\mathbf{P}_{local}$ represents the prediction from AIP with the mode of left PU or above PU. $w_{non-local}$ and w_{local} are weighting factors and $w_{non-local} + w_{local} = 1$. The weighting factors balance the importance of these two predictions. The optimal weights can be estimated at the encoder side by solving a LSE optimization problem which can be formulated as:

$$\underset{w_{local}}{\text{Min}} \; ||\mathbf{O}(i,j) - (1 - w_{local}) * \mathbf{P}_{non-local}(i,j) - w_{local} * \mathbf{P}_{local}(i,j)||^2, \tag{7}$$

where $\mathbf{O}(i,j)$ is the original value in the coding block. Since the original value cannot be obtained in the decoder side, the optimal weight w_{local} needs to be encoded and transmitted. However, the overhead for signaling the weight is large which will degrade the coding efficiency of CLNP. Thus, in CLNP, three different weights are pre-defined which emphasize the different importance of local prediction and non-local prediction. The combined prediction in (6) is rewritten as follows:

$$\mathbf{P}_{CLNP}(i,j) = ((4 - w_{local}) * \mathbf{P}_{non-local}(i,j) + w_{local} * \mathbf{P}_{local}(i,j) + 2) \gg 2, \tag{8}$$

where the weight w_{local} can be 1, 2 or 3. If w_{local} is 1, $w_{non-local}$ will be 3 and the prediction from non-local prediction is more important, and vice versa. If w_{local} is 2, $w_{non-local}$ will also be 2 and these two predictions are averaged to generate the combined prediction. The best combination of ATMP and AIP will be selected from these three combinations though RDO process. In order to signal the best combination, one or two bits are needed. In this paper, CLNP is applied for both luma and chroma blocks. For chroma blocks, only the CLNP mode will be tested when the CLNP mode is selected as the best mode for luma and any other chroma intra prediction modes, such as DC mode and planar mode will be directly skipped.

For the example in Fig. 3, the SSE for the prediction by CLNP is 6762 which is smaller than the SSEs for the predictions by ATMP and AIP. This verifies that the prediction generated by CLNP is more accurate for this block.

C. Combined Neighboring Modes Prediction

The bi-directional intra prediction [14] was firstly proposed as an additional intra prediction mode for H.264/AVC, in which two predictions generated by two intra prediction modes are weighted to generate a new predictor for the coding block. Several combinations of two modes are generated based on the statistics and compared with original intra modes in H.264/AVC. The best intra coding mode for the block is selected from these combinations and the original intra modes. However, there are some disadvantages of this method. Firstly, the number of the combinations is high which leads to high computational complexity for the encoder, especially for HEVC in which the number of modes is up to 35. Secondly, the weights are trained offline which are not adaptive to the content of the block. Besides, the weights for each combination need to be stored which will occupy memory buffers.

Fig. 12. A part of the first frame of *BasketballDrive.*

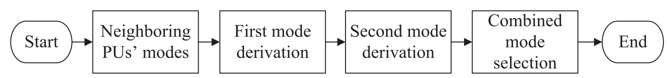


Fig. 13. Flowchart of the proposed CNMP.

In this subsection, a combined neighboring modes prediction (CNMP) is proposed. In CNMP, only two neighboring modes are combined based on the modes of neighboring PUs. The weighting factors for these two predictors are adaptively calculated based on the content of the block and do not need to be encoded. CNMP is mainly used for blocks with blurring edges and boundaries which are common for videos with fast motions for objects and camera. An example is shown in Fig. 12.

The flowchart of CNMP is shown in Fig. 13. In CNMP, two of the 35 modes in HEVC intra prediction are utilized to generate two corresponding predictors and then combined to generate a new predictor for the coding block. As we known, neighboring blocks usually have high correlations, which means the directional information in neighboring blocks is similar. Similar to the intra mode derivation in CLNP, the neighboring blocks used in CNMP are also the above PU and the left PU of current CU. We use one of the neighboring PUs' modes as one of the two modes in CNMP which is referred to as the first mode (*first_mode*). If the neighboring PU is not encoded by AIP, the intra mode of the neighboring PU is decided by using the same method in CLNP. There are two different cases for selecting *first_mode*. When the modes of above PU and left PU are the same, *first_mode* is set to this same mode. Otherwise, *first_mode* can be either the mode of the above PU or the mode of the left PU. If *first_mode* is a directional mode, the second mode (*second_mode*) in CNMP

TABLE I
THE DERIVATION OF THE SECOND MODE

first_mode	*second_mode*
DC	planar
planar	DC
2	3 or 33
34	3 or 33
$3 \sim 33$	*first_mode* -1 or *first_mode* $+1$

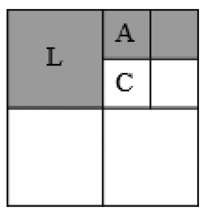


Fig. 14. The locations of neighboring TUs of current TU.

is set to one of its two closest modes. Otherwise, if *first_mode* is a non-directional mode, the *second_mode* is set to the other non-directional mode. The derivation of *second_mode* is shown in Table I. The numbers in Table I represent the intra modes of HEVC intra prediction as shown in Fig. 1.

Based on the derivation of *first_mode* and *second_mode*, a list of mode combinations can be generated. The best combination with the smallest RD cost will be selected. For example, assume the mode of the above PU is 26 and the mode of the left PU is 30, then the combinations of *first_mode* and *second_mode* can be (26, 25), (26, 27), (30, 29) and (30, 31). These four combinations will be compared through RDO process and the best one will be used for prediction.

The prediction of CNMP is represented by

$$\mathbf{P}_{CNMP}(i,j) = w_f * \mathbf{P}_f(i,j) + w_s * \mathbf{P}_s(i,j), \tag{9}$$

where $\mathbf{P}_f$ and $\mathbf{P}_s$ are the predictions generated by *first_mode* and *second_mode*, respectively. w_f and w_s are the weighting factors and $w_f + w_s = 1$. In the HEVC intra coding, TU is the basic unit employed in intra prediction since it allows the reconstructed neighboring TUs to be used as the reference when coding current TU, which reduces the distance between the reference and the block to be predicted. In the proposed CNMP, the weighting factors are adaptively derived by using neighboring reconstructed TUs since there have high correlations between current TU and its neighboring TUs. Assume the whole block in Fig. 14 is a CU and the sub-blocks are TUs. The gray blocks are already encoded and reconstructed TUs. The weighting factor w_f for current TU (**C**) is predicted by its above TU (**A**) and its left TU (**L**) as follows:

$$w_f = (w_f^L + w_f^A)/2, \tag{10}$$

where w_f^A and w_f^L are the weighting factors for *first_mode* which are calculated from the above TU and the left TU respectively. w_f^A and w_f^L can be obtained as follows.

For the left TU **L**, the two predictors in CNMP and its reconstructed block are both available before predicting current TU **C**. A linear model is used to derive the weighting factor, which is represented by

$$\mathbf{L}(i,j) = w_f^L * \mathbf{P}_f^L(i,j) + (1 - w_f^L) * \mathbf{P}_s^L(i,j), \tag{11}$$

where $\mathbf{P}_f^L$ and $\mathbf{P}_s^L$ are the two predictors for the left TU, and w_f^L is the weighting factor. $\mathbf{L}(i, j)$ is the pixel value of left reconstructed TU. The parameter w_f^L can be estimated by solving a LSE optimization problem which is formulated as:

$$\underset{w_f^L}{\mathrm{Min}} \, ||\mathbf{L}(i,j) - w_f^L * \mathbf{P}_f^L(i,j) - (1 - w_f^L) * \mathbf{P}_s^L(i,j)||^2. \tag{12}$$

The solution of the LSE optimization problems can be calculated by:

$$\frac{\partial}{\partial w_f^L} \left\| \mathbf{L}(i,j) - \mathbf{P}_s^L(i,j) - w_f^L * (\mathbf{P}_f^L(i,j) - P_s^L(i,j)) \right\|^2 = 0. \tag{13}$$

So the weighting factor w_L^f can be represented by eq. (14) shown at the bottom of this page.

For the above TU **A**, the same process can be employed to derive the weight w_f^A after the reconstruction of TU **A**.

It should be noted that the weighting factor for the first TU of the CU is set to a default value, 0.5. If the above TU (or left TU) of current TU is unavailable, the weighting factor of the above TU (or the left TU) is also set to 0.5. After calculating the weighting factors w_f^A and w_L^f, (10) is used to approximate the weighting factor for current TU when generating the combined prediction in (9).

For luma block, the best combination of modes will be decided by RDO. For the corresponding chroma blocks, the prediction by using the best combination of modes from luma will be performed to generate the prediction. Other chroma prediction modes, such as DC mode and planar mode, will be directly skipped.

For the example in Fig. 4, the SSEs for AIP with mode 11 and mode 12 are 17326 and 20620, respectively. The SSE for the proposed CNMP which combines predictions from mode 11 and mode 12 is 14233 which is much smaller than the SSEs from mode 11 and mode 12. Therefore, CNMP is more efficient for this block which has blurring edges.

D. Signaling for the Proposed Intra Prediction

In the proposed hybrid intra prediction, besides the original intra prediction in HEVC, three additional intra modes, ATMP, CLNP and CNMP are used for intra prediction. The best intra mode for each coding block should be encoded and

$$w_f^L = \frac{\sum_{i,j=0}^{N-1} (\mathbf{P}_f^L(i,j) - \mathbf{P}_s^L(i,j)) * \mathbf{L}(i,j) - \sum_{i,j=0}^{N-1} (\mathbf{P}_f^L(i,j) - \mathbf{P}_s^L(i,j)) * \mathbf{P}_s^L(i,j)}{\sum_{i,j=0}^{N-1} (\mathbf{P}_f^L(i,j) - \mathbf{P}_s^L(i,j))^2}. \tag{14}$$

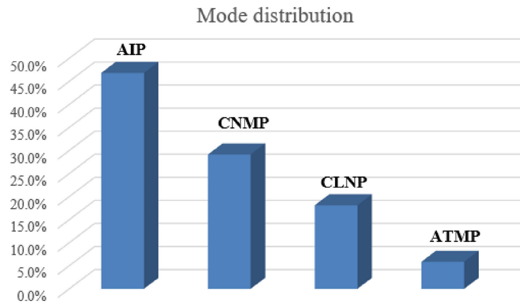


Fig. 15. Mode distributions for AIP, CNMP, CLNP and ATMP.

TABLE II
CODEWORDS FOR EACH INTRA MODE

Mode type	Codewords
AIP	0
CNMP	10
CLNP	110
ATMP	111

transmitted. Thus, efficient coding of the mode information is crucial for intra coding efficiency. In this paper, in consideration of the balance between the coding complexity and coding efficiency, the proposed modes are all implemented at the CU level which means the CU size is equal to the size of the PU.

In order to get an appropriate binarization method for the proposed intra prediction modes, we calculate the distributions for different intra modes: ATMP, CLNP, CNMP and original HEVC intra prediction represented by AIP for test sequences in Class A-Class E [22]. The distributions are shown in Fig. 15. We can see that AIP mode occupies the highest proportion which is about 45% and CNMP with 30% has the second highest occupation, followed by CLNP and ATMP. Therefore, the truncated unary code method can be utilized to encode the mode type. The codewords for each mode type are given in Table II.

When ATMP is selected for the CU, another bit is used to indicate whether the candidate block which has the smallest SSE for the template or the weighted average block is used to predict the CU. When CLNP is used to predict the CU, one bit is used to indicate which mode of the two modes from the above PU and the left PU is used for AIP when these two modes are different. No bit will be used if these two modes are the same. Besides, one of the three weights for CLNP needs to be encoded. The bits are set to 1, 00 and 01 when w_{local} are 2, 1 and 3, respectively. For CNMP, a flag with one bit is employed to indicate the first mode is from the above PU or from the left PU when these two modes are different. No bit is needed if these two modes are the same. For the second mode, one more bit is used to indicate which one of the two closest modes of the first mode is selected as the second mode when the first mode is a directional mode. Otherwise, no bit is needed for signaling the second mode.

IV. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed hybrid intra prediction method, we integrated the proposed method into HEVC reference software HM-14.0 [20], [21]. All intra (AI) main configuration is used in the experiments, and the sequences [22] recommended by JCT-VC are tested. The QPs used are 22, 27, 32 and 37. The coding performance is measured by BD-rate under PSNR [23] compared to the original HEVC encoder. The proposed hybrid intra prediction method, TMP [17], ATMP, ATMP+CLNP and CNMP are compared with HEVC default intra prediction (AIP). Besides, the performance of the proposed overall method is also evaluated in random access (RA) configuration. Furthermore, the comparisons with other two latest intra prediction methods are conducted and the coding gains of the proposed overall method when integrated into JEM-1.0 [27], [28] are also given in this section.

A. *Rate-Distortion Performance in HM*

The RD performance of TMP, ATMP, ATMP+CLNP, CNMP and the proposed overall method for AI configuration is given in Table III. The proposed overall method achieves 2.8% BD-rate reduction on average and the gain is up to 6.9% compared to HEVC default intra coding. It also can be seen the BD-rate reduction is larger than 2.0% for most of the sequences which verifies the robust of the proposed method. In the last column of Table III, we also give the BD-rates under MS-SSIM [36] for the proposed overall method. It can be observed that the results are similar to the BD-rates under PNSR.

For ATMP method, the average BD-rate reduction is 1.7% which is larger than the gain of TMP and the improvement compared to TMP can be up to 1.3% for sequence *BasketballDrill*. The combination of ATMP and CLNP achieves 2.3% BD-rate reduction on average. The gain for CNMP is 0.9% on average. It is noted that the maximum gain for CNMP is 3.0% for sequence *BasketballDrive* which has many regions with blurring boundaries and edges because of the fast movement of people and camera. The proposed CNMP can efficiently predict this blurring boundaries and edges well by combing two neighboring modes.

Fig. 16 plots the rate distortion curves for sequence *BasketballDrill*, *BQTerrace* and *Johnny*. We can see that the proposed method outperforms original HEVC at different quantization levels, especially at medium to small QPs. When QP becomes large, the template used in ATMP and CLNP will be largely distorted which influences the accuracy of the prediction in ATMP and CLNP. However, for the proposed CNMP, the prediction from CNMP can be better for large QP. This is because the reference pixels are quantized too much when QP is large and single prediction from AIP can be much affected by the distorted reference while the proposed CNMP exploiting more local correlations by weighted average of two predictions can reduce the effect of distorted reference. The efficiency of CNMP can be also verified by analyzing the mode usage of CNMP which is given in the next sub-section. Overall, the proposed overall method can provide consistent coding gains among all QP levels.

TABLE III
LUMA BD-RATE COMPARISONS OF TMP, ATMP, CLNP, CNMP AND THE PROPOSED OVERALL METHOD UNDER AI CONFIGURATION

Class	Resolution	Sequences	TMP	ATMP	ATMP + CLNP	CNMP	Overall	Overall (MS-SSIM)
A	2560 × 1600	Traffic	−1.2%	−1.6%	−2.3%	−0.9%	−2.8%	−2.8%
		PeopleOnStreet	−1.5%	−1.8%	−2.3%	−0.6%	−2.6%	−2.5%
B	1920 × 1080	Kimono	−0.5%	−0.4%	−0.9%	−0.7%	−1.4%	−1.2%
		ParkScene	−0.7%	−0.8%	−1.4%	−0.7%	−2.0%	−1.9%
		Cactus	−1.6%	−2.4%	−2.9%	−0.9%	−3.4%	−3.6%
		BasketballDrive	−4.2%	−4.3%	−5.6%	−3.0%	−6.9%	−6.7%
		BQTerrace	−3.2%	−4.1%	−4.8%	−0.9%	−5.2%	−5.7%
C	832 × 480	BasketabllDrill	−3.5%	−4.8%	−5.2%	−1.1%	−5.8%	−5.0%
		BQMall	−0.3%	−0.6%	−1.0%	−0.6%	−1.4%	−1.3%
		PartyScene	−0.3%	−0.5%	−0.8%	−0.3%	−1.1%	−1.4%
		RaceHorsesC	−0.3%	−0.3%	−0.7%	−0.5%	−1.2%	−1.3%
D	416 × 240	BasketballPass	−0.6%	−1.2%	−1.6%	−0.8%	−2.1%	−2.0%
		BQSquare	0.0%	−0.3%	−0.4%	0.0%	−0.6%	−0.4%
		BlowingBubbles	0.0%	−0.1%	−0.3%	−0.5%	−0.7%	−0.7%
		RaceHorses	−0.1%	−0.1%	−0.4%	−0.7%	−0.7%	−0.7%
E	1280 × 720	FourPeople	−0.8%	−1.2%	−1.7%	−0.9%	−2.2%	−2.2%
		Johnny	−3.3%	−4.3%	−4.9%	−1.4%	−5.5%	−5.5%
		KristenAndSara	−1.5%	−2.6%	−3.0%	−1.3%	−3.6%	−3.7%
Average			−1.3%	−1.7%	−2.3%	−0.9%	−2.8%	−2.7%
Enc. Time			154%	166%	218%	127%	248%	248%
Dec. Time			189%	193%	270%	108%	259%	259%

The summary of coding gains for the proposed overall method under AI and RA configurations is given in Table IV. The results show that the proposed method is more efficient for high resolution sequences (Class A-C and Class E) since the local and non-local correlations are higher in high resolution sequences. For RA configuration, the proposed method still has coding gain which is 1.0%, 1.3% and 1.5% for Y, U and V, respectively. The reason is that both the quality of intra frame which will be referenced by the following inter frames and the prediction efficiency of intra coded CUs in inter frames are improved. It also can be noted that the proposed method is efficient for both luma and chroma components.

Besides the objective coding improvements, the proposed method can also achieve some subjective improvements. Fig. 17 shows enlarged regions of the first frames in *BasketballPass*, *BQSquare* and *Traffic* when the frames are coded by original HEVC encoder and the proposed hybrid intra prediction method. It can be observed that the quality of the marked regions is higher for the frames coded by the proposed hybrid intra prediction than the frames coded by original HEVC encoder. For example, the textures in the regions marked in green are remained in some degree for the proposed method while more textures are removed for original HEVC encoder. The color of textures for the regions marked in blue is more similar to the original frame for the proposed method.

B. Coding Mode Analysis

The mode usages for the proposed method under AI configuration when QP is 22 and 37 are given in Tables V and VI, respectively. The mode usage of each proposed method is determined by the number of blocks used in the prediction. The average usages of CNMP, CLNP and ATMP are 16.0%, 16.0%, and 5.9% when QP is 22, respectively. The usage rate of CNMP increases from 16.0% to 27.0% when QP is 37 which verifies the efficiency of the proposed CNMP when QP is large since CNMP can reduce the distortion caused by quantization of reference by averaging two predictions generated by two neighboring intra modes. However, the mode usage of CLNP decreases when QP becomes large. The reasons are given as follows. When QP is large, more blocks will be coded in large block size. However, the prediction generated by CLNP is not that accurate for large blocks. Besides, mode overhead becomes more significant in large QP condition. As shown in Table II, the overhead for signaling CLNP mode is 3 bits which is larger than AIP (1 bit) and CNMP (2 bits) and one of the three weights for CLNP also needs to be encoded which costs additional 1 or 2 bits. We also give the distributions for CUs coded by different intra prediction modes when QP = 22 and 37 as shown in Fig. 18. We can see the proposed CNMP modes are usually used for regions with blurring boundaries and edges. CLNP and ATMP are mainly used for regions with complex textures, e.g., the textures of the wall and the floor, which have some similar textures in their neighboring regions.

C. Computational Complexity

Table III also shows the encoding and decoding complexities of the proposed hybrid intra prediction method when compared to the HEVC default intra coding. The encoding and decoding complexities for the proposed overall method are 248% and 259%, respectively. The encoding complexity mainly comes from the template matching process in ATMP and CLNP, and the RDO process for determining the best prediction mode from AIP, ATMP, CLNP and CNMP. For the proposed CNMP, the encoding complexity is 127%. For the decoder side, the complex-

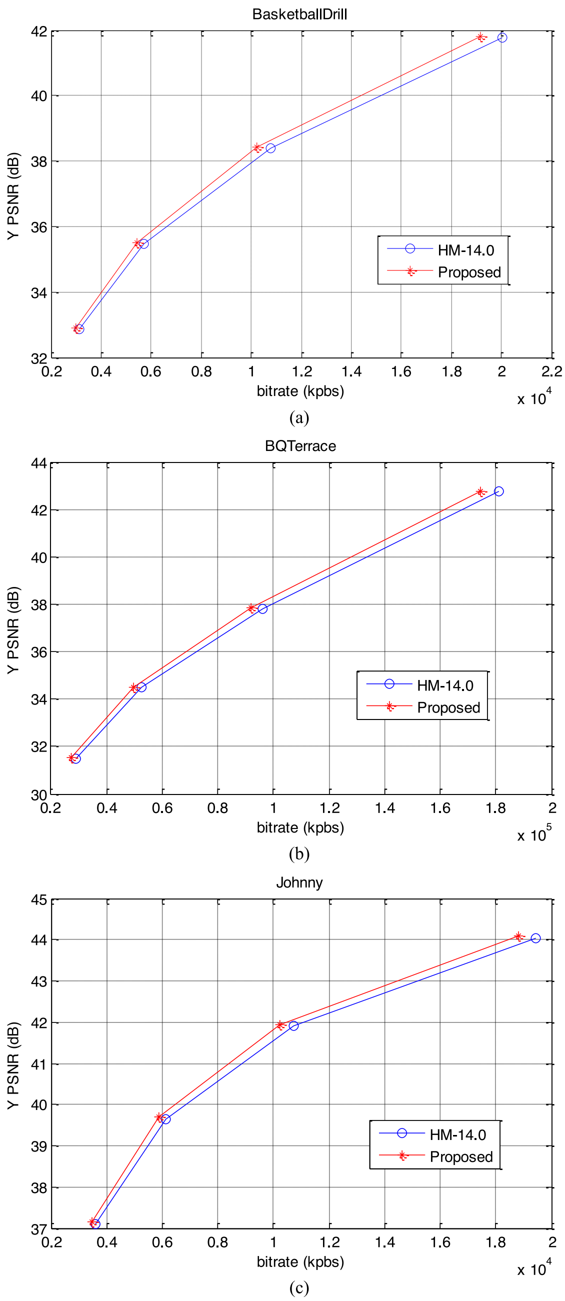


Fig. 16. Rate distortion curves for (a) *BasketballDrill*, (b) *BQTerrace* and (c) *Johnny*.

TABLE IV
BD-RATE FOR THE PROPOSED OVERALL METHOD UNDER AI AND RA CONFIGURATIONS

	AI			RA		
	Y	U	V	Y	U	V
Class A	−2.7%	−3.1%	−4.0%	−1.2%	−2.1%	−2.6%
Class B	−3.8%	−3.7%	−4.2%	−1.2%	−1.7%	−1.7%
Class C	−2.4%	−3.6%	−3.5%	−1.1%	−1.3%	−1.6%
Class D	−1.1%	−1.7%	−1.6%	−0.4%	−0.6%	−0.7%
Class E	−3.8%	−5.5%	−5.1%	–	–	–
Average	−2.8%	−3.3%	−3.4%	−1.0%	−1.3%	−1.5%
Enc. Time		248%			109%	
Dec. Time		259%			128%	

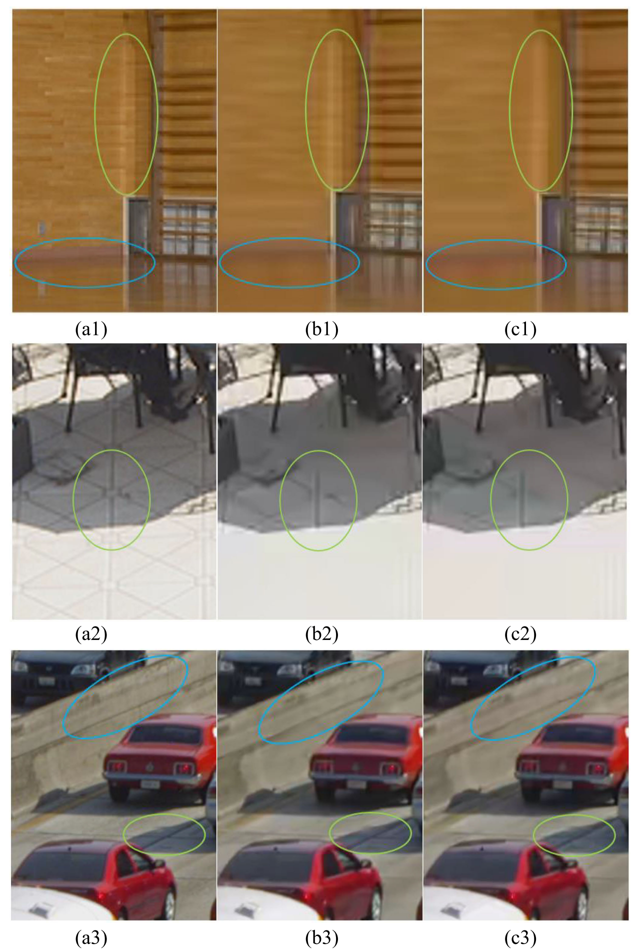


Fig. 17. Enlarged regions of (a1-a3) original frames; (b1-b3) frames coded by the proposed method when QP = 32; (c1-c3) frames coded by original HM-14.0 when QP = 32.

ity for the proposed prediction mainly comes from the template matching process in ATMP and CLNP. For CNMP, the decoding complexity is increased by 8% which is from the calculation of the adaptive weights and the generation of the second prediction in CNMP.

The encoding and decoding complexities of the proposed overall method under RA configuration are shown in the last two rows of Table IV. It can be seen that the increased complexities become much smaller under RA configuration than AI configuration. In RA, the encoding and decoding complexities are increased by 9% and 29%, respectively.

Although the encoding and decoding complexities for the proposed hybrid prediction under AI configuration are high in current HM platform, some work can be done to reduce the complexity. For example, for the encoder, fast mode decision [29] can be designed to early terminate some unnecessary RDO checks for the proposed method. Fast template searching similar to fast motion estimation [30] in inter coding can be also used to accelerate template matching process. Furthermore, considering the powerful hardware in current devices such as personal PC and smartphone, parallel techniques based on multicore CPUs and GPUs [31], [32] can be used in fast template matching.

TABLE V
MODE USAGES OF AIP, CNMP, CLNP AND ATMP WHEN QP = 22

Class	Sequences	ATMP	CLNP	CNMP	AIP
A	Traffic	4.2%	18.2%	19.0%	58.6%
	PeopleOnStreet	6.0%	16.1%	15.7%	62.2%
B	Kimono	3.2%	19.1%	20.1%	57.6%
	ParkScene	2.8%	22.8%	18.6%	55.8%
	Cactus	7.8%	22.1%	14.6%	55.5%
	BasketballDrive	6.8%	28.5%	20.0%	44.7%
	BQTerrace	12.4%	28.6%	15.0%	44.0%
C	BasketablllDrill	8.5%	17.3%	14.3%	60.0%
	BQMall	3.2%	10.0%	15.9%	70.9%
	PartyScene	2.6%	10.1%	9.2%	78.0%
	RaceHorsesC	1.2%	11.6%	16.3%	70.9%
D	BasketballPass	8.3%	16.2%	17.2%	58.4%
	BQSquare	8.8%	7.5%	6.7%	77.0%
	BlowingBubbles	0.7%	8.0%	12.8%	78.5%
	RaceHorses	0.6%	6.5%	15.0%	77.9%
E	FourPeople	6.2%	12.4%	19.3%	62.2%
	Johnny	13.2%	19.4%	19.2%	48.1%
	KristenAndSara	9.7%	13.9%	19.6%	56.8%
	Average	5.9%	16.0%	16.0%	62.1%

TABLE VI
MODE USAGES OF AIP, CNMP, CLNP AND ATMP WHEN QP = 37

Class	Sequences	ATMP	CLNP	CNMP	AIP
A	Traffic	9.8%	16.5%	32.7%	41.0%
	PeopleOnStreet	13.4%	11.6%	24.3%	50.7%
B	Kimono	3.7%	17.7%	37.7%	40.8%
	ParkScene	7.0%	22.0%	43.7%	27.4%
	Cactus	17.9%	10.3%	26.5%	45.3%
	BasketballDrive	13.4%	15.5%	31.5%	39.6%
	BQTerrace	19.9%	14.9%	17.3%	47.9%
C	BasketablllDrill	18.3%	13.4%	20.0%	48.3%
	BQMall	6.1%	8.2%	25.0%	60.8%
	PartyScene	7.8%	10.9%	18.4%	62.9%
	RaceHorsesC	4.4%	12.4%	31.3%	51.8%
D	BasketballPass	13.9%	7.8%	25.3%	53.0%
	BQSquare	9.2%	3.1%	19.4%	68.3%
	BlowingBubbles	2.9%	6.5%	27.4%	63.2%
	RaceHorses	3.2%	7.6%	28.0%	61.3%
E	FourPeople	13.1%	8.7%	23.9%	54.3%
	Johnny	16.2%	8.9%	26.3%	48.5%
	KristenAndSara	14.3%	7.6%	27.3%	50.8%
	Average	10.8%	11.3%	27.0%	50.9%

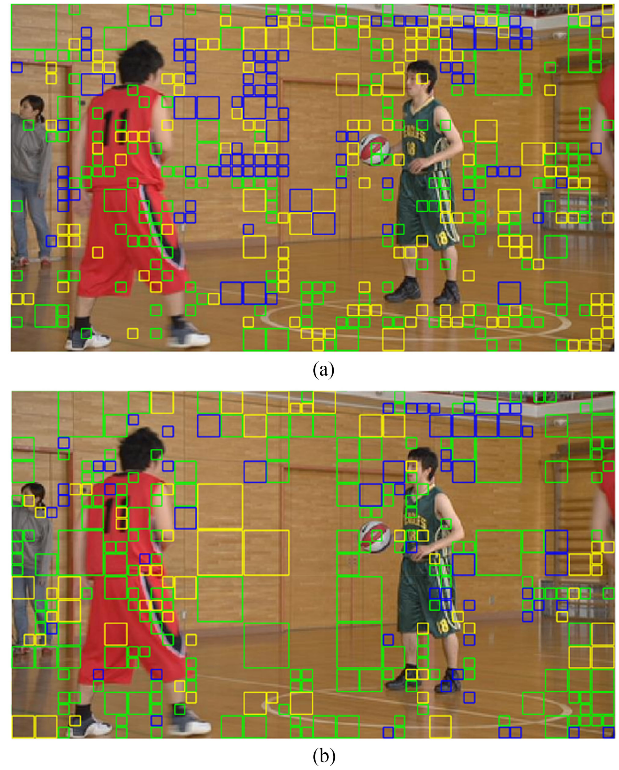


Fig. 18. The distributions of modes for the first frame of *BasketballPass* when a) QP = 22 and b) QP = 37. Blocks marked in green are encoded by CNMP, blocks marked in yellow are encoded by CLNP and blocks marked in blue are encoded by ATMP. Other regions are encoded by HEVC intra prediction. (a) QP = 22 (b) QP = 37.

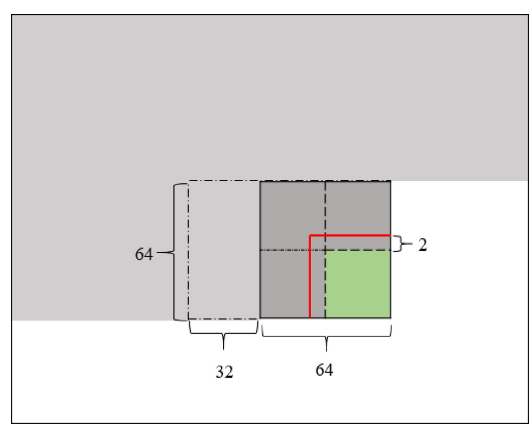


Fig. 19. Constrained search range for the CU (in green) compared with Fig. 8.

In order to reduce the complexity of the proposed method, several strategies are adopted to accelerate the proposed method. The proposed hybrid intra prediction with acceleration is denoted as *proposed_fast*. The strategies are given as follows:

1) *Parameters adjustment*: The search range and the candidate number M are sensitive to the encoding/decoding complexity and coding gain. In the *proposed_fast* method, the candidate number M is reduced from 8 to 4. The search range in vertical direction is constrained to the above boundary of current LCU and the search range in the horizontal direction is constrained to 32 pixels away from the left boundary of current LCU, as shown in Fig. 19.
2) *Early skip for ATMP and CLNP*: In this method, the template of the coding block will be analyzed first before performing ATMP and CLNP. If the template is smooth enough, template matching prediction will be inefficient. Thus, ATMP and CLNP can be early skipped. The variance will be used to detect the smooth template. If the variance of the template is smaller than a threshold, the template is considered to be smooth. The threshold is empirically set to QP–17.
3) *Fast template searching*: A two-step based search method is used for fast template matching. In the first step, the points (circles in Fig. 20) located at the even rows and columns are searched within the search range. The best 4 points can be obtained after the search. In the second

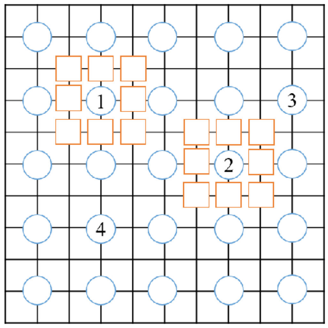


Fig. 20. Search points.

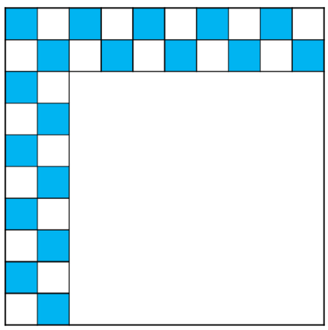


Fig. 21. Used pixels in the template.

step, for the best 2 points (point 1 and point 2 in Fig. 20), the neighboring 8 points (squares in Fig. 20) will be further searched to refine the results. Finally, the SSEs and the corresponding prediction blocks of the best 4 points among all the searched points are saved and will be used in ATMP and CLNP. In order to further reduce the number of pixels used in template matching process, only half of the pixels (blue pixels in Fig. 21) in the template are used for matching.

4) *Fast ATMP*: In ATMP, two predictors are compared by using RD costs in the proposed method, which is time-consuming. In the fast ATMP, we simply compare the SSEs between the prediction and the original block for these two predictors. The one with the smaller SSE cost will be selected as the final predictor for ATMP.
5) *Fast CLNP*: In CLNP, three different weights are used to balance the importance between local and non-local predictions. In the fast CLNP, the prediction method with equal weights ($w_{local} = 2$) is tested first by RDO. If this prediction method is better than all the other tested methods (AIP, CNMP and ATMP), the other two prediction methods ($w_{local} = 1$ and $w_{local} = 3$) will be further tested; otherwise, CLNP will be early skipped.

Table VII shows the results of the *proposed_fast* method. Compared to the results in Table IV, both the coding gain and the encoding/decoding complexity decrease. For AI configuration, the coding gain decreases from 2.8% to 1.9% for Y, while the encoding time decreases significantly from 248% to 147% and the decoding time also decreases significantly, which is from 259% to 126%. For RA configuration, the coding gain decreases from 1.0% to 0.7% for Y, while the encoding and decoding time decreases from 109% to 103% and 129% to 108%, respectively.

TABLE VII
BD-RATE FOR THE PROPOSED_FAST METHOD UNDER AI AND RA CONFIGURATIONS

	AI			RA		
	Y	U	V	Y	U	V
Class A	−1.8%	−1.8%	−2.1%	−0.8%	−1.2%	−1.2%
Class B	−2.4%	−1.7%	−2.4%	−0.8%	−0.4%	−0.8%
Class C	−1.8%	−2.2%	−2.4%	−0.8%	−0.7%	−1.0%
Class D	−0.8%	−1.2%	−1.1%	−0.4%	0.0%	−0.5%
Class E	−2.5%	−3.0%	−3.5%	–	–	–
Average	−1.9%	−1.9%	−2.2%	−0.7%	−0.5%	−0.8%
Enc. Time		147%			103%	
Dec. Time		126%			108%	

TABLE VIII
BD-RATE OF THE TWO COMPARED METHODS UNDER AI CONFIGURATION

	MPI [25]			PDPC [26]		
	Y	U	V	Y	U	V
Class A	−2.1%	−1.5%	−1.5%	−2.1%	−1.4%	−1.6%
Class B	−1.7%	−1.6%	−1.8%	−1.5%	−1.0%	−1.2%
Class C	−1.2%	−1.2%	−1.1%	−1.4%	−1.0%	−0.7%
Class D	−0.9%	−0.5%	−0.7%	−0.9%	−0.7%	−1.5%
Class E	−1.7%	−2.4%	−2.0%	−1.9%	−2.2%	−2.4%
Average	−1.5%	−1.5%	−1.5%	−1.5%	−1.2%	−1.4%
Enc. Time		340%			195%	
Dec. Time		102%			108%	

According to the results, the *proposed_fast* method can get a good balance between the encoding/decoding complexity and the coding gain.

D. Comparisons With Other Methods

In this subsection, the proposed method is compared with two latest intra prediction methods which are multi-parameter intra prediction (MPI) [25] and position dependent intra prediction combination (PDPC) [26]. MPI and PDPC are proposed for future video codecs. Currently, the Joint Video Exploration Team (JVET) of ITU-T VCEG and ISO/IEC MPEG is studying the potential need for standardization of future video coding technology with a compression capability that significantly exceeds that of the current HEVC standard [27]. The Joint Exploration Model (JEM) [27], [28] which includes new coding tools is the test model developed by JVET. PDPC was adopted in JEM due to its high prediction efficiency.

In order to compare the proposed method with MPI and PDPC, we integrated MPI and PDPC into HM-14.0 reference software. Based on HM-14.0, the performance of the proposed method, MPI and PDPC is compared under AI configuration.

Table VIII shows the results of MPI and PDPC. It can be seen that the BD-rate reductions of both MPI and PDPC are 1.5% for Y, which are much smaller than the proposed method (2.8% shown in Table IV and 1.9% shown in Table VII). The encoding complexities of all these three methods are high while the encoding time of the *proposed_fast* method is much smaller

TABLE IX
BD-RATE OF THE PROPOSED OVERALL METHOD UNDER AI AND RA CONFIGURATIONS IN JEM-1.0

	AI			RA		
	Y	U	V	Y	U	V
Class A	−1.0%	−1.6%	−2.2%	−0.4%	−1.3%	−1.7%
Class B	−1.8%	−2.7%	−2.7%	−0.6%	−1.3%	−1.1%
Class C	−0.9%	−1.4%	−1.3%	−0.4%	−1.2%	−0.4%
Class D	−0.4%	−1.5%	−1.6%	−0.1%	−0.7%	0.2%
Class E	−1.9%	−3.5%	−4.1%	–	–	–
Average	−1.2%	−2.1%	−2.3%	−0.4%	−1.1%	−0.6%
Enc. Time		118%			102%	
Dec. Time		151%			101%	

than the other two methods. For the decoder, the complexity of the proposed method is higher than MPI and PDPC since the decoder needs to perform template matching.

E. Performance in JEM

Various new coding tools have been adopted into JEM. For intra prediction, there are 67 intra prediction modes, four-tap intra interpolation filter, boundary prediction filters, cross component prediction (CCP), PDPC and adaptive reference sample smoothing (RSAF) in JEM-1.0. These intra prediction tools have improved the coding efficiency significantly. In order to know the additional coding gain of the proposed method for JEM, we also integrated the proposed method into JEM-1.0. In the test, the anchor is JEM-1.0 with default settings. The coding gains of the proposed method based on JEM-1.0 for AI and RA configurations are shown in Table IX.

Table IX shows that the proposed method still can achieve 1.2%, 2.1% and 2.3% BD-rate reduction for Y, U and V compared to JEM-1.0 under AI configuration, respectively. For RA configuration, the gain is 0.4%, 1.1% and 0.6% for Y, U and V, respectively. Compared to the BD-rate reduction in HM-14.0, the gains become small since the coding gain of the proposed method is overlapped with these new intra prediction methods in JEM-1.0. Table IX also shows that the encoding and decoding complexities of the proposed method become much lower. This is because the complexity of JEM-1.0 is much higher than HM-14.0.

V. CONCLUSION

In this paper, we analyzed the characteristics of complicated blocks which cannot be predicted well by HEVC intra coding. Based on the analysis, a hybrid intra prediction method is proposed to improve the prediction efficiency of HEVC intra coding. In the hybrid intra prediction, three prediction methods are proposed to handle blocks with different characteristics. An adaptive template matching prediction (ATMP) method is proposed for blocks with complex textures by using non-local correlation in which the prediction efficiency is improved by using both the prediction block of the best matched template and the adaptive weighted prediction compared to TMP. For blocks with both complex textures and local features, a combined local and non-local prediction (CLNP) which utilizes both local correlation and non-local correlation is proposed. Besides, in order to predict blocks with blurring edges, a combined neighboring modes prediction (CNMP) method is proposed. Experimental results suggest the proposed hybrid intra prediction method achieves 2.8% BD-rate reduction on average compared to HEVC default intra coding under all intra main configuration. When integrated into JEM, the proposed method still can achieve 1.2% BD-rate reduction. In the future, more work can be done to further reduce the complexity of the encoder and decoder. For the encoder, more efficient mode decision algorithms can be employed for the proposed prediction modes. For both the encoder and the decoder, fast template searching algorithms and parallel techniques based on multicore CPUs and GPUs can be exploited to further reduce the complexity of ATMP and CLNP.

REFERENCES

[1] G. Sullivan, J. Ohm, W.-J. Han, 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.

[2] 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.

[3] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, “Intra coding of the HEVC standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1792–1801, Dec. 2012.

[4] Y. Chen, J. Han, and K. Rose, “A recursive extrapolation approach to intra prediction in video coding,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 1734–1738.

[5] S. Li, Y. Chen, J. Han, T. Nanjundaswamy, and K. Rose, “Rate-distortion optimization and adaptation of intra prediction filter parameters,” in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 3146–3150.

[6] L. Zhang, X. Zhao, S. Ma, Q. Wang, and W. Gao, “Novel intra prediction via position-dependent filtering,” *J. Visual Commun. Image Represent.*, vol. 22, pp. 687–696, Nov. 2011.

[7] X. Qi, T. Zhang, F. Ye, A. Men, and B. Yang, “Intra prediction with enhanced inpainting method and vector predictor for HEVC,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2012, pp. 1217–1220.

[8] H. Chen, T. Zhang, M.-T. Sun, A. Saxena, and M. Budagavi, “Improving intra prediction in high-efficiency video coding,” *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3671–3682, Aug. 2016.

[9] C. Chen, S. Zhu, B. Zeng, and M. Gabbouj, “A new block-based method for HEVC intra coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 8, pp. 1727–1736, Aug. 2017.

[10] L. F. Lucas, N. M. Rodrigues, E. A. da Silva, C. L. Pagliari, and S. M. de Faria, “Image coding using generalized predictors based on sparsity and geometric transformations,” *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 4046–4060, Sep. 2016.

[11] L. F. Lucas, N. M. Rodrigues, C. L. Pagliari, E. A. da Silva, and S. M. de Faria, “Sparse least-squares prediction for intra image coding,” in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2015, pp. 1115–1119.

[12] A. Gabriellini, D. Flynn, M. Mrak, and T. Davies, “Combined intra prediction for high-efficiency video coding,” *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 7, pp. 1282–1289, Nov. 2011.

[13] X. Cao *et al.*, “Short distance intra coding scheme for high efficiency video coding,” *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 790–801, Feb. 2013.

[14] Y. Ye and M. Karczewicz, “Improved H.264 intra coding based on bi-directional intra prediction, directional transform, and adaptive coefficient scanning,” in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 2116–2119.

[15] S.-L. Yu and C. Chrysafis, “New intra prediction using intra macroblock motion compensation,” in *Proc. 3rd Meeting Joint Video Team*, May 2002, Paper JVT-C151.

[16] T. K. Tan, C. S. Boon, and Y. Suzuki, “Intra prediction by template matching,” in *Proc. IEEE Int. Conf. Image Process.*, 2006, pp. 1693–1696.

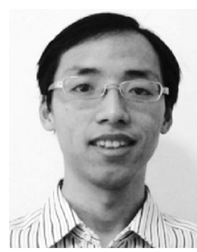
[17] T. K. Tan, C. S. Boon, and Y. Suzuki, “Intra prediction by averaged template matching predictors,” in *Proc. IEEE 4th IEEE Consum. Commun. Netw. Conf.*, 2007, pp. 405–409.

[18] T. Zhang, H. Chen, M.-T. Sun, D. Zhao, and W. Gao, "Hybrid angular intra/template matching prediction for HEVC intra coding," in *Proc. IEEE Visual Commun. Image Process.*, 2015, pp. 1–4.
[19] J. Xu, A. Tabatabai, and O. Nakagami, "On intra block copying in RExt," ITU-T and ISO/IEC, Geneva, Switzerland, Doc. JCTVC-O0232, Oct. 2013.
[20] I. Kim, K. McCann, K. Sugimoto, B. Bross, and W.-J. Han, "High efficiency video coding (HEVC) test model 14 (HM14) encoder description," JCT-VC Doc. P1002, Jan. 2014.
[21] HM Reference Software 14.0, [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware /tags/HM-14.0/
[22] F. Bossen, "Common test conditions and software reference configurations," JCTVC-K1100 Doc., Oct. 2012.
[23] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," VCEG-M33 Document, Apr. 2001.
[24] J. Xu, R. Joshi, and R. A. Cohen, "Overview of the emerging HEVC screen content coding extension," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 50–62, Jan. 2016.
[25] E. Alshina *et al.*, "Known tools performance investigation for next generation video coding," ITU-T SG16/Q6 Doc. VCEG-AZ05, Jun. 2015.
[26] A. Said *et al.*, "Position dependent intra prediction combination," MPEG doc. m37502 and ITU-T SG16 Doc. COM16–C1016, Oct. 2015.
[27] J. Chen, E. Alshina, G. J. Sullivan, J.-R. Ohm, and J. Boyce, "Algorithm description of joint exploration test model 1," 1st Meeting Joint Video Exploration Team of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Oct. 2015, Doc. JVET-A1001.
[28] JEM reference software, [Online]. Available: https://jvet.hhi.fraunhofer.de/svn/svn_HMJEMSoftware/
[29] T. Zhang, M. T. Sun, D. Zhao, and W. Gao, "Fast intra mode and CU size decision for HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 8, pp. 1714–1726, Aug. 2017.
[30] C. Zhu, X. Lin, and L. P. Chau, "Hexagon-based search pattern for fast block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 5, pp. 349–355, May 2002.
[31] W. Xiao, B. Li, J. Xu, G. Shi, and F. Wu, "HEVC encoding optimization using multi-core CPUs and GPUs," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1830–1843, Nov. 2015.
[32] C. Yan *et al.*, "Efficient parallel framework for HEVC motion estimation on many-core processors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 12, pp. 2077–2089, Dec. 2014.
[33] W. Zhu, W. Ding, J. Xu, Y. Shi, and B. Yin, "Hash-based block matching for screen content coding," *IEEE Trans. Multimedia*, vol. 17, no. 7, pp. 935–944, Jul. 2015.
[34] W. Zhu, W. Ding, J. Xu, Y. Shi, and B. Yin, "Screen content coding based on HEVC framework," *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1316–1326, Aug. 2014.
[35] L. Zhao, T. Lin, K. Zhou, S. Wang, and X. Chen, "Pseudo 2D string matching technique for high efficiency screen content coding," *IEEE Trans. Multimedia*, vol. 18, no. 3, pp. 339–350, Mar. 2016.
[36] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. 37th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2003, vol. 2, pp. 1398–1402.

Tao Zhang received the B.S. degree from Xidian University, Xi'an, China, in 2010, the M.S. and Ph.D. degree from Harbin Institute of Technology (HIT), Harbin, China, in 2012 and 2017, all in computer science. He is currently at Tencent. From September 2012 to September 2013, he was with Microsoft Research Asia, Beijing, China, as an Intern. From September 2014 to September 2015, he was a visiting student in Information Processing Lab, University of Washington at Seattle, WA, USA. His research interests include video coding and image processing.

Xiaopeng Fan (S'07–M'09–SM'17) received the B.S. and M.S. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 2001 and 2003, respectively, and the Ph.D. degree from the Hong Kong University of Science and Technology, Hong Kong, in 2009. In 2009, he was in the Department of Computer Science, HIT, where he is currently an Associate Professor. From 2003 to 2005, he was with the Intel China Software Laboratory as a Software Engineer. His current research interests include image/video coding and processing and video streaming and wireless communication. He has authored or co-authored over 50 technical journal and conference papers.

Debin Zhao (M'11) received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology (HIT), Harbin, China, in 1985, 1988, and 1998, respectively. He is currently a Professor in the Department of Computer Science, HIT. He has authored or coauthored more than 200 technical articles in refereed journals and conference proceedings. His research interests include the areas of image and video coding, video processing, video streaming and transmission, and pattern recognition.



Ruiqin Xiong (M'08–SM'17) received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2001, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2007. He was a Research Intern with Microsoft Research Asia, Beijing, from 2002 to 2007, and a Senior Research Associate with University of New South Wales, Sydney, Australia, from 2007 to 2009. He joined the School of Electronic Engineering and Computer Science, Peking University, in 2010, where he is currently a Professor. He has published over 100 technical papers in refereed international journals and conferences. His research interests include statistical image modeling, deep learning, image and video processing, and compression and communications. Dr. Xiong received the Best Paper Award at the 2011 IEEE Visual Communications and Image Processing conference and the Best Student Paper Award at the 2005 SPIE Visual Communications and Image Processing Conference.

Wen Gao (S'87–M'88–SM'05–F'09) received the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991. He is currently a Professor of computer science with Peking University, Beijing, China. Before joining Peking University, he was a Professor of computer science with the Harbin Institute of Technology, Harbin, China, from 1991 to 1995, and a Professor at the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He has authored or coauthored five books and more than 600 technical articles in refereed journals and conference proceedings. His research interests include the areas of image processing, video coding and communication, pattern recognition, multimedia information retrieval, multimodal interface, and bioinformatics. Prof. Gao has served or currently serves on the editorial board for several journals, such as the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT, the *EURASIP Journal of Image Communications*, and the *Journal of Visual Communication and Image Representation*. He chaired a number of prestigious international conferences on multimedia and video signal processing, such as the IEEE ICME and ACM Multimedia, and also served on the advisory and technical committees of numerous professional organizations.