Hybrid Angular Intra/Template Matching Prediction for HEVC Intra Coding

Tao Zhang*, Haoming Chen†, Ming-Ting Sun†, Debin Zhao* and Wen Gao*
*Harbin Institute of Technology, Harbin, China
E-mail: taozhang.hit@hotmail.com
†University of Washington, Seattle, USA
E-mail: sun@ee.washington.edu

Abstract— **In the latest HEVC video coding standard, angular intra prediction (AIP) applies 35 modes including 33 angular modes which can handle blocks with direction information well, and other 2 modes (DC and planar) which are used to predict smooth blocks. However, for blocks with complex texture, these modes may not give good predictions. Some complex blocks can be predicted well by template matching prediction (TMP) which was proposed to predict blocks having similar patterns in the coded regions of the same frame without the cost of large overheads. In this paper, a novel hybrid AIP/TMP is proposed to improve the prediction efficiency. Experimental results show that the proposed method can achieve a coding gain of about 2.5% for high resolution sequences on average compared to the AIP in HEVC. The gain can be up to 4.2%.**

Index Terms— **HEVC, angular intra prediction, template matching, hybrid prediction, intra coding.**

I. Introduction

Intra coding is essential for video coding. In the latest video coding standard HEVC [1], the intra coding process is similar to the previous coding standards such as H.264/AVC [2], which goes through prediction, transform, quantization and entropy coding. The intra prediction is one of the most important processes in intra coding. In the intra prediction of recent video coding standards, it is implemented by copying the reconstructed neighbour pixels of the current block along a specific direction which is indicated by the intra prediction mode.

In the HEVC angular intra prediction (AIP) [3], there are 35 modes in total. Among these intra prediction modes, 33 modes have angular information which is efficient to predict blocks with directional patterns and the other 2 modes, which are DC mode and planar mode, are used to predict smooth blocks. Compared to only 9 modes in the previous video coding standard, H.264/AVC, the prediction efficiency of AIP in HEVC has a significant improvement since it can handle blocks with more directions well.

However, for complex blocks (e.g., blocks with multiple directions or with complicated texture), AIP will not be able to give a good prediction. An example is shown in Fig.1. In Fig. 1(b), the sharp stripes can be predicted well by AIP. However, the complex texture around the stripes are ignored in the prediction block. In these cases, the coding efficiency of AIP is not good.

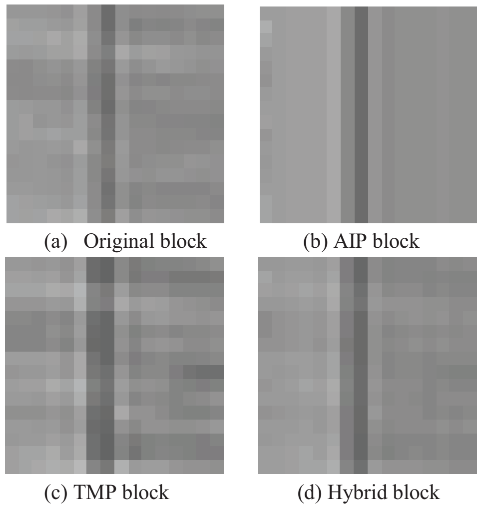

Fig. 1 An example of a 16×16 Luma block with complicated texture from *BasketballPass*. (a) Original block, (b) prediction from AIP with mode 26, (c) prediction from TMP, and (d) prediction from the proposed hybrid method with weight 6.

Some complex blocks can be predicted well by template matching prediction (TMP) [4] [5] which was firstly proposed based on H.264/AVC. The template matching process is shown in Fig. 2. In the TMP process, some reconstructed neighbour pixels of the coding block are used as the template to search one or several best matched template candidates with small SAD values which are similar to the current template from the reconstructed region in the current coding frame. The blocks associated with the matching templates will be used as the prediction blocks for the current coding block.

It should be noted that whether the coding blocks are complex or not, TMP can generate a good prediction if the block and its adjacent template are correlated and similar block patterns as current coding blocks exist in the image. Another important feature of the TMP is that the overhead cost is relatively low since the best candidate can be derived in the decoder sider by using the reconstructed templates. Considering these, the TMP can be used as a coding tool for intra prediction especially for blocks which cannot be well predicted by AIP. Fig. 1(c) shows the prediction block generated by TMP. We can see some complex texture around the vertical stripes can be predicted well by TMP. However, it also introduce some new texture which is not similar to the corresponding parts in the original block.

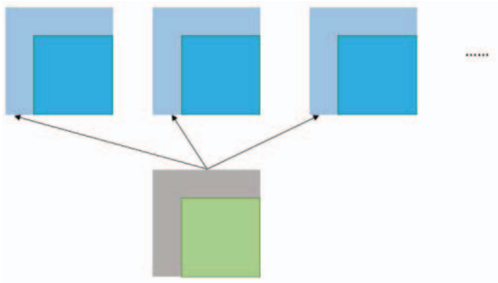

Fig. 2 Template matching process. Inverse-L shaped areas are the templates (in light blue and grey) used in the matching process. The deep blue blocks are the prediction candidates for the current block (in green).

For some blocks, both TMP and AIP can only predict some parts of a block. In this paper, we show that a better prediction can be generated by a weighted average of the predictions from TMP and AIP. As shown in Fig. 1(d), the hybrid prediction keeps the sharp stripes as the original blocks and also retains some complex texture. This combination improves the prediction accuracy and leads to coding improvements.

The rest of the paper is organized as follows. Section II proposes the hybrid AIP/TMP approach. Experimental results are shown and discussed in Section III. Finally, Section IV concludes the paper.

II. PROPOSED METHOD

In this section, the proposed hybrid AIP/TMP prediction will be described in details first, then the signalling method for the added syntax will be analyzed and presented.

A. Hybrid AIP/TMP Prediction

For a coding block O, we assume the prediction from the AIP and TMP is P_a and P_t, respectively. The prediction by hybrid AIP/TMP is generated as:

$$P_h = w * P_a + (1-w) * P_t \quad (1)$$

where P_h is the hybrid prediction and w is the weight used balance the importance between P_a and P_t.

It can be seen that the proposed hybrid prediction covers the pure AIP and pure TMP process. If w is set to 1, the prediction only comes from AIP. On the contrast, if w is set to 0, the prediction will be degraded to TMP. When w is in the range (0, 1), the predictions from AIP and TMP will be combined to derive a hybrid prediction for the current coding block. As a result, the hybrid prediction by (1) is a more general prediction process. Fig. 3 shows the process to generate the hybrid prediction.

In the proposed hybrid AIP/TMP method, a variant version of (1) is used which is expressed as

$$P_h = (w_a * P_a + w_t * P_t + r) >> n \quad (2)$$

where n is set to 3 in this paper, r is the rounding value which is set to 4, and w_a and w_t is the weight factor used to represent the importance of AIP and TMP in this hybrid prediction, respectively. The range of w_a and w_t is from 0 to 8. The sum of w_a and w_t should be 8. Similar to equation (1), when w_a is equal to 0 (8), the hybrid prediction will degrade to TMP (AIP).

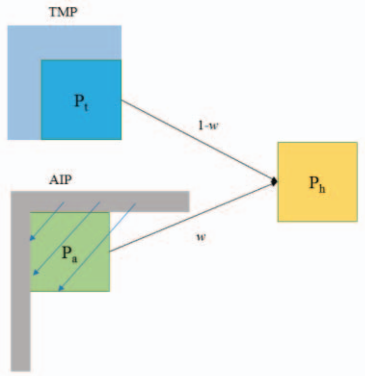

Fig. 3 Hybrid AIP/TMP process

This hybrid prediction method is implemented in the CU level for HEVC intra coding. The best weight factor w_a will be chosen using the RD (Rate-Distortion) optimization process. We divide the weights into three groups: group 1 with weight 0, group 2 with weights 1-7 and group 3 with weight 8. Three different strategies as shown in Fig. 4 are used to generate predictions for these three weight groups. The details will be given as below.

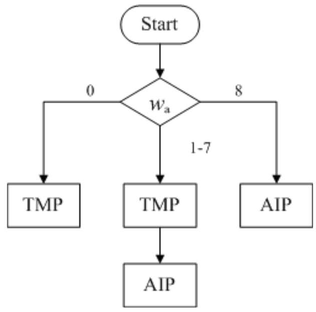

Fig. 4 prediction strategies for different weight groups.

For group 1, the prediction is derived by the TMP process and the CU sizes used in TMP are limited to 8×8 and 16×16 considering that for larger CU sizes, the TMP cannot find good matches and the complexity for searching larger templates is huge according to the experiments. For N×N PU partitions, the TMP will not be used. According to [6], averaging several matching prediction candidates will generate a better prediction than just using one candidate since this averaging can reduce the noise in the reconstructed region caused by quantization. The averaged prediction can be represented as:

$$P_t\,(\text{average}) = (P_{t1} + P_{t2} + \ldots + P_{tN}) / N \quad (3)$$

where P_{ti} are the prediction candidates obtained by the template matching process and N is the allowed maximum number of the candidates. In this paper, N is set to 8.

We found that the candidate with the smallest template's SAD usually is the most reliable candidate, which means this candidate prediction will be chose more frequently than any other candidates. The index of this candidate can be derived by:

$$index = \arg\min\,(\mathrm{SAD}\,(T_1), \mathrm{SAD}\,(T_2), \ldots, \mathrm{SAD}\,(T_N)) \quad (4)$$

where T_i indicate the templates of the candidates and SAD (T_i) is the SAD value between T_i and the template of the current coding block. The prediction from this candidate is P_t (*index*). With this observation, in our TMP, two candidates, the averaged candidate and the candidate with smallest template's SAD are compared. The one which minimizes the SAD between the current coding block and the prediction will be chose as the best prediction for current block. Since there are two prediction candidates, the best one needs to be signaled by a flag with 1 bit. This simple modification improves the TMP prediction accuracy, which we will show in the experimental section.

For group 2, both TMP and AIP process will be conducted to derive a hybrid prediction. Here, the TMP similar to the process for group1 is implemented first to get P_t (average). Differently, for this group, P_t will simply be set to P_t (average), so no comparison needs to be conducted. After that, the AIP process will generate a prediction P_a, and the hybrid prediction P_h can be calculated. To minimize the prediction error from the hybrid prediction, the best weight value and the angular intra mode are chosen by the RD process, and should be coded and transmitted. It should be noted that in this group, the N×N PU partitions will also be used. For N×N PU partitions, the prediction from the TMP process is derived for the CU with size of 2N×2N and the AIP process for each N×N will be conducted with the corresponding N×N prediction part from TMP.

For group 3, the prediction is only generated by AIP which is the same process as the intra prediction in HEVC.

B. Signalling for the Hybrid Prediction

For the hybrid AIP/TMP prediction method, additional two flags, the weight value w_a and the best prediction candidate flag in the pure TMP, need to be encoded and transmitted.

For the weight w_a, it also indicates the prediction group information as mentioned in the previous section. In order to get an appropriate binaryzation method, we calculate the distribution for the weight w_a for 4 selected sequences. This distribution is illustrated in Fig. 5. We can see that pure AIP ($w_a = 8$) is the most frequently used since most of the nature images are composed of many straight edges which can be handled well by AIP. The frequency for weight 0 to 7 is almost the same.

According to the distribution of the weight w_a, two-layer signaling is used for the weight. In the first layer, one bit flag is used to indicate whether the weight is 8 or not. If the weight is 8 which means the prediction is only from AIP and syntax related to AIP needs to be encoded. If the weight is not 8, fixed length binaryzation with 3 bits will be used to represent the 8 different weights from 0 to 7 in the second layer signaling. For weight 0, only a flag with one bit need to be encoded to define which prediction candidate is used in TMP. For weights 1-7, only the syntax related to AIP needs to be signaled.

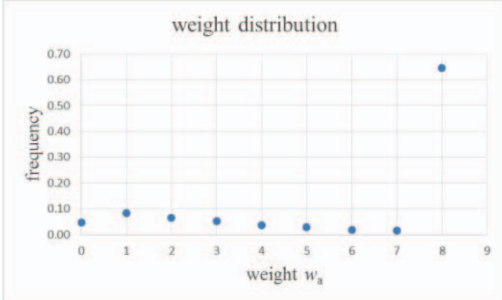

Fig. 5 weight distribution

III. EXPERIMENTAL RESULTS

This section presents the experimental results for the proposed hybrid AIP/TMP prediction method. The proposed method is integrated to HEVC reference software HM14.0.

A. Experimental Settings

The simulation results are provided using the test sequences from JCT-VC [7] encoding with the All-Intra main configuration. QP values of 22, 27, 32, and 37 are used for the evaluation. We measure the coding performance by the Bjontegaard-Delta Bit Rate (BD-BR) [8]. Negative value represents coding gain. In the TMP process, the width of template is two pixels adjacent to the coding block. Several methods are compared. For TMP, the TMP method only using averaged candidate (TMP (average)) as the prediction is also implemented as an additional prediction tool based on AIP to compare with the proposed TMP (TMP (proposed)) where two candidates are used. Moreover, a simplified version (proposed (simple)) of the proposed hybrid prediction is also tested. In the simplified version, the weight value can be only from 0, 4, and 8, which means in group 2 only averaging between TMP and AIP is used. In this case, 1 or 2 bits are needed to indicate the best weight. For all of the comparisons, the anchor is HEVC intra coding with default settings.

B. Performance of the Proposed Hybrid AIP/TMP Method

Table I shows the performance for different prediction methods. The pure TMP using the averaged candidate has a coding gain of 0.6% and 1.4% for low resolution and high resolution sequences, respectively. It can be seen that compared to TMP (average), the proposed TMP have another coding gain, which is 0.1% for low resolution (LR) sequences and 0.2% for high resolution (HR) sequences.

The proposed hybrid prediction method outperforms other methods. The average coding gain compared to the anchor is about 1.2% and 2.5% for LR sequences and HR sequences, respectively. The gain can be up to 4.2% for the sequence *BQTerrace*. There has another 0.7% coding gain on average compared to the proposed pure TMP. The simplified method can achieve 0.9% and 1.9% bit rate reduction. Since the flexible weight is used in the proposed hybrid prediction, the proposed hybrid prediction outperforms the simplified one even though the overhead is larger for the proposed method without simplifications.

TABLE I
PERFORMANCE COMPARISONS FOR THE PROPOSED METHODS

	Test Sequences	TMP (average)	TMP (proposed)	Proposed (simple)	Proposed method
LR Sequences	BasketballDrill	-2.9%	-3.1%	-3.5%	**-3.6%**
	BasketballPass	-0.8%	-0.9%	-1.1%	**-1.4%**
	BlowingBubbles	0.0%	-0.1%	-0.2%	**-0.5%**
	BQMall	-0.4%	-0.5%	-0.9%	**-1.0%**
	BQSquare	-0.1%	-0.3%	-0.6%	**-0.6%**
	PartyScene	-0.3%	-0.4%	-0.6%	**-1.1%**
	RaceHorses	-0.1%	-0.1%	-0.2%	**-0.4%**
	RaceHorses	-0.2%	-0.2%	-0.3%	**-0.7%**
HR Sequences	BasketballDrive	-1.7%	-1.7%	-2.5%	**-3.6%**
	BQTerrace	-3.1%	-3.3%	-3.2%	**-4.2%**
	Cactus	-1.2%	-1.6%	-1.9%	**-2.2%**
	Kimono1	-0.2%	-0.2%	-0.2%	**-0.5%**
	ParkScene	-0.1%	-0.1%	-0.9%	**-1.1%**
	Traffic	-0.9%	-1.1%	-1.4%	**-2.0%**
Average (LR Sequences)		-0.6%	-0.7%	-0.9%	**-1.2%**
Average(HR Sequences)		-1.4%	-1.6%	-1.9%	**-2.5%**
Total Average		-1.0%	-1.2%	-1.4%	**-1.9%**

To further analyse the coding performance of the proposed hybrid AIP/TMP method, weight distributions for four sequences are presented in Table II. From Table II, we can see that AIP (weight = 8) takes up the largest proportion which is 59% on average, while the hybrid prediction with weights 1-7 occupies 37% which is also a large proportion. It demonstrates that the hybrid predication is often used in the intra prediction. We also find that for sequences (e.g., *BQTerrace*) with more blocks coded by hybrid prediction with weights 0-7, the coding gain is higher. This is because for lots of coding blocks in these sequences, this hybrid prediction can provide more accurate prediction than AIP.

TABLE II
WEIGHT DISTRIBUTIONs FOR THE PROPOSED HYBRID PREDICTION

Test Sequences	0	1-7	8
BasketballDrill	3.9%	34.5%	61.6%
BQMall	2.1%	26.6%	71.4%
BQTerrace	7.1%	51.8%	41.1%
Traffic	2.8%	35.3%	61.9%
Average	4.0%	37.0%	59.0%

We also show the distribution of blocks coded by the hybrid prediction in the first frame of sequence *BasketballPass* in Fig. 6. In the figure, the yellow blocks are coded by AIP, the green blocks are coded by TMP and the red blocks are coded by the hybrid prediction with weights 1-7. We can see that for blocks coded by weights in 0-7, similar patterns can be found in the neighbouring area and AIP are often chosen to predict blocks with smooth texture or with strong directions.

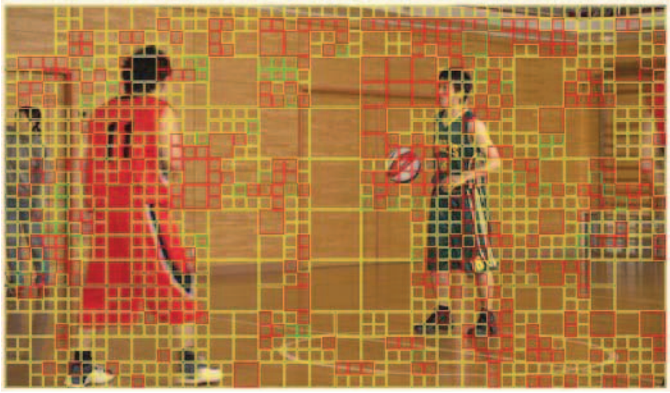

Fig. 6. Distribution of blocks coded by different weights in the proposed hybrid prediction method.

IV. CONCLUSIONS

This paper presents a novel hybrid AIP/TMP prediction method for intra coding. Compared to pure AIP or TMP, the proposed prediction is a more general and accurate prediction for blocks which cannot be predicted well by the AIP prediction in HEVC intra coding. The experimental results demonstrate the prediction efficiency of the proposed method and show that the proposed hybrid prediction achieves 1.9% coding gain on average compared to the original AIP in HEVC. Moreover, the coding gain can be up to 4.2%. In the future, adaptive weight selection without signaling the best weight can be studied to further improve the coding efficiency and also to reduce the coding complexity.

ACKNOWLEDGMENT

This work has been supported in part by the Major State Basic Research Development Program of China (973 Program 2015CB351804), and the National Science Foundation of China under Grant No. 61272386.

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, 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] G. J. 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.

[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] T. K. T. T. K. Tan, C. S. B. C. S. Boon, and Y. S. Y. Suzuki, "Intra Prediction by Template Matching," in 2006 International Conference on Image Processing, 2006, pp. 1693–1696.

[5] Chen, H., Chen, Y.-S., Sun, M.-T., A. Saxena, and M. Budagavi, "Improvements on Intra Block Copy in Natural Content Video Coding," In 2015 IEEE International Symposium on Circuits and Systems (ISCAS), 2015, pp. 2772–2775.

[6] Tan, T. K., Boon, C. S., and Suzuki, Y., "Intra Prediction by Averaged Template Matching Predictors," In 2007 4th IEEE Consumer Communications and Networking Conference, 2007, pp. 405–409.

[7] F. Bossen, Common Test Conditions and Software Reference Configurations, document JCTVC-B300, ITU-T and ISO/IEC, Geneva, Switzerland, Jul. 2010.

[8] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," ITU-T Q.6/SG16, document VCEG-M33, 2001.