Improved Intra Angular Prediction with Novel Interpolation Filter and Boundary Filter

Rujun Wei[1,2,3], Rong Xie[1,3], Li Song[1,2,3], Liang Zhang[1,3], Wenjun Zhang[1,3]
[1]Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University
[2]The State Key Laboratory of Integrated Services Networks, Xidian University
[3]Cooperative Medianet Innovation Center, Shanghai, China
Email: iujean@sjtu.edu.cn, xierong@sjtu.edu.cn, song_li@sjtu.edu.cn, liang.zhang@ieee.org, zhangwenjun@sjtu.edu.cn

***Abstract*—In this paper, two improved intra angular prediction methods are proposed to enhance coding performance. The first method applies new four-tap interpolation filter algorithm. The reference samples at fractional position are interpolated by DCT-based or Gaussian interpolation filter. The second method proposes extended boundary prediction filter to reduce the prediction error. The experimental results show that for AI configuration, the overall coding gain is about 0.85% on average comparing to HEVC reference software while maintaining almost the same coding time.**

I. INTRODUCTION

Video coding technology, which is essential to compress, store and transmit video data, has been widely used in all kinds of fields such as satellite, broadcasting, digital television, surveillance video and so on. In early years, many video coding standards have been developed by ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG), such as MPEG-2, MPWF-4, H.264/AVC, etc. These two organizations then worked together and formed a group named the Joint Collaborative Team on Video Coding (JCT-VC) to formulate an international coding standard HEVC [1], [2]. HEVC adopts a new quad-tree based coding structure with various coding blocks including Coding Unit (CU), Prediction Unit (PU) and Transform Unit (TU). Besides, new coding technologies, like advanced motion vector prediction, sample adaptive offset [3] and wavefront parallel processing, are employed in the encoding process. With these new tools, HEVC achieves about 50% bitrate reduction comparing to H.264/AVC standard.

However, with the fast development of multimedia information and network technology, the requirement for high resolution videos and better visual experience has grown rapidly. Meanwhile, the volume of video data has increased dramatically, which demands for higher compression ratio than ever before. To compress high resolution videos more efficiently, a new video coding standard which accomplishes better coding performance than HEVC is required urgently. In October 2015, ITU-T VCEG and ISO/IEC MPEG organizations jointly found a new group named Joint Video Exploration Team (JVET) to develop Future Video Coding beyond HEVC [4]. The reference software for the JVET group is named Joint Exploration Model (JEM) [5]. The JEM is based on the HEVC Model (HM), which is the reference software for the HEVC standard. So far, the latest JEM software has integrated many new encoding strategies and achieved excellent compression performance comparing to HM software.

Intra frame coding is the key process of video coding. To improve the performance of intra prediction, some approaches have been proposed, which can be divided into two categories. (1) Add new coding block partition methods. K. Kawamura et al. presented an asymmetric partitioning algorithm with a non-power-of-two transform as a prediction and transform unit [6]. In [7], a short distance intra prediction scheme was proposed to improve intra coding efficiency. By dividing the NxN block into lines or non-square blocks, the distance between predicted pixels and its reference pixels can be narrowed, therefore reducing the energy of the prediction residuals. (2) Change the generation process of intra prediction signal. In [8], the predicted signals of the angular prediction mode were generated by blending the predicted signals of the selected angular prediction direction and those of the other mode which has the opposite prediction direction. Method [9] generated weighted prediction signals along the prediction direction and the weight was proportional to the distance between the reference sample and the predicted sample. In [10], C. –H. Yeh et al. proposed a new intra prediction algorithm based on synthesizing two neighboring predictors.

In this paper, novel intra angular prediction algorithms are proposed to enhance the performance of intra coding. Our works can be divided into two parts: four-tap interpolation filter and boundary prediction filter. For four-tap interpolation filter, blending filters are applied to generate the reference samples at fractional position. For boundary prediction filter, the conventional boundary filtering in vertical and horizontal mode is extended to other angular modes.

The rest of the paper is organized as follows. Section II introduces the method of four-tap intra interpolation filter. Section III presents boundary prediction filtering algorithm. The experimental results are shown in Section IV. Finally, a brief conclusion is given in Section V.

II. FOUR-TAP INTRA INTERPOLATION FILTER

In HEVC, intra prediction supports 35 prediction modes, including 33 directional modes, a planar mode and a DC mode.

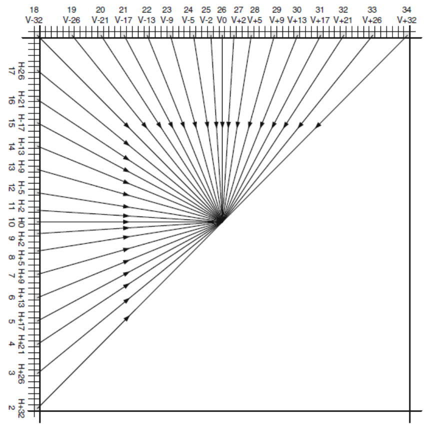

Figure 1. Angular prediction modes and directions [11].

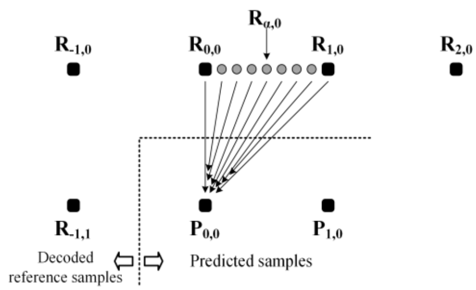

Figure 2. Intra angular prediction [12].

As illustrated in Fig. 1, HEVC defines 33 angular prediction directions with 1/32 sample accuracy. For angular prediction modes, a two-tap linear interpolation filter is used to generate the intra prediction block. Fig. 2 demonstrates the process of intra angular prediction, where $R_{i,j}$ and $P_{i,j}$ represent the reference samples and prediction samples, respectively. For $P_{0,0}$, its reference sample $R_{\alpha,0}$ is generated by two samples $R_{0,0}$ and $R_{1,0}$ with the two-tap interpolation filter, as shown in (1).

$$P_{0,0} = R_{\alpha,0} = ((32-\omega) * R_{0,0} + \omega * R_{1,0} + 16) >> 5 \tag{1}$$

Where ω denotes the weighting factor.

In this paper, a new four-tap intra interpolation filter is proposed to generate the prediction signals with higher prediction accuracy. In Fig. 2, the reference sample $R_{\alpha,0}$ is now interpolated by four sample $R_{-1,0}$, $R_{0,0}$, $R_{1,0}$ and $R_{2,0}$ as described in (2).

$$R_{\alpha,0} = p_0 * R_{-1,0} + p_1 * R_{0,0} + p_2 * R_{1,0} + p_3 * R_{2,0} \tag{2}$$

Where p_i (i=0, 1, 2, 3) are the coefficients of four-tap interpolation filter. In this proposal, two types of interpolation filters are utilized: DCT-based interpolation filter [12] for smaller blocks, and Gaussian interpolation filter [5] for larger blocks. We will introduce the algorithm of four-tap interpolation filter in detail in the following subsections.

A. DCT-based and Gaussian interpolation filters

Suppose $\{r_i\}$ (i=-M+1,…,M) are integral pixel values and r_α ($0 < \alpha < 1$) is the fractional pixel value. For DCT-based interpolation filter, the forward DCT calculates the transformed coefficients set according to (3), where M means the filter tap and is set to 2 in this paper. And then the inverse DCT yields the integral pixel values based on the coefficients set (shown in (4)) [12], [13].

$$c_k = \frac{1}{M} \sum_{i=-M+1}^{M} r_i \cos\left(\frac{(2i-1+2M)k\pi}{4M}\right) \tag{3}$$

$$r_i = \frac{c_0}{2} + \sum_{k=1}^{2M-1} c_k \cos\left(\frac{(2i-1+2M)k\pi}{4M}\right) \tag{4}$$

By replacing i with α in (4), the pixel value of the fractional position r_α can be achieved. Then, DCT-based filter coefficients for any fractional position can be obtained by putting formula (3) into formula (4). After that, the calculated filter coefficients are modified by the phase parameter ρ using equation (5), where c_i denotes the filter coefficients and the parameter i means the position of the filter [14]. In this paper, the phase parameter is set to 0.05.

$$c_i = \begin{cases} (1-\rho)c_i + \rho c_{i+1} & (i=-M+1) \\ \rho c_{i-1} + (1-2\rho)c_i + \rho c_{i+1} & (-M+1 < i < M) \\ (1-\rho)c_i + \rho c_{i-1} & (i=M) \end{cases} \tag{5}$$

For Gaussian interpolation filter, the interpolation function is defined by (6).

$$r_i = \sum_{k=-M+1}^{M} b_k g_k(i) , \tag{6}$$

Where b_k is filter coefficients and g_k is Gaussian normal distribution function. The Gaussian function centered at the interpolation nodes α is represented as (7)

$$g_k(i) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(k-i)^2}{2\sigma^2}) . \tag{7}$$

For each integral pixel value, one equation of b_k can be obtained by putting (7) into (6). And the filter coefficients can be achieved by solving these equations. Then the pixel value at fractional position r_α can be calculated based on (6) by replacing i with α.

In practice, the filter coefficients are scaled by the factor 2^S and rounded to integer for higher accuracy, where s denoted the accuracy factor of the filter coefficients. Besides, the scaled filter coefficients should satisfy the normalization conditions as shown in (8), where *filter*(x) denotes the filter coefficients.

$$\sum filter(x) = 2^s \tag{8}$$

B. Four-tap interpolation filter process

For intra angular prediction, the reference samples at the fractional position can be achieved using DCT-based and Gaussian interpolation filters according to previous section. In proposed algorithm, filters are selected according to the block size. For 4x4 and 8x8 blocks, DCT-based filter is chosen while Gaussian filter is applied to larger blocks. However, the filter coefficients are fixed for each directional mode.

Fig. 3 shows the flow chart of proposed intra prediction process, where the steps in gray represent four-tap interpolation filter algorithm. When PU size is larger than 8x8, Gaussian interpolation filter is used to perform intra angular prediction. Otherwise, DCT-based interpolation filter is employed to generate the prediction signals.

III. EXTENED BOUNDARY PREDICTION FILTER

In vertical mode (mode 26) or horizontal mode (mode 10) of HEVC, after the prediction pixels of the block are calculated, the pixels of left-most column or top-most row in the prediction block are further adjusted using equation (9) or (10).

$$P_{1,y}{}' = P_{1,y} + ((R_{0,y} - R_{0,0}) \gg \tag{9}$$

$$P_{x,1}{}' = P_{x,1} + ((R_{x,0} - R_{0,0}) \gg \tag{10}$$

Where $P_{1,y}$, $P_{x,1}$ are original prediction values and $P_{1,y}'$, $P_{x,1}'$ are the modified ones. $R_{0,y}$, $R_{x,0}$ and $R_{0,0}$ represent the reference samples. The operator >> means right shift operation. Fig. 4 (a) illustrates the conventional boundary filtering for vertical mode.

In this paper, the boundary filtering method is extended to additional intra prediction directions. Considering the directional characteristics of intra angular modes, the proposed algorithm is applied to intra prediction from upper right to lower left (modes 27 to 34) and from lower left to upper right (modes 2 to 9).

For example, Fig. 4 (b) shows the boundary filtering process of mode 34. The prediction pixel $P_{1,y}$ in the left-most column is modified using the difference between $R_{0,y}$ and $R_{x,0}$, where $R_{0,y}$ denotes the reference pixel adjacent to $P_{1,y}$ and $R_{x,0}$ represents the sample next to the reference pixel of $P_{1,y}$. More generally, for modes 27 to 34, the boundary prediction signals are modified using the following equation.

$$P_{1,y}{}' = P_{1,y} + (R_{0,y} - R_{x,0}) * \omega \tag{11}$$

Where $P_{1,y}'$ and $P_{1,y}$ represent the modified and original signals respectively. $R_{0,y}$ has the same meaning to the one of mode 34. $R_{x,0}$ denotes the reference pixel which points to $R_{0,y}$ in the direction of corresponding mode and ω is the weight factor fixed in one experiment.

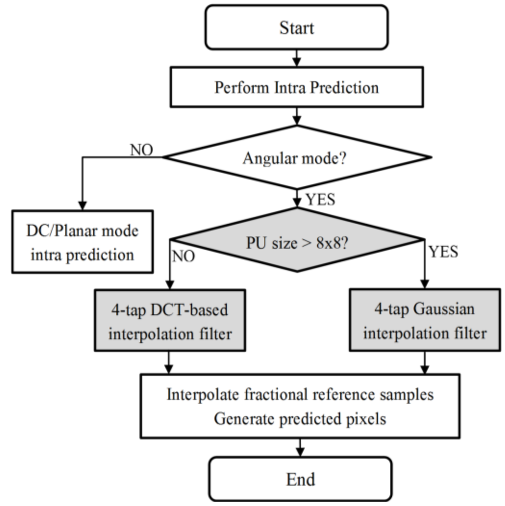

Figure 3. Proposed intra prediction process.

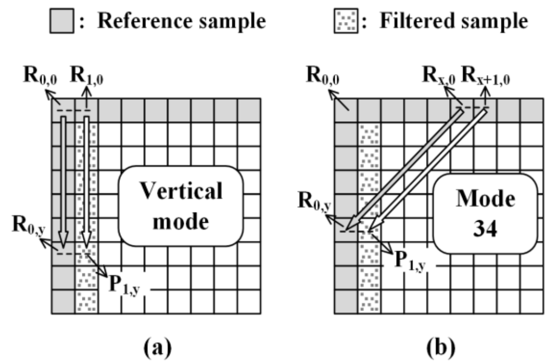

Figure 4. (a) Conventional boundary filtering for vertival mode. (b) Proposed boundary filtering for mode 34.

Similarly, if modes 2 to 9 are selected, the boundary filtering is performed according to (12).

$$P_{x,1}{}' = P_{x,1} + (R_{x,0} - R_{0,y}) * \omega \tag{12}$$

The basis of proposed algorithm is that the difference between two reference pixels represents the variation tendency of pixel values along the prediction direction. Therefore the prediction accuracy can be improved by doing boundary filtering.

IV. EXPERIMENTAL RESULTS

The proposed methods are implemented in HEVC reference software HM-16.6. All intra-main setting is used for experiments. The test platform is Intel(R) Core(TM) i5-4590 CPU@3.30GHz with 8.0 GB RAM. HEVC test sequences [15] including Class A (2560×1600), Class B (1920×1080), Class C (832×480), Class D (416×240) and Class E (1280×720) are simulated for performance evaluation. The quantization parameter values are set to 22, 27, 32 and 37, respectively. The coding performance of the proposed algorithms is measured in terms of Bjøntegaard delta bitrate (BD-Rate) [16].

TABLE I. EXPERIMENTAL RESULTS OF METHOD A COMPARING TO HM-16.6.

Classes	All intra		
	BD-Rate Y	BD-Rate U	BD-Rate V
Class A	-0.17%	-0.26%	-0.20%
Class B	-0.48%	-0.35%	-0.31%
Class C	-0.80%	-0.25%	-0.83%
Class D	-0.66%	-1.57%	-0.75%
Class E	-0.77%	0.25%	-1.48%
Average	**-0.60%**	**-0.47%**	**-0.71%**
Encoding Time	103%		
Decoding Tmie	101%		

TABLE II. PERFORMANCE COMPARISION OF THE PROPOSED METHOD A WITH THE EXISTING METHODS.

Performance	JEM's Method [17]	S. Matsuo et al. [12]	Proposed Method A
Average BD-Rate	**-0.38%**	**-0.41%**	**-0.60%**
Encoding Time	104%	103%	103%
Decoding Time	100%	101%	101%

Table I shows the results of four-tap intra interpolation filter algorithm, called Method A for simplicity. The experimental results indicate that the average coding gains are about 0.6%, 0.47% and 0.71% for Y, U and V component under AI configurations. The performance improvement of low resolution sequences (Class C, Class D and Class E) is higher than that of high resolution sequences (Class A, Class B). And the maximum coding gain is about 2.09% for the sequence "BasketballDill". Besides, the encoding time and decoding time are about 103% and 101%, respectively. Table II shows the performance comparison in terms of BD-Rate and coding time with the existing methods, including JEM [17] and S. Matsuo et al. [12]. JEM's four-tap interpolation filter method has about 0.38% BD-Rate saving on average. And S. Matsuo et al.'s obtains an average 0.41% BD-Rate gain. Compared with these two methods, the proposed Method A shows significantly better RD performance while the computational complexity is almost the same.

In the experiment of boundary filtering, the accuracy of filter coefficients is set to 256. Therefore, the accuracy factor s equals to 8. Table III shows the results of boundary prediction filter (Method B) using different weights. As the weight increases, the average BD-Rate saving gradually grows to a peak and then decreases rapidly. In the case of w=0.35, the maximum BD-Rate reduction of 0.32%, 0.02% and 0.12% (Y, U and V) is obtained. The average encoding time and decoding time are about 101% and 100%. The performance comparison with other methods are shown in Table IV. The JEM's boundary prediction filter [17] achieves about 0.2% coding gains and S. Matsuo et al.'s [8] shows average 0.23% BD-Rate saving. It can be seen that the proposed Method B has a better BD-Rate saving performance than existing methods.

TABLE III. EXPERIMENTAL RESULTS OF METHOD B COMPARING TO HM-16.6.

Weight	All intra		
	BD-Rate Y	BD-Rate U	BD-Rate V
0.10	-0.16%	0.14%	-0.21%
0.15	-0.27%	-0.14%	-0.30%
0.20	-0.27%	-0.10%	-0.16%
0.25	-0.22%	-0.05%	-0.22%
0.30	-0.29%	-0.16%	-0.19%
0.35	**-0.32%**	**-0.02%**	**-0.12%**
0.40	-0.24%	-0.03%	-0.07%
0.45	-0.14%	0.05%	0.02%
0.50	-0.04%	0.12%	0.02%
Encoding Time	101%		
Decoding Tmie	100%		

TABLE IV. PERFORMANCE COMPARISION OF THE PROPOSED METHOD B WITH THE EXISTING METHODS.

Performance	JEM's Method [17]	S. Matsuo et al. [8]	Proposed Method B
Average BD-Rate	**-0.20%**	**-0.23%**	**-0.32%**
Encoding Time	100%	100%	101%
Decoding Time	99%	101%	100%

TABLE V. OVERALL CODING PERFORMANCE OF METHOD A WITH METHOD B.

Classes	All intra		
	BD-Rate Y	BD-Rate U	BD-Rate V
Class A	-1.03%	-0.64%	-0.69%
Class B	-0.74%	-0.26%	-0.44%
Class C	-0.84%	-1.30%	-1.03%
Class D	-0.68%	-1.23%	-0.69%
Class E	-1.18%	0.24%	-0.25%
Average	**-0.85%**	**-0.63%**	**-0.62%**
Encoding Time	104%		
Decoding Tmie	102%		

Table V shows the overall coding performance using Method A and Method B. The weight is set to 0.35 for Method B. Compared with the reference software HM-16.6, the combined method has about 0.85%, 0.63% and 0.62% BD-Rate gains for Y, U and V component, respectively. The encoding time and decoding time are 104% and 102%, respectively.

Therefore, the proposed methods achieve significant coding gains with negligible increment of coding time.

V. CONLUSIONS

In this paper, we propose two novel intra angular prediction methods. The first method employs four-tap interpolation filter to generate the reference samples at the fractional position. For 4x4 and 8x8 blocks, DCT-based interpolation filter is used. For larger blocks, Gaussian interpolation filter is applied. The second method extends original boundary filtering to additional angular prediction modes and experimentally chooses a proper weight factor for better RD performance. HEVC test sequences with various resolutions are tested under AI configuration. Experimental results show that the average BD-rate saving is about 0.60% for the first method. And the maximum BD-Rate gain is about 0.32% in the case of w=0.35 for the second method. When two methods are combined, the overall coding gain is about 0.85%, and the encoding and decoding time are about 104% and 102%. The results verify that the proposed methods can achieve excellent coding performance with tiny increment of coding time.

ACKNOWLEDGEMENT

This work is supported by NSFC (61671296, 61527804, and 61521062) and Shanghai Zhangjiang national independent innovation demonstration zone development fund (201501-PD-SB-B201-001).

REFERENCES

[1] G. J. Sullivan, J.-R. Ohm et al., "Overview of the High Efficiency Video Coding (HEVC) Standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[2] J.-R. Ohm, G. J. Sullivan, H. Schwarz et al., "Comparison of the Coding Efficiency of Video Coding Standards-Including High Efficiency Video Coding (HEVC)," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[3] C.-M. Fu, A. Elena, A. Alexander, et al., "Sample adaptive offset in the HEVC standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1755–1764, Dec. 2012.

[4] JVET JEM software. [Online]. Available: https://jvet.hhi.fraunhofer.de

[5] J. Chen, E. Alshina, G. J. Sullivan et al., "Algorithm Description of Joint Exploration Test Model 1," JVET-B0021, JVET 2nd Meeting: San Diego, Feb. 2016.

[6] K. Kawamura, H. Kato and S. Naito, "Asymmetric Partitioning with Non-power-of-two Transform for Intra Coding," Picture Coding Symposium (PCS), pp. 429-432, May 2012.

[7] X. Cao, X. Peng, C. Lai et al., "CE6.b1 Report on Short Distance Intra Prediction Method," JCTVC-E278, JCT-VC 5th Meeting: Geneva, Mar. 2011.

[8] S. Matsuo, S. Takamura and A. Shimizu, "Intra Angular Prediction with Weight Function and Modification Filter," Picture Coding Symposium (PCS), pp. 77-80, Dec. 2013.

[9] S. Matsuo, S. Takamura and Y. Yashima, "Intra prediction with spatial gradients and multiple reference lines," Picture Coding Symposium (PCS), pp. 1-4, May 2009.

[10] C.-H. Yeh, T.-Y. Tseng, C.-W. Lee et al., "Predictive Texture Synthesis-Based Intra Coding Scheme for Advanced Video Coding," IEEE Transactions on Multimedia, vol. 17, no. 9, pp. 1508-1514, Sep. 2015.

[11] V. Sze, M. Budagavi and G. J. Sullivan, High Efficiency Video Coding (HEVC) algorithm and architecture, Integrated Circuits and Systems, Swizerland, 2014, pp. 97-98.

[12] S. Matsuo, S. Takamura and A. Shimizu, "Modification of Intra Angular Prediction in HEVC," 2012 Asia-Pacific Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), pp. 1-4, Dec. 2012.

[13] S. Matsuo, S. Takamura and H. Jozawa, "Improved intra angular prediction by DCT-based interpolation filter," 20th European Signal Processing Conference (EUSIPCO), pp. 1568-1572, Aug. 2012.

[14] E. Alshina, J. Chen, A. Alshin et al., "CE3: Experimental results of DCTIF by Samsung," JCTVC-D344, JCT-VC 4th Meeting: Daegu, Jan. 2011.

[15] F. Bossen, "Common test conditions and software reference configurations," JCTVC-L1100, JCT-VC 12th Meeting: Geneva, Jan. 2013.

[16] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," VCEG-M33, Video Coding Experts Group (VCEG) Thirteenth Meeting: Austin, Texas, Apr. 2001.

[17] E.Alshina, A.Alshin, K. Choi et al., "Performance of JEM1.0 tools analysis by Samsung" JVET-B0022, JVET 2nd Meeting: San Diego, Feb. 2016.