Adaptive Loop Filter with Temporal Prediction

Xinfeng Zhang[1,2]
[1]Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[2]Graduate University of Chinese Academy of Sciences Beijing 100049, China

Ruiqin Xiong[3], Siwei Ma[3], Wen Gao[3]
[3]Institute of Digital Media, Peking University, Beijing 100871, China
{xfzhang, rqxiong, swma, wgao}@jdl.ac.cn

Abstract—**In this paper, we propose a method to improve adaptive loop filter (ALF) efficiency with temporal prediction. For one frame, two sets of adaptive loop filter parameters are adaptively selected by rate distortion optimization. The first set of ALF parameters is estimated by minimizing the mean square error between the original frame and the current reconstructed frame. The second set of filter parameters is the one that is used in the latest prior frame. The proposed algorithm is implemented in HM3.0 software. Compared with the HM3.0 anchor, the proposed method achieves 0.4%, 0.3% and 0.3% BD bitrate reduction in average for high efficiency low delay B, high efficiency low delay P and high efficiency random access configuration, respectively. The encoding and decoding time increase by 1% and 2% on average, respectively.**

Keywords- Adaptive loop filter, HEVC, video coding, temporal prediction.

I. INTRODUCTION

High Efficiency Video Coding (HEVC) is the upcoming video coding standard under development by the ISO and ITU-T. HEVC has achieved significant coding efficiency improvement beyond existing video coding standard, e.g. MPEG-2 and H.264/AVC, by employing many new coding tools. Adaptive loop filter is one of the most efficient tools. Its basic idea is to estimate wiener filter parameters at the encoder and filter the reconstructed image with them to depress coding artifacts. The ALF used in coding loop is also able to provide a higher quality reference frame for future coding frames. These filter parameters need to be transmitted to the decoder.

There are three types of ALF, i.e. frame-based, block-based and quadtree-based ALFs. In [1], a frame-based ALF is proposed and one picture level flag is employed to control the filter on/off. Although ALF is able to improve the reconstructed image quality on the whole, it may degrade image quality in some local areas. Therefore, a block-based ALF (BALF) is proposed in [2]. BALF applies a filter to luminance blocks and a flag to indicate whether the block is filtered is signaled for each block. Therefore, BALF improves the flexibility of ALF to adapt to different areas. The quadtree-based ALF (QALF) further improves ALF efficiency by signaling the filter switch flag in a quadtree data structure to carry out the variable block size filtering [3].

In HEVC, a modified QALF is proposed in [4]. On one hand, instead of applying a single filter, multiple filters are employed for each frame. For each pixel in a block, a specific filter is assigned, which is based on the local characteristics around the pixel. The number of filters for each frame is also adaptively determined based on the rate distortion optimization at encoder side. Therefore, adaptive filter parameters merging process is introduced at the encoder. In order to remove the redundancy among multiple filter coefficients in each frame, differential coding is applied in filter coefficients coding. On the other hand, since quadtree based coding unit (CU) structure is adopted in HEVC, the basic filter unit for QALF in HEVC is limited not larger than the size of CU.

As known to all, the neighboring frames in a sequence have high correlation. Although the above mentioned methods improve the ALF efficiency greatly, the redundancy of ALF parameters between frames still exists, which limits the potential performance of ALF. In this paper, we propose a simple and effective method to reduce the temporal redundancy of ALF parameters by reusing the prior transmitted filter parameters. In order to get best performance, two sets of filter parameters, namely the prior transmitted filter set and filter set estimated from current frame, are selected based on rate distortion optimization. Only one picture level flag is signaled to indicate which ALF parameter set is employed.

The rest of the paper is organized as follows. A brief overview of ALF in HEVC is provided in Section II. Our proposed temporal prediction ALF and its implementation details are elaborated in Section III. The experimental results are illustrated in Section IV and conclusions are drawn in Section V.

II. ADAPTIVE LOOP FILTER IN HEVC

In the upcoming video coding standard HEVC, a quadtree based adaptive loop filter (QALF) is adopted. It employs the local direction and texture characteristic in reconstructed image to divide the pixels into different categories. For pixels in each category, a specific wiener filter is generated based on the least square estimation. In order to get the best coding efficiency, an adaptive merging process of pixel categories is carried out to decrease bits used in transmitting filter parameters. Thereafter, ALF switch for each CU and its taps are decided based on the rate distortion cost. Finally, the filter parameters are coded into bit stream with entropy coding. Therefore, The ALF process in HEVC is summarized as the following five stages, pixel classification, filter parameters mergence, filter switch decision (ALF CU adaptive) and filter tap decision and filter parameters entropy coding as illustrated in Fig.1.

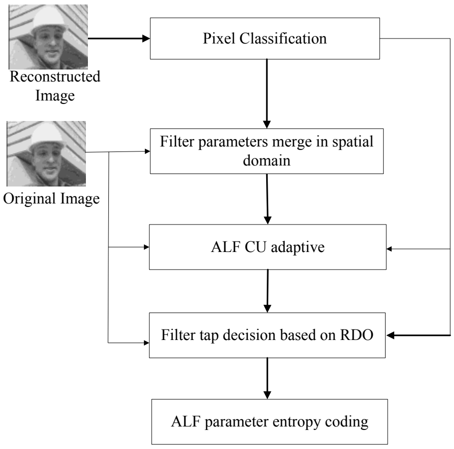


Fig. 1: The adaptive loop filter process in HEVC

A. Pixel classification and adaptive mergence

In HEVC, two pixel classification methods are employed, block-based (BA) and region-based (RA). In BA, the pixels in a 4x4 block are grouped according to local direction and textural characteristics. The 1D Laplacian based gradients [5] are used to represent the direction information and the average of these gradients in a 4x4 block is used as the texture measurement. To be specific, gradients and textual measurement are calculated by (1), (2) and (3) respectively.

$$H = \sum_{i,j} \left| \left(r(i,j) \times 2 - r(i,j-1) - r(i,j+1) \right) \right| \quad i,j = 0,\cdots,3 \quad (1)$$

$$V = \sum_{i,j} \left| \left(r(i,j) \times 2 - r(i-1,j) - r(i+1,j) \right) \right| \quad i,j = 0,\cdots,3 \quad (2)$$

$$T = {(H+V)}/{16} \quad (3)$$

where H and V represent the horizontal and vertical gradients, T is the measurement accounting for texture (also can be regarded as an activity measurement) and $r(i,j)$ denotes reconstructed pixel value at location (i,j). Then, the classification metric C is calculated as follows in (4),

$$C = 5 \times D + T \qquad T = 0,\cdots,4 \quad (4)$$

$$D = \begin{cases} 1 & V > \alpha H \\ 2 & H > \alpha V \\ 0 & else \end{cases} \quad (5)$$

where α is a threshold. In RA, the pixels are classified according to their locations with a predefined table. This classification method is introduced to reduce the ALF complexity at decoder [6].

For pixels in i-th category, the wiener filter coefficients are estimated by solving the following optimization problem,

$$\boldsymbol{h}_i = arg\,\min_{\boldsymbol{h}_i} \left\| \mathbf{R}\boldsymbol{h}_i - \boldsymbol{x}_i \right\|_2^2 \quad (6)$$

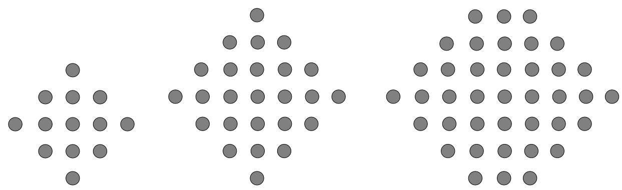
where the vectors $\boldsymbol{x}_i$ and $\boldsymbol{h}_i$ represent the pixel values of the i-th category in original image and the corresponding ALF coefficients to be estimated, the matrix $\boldsymbol{R}_i$ is composed by the reconstructed pixels and each row of $\boldsymbol{R}_i$ are the pixel values used in filter $\boldsymbol{h}_i$ to predict the correspondent value in $\boldsymbol{x}_i$. Therefore, for each pixel category, a special adaptive loop filter is estimated from (6).

In order to repeat the filter process at decoder, multiple groups of filter parameters are needed to transmit, e.g. 15 categories in BA and 16 categories in RA, which will consume lots of coding bits. Therefore, an adaptive merging process is employed to reduce the filter numbers needed to transmit. Merging is conducted on pixel categories and their corresponding ALF coefficients based on the rate distortion optimization. On one hand, the merging procedure indeed generates a new pixel classification result, which is also transmitted to decoder. On the other hand, the merging process makes different frames may have different pixel classification results. Therefore, it makes ALF prediction in temporal difficult, because ALF coefficients vary dramatically for different pixel characteristic.

B. Filter CU adaptive, tap decision and coefficients coding

Although the ALF is useful for the whole category of reconstructed pixels, it may degrade the quality of reconstructed pixels in some local areas. In early versions of quadtree-based ALF, adaptive filter switching scheme is utilized based on different block sizes. It splits the whole image into a quadtree structure and makes a decision on each region whether filter is on or off. For the region that the filter is used, the split process iterates until the region size equals to a predefined minimum value. In HEVC, the coding unit (CU) also employs the iterative quadtree splitting structure. Therefore, The QALF in HEVC takes advantage of CU splitting results and makes a decision for filter switching on each CU.

In order to adapt different image characteristics, three types of filter with different taps are defined in HEVC, which are 5x5, 7x7 and 9x7 illustrated in Fig.2. The filter taps are decided according to the rate distortion cost. Finally, the difference of ALF coefficients between neighboring categories is coded with Exponential-Golomb code.



(a) (b) (c)

Fig.2: (a) 5x5 filter shape, (b) 7x7 filter shape, (c) 9x7 filter shape

III. ADAPITIVE LOOP FILTER WITH TEMPORAL PREDICTION

A. Motivation

Although ALF is able to improve the quality of reconstructed images significantly, the side information for each frame, e.g. the ALF coefficients, pixel classification information and CU adaptive flag, need quite a few bits. As a result, the ALF may be disabled based on rate distortion decision due to the burden of side bits in some cases, especially low bit rate circumstance. As a result that the neighboring video frames usually have similar content and are encoded with approximate quantization parameters (QP), the ALF coefficients for the same category of reconstructed pixels may have high correlation. In TABLE I, we illustrate the percentage of reuse the prior coded ALF parameters for current frame with minimum rate distortion cost using HM3.0 test mode. Only the first frame is coded as intra frame. From TABLE I, we can see that the percentage of reuse ALF parameters is more than 70% for most cases and for *RaceHorses*@416x240, the percentage is up to 96%. Therefore, it illustrates that there is high correlation between ALF parameters in neighboring frames and it is able to improve ALF performance further [7] [8]. In next section, we will introduce our proposed method in detail.

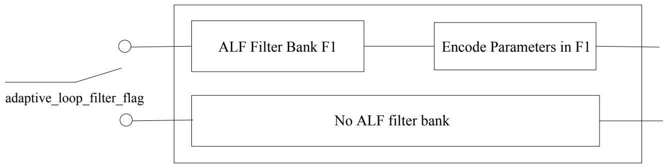
TABLE I. THE PERCENTAGE OF REUSE THE PRIOR CODED ALF PARAMETERS FOR CURRENT FRAME

Sequence	*QP=22*	*QP=27*	*QP=32*	*QP=37*
BasketballDrill_832x416_50	0.61	0.63	0.66	0.63
PartyScene_832x416_50	0.61	0.62	0.71	0.58
BQMall_832x416_60	0.55	0.55	0.54	0.44
RaceHorses_832x416_30	0.81	0.83	0.81	0.81
BasketballPass_416x240_50	0.75	0.72	0.73	0.70
BlowingBubbles_416x240_50	0.80	0.75	0.79	0.74
BQSquare_416x240_60	0.74	0.67	0.78	0.70
RaceHorses_416x240_30	0.96	0.94	0.96	0.89
average	0.73	0.71	0.75	0.69

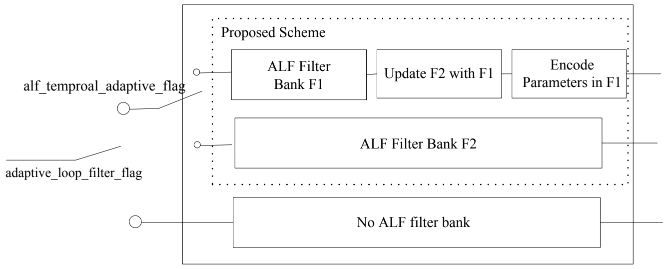
B. Proposed Algorithm and Implement

In order to exploit the temporal correlation between ALF parameters, we propose to reuse the previously transmitted ALF coefficients and their corresponding pixel classification results as an alternative candidate directly, denoted as filter bank F2. The filter bank generated according to the current frame as filter bank F1. The two filter banks are selected according to the rate distortion performance. Therefore, only a picture level flag is needed to indicate which filter bank is adopted for each frame.

The ALF filter process of HM3.0 can be illustrated in Fig.4 (a) and the proposed filter process is illustrated in Fig.4 (b). For intra frame coding, if ALF is enable, the filter bank is set as filter bank F1 and the filter bank F2 is updated with the new ALF parameters in filter bank F1, otherwise we do not change the filter bank F2. For inter frame coding, if ALF filter bank is enable, the filter bank with minimal rate distortion cost is selected as the final one. If filter bank F1 is selected, the ALF parameters of F2 are updated with parameters in filter bank F1. Otherwise, the filter parameters of F2 do not changed. In the latter case, the filter coefficients and pixel classification parameters are not needed to transmit to the decoder, since the parameters in filter bank F2 have been transmitted in prior frame. In our proposed algorithm, it does not increase the calculation complexity at decoder but only needs another buffer to store the parameters in filter bank F2.



(a)



(b)

Fig.4: (a) the simplified ALF structure in HEVC, (b) the proposed ALF structure

IV. EXPERIMENTAL RESULTS

In this section, extensive experiments are conducted to verify the performance of the proposed algorithm. We implement our algorithm in HM3.0 software. Simulations are conducted with the class C and class D test sequences [9]. The performance of original software HM3.0 is used as anchor, and the common test condition is defined in [10]. We carry out our experiments on the random access, low delay B slice and low delay P slice in high efficiency, which are denoted as HE_RA, HE_LD_B and HE_LD_P, respectively. Since our algorithm does not take effects in intra frame coding, the all intra configuration is not conducted in our experiments. For each video sequence, four quantization parameter values are tested: 22, 27, 32 and 37. All the frames in the test sequences are encoded. The BD-rate is calculated according to the method in [11].

From the experimental result in TABLE II~IV, we can see that our proposed algorithm improves the ALF performance in all the test sequence. It achieves about 0.4%, 0.3% and 0.3% BD bitrate saving for luminance component. The largest BD bitrate saving is up to 0.8% for luminance and 1.4% for chroma. The improvement is larger on low resolution sequences. The reason is that the low resolution sequences are usually coded into a relative lower bit rate and the percentage of ALF parameters bit rate saving accordingly increases.

We also evaluate the increased complexity by comparing the average encoding time and decoding time. The executable file was compiled by Microsoft Visual Studio 2010, 64 bit. The tests are run on Windows Sever 2008 R2 64 bit and the CPU in the test is 2xIntel Xeon L5420@2.5GHz.The results are illustrated in TABLE V. From TABLE V, we can see that the encoding time increases only by 3%, 1% and 3% for different configurations and the decoding time increases by 1%. The increased complexity is negligible.

V. CONCLUSION

In this paper, we introduce the adaptive loop filter in HEVC and propose a simple but effect temporal prediction method to save the ALF parameters to be transmitted. Besides the ALF parameters estimated from current frame, the prior coded ALF parameters are employed as another filter set for current frame. The two filter sets are adaptively selected according to rate-distortion performance. A picture level flag is transmitted to indicate which filter set is adopted for each frame. Extensive experiments are conducted to verify the performance of the proposed method. Compared with HM3.0 anchor, the proposed ALF method achieves an average of 0.4%, 0.3% and 0.3% BD bitrate saving for luminance component in high efficiency random access, low delay B slice and low delay P slice configuration, respectively. At the same time, only 1% decoding time increasing and 1%-3% encoding time increasing are brought, which could be negligible. Therefore, the proposed method is a useful tool to improve the ALF performance for HEVC.

TABLE II. BD BITRATE SAVING (%) PROPOSED METHOD OVER ANCHOR WITH HE_RA CONFIGRATION

Sequence	*Y BD-rate*	*U BD-rate*	*V BD-rate*
BasketballDrill_832x416_50	-0.3	-0.2	-0.3
PartyScene_832x416_50	-0.4	-0.1	-0.1
BQMall_832x416_60	-0.1	-0.2	-0.1
RaceHorses_832x416_30	-0.4	-0.3	-0.3
BasketballPass_416x240_50	-0.2	-0.5	-0.6
BlowingBubbles_416x240_50	-0.2	-0.3	-0.2
BQSquare_416x240_60	-0.8	-0.7	-0.7
RaceHorses_416x240_30	-0.6	-0.5	-0.6
average	-0.4	-0.4	-0.4

TABLE III. BD BITRATE SAVING (%) PROPOSED METHOD OVER ANCHOR WITH HE_LD_B CONFIGRATION

Sequence	*Y BD-rate*	*U BD-rate*	*V BD-rate*
BasketballDrill_832x416_50	-0.3	-0.9	-1.0
PartyScene_832x416_50	-0.1	0.2	-0.2
BQMall_832x416_60	-0.2	-0.4	0
RaceHorses_832x416_30	-0.1	-0.2	-0.3
BasketballPass_416x240_50	-0.2	-0.9	-0.7
BlowingBubbles_416x240_50	-0.2	-0.7	-0.2
BQSquare_416x240_60	-0.7	-1.1	1.2
RaceHorses_416x240_30	-0.4	-0.6	-0.1
average	-0.3	-0.6	-0.2

TABLE IV. BD BITRATE SAVING (%) PROPOSED METHOD OVER ANCHOR WITH HE_LD_P CONFIGRATION

Sequence	*Y BD-rate*	*U BD-rate*	*V BD-rate*
BasketballDrill_832x416_50	-0.1	0.2	-0.5
PartyScene_832x416_50	-0.1	-0.3	-0.2
BQMall_832x416_60	-0.2	-0.3	-0.2
RaceHorses_832x416_30	-0.2	-0.2	-0.3
BasketballPass_416x240_50	-0.2	-0.4	-0.8
BlowingBubbles_416x240_50	-0.2	-0.1	-0.6
BQSquare_416x240_60	-0.5	-1.4	-1.3
RaceHorses_416x240_30	-0.5	-0.2	-0.8
average	-0.3	-0.3	-0.6

TABLE V. COMPLEXITY COMAPRISION WITH ACHOR

Sequence	*HE_RA*	*HE_LD_B*	*HE_LD_P*
Encoding time	103%	101%	103%
Decoding time	101%	101%	101%

ACKNOWLEDGEMENT

This work was supported in part by National Basic Research Program of China (973 Program, 2009CB320903, and 2009CB320904), National Natural Science Foundation of China (No. 61073083 and 60833013), Beijing Natural Science Foundation (No. 4112026) and Specialized Research Fund for the Doctoral Program of Higher Education (No. 20100001120027).

REFERENCES

[1] Y.-J. Chiu and L. Xu, "Adaptive (Wiener) Filter for Video Compression," ITU-T Q.6/SG16 Doc., VCEG-M35, Berlin, Jul. 2008.

[2] T. Chujoh, G. Yasuda, N. Wada, T. Watanabe and T. Yamakage, "Block-based Adaptive Loop Filter," ITU-T Q.6/SG16 Doc., VCEG-M35, Berlin, Jul. 2008.

[3] T. Chujoh, N. Wada and G. Yasuda, "Quadtree-based Adaptive Loop Filter," ITU-T SG16 Contribution, C181, Geneva, Jan. 2009.

[4] M. Karczewicz, P. Chen, R. Joshi, X. Wang, W.-J. Chien and R. Panchal "Video coding technology proposal by Qualcomm Inc.," ITU-T SG16 Contribution, JCTVC-A121, Dresden, Apr. 2009.

[5] I. S. Chong, M. Karczewocz, C.-Y. Chen, C.-M. Fu, C.-Y. Tsai, Y.-W. Huang, S. Lei, T. Yamakage, T. Chujoh and T. Watanabe "CE8 Subtest 2: Block based adaptive loop filter (ALF)," ITU-T SG16 Contribution, JCTVC-E323, Geneva, Mar. 2011.

[6] C.-Y. Chen, C.-M. Fu, Y.-W. Huang and S. Lei "CE8 Subtest 2: Adaptation between Pixel-based and Region-based Filter Selection," ITU-T SG16 Contribution, JCTVC-E046, Geneva, Mar. 2011.

[7] T. Yamakage, T. Watanabe, T. Chujoh, C.-Y. Chen, C.-M. Fu, C.-Y. Tsai, Y.-W. Huang S. Lei, M. Karczewicz and I. S. Chong "CE8.1: Block based Adaptive Loop Filter by MediaTek, Qualcomm and Toshiba," ITU-T SG16 Contribution, JCTVC-F321, Torino, Jul. 2011.

[8] Xinfeng Zhang, Ruiqin Xiong, Siwei Ma, Wen Gao "Adaptive Loop Filter Merge in Temporal Domain," ITU-T SG16 Contribution, JCTVC-E498, Torino, Jul. 2011.

[9] Test sequences are available on ftp://hevc@ftp.tnt.uni-hannover.de/testsequences/

[10] Frank Bossen "Common test conditions and software reference configurations," ITU-T SG16 Contribution, JCTVC-E700, Daegu, Jan. 2011.

[11] S. Pateux and J. Jung, "An Excel Add-in Computing Bj ϕ ntegaard Metric and its Evolution," ITU-T SG16 Q.6 Document, VECG-AE07, Marrakech, Jan. 2007