Mode Dependent Deblocking Filter for Video Coding

Qingbo Wu*, Hongliang Li†
School of Electronic Engineering, University of Electronic Science and Technology of China
Chengdu, China 611731
*Email: wqb1985@yahoo.cn
†Email: hlli@uestc.edu.cn

Abstract— **In this paper, a mode dependent deblocking filter is proposed for suppressing the effects of quantization noise. The proposed filter employs Wiener Filter as an in-loop filter which can minimize the mean square error between the original image and the decoded image. In addition, to adapt to different local features of the decoded image efficiently, we design our filter based on the intra mode combination of each pair of 4x4 blocks. Experiment results show that the proposed filter achieves superior coding gains relative to H.264/AVC high profile with negligible complexity increase.**

I. INTRODUCTION

During recent decades, most video coding strategies employ block-based transforms among their compression tools. Coarse quantization of transform coefficients inevitably introduces visual artifact of disturbing discontinuities at the block boundaries [1]. To remove the blocking artifacts, the deblocking loop filter (DLF) [1] is adopted in H.264/AVC [2] which is the state-of-the-art hybrid video coding standard. In DLF, a bank of pre-defined low-pass filters are applied to the block boundaries adaptively along horizontal and vertical directions. Although the DLF has achieved outstanding performance both on subjective and objective quality, two constraints still restrict its capability of restoring the degraded image. On the one hand, the achievement of DLF depends on the smoothing operation, which induces the details information loss in the decoded image. On the other hand, fixed filter coefficient is insufficient to adapt to various local features.

Recently, a number of improved loop filter methods were proposed. As a well-known optimal linear filter to cope with the image degraded by Guassian noise, Wiener filter has been widely applied in various image/video denoising applications. To restore the degraded image by compression, the adaptive loop filter (ALF) is designed by using Wiener filtering in [3]-[4]. However, as the filter coefficients designed for the whole frame, Wiener filter may intensify image distortion in some local region. So additional overhead will be needed to indicate whether to enable the filtering process for current pixel or block. In [5], the Quadtree-based adaptive loop filter (QALF) introduces a more efficient structure to signal the flag which determines current block is filtered by ALF or DLF. And the joint deblocking filter (QALF+DLF) scheme has been employed in the KTA software. Even though an outperforming performance has been achieved by QALF, the increased computational complexity is also striking.

Unlike previous methods to produce the coefficients for all pixels in a frame, we propose a mode dependent deblocking filter (MDDF) scheme which considers the distribution feature of local texture information on the block boundary. To accurately capture local characteristics of decoded image, in our MDDF scheme, a set of coefficients are derived corresponding to different intra prediction mode combinations of the blocks on both sides of the boundary. And the coefficients are sent as side information to the decoder.

In the following, the algorithm of our proposed MDDF is described in Section II. Experiment results and conclusions are provided in Section III and IV, respectively.

II. MODE DEPENDENT DEBLOCKING FILTER

A. Distribution feature of texture directions on both sides of the block boundary

There are two features hidden in intra prediction mode often providing helpful information for improving intra coding. One is the texture information in current block. The other one is the appearance probability of different texture directions.

As shown in Fig. 1, the general direction of the texture in each 4x4 or 16x16 block is consistent with the direction specified by intra mode.

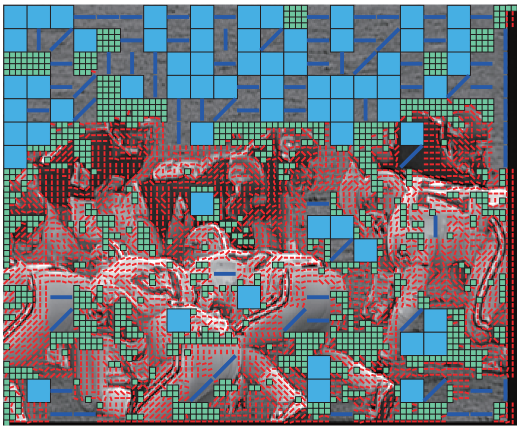


Fig. 1. Intra mode direction in sequence *FOOTBALL* as QP=37. The red lines denote directional intra 4x4 mode, the blue lines represent directional intra 16x16 mode, the green and light blue squares represent the DC mode for 4x4 and 16x16 blocks, respectively.

978-1-4673-0219-7/12/$31.00 ©2012 IEEE

TABLE I

CORRESPONDENCE BETWEEN TEXTURE CLASSES AND INTRA MODE

Texture direction class	intra 4x4/16x16 mode
0	0, 5, 7
1	1, 6, 8
2	2
3	3
4	4

In addition, in H.264/AVC standard, the minimum mode number between the upper block and left block is referred as $most\_probable\_mode$, which means the mode with smaller number would be chosen more probably. Considering the fact that blocking artifact concentrates on the block boundary, we investigate the distribution of different texture direction combinations where the correspondence between texture direction and intra mode is shown in Table I. Since we design the filter along 4x4 block boundary, for the macroblock with intra_16x16 type, the intra mode of each 4x4 block are set to the same mode with intra 16x16 prediction mode. And the combination number of different classes of texture can be defined in equation (1) as follows:

$$C_{num} = 5 \times T_1 + T_2 + 1 \tag{1}$$

where C_{num} represents the combination number, T_1 and T_2 represent the texture direction classes on both sides of the block boundary. For a vertical edge, T_1 and T_2 represents the texture class on the left and right side respectively. And for the horizontal boundary, they will represent upside and downside texture class correspondingly.

The probability distribution of different combination number has been shown in Fig. 2. From these statistics we can find that smaller texture classes prefer to group in pairs, and 3 combination numbers of 1, 7 and 13 (corresponding to texture pairs $(0, 0)$, $(1, 1)$ and $(2, 2)$ respectively) take majority of proportion for all these combinations which means only 4 kinds of texture combination can reflect the main local features in a frame.

B. Filter design

Since Wiener Filter is an optimal linear restoration filter to cope with the degraded pictures. We will describe the method of calculating the filter coefficients firstly. Suppose that $X = (x_1, x_2, \cdots, x_M)$ is the support vector of the Wiener filter in decoded image, the output of linear Wiener filter is given by

$$y = XW \tag{2}$$

where W is a vector of M optimal filter coefficients.

To optimize the filter coefficients, the classical LMSE (least mean square error) framework can be exploited as follows

$$\tilde{W} = \arg\min_{\tilde{W}} E[(y_o - \sum_{i=1}^{M} w_i x_i)^2] \tag{3}$$

where y_o is the intensity value in original frame.

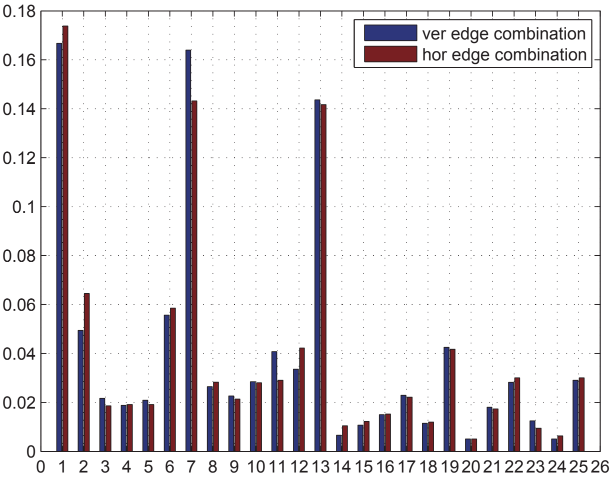
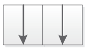
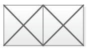


Fig. 2. Probability distribution of different combination number in sequence $FOOTBALL$ as QP=37

To adapt to local texture characteristics in decoded image, we design a set of filter coefficients corresponding to 4 classes texture direction combination. The combination for vertical edge has been listed in Table II where the gray square represents the homogeneous block and the cross line represents all of other texture directions other than above cases. The same manner is also employed for horizontal boundary.

TABLE II

SELECTED TEXTURE DIRECTION COMBINATION CLASS

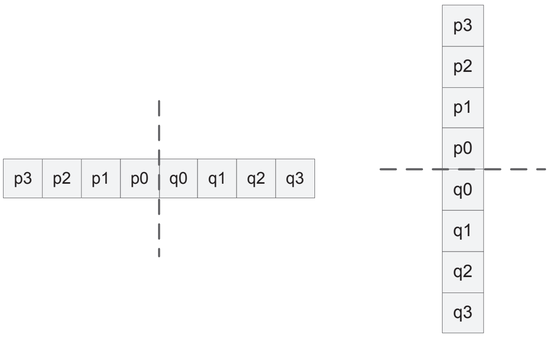
Combination Class\Texture direction pair	Texture direction diagram
1 \ (0,0)	
2 \ (1,1)	
3 \ (2,2)	
4 \ others	

Similar with the filtering process of DLF, we also design our filter coefficients along vertical and horizontal boundaries for each combination class where the 8 pixels on both sides of the boundary construct the support vector X as shown in Fig. 3.

And the train samples along each direction can be denoted as $A_k = (X_1(k), X_2(k), \cdots, X_N(k))^T$ where k represents the texture direction combination class, N represents the number of support vectors belong to combination class k. Accordingly, the LMSE equation (3) can be rewritten as

$$\tilde{W}_l(k) = \arg\min_{\tilde{W}_l(k)} E[\sum_{j=1}^{N}(y_j(l) - \sum_{i=1}^{M} w_i(k,l) x_{i,j}(k))^2] \tag{4}$$

where l denotes the lth column of a line train samples like $p2 \sim q2$ as shown in Fig. 3, j represents the jth row location in A_k. So $\tilde{W}_l(k)$ represents the optimal filter coefficients for



(a) (b)

Fig. 3. Pixels adjacent to vertical and horizontal boundaries

combination class k at column l, and $y_j(l)$ represents the original intensity value of train sample which is at the jth row, lth column of A_k.

At last, the optimal filter coefficient vector can be derived as

$$\tilde{W}_l(k) = (A_k^T A_k)^{-1} A_k^T Y_l \tag{5}$$

where $Y_l = (y_1(l), y_2(l), \cdots, y_N(l))^T$.

However, since the train samples belong to the same combination class are highly correlated, we employ Tikhonov regularization to cope with ill-posed problem by revising the solution in equation (5) as

$$\hat{W}_l(k) = (\hat{A}_k^T \hat{A}_k)^{-1} \hat{A}_k^T \hat{Y}_l \tag{6}$$

where $\hat{A}_k = (A_k, \lambda I)^T$, I represents the identity matrix and λ denotes the scale parameter. $\hat{Y}_l = (Y_l, \mathbf{0})$ where $\mathbf{0}$ represents the zero vector. The effect of regularization may be varied via the scale parameter λ. And in our simulation, this parameter λ is set to 8 in terms of the empirical results.

After obtaining the filter set, we will filter the decoded image along vertical and horizontal direction, respectively.

III. EXPERIMENT RESULTS

To verify the performance of our proposed deblocking filter, the proposed MDDF scheme is implemented on VCEG KTA2.4r1 reference software [6]. Our test configurations are Intra only and IPPP using $QP = \{22, 27, 32, 37\}$ on High Profile. The test sequences cover CIF and HD (720p) resolutions, where the CIF sequences code 150 frames and 720p sequences code 10 frames. The H.264/AVC High Profile with DLF ("Anchor") is utilized as the benchmark. Our proposed MDDF and the QALF are all involved in the comparison. Based on the assumption that the whole sequence exhibits stationary Markov characteristics, only the first frame is used to design the optimal filter coefficients set for MDDF and each coefficient is encoded with 10 bits. As a metric to evaluate coding efficiency, BD-PSNR [7] is used. In addition, to further compare the computational complexity between the proposed scheme and the anchor, the percentage of difference of coding time ($\Delta T\%$) is employed as follows:

$$\Delta T = \frac{T_{pro} - T_{anc}}{T_{anc}} \times 100 \tag{7}$$

where T_{pro} and T_{anc} denote the coding time of the proposed scheme and the anchor respectively.

A. Performance Comparison

TABLE III

CODING GAIN COMPARISON OF DIFFERENT CODING SCHEMES IN TERMS OF Δ BITRATE (%)

Sequence		BDBR	
		MDDF	QALF+DLF
CIF	mobile	-4.07	-5.38
	FOOTBALL	-8.14	-8.35
	mother-daughter	-7.13	-7.3
	news	-3.14	-6.89
Average	CIF	**-5.62**	**-6.98**
720p	park_joy	-2.54	-2.93
	ducks_take_off	-6.04	-6.42
	old_town_cross	-5.07	-5.19
	shield	-5.6	-6.61
Average	720p	**-4.81**	**-5.29**

Table III presents the percentage of BD bitrate reduction at the same PSNR of luminance of different coding schemes compared with the anchor scheme. From Table III we can find that both our proposed MDDF scheme and the QALF+DLF scheme can achieve superior coding gains relative to H.264/AVC. Especially for the sequence $FOOTBALL$, up to -8.14% and -8.35% bitrate reductions have been achieved for MDDF and QALF+DLF, respectively.

In addition, although our proposed MDDF can't attain the same performance as QALF+DLF for some sequences, such as $news$, we also achieve comparable coding gains on average. In Table III, the average differences of BDBR performance between MDDF and QALF+DLF are both close to 1% for CIF and 720p sequences. From the objective quality comparison shown in Fig. 4, we can see that the RD-curves of MDDF and QALF+DLF are very close to each other.

B. Complexity Comparison

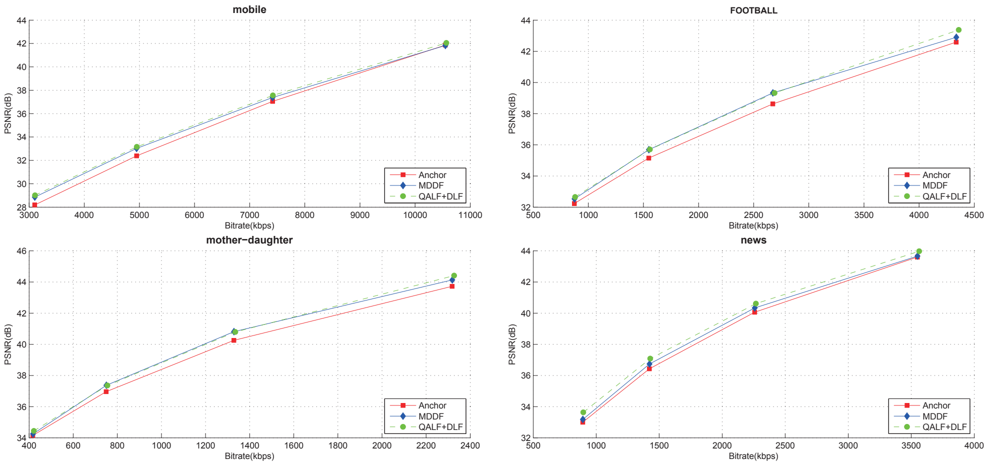
In our experiment, the the system platform is the Intel Pentium Dual-Core Processor of speed 2.8 GHz and 2 GB RAM. The average encoding execution time of different coding schemes for various image resolutions has been shown in Table IV.

TABLE IV

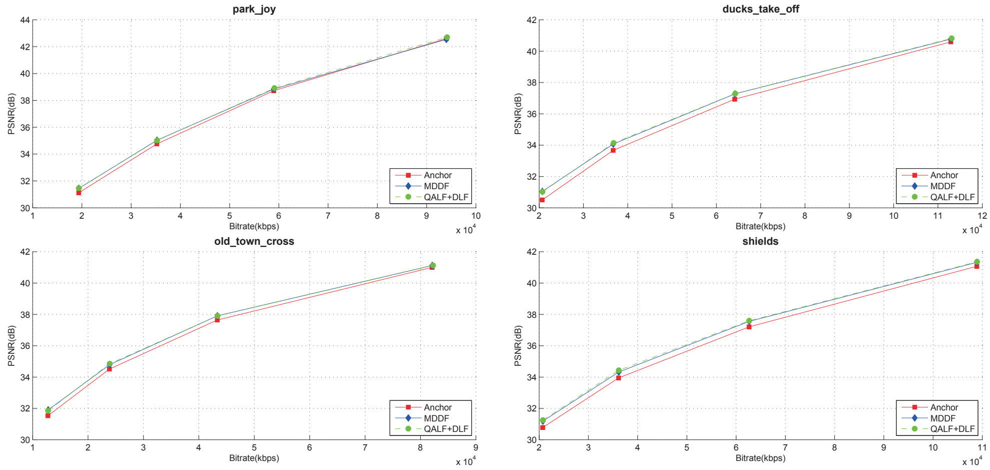
AVERAGE ENCODING TIME COMPARISON IN TERMS OF ΔT (%)

Sequence		∆T	
		MDDF	QALF+DLF
CIF	mobile	+2.08	+194.08
	FOOTBALL	+5.47	+307.33
	mother-daughter	+4.51	+330.49
	news	+6.4	+294.18
Average	CIF	**+4.62**	**+281.52**
720p	park_joy	+18.51	+189.46
	ducks_take_off	+16.63	+207.68
	old_town_cross	+15.52	+225.77
	shield	+10.67	+236.53
Average	720p	**+15.33**	**+214.86**

Since only the first frame is utilized to calculate optimal filter coefficients set, it is clear that the complexity of our



(a) CIF sequences



(b) 720p sequences

Fig. 4. RD-curve of test sequences with different resolution

proposed MDDF scheme is similar with the anchor scheme and significantly lower than QALF+DLF scheme from Table IV. In our proposed MDDF, the added execution time for CIF and 720p sequences are both less than 20% on average. And for CIF sequences, it is even less than 10%. However, for QALF+DLF, the excess encoding time are all more than 200% for various resolutions test sequences on average. So, it can be seen that our proposed MDDF scheme outperforms the QALF+DLF scheme on efficiency.

IV. Conclusions and Future Work

In this paper, a high performance mode dependent deblocking filter is proposed to improve intra coding efficiency. The proposed MDDF scheme designs the optimal Wiener filter coefficients set based on different intra mode combinations on both sides of the block boundary. The simulation results demonstrate that the proposed scheme provides superior results relative to H.264/AVC high profile with negligible complexity increase. In our future work, a more robust strategy to detect scene change and adaptively update the coefficients set will be studied.

Acknowledgment

This work was partially supported by NSFC (No. 60972109 and 61101091), the Program for New Century Excellent Talents in University (NCET-08-0090), and the Ph.D. Programs Foundation of Ministry of Education of China (No. 20110185110002).

References

[1] P. List, A. Joch, J. Lainema, G. Bjontegaard, and M. Karczewicz, “Adaptive Deblocking Filter,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614-619, Jul. 2003.
[2] *ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4) AVC:* “Advanced Video Coding for Generic Audiovisual Services,” Mar. 2005.
[3] T. Chujoh, A. Tanizawa, and T. Yamakage, “Adaptive loop filter for improving coding efficiency,” *ITU-T SG16 Contribution*, C402, Geneva, Switzerland, Apr. 2008.
[4] Y. Chiu and L. Xu, “Adaptive (Wiener) filter for video compression,” *ITU-T SG16 Contribution*, C437, Geneva, Switzerland, Apr. 2008.
[5] T. Chujoh, N. Wada, and G. Yasuda, “Quadtree-based adaptive loop filter,” *ITU-T SG16 Contribution*, C181, Geneva, Switzerland, Jan. 2009.
[6] *ITU-T VCEG KTA Reference Software* [Online], Available: http://iphome.hhi.de/suehring/tml/download/KTA
[7] G. Bjontegaard, “Calculation of average PSNR differences between RD-Curves,” ITU-T VCEG, VCEG-M33, Apr. 2001.