Adaptive Perceptual Preprocessing for Video Coding

Guoqing Xiang, Huizhu Jia*, Jie Liu, Binbin Cai, Yuan Li, Xiaodong Xie

EECS of Peking University

Beijing, 100871

P.R. China

Email: {gqxiang, hzjia, liuzimin, bbcai, yuanli, xdxie}@jdl.ac.cn

Abstract —The quantization of block DCT coefficients is too coarse, and will result in much non-uniform artifacts, therefore, blocking and ring artifacts are usually visible in reconstructed video frames especially when the bitrate is not sufficient. In this paper, we present an adaptive perceptual preprocessing (APP) method to reduce the probability of these artifacts generated in video encoding. The APP algorithm employs a novel filter based on just noticeable distortion filter (JNDF) and adaptive bilateral filter (ABF), and they can be adaptively chosen by block characteristics and quantization parameters. Experimental results demonstrate our proposed algorithm can significantly improve the subjective quality of reconstructed image.

Keywords—Blocking arctifact; just noticeable distortion filter; bilateral filter; quantization; subjective quality

I. INTRODUCTION

Traditional hybrid video coding aims to remove spatial and temporal statistical redundancies for signal compression based on block discrete cosine transform and quantization. All these video compression tools will suffer noticeable discontinuities between neighboring blocks, such as blocking and ring artifacts. Since DCT coefficients are quantized to zero at low bitrate, the subject quality degradation of homogeneous area is inevitable. To alleviate these problems, some post-processing algorithms have been proposed [1] in time domain or frequency domain [2] and integrated in mainstream coding tools, such as deblock and SAO [3]. In spite of these tools, these artifacts still can appear when the bitrate is not sufficient for compression.

Insufficient bitrate for video compression usually occurs when information channel bandwidth is low for video source or the image scene becomes complex for current bitrate. For the first aspect, improving the channel bandwidth maybe the main method to solve the problem, which will need more cost for application. However, at most cases, the information channel bandwidth is limited and cannot be always adaptively adjusted. For the second situation, if we can reduce the video source complexity, it will alleviate the deficiency problems at the same bitrate, which inspires video preprocessing research

With useful preprocessing algorithms, we can remove some relatively insignificant high frequency components to reduce image complexity before compression. Even if there are numerous methods, such as VB3D filter [4], bilateral filter [5]

---

etc., they cannot be utilized to solve these reconstructed image artifacts directly for the following reasons. Firstly, most video preprocessing algorithms aim to reduce noise and they are modeled with noisy images, which are different for alleviating reconstructed artifacts with preprocessing method. Secondly, these methods are entirely independent from video coding tools, and the spatial or temporal filter they proposed cannot be adaptive with different bitrates, which will be over or short of filtering. Thirdly, even though there are few methods combine these filters with video quantization parameters [6], the method directly filter at frame level without considering local block characteristics, which will over smooth in some flat or texture sparse areas. Finally, most current filter methods will bring more or less blur into image and could be visible when filtering too strong. Even though the blocking and ring artifacts are disappeared, the reconstructed picture become vaguer. What's more, to alleviate the bitrate insufficiency problem but not video denoise, the noticeable blur artifacts may be not necessary when the bitrate is not very low.

In order to solve these problems, we have proposed a novel adaptive perceptual preprocessing method (APP) exploiting both visual and video compression characteristics. As concerned above, when bitrate is not very low, for some complex image, we can choose filter unnoticeable high frequency components visually which will not produce noticeable blur. But when the bitrate is low enough, we can only improve some blurriness with low pass filtering controlled adaptively by quantization to replace the rebarbative blocking artifacts. The APP algorithm contains mainly two parts, the DCT domain based just noticeable distortion (i.e. the distortion just can be noticed by viewers) filter (JNDF) and spatial domain adaptively bilateral filter (ABF), respectively. We utilize the JNDF to remove unnoticeable information for each frame to alleviate coding pressure and ABF is responsible for stronger low pass filtering when necessary together with JNDF. Many experiments have been done to verify our method, and results demonstrate that the APP algorithm can significantly improve the subjective quality of reconstructed image.

The reminder of this paper is organized as follows. In Section Ⅱ, the main structure and details of the proposed APP model are introduced. Then the model is verified with experiment and results are shown and discussed in Section Ⅲ. The Section Ⅳ draws the conclusions of our work finally.

II. ADAPTIVE PERCEPTUAL PREPROCESSING METHOD

The proposed APP model includes JNDF and ABF two parts, and work flow is shown in Fig. 1. There are two paths in

the diagram, in the Path1, only JNDF works and the ABF will join in together in Path2. More details will be described later.
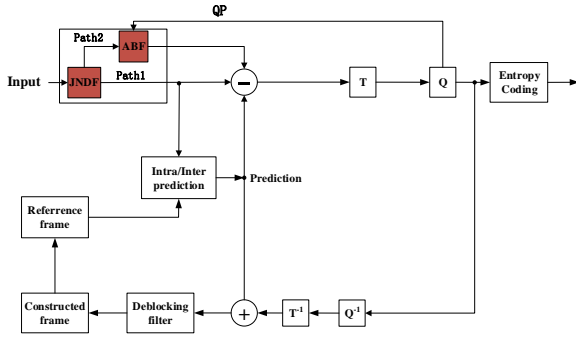

Fig. 1. The proposed Adaptive Perceptual Preprocessing (APP) Video Coding diagram

A. Just Noticeable Distortion Filter (JNDF)

The existing JND models can be classified into pixel domain and transform domain, respectively. In the pixel domain, most JND models use luminance adaptation and texture masking to compute pixel-level JND [7]. While among the popular DCT domain JND models, researchers focus on luminance adaptation, the spatial CSF (contrast sensitivity function) and temporal CSF effects [8] to get useful spatial-temporal JND models. In respect of noticeable distortion of human eyes, the pixel domain JND means the just noticeable distortion of each pixel directly and the DCT domain JND can reflect the unnoticeable energy which can be removed in the frequency domain without introducing visible difference. For DCT domain JND of each block, it can be expressed as,

$$JND_{DCT}(n,u,v) = T(n,u,v) \cdot \alpha_{lum}(n)$$
$$\cdot \alpha_{intra}(n,u,v) \cdot \alpha_{inter}(n) \quad (1)$$

where u and v are the index of $N_1 \times N_1$ each block, n is the block index, $T(n,u,v)$ represents base threshold for a DCT subband, $\alpha_{lum}(n)$ denotes the luminance adaptation factor for the block, $\alpha_{intra}(n,u,v)$ and α_{inter} account for the effects of intra-band masking and inter-band masking. More details can be find in [8].

The JND model of DCT domain can avoid yielding filtering values larger than the human visual thresholds, and we can introduce lots of unnoticeable noise energy without jeopardizing picture quality by injecting or removing the JND energy and here we define it as,

$$C_{JNDF}(n,u,v)$$
$$= \begin{cases} \sqrt{C^2(n,u,v) - p(u,v) \cdot JND_{DCT}^2(n,u,v)} \cdot sgn(C(n,u,v)), else \\ 0, \ if \ |C(n,u,v)| < JND_{DCT}(n,u,v) \end{cases} \quad (2)$$

where $C(n,u,v)$ denotes the (u,v) coefficient of the nth original DCT block, the $p(u,v)$ represents different weights for different positions of each DCT block and $C_{JNDF}(n,u,v)$ is the filter result of every coefficient. The sgn(x) is sign function returning 1 or -1 when x is positive or negative respectively.

On one hand, we can see that if the energy of (u,v) in the original block DCT is smaller than the JND value, it means this energy cannot be perceived by human eyes, then it can be removed directly. On the other hand, when the original DCT coefficient is larger than JND threshold, we can still filter part of them to reduce the block complexity. Here we do not filter all the JND energy directly because different areas of DCT block should have different influences on human eyes. For example, the DC coefficient should keep the same for better inverse DCT, even though it is large than its responding JND threshold always.

Therefore, in terms of the filter weight $p(u,v)$, it can control the filter strength, because over filtering in DCT domain will easily result in blocking effect in pixel domain when inverse DCT transform has performed. Here the control weight of each position in DCT block is only related with frequency characteristics defined as Fig. 2. With our large experiments, for DC area, low frequency area, medium frequency area and high frequency area, when the weight is 0, 0.25, 0.75 and 1.0 respectively we can achieve relatively good perceptual results when inverse DCT done.
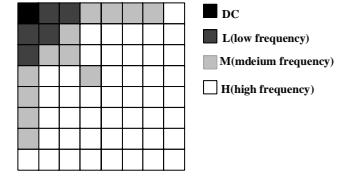

Fig. 2. Frequency characteristics of DCT block

Therefore, with the JNDF in DCT domain, the image energy will become lower, especially the high frequency energy. Even though we has removed much information from the block, the removed information is unnoticeable and it will not produce any visual artifacts when the block has been inverse transformed into pixel domain as showed in Fig.3. Thus, the block complexity has been reduced, and it becomes looser for the bitrate requirement and can get better compression efficiency.
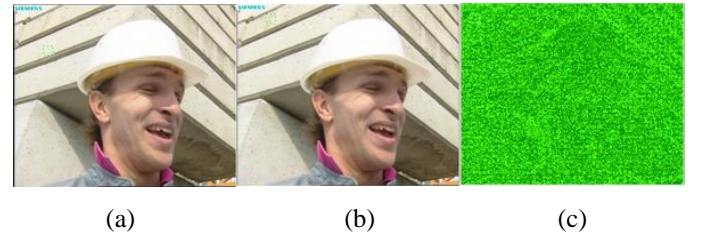

Fig. 3. JNDF result comparison (a) original image (b) JNDF filter result (c) Difference map

B. Adaptive Bilateral Filter (ABF)

When the JND filter cannot fit the bitrate requirement, we have to use some strong filters to smooth the image. Among numerous spatial filters, the bilateral filter possesses well done performance with smoothing images meanwhile preserving edges by means of a nonlinear combination of nearby window values. It utilizes a weighted average of local samples, in which higher weights are given to samples that are

closer in both space distance and intensity to center sample [5]. Supposed the filter center is (*x, y*), the filter result can be showed as,

$$\hat{f}(x,y) = \frac{1}{\omega} \sum_{i,j \in \Omega} \omega_d(i,j) \cdot \omega_r(i,j) \cdot I(i,j) \quad (3)$$

where $\hat{f}(x,y)$ is filtered image, $\omega_d(i,j)$ is spatial domain weight, $\omega_r(i,j)$ is the range domain weight, $I(i,j)$ is the input image, and Ω is neighborhood range of current center position, ω is the weight normalize factor as,

$$\omega = \sum_{i,j \in \Omega} \omega_d(i,j) \cdot \omega_r(i,j) \quad (4)$$

The spatial and range domain weight are both Gaussian functions with filter strength σ_d and σ_r respectively as (5) and (6). The spatial weight depicts a low-pass nature of the bilateral filter and the range weight suppresses the contributions of pixels far from edge, which performs as a truncation Gaussian bell weight. Thus, the edge is not diffused and high-frequency noise and insignificant components can be suppressed.

$$\omega_d = \exp[-\frac{|i-x|^2 + |j-y|^2}{2\sigma_d^2}] \quad (5)$$

$$\omega_r = \exp[-\frac{|I(i,j) - I(x,y)|^2}{2\sigma_r^2}] \quad (6)$$

Although the edge-preserved feature of bilateral filter, it is still a smooth filter essentially, the edge- preserved performance is limited, large σ_d and σ_r will produce oversmooth into image and edges maybe blur as well. As a consequence, the key factor of using bilateral filter is to explore an effect method to adaptively adjust the filter strength. From the perspective of bitrate requirement, when the bitrate is enough for current block, the filter strength should be small, or else we need improve the filter strength, which means visible blur is inevitable in some large filter strength circumstance.

There are two filter strengths of bilateral filter which should be determined. The spatial strength σ_d is related with the radius of filter window. Based on the Gaussian surface property, more than 95% components can be included in the area [$-2\sigma_d$, $2\sigma_d$], thus we can set $2\sigma_d + 1 = R$, namely

$$\sigma_d = \frac{R-1}{2} \quad (7)$$

where R denotes the filter window width and height.

Considering the range domain strength σ_r, it reflects the degree of closeness between center and neighborhood pixel. Some researchers [5, 6] have done much effort to adaptively modulate the range domain filter strength, however, none of them consider the human visual system characteristics for filtering, which results in large likelihood of visible distortion. Here we combine the perceptual considering and bitrate requirement together to define the adaptive filter strength σ_r as,

$$\sigma_r(t) = k_1 \cdot \sqrt{JND_{noise}} + k_2 \cdot (QP(t-1) - Th) \quad (8)$$

where JND_{noise} is the perceptual noise in the image, the $QP(t-1)$ denotes previous frame average quantization, which reflects the bitrate requirement of each frame under

bitrate control circumstance, k_1 and k_2 are experimental parameters and Th is a quantization threshold for adjustment for all macroblocks on a whole frame.

When the previous quantization level is higher than the threshold Th, it means the bitrate is very low, more visual blocking effects will emerge in reconstructed image, so we use both JNDF and ABF in Path2 to filter more energy. Thus the path choose method can be as showed below.

$$APP_{Path} = \begin{cases} Path1, & otherwise \\ Path2, & QP(t-1) > Th \end{cases} \quad (9)$$

In order to get the term JND_{noise} above, firstly, we choose pixel domain JND model [7] to calculate each value JND_{pixel} of every $N_2 \times N_2$ macroblock as follow,

$$JND_{pixel}(x,y) = T_l(x,y) + T_t(x,y)$$
$$- C_{l,t} \cdot \min\{T_l(x,y), T_t(x,y)\} \quad (10)$$

where $T_l(x,y)$ and $T_t(x,y)$ are the visibility thresholds for background luminance adaptation and texture masking, respectively; $C_{l,t}$ refers to the overlapping effect in masking, the determination details can be found in [7]. Then we calculated the variance of $N_2 \times N_2$ JND values and set it as perceptual noise JND_{noise} as equation (11). The more complex the block is, the larger JND_{pixel} will be, which means more noise or artifacts can be hidden in the block without noticeable difference.

$$JND_{noise} = Cal\_variance(JND_{pixel})_{N_2 \times N_2} \quad (11)$$

It is noted that the JND_{noise} is also limited to block property. Therefore, it can be adjusted adaptively with block itself JND variance and work well when the bitrate is enough. If the bitrate is very low, the filter strength will be dominated by quantization parameters and more blur will be inescapable.

Therefore, we can adaptively reduce image complexity according to image visual properties and video coding bitrate requirement with the APP method.

III. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed APP video coding scheme, the integration procedure is implemented on x264-encoder [9] platform. To produce encodings of both original and filtered content with closet possible amounts of distortion. The bitrate has been approximated to integer for each resolution listed in TABLE I. All sequences are tested with IPPP structure. The parameter N_1 and N_2 is 8 and 16, respectively, which means we use 8x8 DCT block transform in JNDF and 16x16 macroblock for ABF.

We have compared the quantization level with original ones in Fig. 4, and all other decrements of QPs are listed in TABLE I. According to these, we can take up several discussions as follows. First, with our APP algorithm, average QP can be reduced down for each sequence due to the lower complexity compared with original one, which means less quantization distortion will appear under the same bitrate. Second, we can also find that average QP of each sequence has decreased not so much while QPs varies great in different frames. The phenomenon implies that our APP method will not force too much smooth into original image on the whole, However, when the frame become complex or bitrate

insufficient, the APP will reduce more information resulting smaller QP necessary. So the more bitrate will need less QP decrement on the whole while vice versa, the final QP decrement depends both on bitrate and sequence complexity. At last, we can see that the APP algorithm has improved the PSNR of all sequences, which can indicate blocking artifacts have disappeared to some extent. Finally, we show our subjective performance in Fig. 5 and Fig. 6, which can demonstrate that the reconstructed video has much less blocking and ring effects with most details kept on the whole, becoming more easy on the eyes and "good-looking" for viewers.
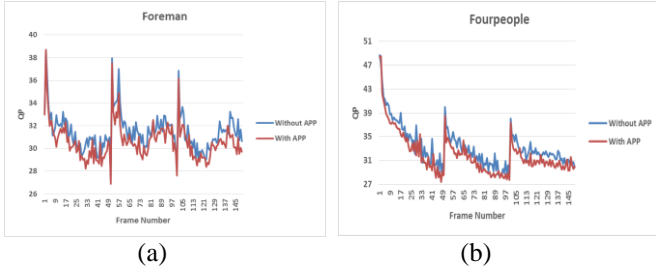

(a)   (b)

Fig. 4. Quantization level comparison (a) Foreman at 300kbps (b) Fourpeople at 1250 kbps

(a)   (b)

Fig. 5. Subjective quality comparison of 24th frame of Foreman at 300kbps (a) Without APP (b) With APP
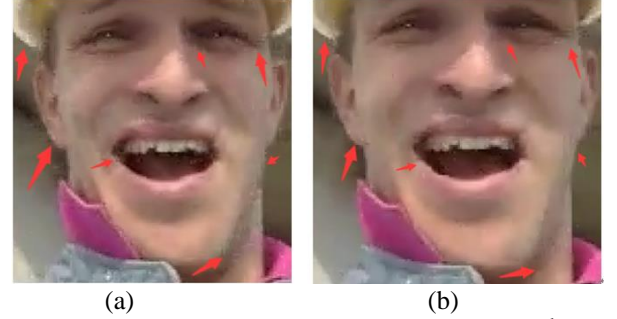

(a)   (b)

Fig. 6. Segments subjective quality comparison of 24th frame of Foreman at 300kbps (a) Without APP (b) With APP

TABLE I
DIFFERENT PERFORMANCE FOR SEQUENCES IN RC CONTROL

Sequence	Bitrate (kbp/s)	Average QP		Average PSNR	
		Without APP	With APP	Without APP	With APP
Foreman (352x288)	1100	23.45	23.21	38.10	38.44
	700	26.05	25.27	36.47	36.81
	400	29.58	28.58	34.58	34.89
	300	31.48	30.60	33.59	33.80
Paris (352x288)	1100	24.67	24.50	36.55	36.83
	700	28.57	28.07	33.51	33.73
	400	32.81	32.12	30.42	30.50
	300	34.86	34.15	29.12	29.25
BQMall (832x480)	17000	21.93	20.89	39.22	40.45
	11000	24.12	23.05	37.59	38.52
	6000	27.36	27.05	35.18	35.75
	3000	31.67	31.22	32.17	32.48
PartySence (832x480)	17000	26.85	26.55	34.11	34.57
	11000	29.55	28.49	31.51	31.98
	6000	33.67	32.51	28.50	28.89
	3000	36.87	36.56	26.10	26.43
Fourpeople (1280x720)	10000	21.18	20.84	40.41	40.95
	5000	24.16	23.30	38.46	39.09
	2500	27.78	26.60	36.41	36.66
	1250	33.23	31.94	33.51	33.60
Raven (1280x720)	10000	23.04	22.83	40.57	40.87
	5000	26.67	26.11	38.58	38.68
	2500	30.49	29.57	36.37	36.41
	1250	34.88	33.81	34.06	34.19

IV. CONCLUSION

In this paper, we have proposed an adaptive perceptual preprocessing (APP) algorithm. The APP preprocessing method contains two parts, namely JNDF and ABF, and each part will work in different conditions to reduce video complexity for easier compression under bitrate control. The APP is fully compatible with current mainstream video coding standard and also can be applied for HEVC coding framework Experiments have demonstrated obliviously subjective quality improvement on reconstructed video. In future, we plan to explore more accurate APP model and integrate its applications into hybrid video coding to further enhance the coding subjective performance.

REFERENCES

[1] Yang X K, Yao S S, Lim K P, et al. An adaptive edge-preserving artifacts removal filter for video post-processing[C]//Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on. IEEE, 2005: 4939-4942.

[2] Chen T, Wu H R, Qiu B. Adaptive postfiltering of transform coefficients for the reduction of blocking artifacts[J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2001, 11(5): 594-602.

[3] Sullivan G J, Ohm J R, Han W J, et al. Overview of the high efficiency video coding (HEVC) standard[J]. Circuits and Systems for Video Technology, IEEE Transactions on, 2012, 22(12): 1649-1668.

[4] Dabov K, Foi A, Egiazarian K. Video denoising by sparse 3D transform-domain collaborative filtering[C]//Proc. 15th European Signal Processing Conference. 2007, 1(2): 7..

[5] Pham T Q, Van Vliet L J. Separable bilateral filtering for fast video preprocessing[C]//Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on. IEEE, 2005: 4 pp.

[6] Li M, Xu Z. An adaptive preprocessing algorithm for low bitrate video coding[J]. Journal of Zhejiang University SCIENCE A, 2006, 7(12): 2057-2062..

[7] C.H. Chou, Y.C. Li, "A perceptually tuned subband image coder based on the measure of Just-Noticeable-Distortion Profile," IEEE Transaction on Circuits and Systems for Video Technology, vol. 5, no. 6, pp. 467-476 (1995)

[8] Yuting, J., L. Weisi and A.A. Kassim, Estimating Just-Noticeable Distortion for Video. Circuits and Systems for Video Technology, IEEE Transactions, 16(7): p. 820-829 (2006)

[9] "x264 encoder," http://www.videolan.org/developers/x264.htm