

A PERCEPTUAL BASED DEAD-ZONE ADJUSTMENT ALGORITHM FOR H.264

Wei Jiang, Jun Sun, Songyu Yu, Jia Wang

Institute of Image Communication & Information Processing Research, Shanghai Jiao Tong Univ.
Shanghai 200030, China

Email: sjtjiangwei@sjtu.edu.cn; sunjun@cdtv.org.cn; syyu@cdtv.org.cn; jiawang@cdtv.org.cn.

ABSTRACT

A new adaptive dead-zone adjustment algorithm has been developed based on a human visual model for H.264 video coding. In this paper, the proposed scheme utilizes the Rate-Distortion (RD) optimization technology to adaptively adjust dead-zone size. Also incorporation of the human visual system model into RD cost computation improves performance. Compared with the H.264 reference coder, the proposed algorithm can efficiently improve the subjective viewing quality and reduce the bit rate 10% on average.

1. INTRODUCTION

H.264 is a new international video coding standard and is jointly developed by the ITU-T Video Coding Experts Group and the ISO/IEC Motion Picture Expert Group. Compared with other video standards, H.264 improves prediction methods and coding efficiency. Its additional feature is multiple-free transform and scalar quantization that avoids division and floating point arithmetic. This algorithm can reduce complexity but it will not decrease Peak Signal to Noise Ratio (PSNR). However, its fixed dead zone size does not help to reconstruct film grain which is very important for the subjective picture quality, especially for the movie industry. Moreover since human observer is the end user of most image information and our human visual system (HVS) does not perceive quality in the Mean Squared Error (MSE) sense, the classical coding theory of MSE is not sufficient to indicate the picture quality.

Recently Thomas Wedi and Steffen Wittmann[1] proposed an adaptive dead zone size quantization algorithm which allows to control the size of the quantization interval around zero by an additional dead-zone parameter. However, the extra expense introduced by transmission of dead-zone parameter and increase of bit rate caused by smaller dead-zone size are not considered.

In this paper, we present a Perceptual Dead-Zone

Rate-Distortion Optimization (PDRDO) algorithm using JM8.0[2] as a platform. Instead of the fixed dead-zone in fixed quantization step, the proposed scheme adaptively adjusts dead-zone size by dropping some coefficients to zero or adding some coefficients to one according to perceptual weighted rate-distortion optimization. This algorithm can greatly improve the visual quality with the similar bit rate.

The remaining part of this paper is organized as follows. Section 2 briefly introduces the DRDO algorithm. Then section 3 provides the PDRDO algorithm in detail. Furthermore, performance study is carried out by experimental results in section 4. Finally section 5 provides some conclusions.

2. DEAD-ZONE RATE-DISTORTION OPTIMIZATION ALGORITHM (DRDO)

In the reference software JM8.0 from JVT, the scalar quantization operates on 4×4 blocks of transform coefficients. The basic forward quantizer operation can be implemented by the following equation

$$Z = \text{ROUND}((Y + f)/Q_{\text{step}}) \quad (1)$$

where Y is the 4×4 prediction error transform block, Z is the quantization block of Y , Q_{step} is the quantization step size ranging from 0.625 to 224 in H.264 and f is the rounding control parameter. Then the defined inverse quantizer operation is given by

$$Y = Z \times Q_{step} \quad (2)$$

From Eqn.(1) it can be seen that the dead zone of the quantizer is $[0, Q_{step}f]$. It is clear that all the coefficients falling into the dead zone will be quantized to zero, which makes the reconstructed pixels be equal to the predicted values. Therefore distortion is generated between the original pixel and its reproduction. DRDO algorithm adaptively adjusts the size of the dead zone for every coefficient. Compared with the fixed dead zone size, it can not only improve the picture quality but also effectively avoid the great increase of encoded bits.

In DRDO algorithm an adaptive zone is defined to judge whether the dead zone size will be adaptively changed for the coefficient. That is to say, the coefficients falling into this zone will be re-quantized to zero or one according to the rate-distortion cost. Therefore the optimization quantization block is obtained by minimizing rate-distortion cost $J(Y, Z)$.

$$J(Y, Z) = D(Y, Z) + \lambda \times R(Y, Z) \quad (3)$$

where λ is the Lagrange multiplier $\lambda \geq 0$, $R(Y, Z)$ is the actual number of bits spent on Y with quantization block Z and $D(Y, Z)$ is the distortion between the original block Y and the decoded block Y' .

$$D(Y, Z) = \sum_{(i,j) \in Y} |Y_{ij} - Y'_{ij}|^2 \quad (4)$$

In order to clearly explain this idea an example is given as follows. Suppose coefficients x and y respectively locate in the dead-zone and $[Q_{step}f, 2Q_{step}f]$ as shown in Fig.1 so that x is quantized to zero and y is quantized to one originally. In DRDO algorithm x may be requantized to one and y may be requantized to zero according to the rate-distortion cost.

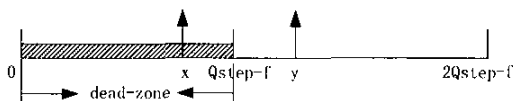


Fig.1. An example of the dead-zone

For the purpose of higher compression performance this algorithm also makes improvement on the encoding process. Instead of using fixed thresholds as JM8.0 DRDO algorithm applies rate-distortion optimization technique to judge whether encode the 8×8 block or

macro-block or not. Moreover, RD optimization is also applied in coding the 4×4 block to form the bottom-up rate-distortion optimization.

3. PERCEPTUAL DEAD-ZONE RATE-DISTORTION OPTIMIZATION ALGORITHM

Since the human observer is the end user of most image information, it is more appropriate to consider the human vision characteristics in video compression. The error sensitivity of human eyes is a function of spatial frequency. Basically, the function can be viewed as a band pass filter with a frequency response reaching the highest value at some point and decreasing very fast with increasing spatial frequency. That is, the eye is more sensitive to errors at low frequency and less sensitive to errors at high frequency. We utilize this characteristic in dead-zone rate-distortion optimization algorithm to acquire higher subjective image quality and lower bit rate by making greater distortion and less coded bits at high frequency.

It is known that the amount of luminance energy in an image block represents its inherent noise masking capability[4]. This masking capability relates the amount of quantization noise that is not perceptible to the eye. In general, the HVS sensitivity function $\hat{H}(f)$

which measures the relative sensitivity of the eye at different frequency is given by

$$\hat{H}(f) = |A(f)| \times H(f) \quad (5)$$

where $H(f) = (0.31 + 0.69f) \times \exp(-0.29f)$

$$A(f) = \left[\frac{1}{4} + \frac{1}{\pi^2} \left\{ \ln \left(\frac{2\pi f}{\delta} + \sqrt{\frac{4\pi^2 f^2}{\delta^2} + 1} \right) \right\}^2 \right]^{0.5} \quad (6)$$

In Eqn.(5-6) $\delta = 11.636 \text{ degree}^{-1}$ and the frequency variable f is in cycles/degree. To compute f the following conversion formula is employed

$$f(\text{cycle/degree}) = f_d(\text{cycles/pixel}) \times f_s(\text{pixels/degree}) \quad (7)$$

Where

$$f_d = \frac{\sqrt{i^2 + j^2}}{2N} \quad i, j = 0, 1, \dots, N-1 \quad (8)$$

f_s is dependent on the viewing distance; N is the DCT block size and is chosen to be 4. For CIF image with a height of 288 pixels and viewed at a distance of *four* times the image height, f_s is 20pixels/degree.

According to the above HVS sensitivity function $\hat{H}(f)$ the energy of pixel at i line and j column is masked as $\hat{H}_{ij}^2 Y_{ij}^2$. Thus the perceptual weighted distortion $D_H(Y, Z)$ between the original block and the decoded block is defined as

$$D_H(Y, Z) = \sum_{(i,j) \in T} \hat{H}_{ij}^2 |Y_{ij} - Z_{ij}|^2 \quad (9)$$

And the weighted rate-distortion cost $J_H(Y, Z)$ is

$$J_H(Y, Z) = D_H(Y, Z) + \lambda \times R(Y, Z) \quad (10)$$

From Eqn.(9-10), we can see that the ratio of distortion of a pixel to total distortion $D_H(Y, Z)$ reduces with the increase of the spatial frequency. Thus the pixels at higher frequency have higher possibility to be requantized from *one* to *zero* and the pixels at lower frequency have lower possibility to be requantized from *zero* to *one*. In this way it can effectively reduce the bit rate at the same time maintain the same subjective picture quality. With above explanation the proposed PDRDO algorithm takes the following main steps

1. Determine the adaptive zone.
2. Find the coefficients that fall into the adaptive zone.
3. Rank all the possible combinations as the candidate quantization blocks. There are 2^n combinations, if n coefficients are recorded in step 2.
4. For all the combinations ranked in step3 compute and store the rate-distortion cost $J_H(Y, Z)$.
5. Find the optimization quantization block that minimizes the $J_H(Y, Z)$.

4. EXPERIMENTAL RESULTS

DRDO algorithm is tested on two QCIF sequences, i.e. Container and Silent, and one CIF sequence Mobile. For each sequence, 300 frames are encoded with IPPP frame structure and with fixed quantization factors Q_p . Various Q_p factors ranged from 6 to 40 are tested. Fig.2 gives the results of PSNR and bit rate of the above testing sequences using $Q_p=[6,8,10,12,14]$. It can be

seen that average PSNR gains of 0.6 dB can be observed for DRDO algorithm with the same rate. At high rates gains of more than 1.0 dB can be observed. However, when the large Q_p is used, the improvement is not so obvious and DRDO algorithm almost has the same performance with JM8.0. RD plot for PDRDO algorithm is given for the sequence Mobile using $Q_p=[6,8,10,12,14,16,18,20,22,24,26,28,30]$ in Fig.3. Although the PSNR numbers indicates inferior objective performance for PDRDO algorithm, the coded video reveals the same subjective quality with average rate savings of about 10%. Table 1 gives the results of reduction in bit rate and slight increase in PSNR of the proposed PDRDO algorithm at high bit rate with the same subjective quality. When $Q_p=30$ is used for JM8.0 and $Q_p=29$ is used for PDRDO algorithm, PDRDO algorithm has about 6% bit rate savings and 0.13dB PSNR gains. In Fig.4, a frame of the CIF sequence Mobile encoded at $Q_p=30$ is shown. As can be seen, the subjective quality is the same while the rate is saved.

5. CONCLUSION

An adaptive dead-zone adjustment algorithm has been designed based on perceptual criteria. The proposed algorithm is able to adaptively adjust the dead-zone size in terms of RD optimization. The second contributing factor to the improved performance is the utilization of the human visual model, which discards more coefficients at high frequency without affecting the reconstructed subjective quality. Subjective viewing tests have shown that combining of an HVS model with the adaptive dead-zone adjustment method results in better performance in video compression and picture quality assessment applications. When it is measured at the same subjective quality, PDRDO algorithm is capable of 10% bit rate saving on average.

6. REFERENCES

- [1] Thomas Wedi, Steffen Wittmann, "Quantization with an Adaptive Dead Zone Size for H.264/AVC FExt", JVT-K026, Munich, Germany, March 2004
- [2] Joint Video Team(JVT), reference software JM8.0, <http://bs.hhi.de/~suehring/tml/download/>
- [3] Shanika A.Karunasekera, Nick G.Kingsbury, "A

Distortion Measure for Blocking Artifacts in Images Based on Human Visual Sensitivity", IEEE Transactions on Image processing, vol 4, pp. 713-724, June 1995

- [4] Soon Hie Tan, Khee K.Pang, "Classified Perceptual Coding with Adaptive Quantization", IEEE Trans. Circuits Syst. Video Technol., vol 6, No 4, pp. 375-388, Aug 1996

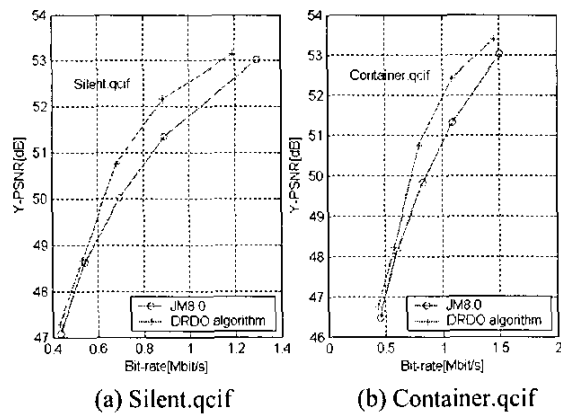


Fig.2. Comparison of the RD performance with and without DRDO

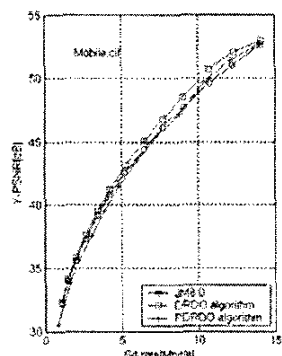
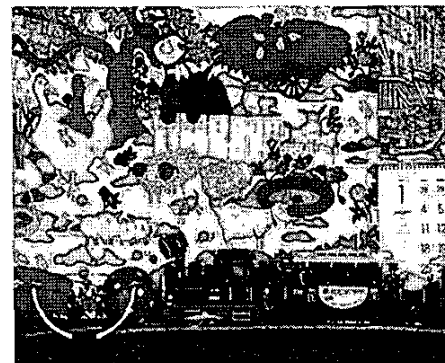


Fig.3. Comparison of the RD performance for H.264, DRDO and PDRDO algorithm

Mobile.cif		
Q_p	Bit rate saving	PSNR difference(dB)
6	1%	0.047
8	1.5%	0.385
10	4%	0.046

Table 1. Comparison of PSNR and bit rate for H.264 and proposed PDRDO algorithm



(a)



(b)



(c)

Fig.4. Subjective Comparison. (a) Original. (b) Encoded without HVS. (c) Encoded with HVS