Multi-level Low-complexity Coefficient Discarding Scheme for Video Encoder

Chuang Zhu, Huizhu Jia*, Jie Liu, Xianghu Ji, Hao Lv, Xiaodong Xie and Wen Gao
School of Electronics Engineering and Computer Science, Peking University
100871 Beijing, P. R. China
Email: {czhu, hzjia, liuzimin, xhji, hlv, xdxie, wgao}@jdl.ac.cn

***Abstract*—Rate-Distortion (R-D) optimization technique plays an important role in video coding. R-D sense discarding (thresholding) technique can make great improvement on the coding efficiency. This work first proposes a multi-level coefficient discarding scheme, which is composed of coefficient-level (CL), block-level and macroblock-level discarding. In CL, coefficient-level R-D cost function is formulated and then CL discarding scheme is developed. At last, an effective implementation method is proposed to reduce the complexity of the proposed scheme. The experimental results show that our proposed multi-level discarding scheme can improve the coding performance of video encoder by 0.15db in average.**

Keywords—R-D optimization; multi-level coefficient discarding; cost function

I. INTRODUCTION

Within the last decade, the use of data compression has become ubiquitous and various international standards are drafted to promote the development and application of video coding technique [1].

Popular standards, like H.264 and AVS, are DCT-based (discrete cosine transform) and adopt block-based hybrid coding framework. R-D optimization technique plays an important role in the hybrid coding framework. Discarding the less significant DCT coefficients may be desirable in the R-D sense because it may lead to a significant reduction in coding bitrate at a marginal sacrifice of coding quality. Kannan Ramchandran and Martin Vetterli first propose an R-D optimal way to drop the DCT coefficients of JPEG and MPEG compression standards [2]. However, the entropy coding scheme in H.264 or AVS is quite complex and performing R-D optimized thresholding in [2] for H.264 and AVS is difficult [3]. In H.264 reference model, the R-D optimized thresholding is simplified to the following problem: whether to discard or to keep the coefficients in an 8×8 block consisting of four transform blocks. We call this simplified case block-level coefficient discarding (BLD) problem. This paper conducts further research and proposes a low-complexity multi-level coefficient discarding (MLD) scheme.

The rest of this paper is organized as follows. Section II reviews BLD technique in H.264 and AVS, and then proposes a MLD scheme. To achieve MLD, section III formulates the single coefficient R-D cost function and presents the proposed coefficient-level discarding (CLD) scheme. Section IV shows the experimental results and section V concludes this paper.

This work is partially supported by grants from the Chinese National Natural Science Foundation under contract No.61171139 and No. 61035001, and National High Technology Research and Development Program of China (863 Program) under contract No.2012AA011703. *The Corresponding author.

II. PROPOSED MULTI-LEVEL COEFFICIENT DISCARDING SCHEME

A. BLD review

In H.264, the entire transform coefficients of one 8×8 block will be discarded if their total cost ($COST_{BL}$) is below T_{BLD}, which is set to 4 in JM reference model. On considering the 4×4 transform size of H.264, the BLD is described by (1) in the following [3].

$$COST_{BL} = \sum_{i=0}^{3}\sum_{j=0}^{15} cost(i,j) > 4 \tag{1}$$

where i is the transform block index in raster scan order and j is the transform coefficient index in zigzag scan order of one 4×4 block. $cost(i, j)$ is a cost function defined in the following,

$$cost(i,j) = \begin{cases} \infty & |\hat{c}(i,j)| > 1 \\ 3 & |\hat{c}(i,j)| = 1 \,\&\&\, j = 0 \\ 2 & |\hat{c}(i,j)| = 1 \,\&\&\, 1 \le j \le 2 \\ 1 & |\hat{c}(i,j)| = 1 \,\&\&\, 3 \le j \le 5 \\ 0 & otherwise \end{cases} \tag{2}$$

where $\hat{c}(i,j)$ is the quantized transform coefficient.

AVS has the similar BLD technique and provides an alternative cost function except for (2), as shown in (3).

$$cost(i,j) = \begin{cases} \infty & |\hat{c}(i,j)| > 1 \\ 1 & |\hat{c}(i,j)| = 1 \\ 0 & otherwise \end{cases} \tag{3}$$

Therefore, for AVS if we choose (3) as the cost function, one 8×8 block will be discarded when the number of ±1 is equal or less than 4 (assuming T_{BLD}=4).

B. Proposed multi-level discarding scheme

In this paper, we propose an MLD which performs different levels of discarding and as shown in Fig. 1, MLD module is placed between Quantization (Q) and point A in the encoder part. The proposed MLD scheme is described in the following:

- In the coefficient level (CL), each quantized coefficient in a block will be judged according to the CL R-D cost function. Once the discarding function is satisfied, the quantized coefficient will be discarded directly. The CLD scheme will be detailedly described in Section III.

- In the block level (BL), after the processing of CLD scheme, each quantized coefficient block will be judged as a whole. If the total cost ($COST_{BL}$) is less than T_{BLD}, then the whole block will be discarded directly.
- In the macro-block level (MBL), after the processing of CLD and BLD, each macro-block will be judged as a whole. If the total cost ($COST_{MBL}$) is less than T_{MBLD}, then the quantized coefficients of the whole macro-block will be discarded directly.

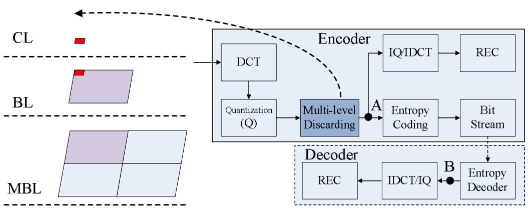

Fig. 1. Proposed multi-level discarding scheme

BLD and MBL discarding (MBLD) schemes can be easily implemented, as long as we set cost thresholds (T_{BLD} and T_{MBLD}) for them, respectively. Actually, some schemes about BLD and MBLD have been integrated into the R-D optimization process of the H.264 and AVS test model. However, to the best of our knowledge, there is no proposed low-complexity CLD scheme in the literature. In this work, we propose a low-complexity CLD scheme based on the coefficient-level R-D cost function.

III. Proposed CLD Scheme

A. Coefficient-level R-D cost function

In work [4], the authors propose a novel statistical model of R-D estimation for H.264/AVC coders and use it to accelerate mode decision process. The R-D estimation in the above work is block-level based and we will formulate the coefficient-level R-D cost function based on some conclusions of [4].

In this work, we assume each coefficient of the quantized DCT coefficient is independent and can be processed singly no matter what entropy coding method is used.

The probability density function (PDF) of a single DCT coefficient C_{uv} is modeled with the generalized Gaussian function (GGF) which includes the Gaussian and the Laplacian PDF as special cases [5], as shown in Fig. 2.

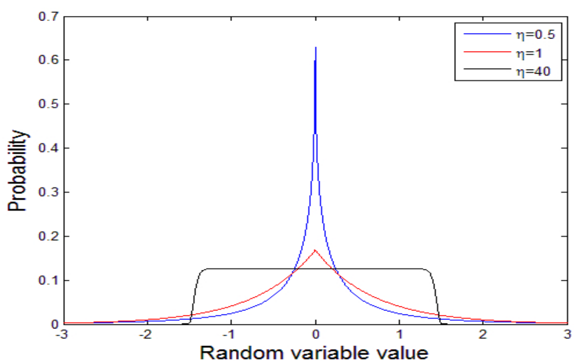

Fig. 2. Generalized Gaussian function

The GGF is described as follows:

$$P_{C_{uv}} = f_{uv}(C_{uv}) = \frac{\eta_{uv}\alpha_{uv}(\eta_{uv})}{2\sigma_{uv}\Gamma(1/\eta_{uv})}\exp\left\{-\left[\alpha_{uv}(\eta_{uv})\frac{|C_{uv}|}{\sigma_{uv}}\right]^{\eta_{uv}}\right\} \qquad (4)$$

with

$$\alpha_{uv}(\eta_{uv}) = \sqrt{\frac{\Gamma(3/\eta_{uv})}{\Gamma(1/\eta_{uv})}} \qquad (5)$$

where $\Gamma(\bullet)$ is the gamma function, η_{uv} and σ_{uv} are positive real-valued distribution parameters which control the shape and scale of the GGD, respectively.

Corresponding to C_{uv}, the quantized value is $\hat{C}_{uv}$. According to the design rules of optimal entropy coders, the entropy coding bits of a quantized coefficient $\hat{C}_{uv}$ with occurrence probability $P_{\hat{C}_{uv}}$ is directly dependent on the self-information [4]. This work directly uses self-information as the coding bits r_{uv} corresponding to $\hat{C}_{uv}$. According to the definition of self-information, we can get

$$r_{uv} = -\log_2(P_{\hat{C}_{uv}}) \qquad (6)$$

Frankly speaking, the solution of (6) is hard to yield, however, we can first find out the solution "form" (not the exact solution) to (6) and then use the least squares technique to make roughly estimation of r_{uv}.

According to (6), the key is to find out $P_{\hat{C}_{uv}}$ to formulate r_{uv}. In the recent standards, like H.264 and AVS, uniform scalar quantization method is used. Supposing the quantization step size is Q_{step}, the relationship between C_{uv} and $\hat{C}_{uv}$ can be roughly described as (7).

$$\hat{C}_{uv} = C_{uv} / Q_{step} \qquad (7)$$

Then, we can get (8). Here, we use the reconstructed value to approximate the transform coefficient before quantization.

$$C_{uv} \approx \hat{C}_{uv}\bullet Q_{step} \qquad (8)$$

Based on the theory of probability, we will have

$$\begin{aligned} P_{\hat{C}_{uv}} &\approx f_{uv}(C_{uv})\bullet\left|(C_{uv})'\right| \\ &= f_{uv}(\hat{C}_{uv}\bullet Q_{step})\bullet Q_{step} \end{aligned} \qquad (9)$$

Combining (4), (6) and (9), the following (10) can be developed. We should point out that $P_{\hat{C}_{uv}}$ is no longer a strictly continuous function (discrete function in fact) in quantization process of video coding. Besides, in actual coding standards, both transform coefficients and quantized coefficients are discrete values. Many transform coefficients are mapped to one quantized coefficient point after quantization. Therefore, we can treat each point in $P_{\hat{C}_{uv}}$ as integration of an interval in $P_{C_{uv}}$ and thus the point in $P_{\hat{C}_{uv}}$ can be treated as the probability of a quantized coefficient $\hat{C}_{uv}$.

$$\begin{aligned} r_{uv} &= -\log_2(P_{\hat{C}_{uv}}) = -\log_2\left(\left(f_{uv}(\hat{C}_{uv}\bullet Q_{step})\right)\bullet Q_{step}\right) \\ &= -\log_2\left\{\frac{\eta_{uv}\alpha_{uv}(\eta_{uv})Q_{step}}{2\sigma_{uv}\Gamma(1/\eta_{uv})}\exp\left\{-\left[\alpha_{uv}(\eta_{uv})\frac{|\hat{C}_{uv}\bullet Q_{step}|}{\sigma_{uv}}\right]^{\eta_{uv}}\right\}\right\} \end{aligned} \qquad (10)$$

For convenience, we introduce (11) and (12).

$$m = \frac{\eta_{uv}\alpha_{uv}(\eta_{uv})Q_{step}}{2\sigma_{uv}\Gamma(1/\eta_{uv})} \tag{11}$$

$$n = \left[\alpha_{uv}(\eta_{uv})\frac{Q_{step}}{\sigma_{uv}}\right]^{\eta_{uv}} \tag{12}$$

Then,

$$\begin{aligned} r_{uv} &= -\log_2\left\{m\exp\left\{-n\left|\hat{C}_{uv}\right|^{\eta_{uv}}\right\}\right\} \\ &= -\left\{\log_2 m + \log_2\left\{\exp\left\{-n\left|\hat{C}_{uv}\right|^{\eta_{uv}}\right\}\right\}\right\} \\ &= -\left\{\log_2 m + \ln\left\{\exp\left\{-n\left|\hat{C}_{uv}\right|^{\eta_{uv}}\right\}\right\}/\ln 2\right\} \\ &= -\log_2 m + n\left|\hat{C}_{uv}\right|^{\eta_{uv}}/\ln 2 \\ &= a\left|\hat{C}_{uv}\right|^{\eta_{uv}} + b \end{aligned} \tag{13}$$

with $a = n/\ln 2$ and $b = -\log_2 m$.

For distortion part, as work [4] tells that for a quantized coefficient $\hat{C}_{uv}$, the distortion d_{uv} can be calculated in the transform domain after quantization, as shown in (14).

$$d_{uv} = \left[\left(\left|offset_{uv} - low\_qbits_{uv}\right|/2^{q\_bits}\right)\bullet Q_{step}\right]^2$$
(14)

where offset is quantization rounding offset and low_qbits_{uv} is the discarded low q bits in quantization process [4]. We use (14) as distortion for a quantized coefficient C_{uv} because d_{uv} can be calculated directly after quantization process.

Besides, for simplicity, all the quantized coefficients use the same Lagrange multiplier λ and then the R-D cost function for one quantized coefficient C_{uv} can be formulated as

$$\begin{aligned} j_{uv} &= d_{uv} + \lambda r_{uv} \\ &= \left[\left(\left|offset_{uv} - low\_qbits_{uv}\right|/2^{q\_bits}\right)\bullet Q_{step}\right]^2 + \lambda\left(a\left|\hat{C}_{uv}\right|^{\eta_{uv}} + b\right) \end{aligned}$$
(15)

In the following, we will present the proposed CLD scheme based on (15).

B. Proposed CLD scheme

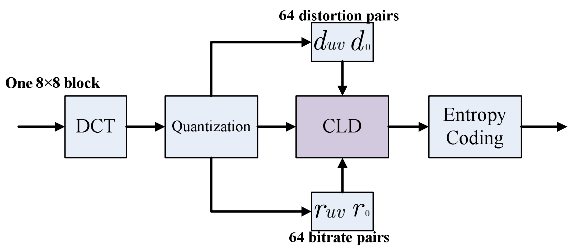

Fig. 3. Proposed CLD framework

We propose a CLD scheme and the discarding framework is shown as Fig. 3.

Take an 8×8 block for example, after quantization, we calculate two distortions (d_{uv} and d_0) and two bitrates (r_{uv} and r_0) for each quantized coefficient. d_0 and r_0 denote the distortion and rate introduced by C_{uv} when quantizing it to 0. In fact, r_0 is very small (b in (15)) and can be treated as zero. Therefore, for each DCT coefficient C_{uv}, we have two choices, quantizing it to $\hat{C}_{uv}$ or quantizing it to 0 (discarding). In this paper, we make the decision in an R-D optimized manner and select (16) as our decision function.

$$j_{uv} > j_0 \;\Rightarrow\; d_{uv} + \lambda r_{uv} > d_0 + \lambda r_0 \tag{16}$$

When (16) satisfies, we will discard the quantized coefficient $\hat{C}_{uv}$.

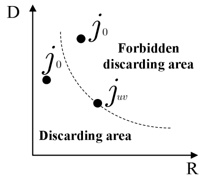

Fig. 4. R-D optimized CLD Schematic

From Fig. 4 we can see that when the R-D cost j_0 of a single coefficient C_{uv} falls in the discarding area, the quantized coefficient $\hat{C}_{uv}$ could be discarded. However, if j_0 falls in the forbidden discarding area, we should not discard the corresponding coefficient. It is clear that by discarding the quantized coefficient when it falls in the discarding area of Fig. 4, the coding performance will be improved: a significant reduction in coding bitrate at a marginal sacrifice of coding quality.

C. Low-complexity implementation of proposed CLD scheme

The key issue of implementing our proposed CLD scheme is the calculation of j_{uv} and j_0 for each single quantized coefficient C_{uv}.

For d_{uv} and d_0, we can calculate them according to (14) and the value of C_{uv} itself directly. It is hard to calculate r_{uv} and r_0 based on (13), because we need to update the parameters of GGF frequently.

For simplicity, we simplify r_{uv} in (13) to the following linear form (the experimental results tell that this kind of simplicity does not introduce much coding efficiency loss).

$$r_{uv} = a_{uv}\left|\hat{C}_{uv}\right|^{\eta_{uv}} \tag{17}$$

In the following, we use least squares method to fix the weighting factor a_{uv} for each quantized coefficient $\hat{C}_{uv}$. According to work [5], we choose η=0.5 for simplification. We fetch t num of 8×8 coding blocks with R1, R2, … , Rt (which can be easily generated from reference test model) being the real coding bits of each coding block after entropy coding to fix a_{uv}.

Let x_{uv} be $\left|\hat{C}_{uv}\right|^{\eta_{uv}}$, then we have

$$\begin{aligned} &\sum_{u=0}^{7}\sum_{v=0}^{7} a_{uv}x_{1uv} = R_1 \\ &\sum_{u=0}^{7}\sum_{v=0}^{7} a_{uv}x_{2uv} = R_2 \\ &\qquad\ldots\ldots \\ &\sum_{u=0}^{7}\sum_{v=0}^{7} a_{uv}x_{tuv} = R_t \end{aligned} \tag{18}$$

Rewrite (18) in matrix form, the following (19) can be developed,

$$X \bullet A^T = R \tag{19}$$

with

$$X = \begin{bmatrix} x_{100} & x_{101} & \dots & x_{177} \\ x_{200} & x_{201} & \dots & x_{277} \\ \dots & \dots & \dots & \dots \\ x_{t00} & x_{t02} & \dots & x_{t77} \end{bmatrix} \quad \begin{array}{l} A = [a_{00}, a_{01}, \dots, a_{77}] \\ R = [R_1, R_2, \dots, R_t]^T \end{array}$$

Then we can get A^T as

$$A^T = (X^T \bullet X)^{-1} \bullet X^T \bullet R \tag{20}$$

Once we get A, then all the weighting factors (a_{uv}) are generated, and r_{uv} can be yielded according to (17) directly.

Actually, in practice we train different weighting factors for intra and inter modes, respectively. In this work, we adopt the following trained A_{intra} and A_{inter} for intra modes and inter modes, respectively.

$$\begin{array}{l} A_{\text{intra}} = [2.8522, 2.7943, 3.8530, 4.0431, 5.4804, 6.1372, 6.1372, 8.6949, \\ 3.4171, 3.2302, 3.7109, 4.4696, 6.1372, 6.1372, 8.6949, 8.6949, \\ 3.5170, 4.0278, 4.8078, 6.1372, 6.1372, 8.6949, 8.6949, 8.6949, \\ 4.9384, 4.1153, 6.1372, 6.1372, 8.6949, 8.6949, 8.6949, 11.4159, \\ 4.9234, 6.1372, 6.1372, 8.6949, 8.6949, 8.6949, 11.4159, 11.4159, \\ 6.1372, 6.1372, 8.6949, 8.6949, 8.6949, 11.4159, 11.4159, 11.4159, \\ 6.1372, 8.6949, 8.6949, 8.6949, 11.4159, 11.4159, 11.4159, 11.4159, \\ 8.6949, 8.6949, 8.6949, 11.4159, 11.4159, 11.4159, 11.4159, 11.4159] \end{array}$$

$$\begin{array}{l} A_{\text{inter}} = [2.4216, 2.8248, 3.7599, 4.1517, 4.6482, 5.0337, 5.0337, 6.5898, \\ 3.3364, 3.5259, 4.0770, 4.5610, 5.0337, 5.0337, 6.5898, 6.5898, \\ 3.4069, 3.9707, 4.5525, 5.0337, 5.0337, 6.5898, 6.5898, 6.5898, \\ 4.2071, 4.2134, 5.0337, 5.0337, 6.5898, 6.5898, 6.5898, 8.2926, \\ 4.1612, 5.0337, 5.0337, 6.5898, 6.5898, 6.5898, 8.2926, 8.2926, \\ 5.0337, 5.0337, 6.5898, 6.5898, 6.5898, 8.2926, 8.2926, 8.2926, \\ 5.0337, 6.5898, 6.5898, 6.5898, 8.2926, 8.2926, 8.2926, 8.2926, \\ 6.5898, 6.5898, 6.5898, 8.2926, 8.2926, 8.2926, 8.2926, 8.2926] \end{array}$$

Fig. 5. Adopted weighting factors of intra and inter modes

IV. EXPERIMENTAL RESULTS

We integrate our proposed multi-level coefficient discarding scheme to AVS test model and apply the discarding scheme to non-reference frames. We set T_{BLD} to 4 and T_{MBLD} to 5. Besides, we choose (3) as our cost function.

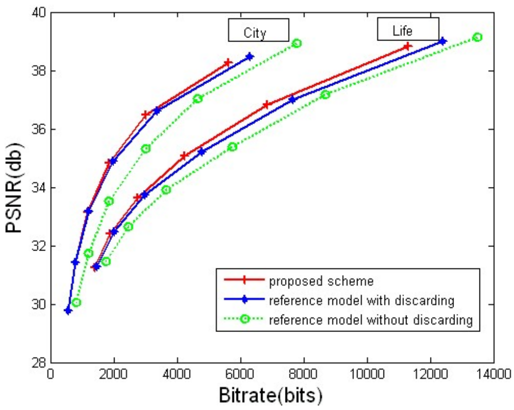

Fig. 6. Performance Comparisons for "City" and "Life"

Two selected HD sequences "City" (720P) and "Life" (1080P) were tested and Fig. 6 shows the corresponding RD curves. From Fig. 6, we can see our proposed scheme can improve the coding performance about 0.2db when compared with AVS reference model. Through advanced analysis, we find that the main improvement of our proposed MLD scheme happens in the high bitrate part. That is because in our MLD scheme we introduced CLD technique which is effective in high bitrate situation, while BLD and MBLD do not work at all due to the fact that the $COST_{BL}$ and $COST_{MBL}$ are so easy to exceed T_{BLD} and T_{MBLD}, respectively. More test results are tabulated in Table I and the average PSNR gain is 0.15db compared with the reference model (discarding open). The comparisons between the proposed scheme and the reference model with no discarding scheme are also presented in Table I to show the importance of discarding technique.

TABLE I. PERFORMANCE COMPARISONS.

Format	Sequence	Proposed method VS Test model of AVS		Proposed method VS No discarding	
		PSNR Gain (dB)	***Bit-rate change (%)***	***PSNR Gain (dB)***	***Bit-rate change (%)***
CIF	*bike*	0.23	-2.85	0.37	-4.58
	container	0.21	-5.34	1.55	-43.5
	flower	0.32	-5.56	0.49	-8.93
720P	*city*	0.15	-4.79	1.02	-34.29
	cyclists	0.07	-2.71	0.66	26.93
	night	0.15	-4.37	0.50	-14.83
1080P	*mobcal*	0.20	-9.66	0.74	-38.80
	park_join	0.12	-2.67	0.17	-3.76
	life	0.26	-7.01	0.68	-18.72

V. ACKNOWLEDGMENT

The authors are grateful to Dr. Xin Zhao for his valuable discussions related on R-D estimation model.

REFERENCES

[1] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification, document JVT-G050.doc, ITU-T Rec. H.264 and ISO/IEC 14496-10, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Mar. 2003.

[2] Kannan Ramchandran and Martin Vetterli, Rate-Distortion Optimal Fast Thresholding with Complete JPEGMPEG Decoder Compatibility, IEEE Transactions on Image Processing, vol. 3, no. 5, Sep. 1994.

[3] Carlsson Pontus, Pan F., and Chia L. T., " Coefficient thresholding and optimized selection of the Lagrangian multiplier for non-reference frames in H.264 video coding. ," presented at IEEE Int. Conf. on Image Processing (ICIP 2004) , pp. 62–67 ((2004)).

[4] Xin, Z., Sun Jun, Siwei Ma, Wen Gao., "Novel Statistical Modeling, Analysis and Implementation of Rate-Distortion Estimation for H.264/AVC Coders," IEEE Trans. Circuits Syst. Video Technol. 20(5): pp. 647-660, 2010.

[5] F. Müller, Distribution shape of two-dimensional DCT coefficients of natural images, Electron. Lett., vol. 29, no. 22, pp. 1935−1936, Oct. 1993.