An Effective Post Quantization Rate Estimation for HEVC Intra Encoder

Hongbin Cao, Ronggang Wang, Zhenyu Wang, Ge Li, Wenmin Wang
School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University
hbcao@pku.edu.cn, {rgwang, wangzhenyu, gli, wangwm}@pkusz.edu.cn

***Abstract*—In high efficiency video coding (HEVC), the encoder employs a flexible quad-tree coding structure as well as a large number of prediction modes. For each size of coding unit (CU), transform unit (TU) and each prediction mode, the rate distortion optimization (RDO) is performed to select the best CU, TU and the best prediction mode. Although better coding efficiency is achieved, the computational complexity increases dramatically. In order to reduce the burden of RDO in the HEVC intra encoder, in this paper, we propose an effective approach, which is based on the generalized Gaussian distribution (GGD) model, to estimate the block level bit-rate. The weaknesses of the conventional GGD model are analyzed and relevant improvements are exploited. Our experiments show that, compared with the original RDO procedure in HM16.0, the proposed algorithm reduces RDO time by 37.7% with 0.64% BD-rate loss.**

***Index Terms*—HEVC, GGD, intra prediction, rate estimation, video coding.**

I. Introduction

The newest international video coding standard HEVC has achieved a significant improvement in rate-distortion efficiency over the previous H.264/AVC. However, the encoder complexity increases greatly due to the highly flexible quad-tree coding structure as well as a large number of prediction modes [1]. Although several methods (i.e. [2], [3]) are adopted to reduce the computational complexity of rate distortion optimization (RDO), it is still a time-consuming work as entropy coding is required to get the coding bits for each mode for each block. Therefore, it is a good choice to find a simple way to predict the coding bits, which may save the large time spent in entropy coding and enhance the speed of the HEVC intra encoder.

Several approaches have been proposed to estimate the coding bits. Table-based rate estimation is used to provide a simplified CABAC engine in [4]–[6]. Other methods are based on block level rate estimation in discrete cosine transformation (DCT) domain. Chen et al. took advantage of the l_0 and l_1-norm to model the bit-rate in [7]. Li et al. used $l_{1/2}$-norm to form their model in [8]. Tu et al. proposed a more dedicated model which combined l_0 and l_p-norm to estimate the bit-rate of quantized transform coefficients [9]. Zhao et al. in [10] proposed a novel statistical model based on the generalized Gaussian distribution (GGD) which utilized the different positions of transform coefficients, and achieved a better performance. Sheng et al. also took into account the different positions of transform coefficients and modeled them with the l_1-norm to predict the coding bits [11].

Among all these methods, [4]–[6] tend to provide less time saving, since they only simplify the arithmetic coding and still need to perform the binarization and context modeling. [7]–[9] neglect the positions of coefficients in each blocks. [11] takes into account the coefficients' positions, but its model is not dedicated enough. Zhao's model [10] performs better in H.264/AVC, however its loss raises greatly if we directly apply it in HEVC (3.0% BD-rate loss from our experiment), because of the difference in the coding structures between the two standards. With various sizes of coding unit (CU), prediction unit (PU) and transform unit (TU) in HEVC, Zhao's model cannot achieve accurate decisions for CU and TU depth.

In this paper, we first review the algorithm of Zhao's model, and analyze its weaknesses when it is applied to HEVC. Based on our analysis, we propose several improvements on Zhao's algorithm and make it practical in HEVC. Quality and coding speed are considered in our algorithm and experimental results show that our improved algorithm achieves less quality loss on the same acceleration of RDO period than other existing models.

The rest of the paper is organized as follows. A brief outline of Zhao et al.'s algorithm is reviewed in Section II. The problems when it is applied to HEVC are analyzed and related solutions are proposed in Section III. The experimental results are given and discussed in Section IV. Finally, the paper is concluded in Section V.

II. Review of Zhao's approach

In Zhao's proposal, three steps need to be taken for the bit estimation, namely building models with the transform coefficients, estimating the coding bits with the quantized transform coefficients on the calculated model, and revising the estimated coding bits through linear fit according to the relationship between the previously estimated bits and the corresponding actual coding bits.

A. Model Selection

Zhao in his algorithm employed a zero-mean generalized Gaussian distribution (GGD) to model the distribution of

Thanks to National Natural Science Foundation of China 61370115, China 863 project of 2015AA015905, Shenzhen Peacock Plan, Shenzhen Research Projects of JCYJ20150331100658943 and JCYJ20160506172227337, and Guangdong Province Projects of 2014B010117007 and 2014B090910001 for funding.

978-1-5090-5316-2/16/$31.00 © 2016 IEEE

transform coefficients. He pointed out that each position in a transform block should have a distinct model, since the same level in different positions may contribute differently to the final coding bits. The distribution for a single transform coefficient C_{uv} can be described as

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

with

$$\alpha_{uv}(\eta_{uv}) = \sqrt{\frac{\Gamma(3/\eta_{uv})}{\Gamma(1/\eta_{uv})}} \quad (2)$$

where $f_{uv}(x)$ indicates the probability density function (PDF) of C_{uv}, $\Gamma(\cdot)$ is the gamma function, η_{uv} and σ_{uv} are position real-valued distribution parameters that control the shape and scale of the GGD [10].

B. Coding Bits Estimation

Self-Information is used to estimate the coding bits from quantized transform coefficients with the selected models. As Fig. 1 shows, the curve $f_{uv}(x)$ represents the distribution of transform coefficients, and the shadow volume represents the probability of transform coefficients which will be quantized to $\hat{x}$.

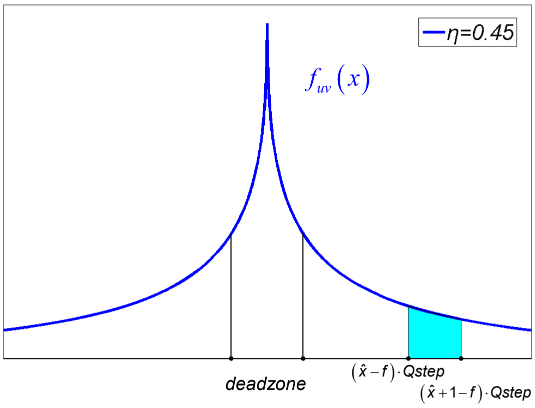

Fig. 1. Generalized Gaussian distribution with shape parameter $\eta = 0.45$

The self-information of $\hat{C}_{uv}$ (quantized value of C_{uv}) can be formulated by

$$r_{uv} = -\log_2 P\{\hat{C}_{uv} = \hat{x}\} \quad (3)$$

where $P\{\hat{C}_{uv} = \hat{x}\}$ represents the probability of an coefficient which will be quantized to $\hat{C}_{uv}$. We can calculate it with the following expression

$$P\{\hat{C}_{uv} = \hat{x}\} = \begin{cases} 2\cdot\int_0^{(1-f)\cdot Qstep} f_{uv}(x)\mathrm{d}x & \hat{x}=0 \\ \int_{(|\hat{x}|-f)\cdot Qstep}^{(|\hat{x}|+1-f)\cdot Qstep} f_{uv}(x)\mathrm{d}x & \hat{x}\neq 0 \end{cases}$$
$$\approx \begin{cases} 2(1-f)\cdot Q_{step}\cdot f_{uv}(f\cdot Qstep) & \hat{x}=0 \\ Q_{step}\cdot f_{uv}(|\hat{x}|\cdot Qstep) & \hat{x}\neq 0 \end{cases} \quad (4)$$

which uses $f\cdot Qstep$ (f denotes the quantization offset) to approximate the average probability of the first dead zone quantization interval ($\hat{x}=0$), and $|\hat{x}|\cdot Qstep$ to approximate of the rest corresponding quantization interval ($\hat{x}\neq 0$) [10].

C. Linear Fit Revision

In Zhao's algorithm, he verified that it is improper to use self-information to estimate the entropy coding bits. However, he found strong linear correlation between self-information and the actual entropy coding bits. The final estimator is formulated as

$$r_B = \sum\nolimits_u \sum\nolimits_v r_{uv}, \quad (5)$$

$$R_B = \alpha\cdot r_B + \beta \quad (6)$$

where r_B is the self-information of each block and R_B is the final estimated coding bits.

III. Proposed Revised Algorithm

A. Model Improvement

In many cases, we observe that the DCT coefficients have a heavy tail, and consequently it is ineffective to model those coefficients by an exponentially decaying function, i.e. the GGD model. The binarization strategy for an absolute coefficient level in HEVC relies on a concatenated application of truncated unary (TrU), k-th order truncated Rice (TRk), and $(k+1)$-th order Exp-Golomb (EGk). EGk, which has a slow growth of the code word length, is applied to large absolute level coefficients. The length of binarization bits (as well as actual coding bits) tends to be the same. As a result, uniform distribution is more suitable for modelling the transform coefficients in DCT domain when they are large. As an improvement, we model the transform coefficients with the original model (GGD) in the main body, and use uniform distribution to model the DCT coefficients in the heavy tail. We use a simple but effective way to approximate the combined model, which is formulated as

$$f'_{uv}(x) = \begin{cases} \theta_{uv}\cdot f_{uv}(x) & |x| \leq b_{uv} \\ f'_{uv}(b_{uv}) & b_{uv} < |x| \leq m_{uv} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

where θ_{uv} is a scaling factor of the original GGD probability density function $f_{uv}(x)$, b_{uv} is the boundary between the GGD and the uniform distribution, and m_{uv} represents the largest magnitude a sample x can take. As we mentioned above, the binarization bits could be probably the same when the coefficients are large. As a result, we can use the following empirical formula to estimate the value b_{uv} in our model:

$$f'_{uv}(b_{uv})\cdot(m_{uv}-b_{uv}) \leq 1/1024. \quad (8)$$

The boundary b_{uv} and the scaling factor θ_{uv} of the model can be solved together with the following equation:

$$\int_0^{b_{uv}} f'_{uv}(x)\mathrm{d}x + f'_{uv}(b_{uv})\cdot(m_{uv}-b_{uv}) = 1/2. \quad (9)$$

The combined model and the conventional GGD model are displayed in Fig. 2. As we can see, the distribution turns into uniform distribution when x exceeds the boundary.

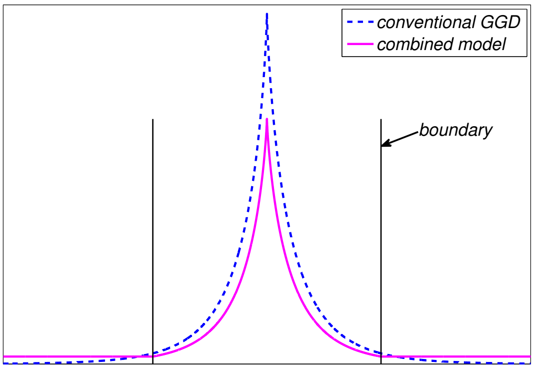

Fig. 2. The conventional GGD model and the proposed combined model with GGD and uniform distribution.

B. Improvement In Coding Bits Estimation Period

In Zhao's algorithm, in order to avoid the repetitive calculation of formula (4) during the RDO process, a look-up table is used to preserve the estimated coding bits for most of the quantized transform coefficients [10]. A const threshold value 200 is set in Zhao's algorithm, which means the estimated coding bits of quantized coefficients lower than 200 should be pre-calculated in the look-up table. We can see that 32000 (4×4×200) computations are needed for 4×4 block, but when it comes to 32×32 block, the needed computations turn out to be 204800 (32×32×200). The cost for updating the look-up table for each intra frame increases dramatically as various block sizes are introduced in HEVC.

However, in our model, we only need to apply the time-consuming calculation for the main body of the model. The tail part is uniform distribution, which means the quantized coefficients in this area share the same estimated coding bits. The boundary of the two model is far less than 200 in most cases. Fig. 3 shows the boundary of 4x4 block of BQSquare with WQVGA (Qstep is set to 22).

45	23	14	8
21	13	8	5
12	8	5	4
7	5	4	3

Fig. 3. Boundary between GGD and uniform distribution of 4×4 block. The data above represent the boundary value divided by Qstep for each positon.

As we can see in Fig. 3, the computations for model update of 4×4 block drop from 32000 (4×4×200) to 185. For larger size of block, more decrease in computation is obtained compared to the direct application of Zhao's algorithm.

Other improvements are stated as follows:

1) The update of look-up tables is processed for each frame in Zhao's algorithm, but when it comes to huge block sizes, such as 32×32 and 64×64 blocks, the sample number could be rather rare, especially for low resolution sequences. The calculated model might be inaccurate. As a solution we set a counter for each block size to decide when to update the corresponding look-up tables.

2) Fixed point number is used for look-up tables other than the original double precision floating point number. Experiments show that when we use the low 12 bit of an integer number as the decimal part of the estimated bits in the look-up tables, acceleration is achieved with insignificant loss in the final results.

3) In order to simplify the estimator, we only calculate the coding bits when the absolute level of a coefficient is greater than 0. The formula (3) is revised as

$$r_{uv} = \begin{cases} -\log_2 P\{\hat{C}_{uv} = \hat{x}\} & |\hat{x}| \neq 0 \\ 0 & |\hat{x}| = 0. \end{cases} \qquad (10)$$

C. Improvement In Revision Period

Zhao in his algorithm use linear relation to fit the self-information and the final estimated coding bits, as is shown in formula (6). The purpose is to help judge the best CU depth in the RDO process, since the best mode in a certain CU depth is determined by formula (5). It works well in H.264/AVC as there are only 4×4 blocks, 8×8 blocks and 16×16 blocks in the encoder. However, in HEVC, the coding unit (CU) varies from 8×8 to 64×64, and for each CU, different sizes of transform unit (TU) could be chosen. The original algorithm fails for the dedicated partition of CU and TU.

In order to get more rational partition of CU and TU, a final entropy coding might be helpful after we get the best mode among the candidates for each depth. However, the cost of entropy coding is huge for both software and hardware implementation in the encoder. As alternative, we use the binarization bits as the judge to select the rational size of CU and TU. As a result, the linear fit process in the original algorithm can be removed, since we will perform the binarization for the best mode in each depth, and use the binarization bits as the judge for different CU and TU sizes. Through these improvements, more accurate decisions can be made for CU and TU partition in the RDO process.

In our implementation, the coefficients of the best mode for each depth should be stored as we need to perform a simple entropy coding (actually only binarization) afterwards. In order to avoid the copy operation when we get a better mode, we apply dual buffer strategy. Buffer A is for temporarily transform and quantization process and buffer B is to store the best coefficients at the initial time. When a better mode comes (it must be quantized in buffer A), then buffer A turns to store the best coefficients and buffer B becomes the temporary buffer for transform and quantization.

IV. Experimental Results

To verify the performance of the proposed algorithm, we implement it in HM16.0. In the experiment, sequences are tested in All-Intra (AI) configuration. The test condition is based on common test conditions of HEVC [12] and the configuration of RDOQ is turned off. QP is set as 22, 27, 32,

37. The complexity reduction is measured by time reduction radio of RDO (ΔT_{RDO}), which is defined as

$$\Delta T_{\text{RDO}} = \frac{T_{OriginalRDO} - T_{ProposedRDO}}{T_{OriginalRDO}} \times 100\%. \quad (11)$$

The time cost for the RDO module is measured by Intel Vtune Amplifier, and the estimation accuracy is measured by BD-rate [13].

In this experiment, the proposed algorithms are compared with Sheng et al.'s rate estimation algorithm which is based on quantized coefficients described in [11]. Sheng et al.'s algorithm also takes into account the coefficients' positions. What's different, their algorithm regards the positions as variables in the expression of the estimator, which is formulated as

$$r_B = \sum\nolimits_{\hat{C}_{uv} \neq 0} (|\hat{C}_{uv}| + \theta \cdot (u + v)), \quad (12)$$

$$R_B = \alpha \cdot (r_B)^{\beta} \quad (13)$$

where θ is regard as a balance parameter of the position information (u, v). Nonlinear regression is used to to revise the predicted coding bits. Both algorithms are implemented in HM16.0 with the test configuration discussed above.

TABLE I
Performance Of The Proposed Algorithm And Sheng et al.'s Method in [11] (Compared With The Original Rate-Distortion Optimization Technique)

Class	**Sequence**	**Proposed**		**Sheng et al.'s**	
		BD-rate [%]	ΔT_{RDO} [%]	BD-rate [%]	ΔT_{RDO} [%]
Class A (4K)	Traffic	0.91	36.3	1.96	37.1
	PeopleOnStreet	0.94	37.4	1.85	39.9
Class B (1080p)	Kimono	0.82	32.4	2.05	29.9
	ParkScene	0.87	37.2	1.64	38.5
	Cactus	0.77	42.0	2.05	40.4
	BasketballDrive	0.87	37.7	2.58	38.3
	BQTerrace	0.73	31.7	1.61	35.5
Class C (WVGA)	BasketballDrill	0.63	34.1	1.88	37.2
	BQMall	0.72	39.4	1.74	38.8
	PartyScene	0.42	44.5	1.03	44.1
	RaceHorses	0.65	38.4	1.60	40.7
Class D (WQVGA)	BasketballPass	0.67	40.7	1.75	42.0
	BQSquare	0.28	44.6	0.71	45.5
	BlowingBubbles	0.49	44.2	1.14	44.9
	RaceHorses	0.69	43.4	1.61	43.7
Class E (720p)	FourPeople	0.86	32.7	1.96	32.7
	Johnny	0.79	31.6	2.24	30.3
	KristenAndSara	0.76	30.8	1.93	30.8
Class F	BasketballDrillText	0.51	36.3	1.52	35.2
	ChinaSpeed	0.31	39.0	0.91	39.1
	SlideEditing	0.05	43.6	0.26	44.6
	SlideShow	0.35	32.2	0.83	30.5
Total Average		**0.64**	**37.7**	**1.58**	**38.2**

The experimental results are shown in Table I. We can see that our method achieves quality loss of 0.64% in BD-rate increase, which is better than Sheng et al.'s method (BD-rate increases in 1.58%). From the formulas (10, 12) of both methods , we can see that, in the second period of rate estimation, our algorithm seems to act faster, because the values in formula (10) are pre-calculated in the look-up tables. In the third period, our method will perform a simple entropy coding (actually only binarization) for the best mode in each depth, which will afterwards be used for CU and TU depth decision. Although a little more time is cost, better accuracy is achieved than Sheng et al.'s method which uses formula (13) as revision. As for the first period (model building), the proportion is so small in the encoder that the influence could be neglected. In general, our method achieves far less loss in the encoder than Sheng et al.'s method at almost the same acceleration.

V. Conclusion

In this paper, a post quantization rate estimation algorithm is proposed for HEVC intra prediction. We improve the model applied by Zhao et al. in H.264, and make it practical for various sizes of coding units (CU) and transform units (TU) in HEVC. Experiments show that, with the proposed algorithm, the computational complexity of RDO is reduced by 37.7% with negligible coding performance degradation (0.64% in BD-rate increase). Future work includes applying the algorithm to the inter-picture prediction in the HEVC encoder.

References

[1] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, 2012.

[2] L. Zhao, L. Zhang, X. Zhao, S. Ma, D. Zhao, and W. Gao, "Further encoder improvement of intra mode decision," *JCTVC-D283, Daegu*, 2011.

[3] J. Kim, J. Yang, H. Lee, and B. Jeo, "Fast intra mode decision of hevc based on hierarchical structure," in *Information, Communications and Signal Processing (ICICS) 2011 8th International Conference on*. IEEE, 2011, pp. 1–4.

[4] J. Hahm and C.-M. Kyung, "Efficient cabac rate estimation for h. 264/avc mode decision," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 2, pp. 310–316, 2010.

[5] K. Won, J. Yang, and B. Jeon, "Fast cabac rate estimation for h. 264/avc mode decision," *Electronics letters*, vol. 48, no. 19, pp. 1201–1203, 2012.

[6] F. Bossen, "Ce1: Table-based bit estimation for cabac," *JCTVC-G763, Geneva*, 2011.

[7] Q. Chen and Y. He, "A fast bits estimation method for rate-distortion optimization in h. 264/avc," in *Proc. PCS*, 2004.

[8] Y. Li, X. Mou, and C. Wang, "An l1/2-norm based efficient block level rate estimation model for hevc," in *Multimedia Signal Processing (MMSP), 2015 IEEE 17th International Workshop on*. IEEE, 2015, pp. 1–6.

[9] Y.-K. Tu, J.-F. Yang, and M.-T. Sun, "Efficient rate-distortion estimation for h. 264/avc coders," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 16, no. 5, pp. 600–611, 2006.

[10] X. Zhao, J. Sun, S. Ma, and W. Gao, "Novel statistical modeling, analysis and implementation of rate-distortion estimation for h. 264/avc coders," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 5, pp. 647–660, 2010.

[11] Z. Sheng, D. Zhou, H. Sun, and S. Goto, "Low-complexity rate-distortion optimization algorithms for hevc intra prediction," in *MultiMedia Modeling*. Springer, 2014, pp. 541–552.

[12] F. Bossen, "Common test conditions and software reference configurations," *Joint Collaborative Team on Video Coding (JCT-VC), JCTVC-F900*, 2011.

[13] G. Bjontegaard, "Calcuation of average psnr differences between rd-curves," *Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April 2001*, 2001.