Low-Complexity Rate-Distortion Optimization Algorithms for HEVC Intra Prediction

Zhe Sheng, Dajiang Zhou, Heming Sun, and Satoshi Goto

Graduate School of Information, Production and Systems,
Waseda University, Kitakyushu, Japan
shengzhe@fuji.waseda.jp

Abstract. HEVC achieves a better coding efficiency relative to prior standards, but also involves dramatically increased complexity. The complexity increase for intra prediction is especially intensive due to a highly flexible quad-tree coding structure and a large number of prediction modes.
The encoder employs rate-distortion optimization (RDO) to select the optimal coding mode. And RDO takes a great portion of intra encoding complexity.Moreover HEVC has stronger dependency on RDO than H.264/AVC. To reduce the computational complexity and to implement a real-time system,this paper presents two low-complexity RDO algorithms for HEVC intra prediction. The structure of RDO is simplified by the proposed rate and distortion estimators, and some hardware-unfriendly modules are facilitated. Compared with the original RDO procedure, the two proposed algorithms reduce RDO time by 46% and 64% respectively with acceptable coding efficiency loss.

Keywords: HEVC, intra prediction, RDO, video coding

1 Introduction

High Efficiency Video Coding (HEVC) [1] is the latest video coding standard developed by JCT-VC. It adopts a flexible quad-tree coding structure using variable sizes of coding unit (CU), prediction unit (PU) and transform unit (TU). For intra prediction, there are several new features. The coding unit can vary from 64x64 down to 8x8 and up to 35 prediction modes are defined in HEVC. The RDO result, rate-distortion Lagrangian cost (R-D cost), is the criterion of choosing the best mode and optimal sizes of CU, PU, TU. Since HEVC has much more prediction modes than H.264/AVC, its dependency on RDO becomes stronger. And the traditional RDO-off method is not integrated in HEVC test model (HM) because of poor performance. The hardware implementation of HEVC encoder becomes critical as a result of the sequential processing of high-complexity RDOand the data dependency of neighboring blocks. Therefore, an efficient algorithm and simplified RDO structure become essential.

The related works about computational complexity reduction of RDO can usually be classified into two categories: (a) Some low-possibility candidate

modes or sizes are excluded based on the spatial or temporal features; (b) Full search of all candidate modes is maintained, while the calculation of each R-D cost value is simplified by estimating the rate and distortion in a low complexity way. Many works proposed methods of category (a), e.g. [2] developed a mode filter by the SAD of original pixels. This work is dedicated in (b), for which some works have been proposed for H.264/AVC, but hardly any for HEVC. In [3], a rate estimator used Generalized Gaussian Distribution to model the coded coefficients. An efficient R-D cost estimation method was proposed in [4], which used the l_1-norm of coefficients and the transform-domain distortion. In [5], the coordinate of non-zero coefficient helped to model rate estimation. Distortion is approximated when the DCT coefficients is modeled by Laplacian distribution in [6]. However, these proposals can not be directly applied to HEVC because they are mostly based on TU size of 4x4, while TU can be 4x4 to 32x32 in HEVC.

The target of this work is to develop a low-complexity RDO algorithm which is suitable for hardware implementation of HEVC encoder. Therefore, it is necessary to consider about replacing some parts of the sequential RDO with low-complexity algorithms. This paper presents two efficient RDO algorithms based on quantized coefficients and transformed coefficients. For rate calculation, the real number of bits coded by CABAC is estimated. For distortion calculation, the distortion caused by quantization is used to estimate the actual distortion. Thus, some of the high-complexity modules in the conventional RDO can be simplified, which benefits the hardware implementation.

The rest of this paper is organized as follows. Section 2 introduces the brief structures of original RDO and the proposed ones. Section 3 and 4 present the proposed RDO algorithms based on quantized coefficients and transformed coefficients, respectively. Section 5 shows the experimental results, followed by the conclusion in Section 6.

2 RDO in HEVC

Fig.1.(a) illustrates the flowchart of original RDO based mode decision. The area in dashed line means R-D cost calculation, and the wide grey line shows the latency for rate and distortion. The rate is counted after the quantized transform pixels are entropy coded. And the distortion is calculated after the block is reconstructed and compared with the original pixels.

In HEVC intra prediction, the PU can vary from 64x64 to 4x4. For each PU, several candidate prediction modes are selected from all the 35 directional prediction modes by calculating the SATD based cost, which is defined by

$$J_{pred,SATD} = SATD + \lambda_{pred} \cdot B_{pred} \quad [7], \tag{1}$$

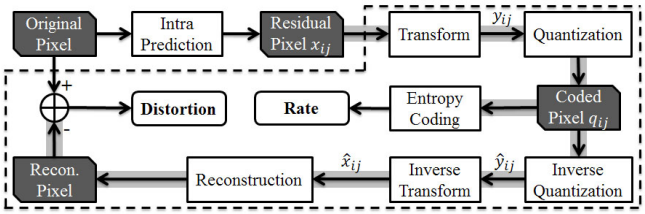
where $SATD$ is the sum of absolute difference of the Hadamard transformed coefficients, B_{pred} specifies bit cost of encoding the mode information.

Then the best prediction mode is selected from these candidates by calculating the R-D cost, which is defined by

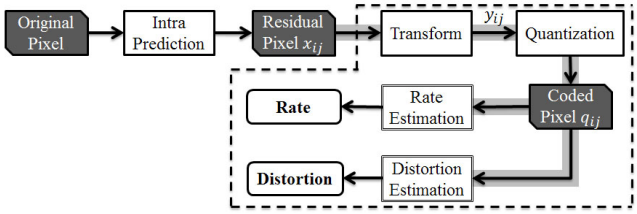
$$J_{mode} = SSE + \lambda_{mode} \cdot B_{mode} \quad [7], \tag{2}$$

where distortion SSE is the sum of square error between the original pixels and reconstructed pixels, B_{mode} specifies bit cost of encoding the whole block by CABAC. This cost is also the criterion of deciding the optimal CU size, PU size and TU size.

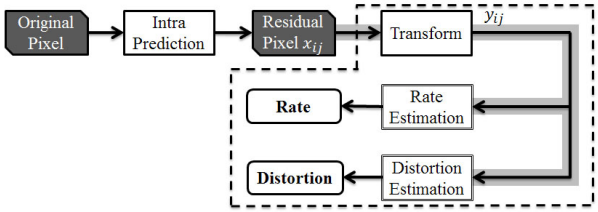
Even though only several number of candidates are selected to calculate the R-D cost, the procedure of rate-distortion optimization still occupies an essential portion in the computational complexity.



(a) RDO in HM



(b) Proposed RDO based on Quantized Coefficients



(c) Proposed RDO based on Transformed Coefficients

Fig. 1. The Flowchart of R-D cost Based Mode Decision

In Fig.1.(a), the successive modules including Transform (T), Quantization (Q), Inverse Quantization (IQ) and Inverse Transform (IT) would cause a strong data dependency, but relatively easy to be parallelized and pipelined in hardware. Whereas the module of Entropy Coding, in which coefficients are processed sequentially, is disadvantageous to hardware implementation.

The flowchart of the proposed RDO based on quantized coefficients is shown in Fig.1.(b). The module of entropy coding is replaced with a simple rate estimation module. The inverse quantization, inverse transform and pixel reconstruc-

tion are replaced with a low-complexity distortion estimation module. Therefore, the processing of RDO is simplified and the data dependency between blocks is alleviated, which brings a lot of convenience to hardware implementation.

The flowchart of the proposed RDO based on transformed coefficients is shown in Fig.1.(c). Compared with previous one, the module of quantization is also skipped. R-D cost calculation is further facilitated and the data dependency is further alleviated.

3 Proposed RDO based on Quantized Coefficients

3.1 Post-quantization Rate Estimation

The rate B_{mode} can be divided into two parts as

$$B_{mode} = B_{hdr} + B_{coef}, \tag{3}$$

where B_{hdr} is the number of header bits and B_{coef} means the number of coefficient bits.

The l_p-norm of the quantized transform coefficients $q_{ij}(i, j = 0..N-1)$ is defined by

$$E = \left(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |q_{ij}|^p \right)^{1/p}. \tag{4}$$

According to the residual coding syntax described in [8], the coefficient bits are coded from the information of non-zero map, greater-than-one map, sign, remaining level and so on. As the most significant part, the binarization of remaining level uses Golomb-Rice method, in which the rice parameter cRP may increase when scanning from the high frequency coefficients to low frequency coefficients. The prefix part of the binarization is derived by invoking the Truncated Rice binarization process, whose number of bins is $cRP + \lceil Level/2^{cRP} \rceil$. The suffix is derived using the k-th order Exponential Golomb (EGk) binarization, whose number of bins is $k + \lceil \log_2(Level/2^k + 1) \rceil$[9], where $k = cRP + 1$.

Based on these processes, we draw the conclusion that a small level would be coded into longer bins in lower frequency positions than that in higher frequency, and a large level would be coded into shorter bins in lower frequency positions than that in higher frequency. By adding the position information of non-zero coefficients into (4), we model the bit consumption E_{pos} as

$$E_{pos} = \left(\sum_{q_{ij} \neq 0} \left(|q_{ij}| + \theta \cdot |q_{ij}| \, (i+j) \right)^p \right)^{1/p}, \tag{5}$$

where θ is a balance parameter of the position information. For computation reduction, we choose $p = 1$. Taking the fact that most non-zero coefficients are 1 into consideration, E_{pos} is simplified as

$$E_{pos} = \sum_{q_{ij} \neq 0} \left(|q_{ij}| + \theta \cdot (i+j) \right). \tag{6}$$

Fig. 2 shows the relationship between the number of actual bits B_{coef} and E_{pos} when $\theta = 1$. There exists high correlation between the number of actual bits and E_{pos}, while the linearity is not perfect. And it is a problem that how to estimate the number of bits for different TU sizes. Based on the observation of Fig. 2, we propose a rate estimator designed for different TU sizes, which is formulated as

$$\hat{B}_{coef} = \alpha \cdot E_{pos}^{\beta} = \alpha \cdot \left(\sum_{q_{ij} \neq 0} (|q_{ij}| + \theta \cdot (i+j)) \right)^{\beta}, \tag{7}$$

where $\hat{B}_{coef}$ represents the estimated number of coefficient bits, α and β are model parameters which are different for different TU size N and different quantization parameter (QP). With proper $\alpha(QP, N)$, $\beta(QP, N)$ and $\theta(QP, N)$, the proposed estimator model will achieve accurate prediction.

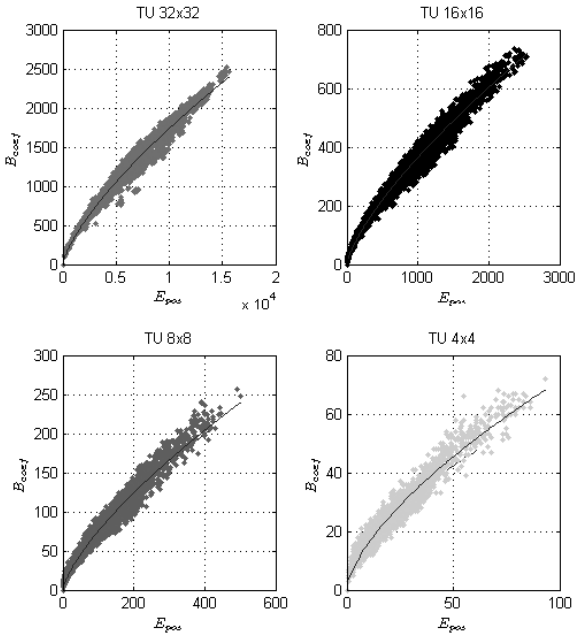


Fig. 2. The number of actual bits B_{coef} versus E_{pos}

Benefiting from the rate estimator, the module of Entropy Coding for R-D cost calculation is skipped. Since the estimator in (7) can be calculated by a look-up table, it is easy to be implemented in hardware. As $\hat{B}_{coef}$ is only calculated once for each block, a little additional complexity would be caused.

For intra prediction, the number of header bits B_{hdr} is usually much smaller than the coefficient bits. According to [8], the header bits of intra prediction mainly include the information about prediction mode, partition size, transform unit splitting and so on. And the prediction mode is coded based on the predicted modes inferred from the neighboring blocks. Since the calculation of header bits is not complex, we directly use the method in HEVC test model.

3.2 Post-quantization Distortion Estimation

As described in Section 2, the distortion is measured by the SSE between the original pixels and the reconstructed pixels. The data flows through modules of Intra Prediction, Transform, Quantization, Inverse Quantization, Inverse Transform and Reconstruction.

In forward transform, the transform coefficients y_{ij} are derived from the residual samples x_{ij}, which is formulated as

$$Y = T(X) = C_f X C_f^T \otimes E_f, \tag{8}$$

where C_f is the transform matrix and E_f represents the scaling matrix. Although HEVC uses more accurate C_f than H.264, the integer matrix is still different from the theoretical transform matrix. So the forward transform would cause some distortion, and the module of inverse transform is similar. We define the distortion caused by T and IT as D_T.

In quantization module, the transform coefficients y_{ij} are scaled into smaller quantized coefficients q_{ij}.The division brings distortion because of the right shift operation. Let D_Q be the distortion resulted from quantization.

Each coefficient would be clipped when the value exceeds the data range. The distortion resulted from clipping is denoted by D_C. Then the total distortion D is a co-effect result of D_T, D_Q, and D_C, which can be formulated as

$$D = G(D_T, D_Q, D_C). \tag{9}$$

However, G is not same as the sum function because of the counteraction. According to [4], the transform-domain distortion is close to the actual spatial-domain distortion D. That is to say, D_Q takes the majority part of D.

The conventional quantization can be formulated as

$$q_{ij} = (y_{ij} * f[QP\%6] + offset) \gg iQBits \quad [7], \tag{10}$$

where f is the scaling parameter related to QP, $iQBits$ is the number of shifted bits related to QP and TU size. And we define the scaled coefficients $s_{ij} = y_{ij} * f[QP\%6] + offset$.

In HEVC, the default quantization module is Rate-Distortion Optimized Quantization (RDOQ) [7], in which the quantized level is selected from three possible values by calculating each cost. The three possible quantized values are $0, l_{ij}^{floor}, l_{ij}^{ceil}$, defined by

$$l_{ij}^{ceil} = s_{ij} \gg iQBits, \quad l_{ij}^{floor} = l_{ij}^{ceil} - 1. \tag{11}$$

We calculate the difference between the scaled coefficient and the quantized coefficient, noted by d_{qs}, as

$$d_{qs_{ij}} = |s_{ij} - q_{ij} \ll iQBits| \,. \tag{12}$$

And the total scaled quantization distortion D_{qs} can be derived as

$$D_{qs} = \sum_{i,j} {d_{qs}}_{ij}^{2} = \sum_{i,j} (s_{ij} - q_{ij} \ll iQBits)^2. \tag{13}$$

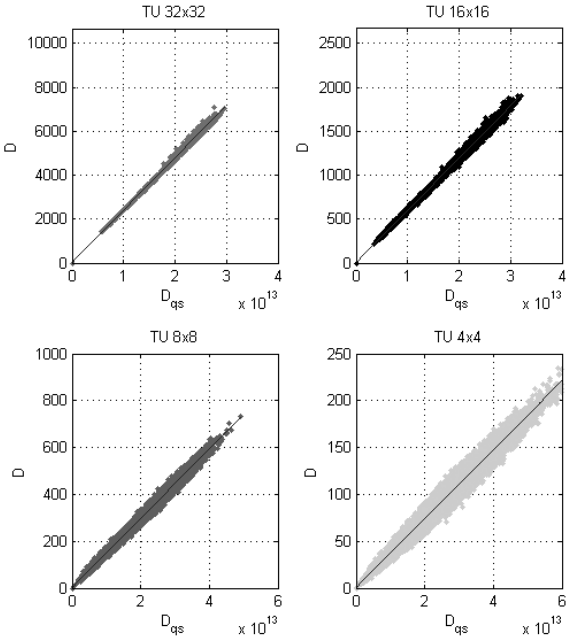


Fig. 3. The real distortion D versus D_{qs}

Fig.3 shows the relationship between the real distortion D and the scaled quantization distortion D_{qs}, which possesses strong linearity. Since inverse quantization is formulated as

$$\hat{y}_{ij} = (q_{ij} * scale + iAdd) \gg iShift, \tag{14}$$

where *scale* is a scaling parameter correlates with the scaling parameter in quantization procedure, $iAdd$ is an offset and $iShift$ is the number of shifted bits related to TU size, we model the estimated transform-domain distortion $\hat{D}_Q$ as

$$\hat{D}_Q = \eta \cdot D_{qs} \gg 2(k - \log_2 N), \tag{15}$$

where η is the parameter caused by *scale*, k is the parameter caused by shifting and N is the TU width. Finally, the distortion estimator is formulated as

$$\hat{D} = \hat{D}_Q = \mu \cdot D_{qs} \cdot N^2, \tag{16}$$

where $\hat{D}$ represent the estimated distortion and μ ($\mu = \eta \gg 2k$) is the model parameter related to QP. With proper $\mu(QP)$, the total distortion would be estimated accurately.

With the help of distortion estimator, inverse quantization, inverse transform and pixel reconstruction can be skipped for R-D cost calculation. And reconstruction is only needed for the best mode. Therefore, the number of pixel processing steps is decreased. And in hardware implementation, the low-complexity estimator would hide the data dependency.

4 Proposed RDO based on Transformed Coefficients

Although the algorithm described in Section 3 is much simpler than the original RDO, the module of quantization still brings much complexity. Since our target is to develop a RDO algorithm suitable for hardware, we tried to further simplify RDO based on the transformed coefficients, as shown in Fig.1.(c).

4.1 Pre-quantization Rate Estimation

From equation (10), the procedure of conventional quantization is actually a division, which can be formulated as

$$q_{ij} = \frac{|y_{ij}| + r}{\Delta}, \tag{17}$$

where the divisor $\Delta = 2^{iQBits}/f[QP\%6]$, and $r = offset/f[QP\%6]$ (in HM $r = \Delta/3$). And we define a threshold value function $T[k]$ as

$$T[k] = k \cdot \Delta - r \quad (k \geq 1). \tag{18}$$

The analysis of rate is similar to the post-quantization rate estimation model described in Section 3.1. The transformed coefficients with the absolute value smaller than $T[1]$ would become 0 after quantization. So these coefficients have no effect on the final rate and can be ignored in the rate estimation model. Take this fact into consideration, we model the bit consumption E_{pos} as

$$E_{pos} = \sum_{|y_{ij}| \geq T[1]} \left(|y_{ij}| + \theta \cdot (i+j) \right). \tag{19}$$

Similarly, the estimated number of coefficient bits $\hat{B}_{coef}$ is formulated as

$$\hat{B}_{coef} = \alpha \cdot E_{pos}^{\beta} = \alpha \cdot \left(\sum_{|y_{ij}| \geq T[1]} \left(|y_{ij}| + \theta \cdot (i+j) \right) \right)^{\beta}, \tag{20}$$

where $\alpha(QP, N)$, $\beta(QP, N)$ and $\theta(QP, N)$ are model parameters. The experimental results shows that the modified model can still estimate the rate accurately.

4.2 Pre-quantization Distortion Estimation

Since the distortion is mainly caused by quantization, the final distortion is not easy to be estimated without quantization. It is considered to calculate the real distortion for the majority of coefficients and discard some small portion of coefficients. It is observed that the quantized coefficients with smaller absolute values have much higher possibilities than those with higher absolute values. A comparator-based distortion estimator is proposed.

When the absolute value of transformed coefficient y_{ij} is smaller than $T[k]$, the actual distortion would be calculated after the comparators find out which quantized value q_{ij} it would be. Otherwise, an expectation of distortion would be added instead of the actual distortion. In this case, $\lceil \log_2(k+1) \rceil$ comparators are needed. Then the difference between the transformed coefficient y_{ij} and the inverse quantized coefficient $\hat{y}_{ij}$, noted by d_q, is formulated as

$$d_{q_{ij}} = \begin{cases} |y_{ij} - \hat{y}_{ij}|, & |y_{ij}| \leq T[k] \\ E[d_q], & otherwise \end{cases} \tag{21}$$

$$d_{q_{ij}}^{\ 2} = \begin{cases} |y_{ij} - \hat{y}_{ij}|^2 \\ E[d_q^2] \end{cases} = \begin{cases} (y_{ij} - \Delta q_{ij})^2, & |y_{ij}| \leq T[k] \\ \Delta^2/9, & otherwise \end{cases} \tag{22}$$

The expectation $E[d_q^2]$ would become $\Delta^2/9$ when using the plain distribution. Other distributions can also be considered, such as the Laplacian distribution as described in [6].

Table 1. The Arrangement of k

i', j'	0	1	2	3
0	15	7	7	3
1	7	7	3	3
2	7	3	3	1
3	3	3	1	1

To estimate the distortion accurately, the threshold value $T[k]$ is arranged with different k at different positions, as shown in Table 1. Position coordinate $i' = 4i/N$, $j' = 4j/N$, where N is the TU width. It is basically ensured that more than 90% of coefficients satisfy $|y_{ij}| \leq T[k]$ when $QP = 22$ and their actual distortions are calculated. The needed comparators would cost much less than the original multipliers. The total quantization distortion D_q can be derived by

$$D_q = \sum_{i,j} d_{q_{ij}}^{\ 2}. \tag{23}$$

Similarly to equation (16), the distortion estimator is formulated as

$$\hat{D} = \hat{D}_Q = \mu \cdot D_q \cdot N^2, \tag{24}$$

where $\hat{D}$ represent the estimated distortion and μ is the model parameter related to QP. With proper $\mu(QP)$, the total distortion would be estimated accurately.

5 Experimental Results

5.1 Model Parameter Training

The model parameters are obtained under 4 QP values: 22, 27, 32 and 37 [10]. The rate estimation model parameters $\alpha(QP, N)$, $\beta(QP, N)$ and $\theta(QP, N)$ mentioned in Section 3.1 and 4.1 are trained by Levenberg-Marquardt algorithm, a nonlinear regression algorithm based on least squares as described in [11]. The distortion estimation model parameters $\mu(QP)$ mentioned in Section 3.2 and 4.2 are computed by the simple linear regression method.

Two sequences are used to train the model parameters. For the high resolution sequences in class A and B, one set of parameters is obtained by a randomly selected sequence *BQTerrace*. For the low resolution sequences in class C, D and E, another set of parameters is obtained by *BasketballPass*.

5.2 The Performance of Proposed Algorithm

The proposed R-D cost estimation algorithms are integrated with HM 8.0. In the experiment, all test sequences listed in [10] except the two sequences used to obtain the model parameters are encoded under configuration of all-intra, with QP of 22, 27, 32 and 37. The configuration of RDOQ is turned off. The coding efficiency and complexity is compared between our proposed RDO and that in HM 8.0. The coding efficiency is measured by bit-rate difference and PSNR difference using Bjontegard's method [12]. And the complexity reduction is measured by time reduction ratio of R-D cost calculation ΔT_{RDO}, which is defined by

$$\Delta T_{RDO} = \frac{T_{ProposedRDO} - T_{OriginalRDO}}{T_{OriginalRDO}} \times 100\%. \tag{25}$$

The experimental result of the proposed RDO based on quantized coefficients is shown in Table 2. The proposed algorithm causes an average quality loss of $0.109dB$ and increased BD-rate of 1.93%, which is acceptable and demonstrates that the R-D cost estimation is relatively accurate. Meanwhile it achieves an average of 46% RDO time saving. Almost the same complexity reduction can be achieved regardless of the resolution or the sequence characteristics, which indicates the potential to apply the proposed algorithm into real-time systems.

The experimental result of the proposed RDO based on transformed coefficients is shown in Table 3. The BD-PSNR decreases by $0.135dB$ and BD-rate increases by 2.41% in average. While about 64% of RDO time is reduced. It can achieve more RDO time saving than the first test because that the module of quantization is also skipped. Considering its contribution of complexity reduction, the coding efficiency loss is acceptable.

Note that the contribution of the proposed algorithms to hardware implementation should be larger than that reflected by ΔT_{RDO}. This is because the operations simplified by the proposed algorithm are hardware-unfriendly parts including entropy coding and reconstruction that involves a long pipeline latency. The remaining complexity of RDO is mainly from transform and estimation which are relatively easy for parallel processing and pipelining.

Table 2. Performance of Proposed RDO based on Quantized Coefficients

Class	Sequence	BD-rate (%)	BD-psnr (dB)	ΔT_{RDO} (%)
A (4K)	Traffic	1.75	-0.095	-45.9
	PeopleOnStreet	1.91	-0.108	-46.2
B (1080p)	Kimono	1.42	-0.049	-43.0
	ParkScene	2.15	-0.093	-46.4
	Cactus	1.92	-0.073	-46.7
	BasketballDrive	2.40	-0.105	-47.5
C (WVGA)	RaceHorses	1.83	-0.120	-46.3
	BQMall	2.28	-0.143	-47.2
	PartyScene	2.30	-0.184	-49.0
	BasketballDrill	1.32	-0.064	-45.0
D (WQVGA)	RaceHorses	1.62	-0.115	-47.7
	BQSquare	2.31	-0.202	-48.7
	BlowingBubbles	1.98	-0.123	-47.5
E (720p)	Vidyo1	2.01	-0.096	-43.8
	Vidyo3	1.74	-0.087	-42.8
	Vidyo4	1.91	-0.084	-43.2
Total Average		**1.93**	**-0.109**	**-46.1**

Table 3. Performance of Proposed RDO based on Transformed Coefficients

Class	Sequence	BD-rate (%)	BD-psnr (dB)	ΔT_{RDO} (%)
A (4K)	Traffic	2.16	-0.116	-58.6
	PeopleOnStreet	2.61	-0.147	-60.3
B (1080p)	Kimono	2.23	-0.077	-57.4
	ParkScene	2.37	-0.102	-59.0
	Cactus	2.52	-0.095	-59.4
	BasketballDrive	2.54	-0.157	-66.6
C (WVGA)	RaceHorses	2.22	-0.145	-65.6
	BQMall	2.71	-0.169	-69.7
	PartyScene	2.80	-0.204	-69.4
	BasketballDrill	1.75	-0.086	-67.3
D (WQVGA)	RaceHorses	1.98	-0.138	-68.8
	BQSquare	2.93	-0.227	-70.2
	BlowingBubbles	2.44	-0.154	-69.1
E (720p)	Vidyo1	2.72	-0.129	-59.3
	Vidyo3	2.28	-0.114	-59.9
	Vidyo4	2.27	-0.100	-62.6
Total Average		**2.41**	**-0.135**	**-63.9**

6 Conclusion

We proposed two low-complexity rate-distortion optimization algorithms for HEVC intra prediction with proper rate and distortion estimation models based on quantized coefficients and transformed coefficients, respectively. They facilitate some hardware-unfriendly modules in the original RDO, and the data dependency can be alleviated in hardware implementation. In software simulation, the computational complexity of RDO is reduced by 46% and 64% respectively with acceptable coding efficiency loss. Furthermore, these algorithms can also be used in inter prediction or combined with mode filtering algorithms for HEVC.

7 Acknowledgement

This work is partly supported by KAKENHI and STARC.

References

1. Sullivan, G.J., Ohm, J.R., Han, W.J., Wiegand, T., Wiegand, T.: Overview of the high efficiency video coding (hevc) standard. Circuits and Systems for Video Technology, IEEE Transactions on 22(12), 1649-1668 (Dec 2012)
2. Sun, H., Zhou, D., Goto, S.: A low-complexity hevc intra prediction algorithm based on level and mode filtering. In: Multimedia and Expo (ICME), 2012 IEEE International Conference on. pp. 1085-1090 (Apr 2012)
3. Zhao, X., Sun, J., Ma, S., Gao, W.: Novel statistical modeling, analysis and implementation of rate-distortion estimation for h.264/avc coders. Circuits and Systems for Video Technology, IEEE Transactions on 20(5), 647-660 (May 2010)
4. Tu, Y.K., Yang, J.F., Sun, M.T.: Efficient rate-distortion estimation for h.264/avc coders. Circuits and Systems for Video Technology, IEEE Transactions on 16(5), 600-611 (May 2006)
5. Wang, Q., Zhao, D., Gao, W., Ma, S.: Low complexity rdo mode decision based on a fast coding-bits estimation model for h.264/avc. In: Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on. pp. 3467-3470 Vol. 4 (May 2005)
6. Moon, J.M., Moon, Y.H., Kim, J.H.: A computation reduction method for rdo mode decision based on an approximation of the distortion. In: Image Processing, 2006 IEEE International Conference on. pp. 2481-2484 (2006)
7. JCT-VC: High efficiency video coding (hevc) test model 8 encoder description. JCTVC-J1002 (Jul 2012)
8. JCT-VC: High efficiency video coding (hevc) text specification draft 8. JCTVC-J1003_d7 (Jul 2012)
9. Bardone, D., Carotti, E., De Martin, J.: Adaptive golomb codes for level binarization in the h.264/avc frext lossless mode. In: Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on. pp. 287-291 (Dec 2008)
10. JCT-VC: Common test conditions and software reference configurations. JCTVC-E700 (Jan 2011)
11. Seber, G., Wild, C.: Nonlinear Regression. Wiley Series in Probability and Statistics, Wiley (2003)
12. BJONTEGARD, G.: Calculation of average psnr differences between rd-curves. ITU-T VCEG-M33 (Apr 2001)