# A FAST RATE-DISTORTION OPTIMIZATION ALGORITHM FOR H.264/AVC

*Zhenyu Wei* *King Ngi Ngan*

Electronic Engineering Department
The Chinese University of Hong Kong, Hong Kong, P.R.China

## ABSTRACT

In H.264/AVC video coding standard, Rate distortion optimization (RDO) is a very efficient technique to decide the coding mode of the macroblock and improves the coding performance greatly. But at the same time, RDO introduces a great deal of computation complexity because it needs to do the transform and entropy coding to get accurate distortion and bit-rate to calculate the Lagrange cost. In order to avoid the expensive computational cost, in this paper , a simple and accurate bit estimation model is proposed and used in RDO scheme. Combined with distortion measure method in transform domain and the fast intra mode RDO method, a lot of computation of RDO is reduced. Experimental results prove that our proposed fast RDO algorithm can reduce about 60% total encoding time and save about 76% computation time of RDO module with little degradation in coding performance.

***Index Terms***— H.264/AVC, rate-distortion, bit estimation

## 1. INTRODUCTION

The H.264/AVC is the newest video coding standard recommended by ITU-T and MPEG[1]. Compared with all existing video coding standards, H.264 can achieve superior performance by using many advanced techniques. There are total 7 different inter modes, 9 intra $4\times4$ modes and 4 intra $16\times16$ modes. An efficient rate distortion optimization (RDO) technique is employed in H.264 to choose the best mode for each MB. But it needs to do transform and entropy coding for each mode to get the real distortion and bit rate, so a lot of computational cost is introduced. Therefore, if we can find a good method to predict the bit rate and distortion accurately, a lot of unnecessary computation will be reduced, such as entropy coding, inverse transform etc.

In order to solve this problem, several methods can be used. Because of conservation of energy, we can calculate the distortion in transform domain. But for bit-rate estimation, because entropy coding in H.264/AVC is more complex than before, it is not easy to get accurate bit-rate by using lookup tables for runs and levels. Some previous literatures have already studied the bit-estimation problem. [2]and [3] proposed rate-distortion models based on quantizer-domain. The $\rho$-domain rate model is appeared in [4]where a linear relationship between rate and nonzero quantized DCT coefficients is investigated. But unfortunately, these methods are designed for rate control originally and more suitable for bit-estimation in frame level, but not in macroblock level. [5] proposed a bit-rate estimation model, but because the parameters of this model are fixed and not self-adaptive, the estimation accuracy is limited. The method in [6] mainly focuses on the inter mode decision, but in H.264/AVC, intra mode decision occupies the main part of RDO computation, so the speed of this method is not very remarkable.

By studying the entropy coding method in H.264/AVC, an accurate bit-rate estimation model is proposed in this paper. The parameters in this model are self-adaptive to guarantee the precision of estimation. A fast intra mode RDO method is designed for reducing the unnecessary RDO calculation in intra mode. We also calculate the distortion in transform domain to avoid the MB reconstruction process. By using these methods, a lot of RDO computation is saved while keeping the performance of RDO very well. Our algorithm also can be combined with any other fast motion estimation, fast inter/intra mode decision methods to speedup the coding time further.

The rest of paper is organized as follows: section 2 briefly introduces the RDO in H.264/AVC. Section 3 illuminates our proposed fast RDO algorithm. Experimental results and conclusion will be given in section 4 and 5.

## 2. RDO IN H.264/AVC

To choose the best macroblock mode, H.264/AVC encoder computes the RDcost(Rate Distortion cost)for each possible mode and choose the mode which has minimum RDcost as the best mode.In the view of information theory, the RDO process can be described as looking for the minimal required rate $R$ to achieve a given distortion $D$. RDO uses Lagrange Multiplier method to get the optimal solution. The cost function is shown as follows.

$$RDcost = SSE + \lambda_{mode} \times R \tag{1}$$

Where $SSE$ is the sum of squared error between the original block and the reconstructed block, $\lambda_{mode}$ is the Lagrange

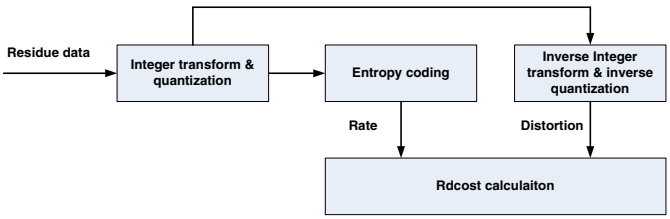


**Fig. 1**. RDO Computation

multiplier, *R* represents the bit number consumed for coding this block. As shown in Fig.1,in order to compute the RDcost for each mode, the block needs to be encoded and decoded to get the bit-rate and the distortion, so a lot of operations of forward/ inverse transform and entropy coding are repeatedly performed. The computational cost of RDO is very high.

In H.264/AVC,$RDO_{off}$ option is also supported. The cost function of this mode is:

$$RDcost = SAD + \lambda_{mode} \times R \tag{2}$$

Where *SAD* is the sum of absolute difference between the original block and the reconstructed block,R stands for the bits for coding motion vector and other side information. The entropy coding and decoding process are not needed in this mode, so the computation complexity is much lower than $RDO_{on}$ mode, but at the same time, the coding performance is also much worse than $RDO_{on}$ mode.

## 3. PROPOSED FAST RDO ALGORITHM

Although RDO improves the coding performance greatly, the computation complexity is too high to be acceptable for real-time application. It is very necessary to design a fast RDO algorithm to reduce the computational cost of this module.In this section, a fast RDO approach is proposed, which is based on the following three techniques.

### 3.1. Precise Bit-rate Estimation Model

In order to predict the bit-rate accurately, we should analyze the components of bits for coding one block. Total bits for coding one block can be expressed by the following formula.

$$R_{total} = R_{coef} + R_{header} + R_{motion} \tag{3}$$

Where $R_{header}$ stands for the bits used for coding head information,such as mode type, coded block pattern(CBP).$R_{motion}$ is the bit number of motion information,including motion vector, reference frame index etc. The bit number of these two parts both can be obtained through look-up tables easily.

$R_{coef}$ represents the bits used for coding quantized coefficients.It is not practical to get the bits of this part by using look-up table, because the entropy coding method in H.264/AVC is much more complex than before. Through studying the entropy coding algorithm in H.264/AVC, it is found that consumed bits for coding quantized coefficients are related to three factors: *N* (the number of nonzero quantized transform coefficients), *Z* (summation of run-before), and *E* (summation of absolute value of total quantized coefficients). We can model the bits for coding quantized coefficients by following function:

$$R_{coef} = \alpha \times N + \beta \times Z + \gamma \times E \tag{4}$$

Where $\alpha\ \beta\ \gamma$ are three parameters of this function and they can be computed and updated by using linear regression method.

$$\alpha = T_\alpha / F \quad \beta = T_\beta / F \quad \gamma = T_\gamma / F \tag{5}$$

Let:

$$\begin{aligned}
S_{nr} &= \sum_{k=1}^{n} N_k R_k \quad S_{zr} = \sum_{k=1}^{n} Z_k R_k \quad S_{er} = \sum_{k=1}^{n} E_k R_k \\
S_{nn} &= \sum_{k=1}^{n} N_k N_k \quad S_{zz} = \sum_{k=1}^{n} Z_k Z_k \quad S_{ee} = \sum_{k=1}^{n} E_k E_k \\
S_{nz} &= \sum_{k=1}^{n} N_k Z_k \quad S_{ze} = \sum_{k=1}^{n} Z_k E_k \quad S_{ne} = \sum_{k=1}^{n} N_k E_k
\end{aligned} \tag{6}$$

Where n stands for the number of encoded macroblocks in past,$N_k\ Z_k\ E_k$ are the values of the three corresponding components of the $k_{th}$ MB, $R_k$ is the real consumed bits for coding the quantized coefficients of the $k_{th}$ MB.
Then:

$$\begin{aligned}
T_\alpha = {} & S_{nr}(S_{zz}S_{ee} - S_{ze}^2) - S_{zr}(S_{ee}S_{nz} - S_{ne}S_{ze}) \\
& + S_{er}(S_{nz}S_{ze} - S_{zz}S_{ne})
\end{aligned} \tag{7}$$

$$\begin{aligned}
T_\beta = {} & S_{nr}(S_{ne}S_{ze} - S_{ee}S_{nz}) - S_{zr}(S_{nn}S_{ee} - S_{ne}^2) \\
& + S_{er}(S_{nz}S_{ne} - S_{nn}S_{ze})
\end{aligned} \tag{8}$$

$$\begin{aligned}
T_\gamma = {} & S_{nr}(S_{nz}S_{ze} - S_{zz}S_{ne}) - S_{zr}(S_{ne}S_{ze} - S_{nn}S_{ze}) \\
& + S_{er}(S_{nn}S_{zz} - S_{nz}^2)
\end{aligned} \tag{9}$$

$$\begin{aligned}
F = {} & S_{nn}S_{zz}S_{ee} - S_{nn}S_{ze}^2 - S_{zz}S_{ne}^2 - S_{ee}S_{nz}^2 \\
& + 2S_{nz}S_{ne}S_{ze}
\end{aligned} \tag{10}$$

Although the above regression function looks very complex, in fact, the $(k+1)_{th}$ group of the parameters $(\alpha\ \beta\ \gamma)$ can be gotten easily by the $k_{th}$ group of the parameters, so little extra computation cost is introduced in this regression process.

In Fig.2(test sequence is foreman.qcif) X-Axis is the actual bits, Y-Axis is our estimated bits. Green line is 45 degree line. We can see that the most of dots lie on the narrow range around the green line.It means that our bits estimation model can predict real bits accurately, then a lot of entropy coding computation is saved.

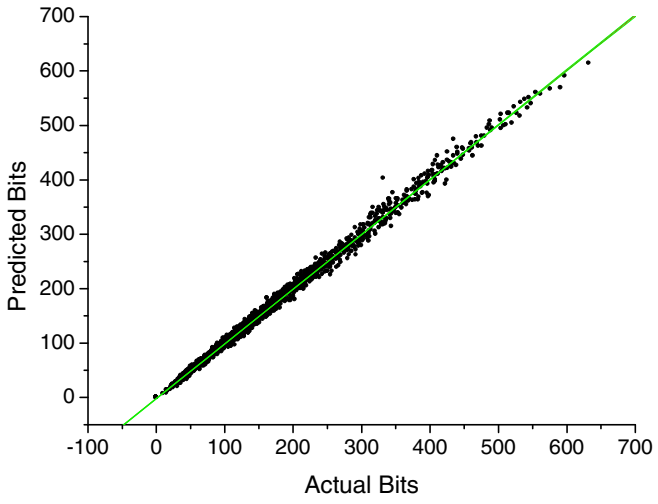


**Fig. 2**. Comparison Between Actual Bits And Predicted Bits

### 3.2. Fast Intra Mode RDO Method

H.264 allows intra modes to be used in inter frames.Therefore, when doing mode selection, not only inter modes but also intra modes need to be checked. We can regard the $4\times 4$ block as the count unit of RDO operation because transform, quantization and entropy coding are all use $4\times4$ as basic unit. After analysis of H.264/AVC reference software JM10.1[7], It is found that total 128 RDO operations are needed for inter mode decision[8].

For intra mode decision, because H.264/AVC combines luma and chroma components together to perform RDO operation. The RDcost also includes luma and chroma components. It means that the best combination of luma mode and chroma mode can give minimum RDcost. When a chroma mode is fixed, all the luma intra modes must be checked to calculate the RDcost.Because there are 4 chroma modes, the total number of RDO operations for intra mode decision is 640[8], which is much higher than that of inter mode decision. So, the intra mode selection occupies the majority of RDO computation and it is very necessary to reduce the intra mode RDO computation.

It is observed that luma and chroma components are approximatively independent when performing intra mode selection, so we can perform RDO computation for chroma and luma components respectively to choose the best chroma mode and luma mode.Experimental results show that a lot of RDO operations are reduced and the coding performance is influenced slightly. It needs to be clarified that this fast intra RDO method differs from other traditional fast intra mode selection algorithms because it does not reduce the mode types which will be checked.So, this method can be integrated with other fast mode selection algorithms to achieve much faster encoding speed.

### 3.3. Distortion Computation in Transform Domain

In original H.264 algorithm, we need to reconstruct the frame to calculate the distortion between the original frame and the reconstructed frame. Because the integer transform in H.264/AVC is a kind of orthogonal transform and distortion is measured by SSE (sum of squared error) in RDO, according to conversation of energy, the distortion in transform domain must be equal to that in spatial domain. We can calculate the distortion in transform domain. Then, a lot of de-quantization and inverse ICT computation can be avoided. This idea was also described in literature [5]and [6].

## 4. EXPERIMENTAL RESULTS

Our proposed algorithm is integrated within H.264 software JM10.1[7]. The system hardware is a PC with 2.8GHz Intel P4 CPU and 1Gb memory. Several test sequences with 150 frames are chosen and covering high, mild and low motion, as well as QCIF and CIF resolution. Quantization parameters are set as 28, 32, 36 and 40, RD optimization is enabled and entropy coding uses CAVLC, coding mode is IPPP...

Table.1 shows the results of $RDO_{on}$ vs. $RDO_{off}$ and $RDO_{on}$ vs. proposed method. In order to evaluate coding efficiency, several terms are compared.The difference of PSNR($\Delta PSNR$) and the bit-rate($\Delta BR$) are calculated by using the RD-curve fitting method in [9]. $\Delta T_{total}$ is the saving ratio of total encoding time. $\Delta T_{RDO}$ stands for the coding time saving ratio of the RDO module and calculated by following equation:

$$\Delta T_{RDO} = -\frac{T_{proposed} - T_{RDO_{on}}}{T_{RDO_{off}} - T_{RDO_{on}}} \times 100\% \qquad (11)$$

From the Table.1, we can see that when RDO is turned off, speed is much faster, but the coding performance is degraded greatly. The PSNR is decreased about 0.23 dB averagely and bit rate is increased 5.62% equivalently.

Our proposed method can keep coding performance very well. Fig.3 shows the RD curve of three algorithms.From Fig.3, we can see that RD curves of $JMRDO_{on}$ and our proposed method almost overlap each other, that means the performance of our approach is very close to $JMRDO_{on}$. Compared with JM 10.1, PSNR of our method is decreased only 0.03 dB averagely and bit rate is increased 0.82% equivalently. At the same time,our algorithm can save about 60% total encoding time and about 75% computational cost of RDO part.

## 5. CONCLUSION

Rate distortion optimization (RDO) plays a vital role in H.264/AVC standard and can improve the coding performance greatly. However, this process is very time-consuming.In this paper,a fast RDO algorithm is proposed. There are two main contributions in this paper: one is to propose an accurate bit-rate

**Table 1**. Experimental Results.

| Format | Sequence | JM10.1(RDO OFF) | | | Proposed Method | | | |
|---|---|---|---|---|---|---|---|---|
| | | $\Delta BR$ [%] | $\Delta PSNR$ [dB] | $\Delta T_{total}$ [%] | $\Delta BR$ [%] | $\Delta PSNR$ [dB] | $\Delta T_{total}$ [%] | $\Delta T_{RDO}$ [%] |
| QCIF | Foreman | 4.20 | -0.19 | -75.87 | 0.18 | -0.01 | -56.33 | -74.26 |
| | Carphone | 2.23 | -0.10 | -76.66 | -0.48 | 0.02 | -56.13 | -73.22 |
| CIF | Foreman | 5.74 | -0.24 | -75.44 | 0.23 | -0.01 | -56.17 | -74.46 |
| | Bus | 6.55 | -0.29 | -75.30 | 0.89 | -0.04 | -58.25 | -77.36 |
| | Paris | 5.12 | -0.24 | -78.74 | 1.50 | -0.07 | -60.10 | -76.32 |
| | Hallmonitor | 10.44 | -0.37 | -78.50 | 2.02 | -0.07 | -58.58 | -74.63 |
| | Mobile | 4.45 | -0.17 | -77.88 | 0.09 | -0.00 | -61.25 | -78.66 |
| | Football | 6.49 | -0.28 | -73.00 | 2.07 | -0.09 | -54.71 | -74.94 |
| | Coastguard | 5.32 | -0.17 | -75.06 | 0.92 | -0.03 | -57.63 | -76.78 |
| Average | | 5.62 | -0.23 | -76.27 | 0.82 | -0.03 | -57.68 | -75.63 |

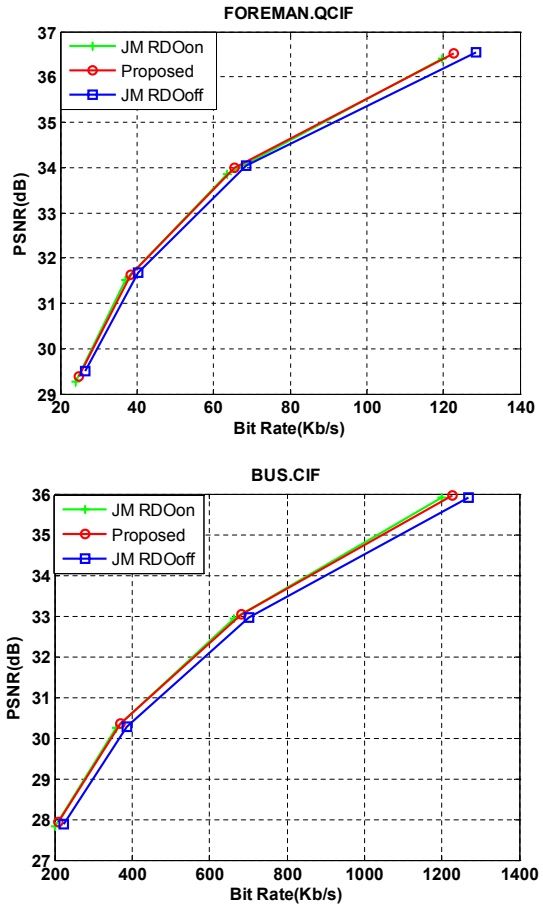


**Fig. 3**. Rate-Distortion Curve

estimation model, based on this model, a lot of entropy coding computation is saved. Another one is to design a fast intra mode RDO scheme. Experiment results prove that our proposed fast RDO algorithm can reduce about 60% total encoding time and saves about 76% compuation time of RDO module with little degradation in coding performance. Our algorithm can be combined with other fast motion estimation algorithms or fast inter/intra mode decision algorithms to improve the speed of encoder further. It is helpful for the real-time application of the H.264 encoder and useful for the low-power applications of video coding.

## 6. REFERENCES


[1] H.264 Standard, "ITU-T recommendation H.264 advanced video coding for generic audiovisual services," 2005.

[2] Tihao Chiang and Ya-Qin Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. CSVT*, vol. 7, pp. 246–250, Feb. 1997.

[3] J.Ribas-Corbera and S. Lei, "A new rate control scheme using quadratic rate distortion model," *IEEE Trans. CSVT*, vol. 9, pp. 172–185, Feb. 1999.

[4] Zhihai He and S.K. Mitra, "A linear source model and a unified rate control algorithm for dct video coding," *IEEE Trans. CSVT*, vol. 12, pp. 970–982, Nov. 2002.

[5] Q Chen and Y He, "A fast bits estimation method for rate-distortion optimization in H. 264/AVC," in *Proceeding of Picture Coding Symp.*, 2004.

[6] YK Tu, JF Yang, and MT Sun, "Efficient rate-distortion estimation for h. 264/avc coders," *IEEE Trans. CSVT*, vol. 16, pp. 600–611, May 2006.

[7] "JVT model JM10.1," downloaded from http://iphome.hhi.de/suehring /tml/download/old_jm/.

[8] Byeungwoo Jeon, "Fast mode decision to jvt," *Joint Video Team(JVT) Docs. JVT-J033*, Dec 2003.

[9] G. Bjontegaard, *Calculation of average PSNR differences between RD-curves*, 13th VCEG-M33 Meeting, Austin, Texas, USA, April 2001.