Adaptive Rate Control in Frame-layer for Real-time H.264/AVC

Myoung-Jin Kim, Kyoung-Hwan Kim, and Min-Cheol Hong
School of Electronic Engineering
Soongsil University
Sangdo5-dong Dongjak-gu, Seoul, 156-743, Korea
webzealer@ssu.ac.kr, dunamis@vipl.ssu.ac.kr, mhong@ssu.ac.kr

Abstract — Rate control is a critical issue in H.264/AVC video coding standard. The purpose of this paper is to improve allocation of the number of bits without skipping the frame by accurately estimating the target bits in H.264/AVC rate control. In our scheme, we propose an enhancement to the target frame rate based H.264/AVC bit allocation method. The enhancement is by using a frame complexity estimation to improve the existing mean absolute difference (MAD) based complexity measure. Bit allocation to each frame is not just computed by target frame rate but also adjusted by a combined frame complexity measure. To prevent an undesirable buffer overflow or underflow in short of channel bandwidth, the computed quantization parameter (QP) for the current frame is adjusted based on actual encoding results at that point. The objective of QP and adjustment is to produce bits as close to the target frame as possible, which is especially important for low bandwidth based real-time applications.

Simulation results show that the H.264 encoder, using our proposed rate control scheme, obtains significant improvement for the mismatch ratio of target bits and actual bits, achieves a 0.17 dB average PSNR improvement, achieves a similar or smaller PSNR deviation, and achieves time saving of 71% when compared to the JM 12.1 rate control algorithm.

Keywords — H.264/AVC, MAD, peak signal-to-noise ratio (PSNR), Quantization Parameter (QP)

1. Introduction

H.264/AVC is the latest international video coding standard developed by Joint Video Team (JVT) of ISO Motion Picture Expert Group (MPEG) and ITU-T Video Coding Expert Group (VCEG), in order to provide an enhanced video coding standard [1–5]. This is mainly intended for video transmission in all areas where bandwidth or storage capacity is limited (e.g. video telephony or video conferencing over mobile channels and devices), supplying an enhanced coding efficiency and an improved network adaptation [6]. Since many target applications concern video transmission over time-varying bandwidth channels, we need to control bit rate algorithms that allow modifying coding parameters according to channel’s variations.

Thus a good rate control scheme is highly desirable for H.264 which must be both accurate and computationally efficient. One fundamental problem in the encoder design is the selection of quantization parameter (QP) to maximize visual quality under constraints imposed by the computational complexity and bandwidth.

H.264 encoder employs more complicated approaches in the coding procedure. One of the important approaches is the utilization of rate distortion optimization (RDO), but this imposes a big problem for rate control in H.264, which is the well-known chicken-and-egg dilemma in the RDO process [7–9]. To perform RDO, QP should be first determined by using the mean absolute difference (MAD) of the current frame and/or MB. However, in order to perform rate control, QP can only be obtained according to the coding complexity and number of target bits that are calculated by motion compensated residues after RDO mode decision. To resolve this dilemma, Li et al. [10] presented a linear model to predict MAD and adopted a fluid flow traffic model to allocate target bit rate for current frame or MB. To meet the hypothetical reference decoder (HRD) requirements, the target bits are further bounded in [9]. However, to estimate the target bits for each frame, a common straightforward way is used, namely, an equal number of bits is allocated to each frame regardless of its complexity. Moreover, the linear MAD model is weak in predicting picture characteristics.

Many rate control schemes have been proposed in previous works [11-13]. However, they are difficult to be applied directly to H.264 rate control since they need the information after actually encoding the current frame to decide the appropriate QP. It does not comply with the H.264 RDO procedure. Kamaci et al in [14] have proposed a Cauchy-Density-Based rate model. Although this model is accurate, it needs additional computations to decide the Cauchy-based rate and distortion model parameters.

To resolve this problem, we propose an enhancement frame estimation model to select the appropriate QP for inter-frames. The enhancement is by using a frame complexity estimation to improve the existing MAD based complexity measure. The proposed model utilizes complexity measure for inter-frames which can be obtained without pre-encoding the frame. Bit allocation to each frame is not just computed by target frame rate but also adjusted by a combined frame complexity measure. Simulation results show that our proposed method achieves better rate control for Inter-coded frames. Improved picture quality without skipping the frame is achieved.

The rest of the paper is organized as follows. In Section 2 the bit rate control of H.264/AVC is simply introduced. Section 3 we describe in detail the development of our proposed frame estimation model. Section 4 demonstrates the experimental results for comparison. Finally, we draw a conclusion in Section 5.
2. Background

Rate control plays a very important role in video coding, although it’s not a normative tool for any video coding standard. However, unlike the existing video coding standards, rate control in AVC standard becomes quite difficult because both rate control and rate-distortion optimization (RDO) will involve the quantization parameters. In this section, we will only review and summarize the method used for estimating the target bits. For complete algorithm description, please refer to [9, 10].

2.1 The Concept of A Basic Unit

In the adaptive rate control scheme, a concept of basic unit is introduced. The basic unit can be a frame, a slice, or an MB. Suppose that a frame is composed of \( N_{mbpic} \) MBs. A basic unit is defined to be a group of contiguous MBs which is composed of \( N_{mbunit} \) MBs where \( N_{mbunit} \) is a fraction of \( N_{mbpic} \). Denote the total number of basic units in a frame by \( N_{unit} \), which is computed by

\[
N_{unit} = \frac{N_{mbpic}}{N_{mbunit}}
\]

Examples of a basic unit can be an MB, a slice, a field, or a frame. For example, consider a video sequence with QCIF size, \( N_{mbpic} \) is 99, \( N_{mbunit} \) can be 1, 3, 9, 11, 33, or 99. The corresponding \( N_{unit} \) is 99, 33, 11, 9, 3, and 1, respectively. It is noted that by employing a bigger basic unit, a higher PSNR can be achieved while the bit fluctuation is also bigger. On the other hand, by using a small basic unit, the bit fluctuation is less severe, but with slight loss in PSNR.

2.2 A Linear Model for MAD Prediction

A linear model is used to predict the MADs of current basic unit in the current frame by that of the basic unit in the co-located position of the previous frame.

\[
MAD_c = C_1 \times MAD_p + C_2
\]

\[
C_1 = \frac{1}{20} \sum_{i=1}^{20} MAD_{i-1}
\]

where \( C_1 \) and \( C_2 \) are two coefficients of prediction model. The initial value of \( C_1 \) and \( C_2 \) are set to 1 and 0, respectively. They are updated after coding each basic unit. The linear model (2) is proposed to solve the chicken and egg dilemma between rate control and RDO.

3. Proposed Rate Control Scheme

Similar to earlier standards, H.264/AVC exploits the spatial, temporal and statistical redundancies in the sequence. Since the level of redundancy changes from frame to frame, the number of bits generated per frame is variable. In general, rate control scheme can be thought of as applied to the frame layer and/or to the macroblock (MB) layer. Frame layer rate control allocates a target number of bits to each frame. For a given frame, rate control determines the QP to achieve the frame target bits.

In this paper, we propose a new frame-layer rate control algorithm for target bit allocation. Considering the target bandwidth and the spatial complexity of the frames, we determine the optimal number of target bits for the current frame. In our scheme, the total number of basic unit is frame.

Figure 1 shows the hierarchical diagram of the proposed rate control algorithm. Specifically, the proposed rate control consists of four main operations:

1) Initialization parameters
2) Control GOP-level and Determination of QP
3) Estimation of the target bit rate
4) Updating modeling parameters

In the first stage, we initialize several parameters such as frame counter, used bits after encoding data, the rest of the available bits from the frame rate, and quantization parameter for first I-frame of the sequence. Using this parameter, we obtain initial QP of the first I-frame from the following Fig. 2.

I-frame is being coded, the desired bit rate is read and the...
target bits-per-pixel (bpp) indicator is computed according to the frame size and frame rate by

\[ \text{bpp} = \frac{\text{TargetFrameBits}}{\text{FrameRate} \times \text{width} \times \text{height}} \]  

(4)

Figure 2 shows the initial QP algorithm for the first I-frame of the sequence. In this case, defined bpp, L1, L2, and L3 are used in JVT [18].

In the second stage, we control the number of remaining frames in GOP, and decide QP of the current frame using QP table in the final stage. In hybrid video coding, the structure of GOP (group of picture) influences the whole coding efficiency. GOP is consists of the one I picture and several other kinds of pictures. IPPP… and IBBP… are good examples of the GOP structures. I picture is most important since it used as a reference picture for P picture.

In our scheme, we assume that the first frame in a GOP is an intra-coded I-frame and the remaining frames are all predicted P-type frames, but I-frame is just once encoded in the first GOP and remaining GOP are all encoded to P-frames. We decide the final frame QP by clipping the computed QP (denoted as \( QP_c \)) obtained from the frame-layer rate control calculation, to no more than or the previous frame’s (which is denoted as \( QP_p \)), to maintain the smoothness of visual quality among successive frames, the computed QP is limited to change within a range. In our scheme, the QP for encoding the current frame, which is denoted as \( nQP \), is decided by

\[ nQP = \min \{ QP_p + \Delta QP, \max \{ QP_p - \Delta QP, QP_c \} \} \]  

(5)

where the increment or decrement of \( \Delta QP \) is limited with \( \pm 2 \).

If the estimated QP leads to a larger measurement bit rate, then we increase the QP else we decrease the QP. We adjust \( nQP \) simply by adding 1,

\[ nQP = nQP + 1 \]  

(6)

In the third stage, we estimate the target bit rate for the current frame by three steps. At first, we compute the total number of coding bits used in the previous frame, and then complexity of the current frame is computed. Finally, the target bits for the current frame are estimated. Over the past few years, several studies have been made on rate control by frame complexity measure, but what seems to be lacking is consideration of the encoding time of each frame at the real transmission application. It may be helpful to consider some important factors of achieving accurate frame complexity here. In order to estimate the number of bits to each frame, it is necessary to find out the average bits of five various sequences. Computed average bits of five CIF sequences (slow and smooth sequence “Container”, “News”, normal sequence “Foreman”, fast and detail sequence “Mobile”, “Stefan”) are reported in Table 1.

Table 1. QP Table of P-frames

<table>
<thead>
<tr>
<th>QP</th>
<th>container</th>
<th>foreman</th>
<th>mobile</th>
<th>news</th>
<th>stefan</th>
<th>QP Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>51395</td>
<td>82768</td>
<td>189550</td>
<td>28541</td>
<td>171396</td>
<td>98231</td>
</tr>
<tr>
<td>18</td>
<td>42484</td>
<td>68656</td>
<td>169702</td>
<td>24412</td>
<td>153408</td>
<td>87274</td>
</tr>
<tr>
<td>19</td>
<td>36361</td>
<td>59772</td>
<td>155991</td>
<td>21704</td>
<td>140248</td>
<td>77201</td>
</tr>
<tr>
<td>20</td>
<td>28845</td>
<td>49106</td>
<td>137400</td>
<td>18580</td>
<td>124006</td>
<td>67587</td>
</tr>
<tr>
<td>21</td>
<td>21139</td>
<td>42086</td>
<td>123580</td>
<td>16430</td>
<td>111703</td>
<td>59867</td>
</tr>
<tr>
<td>22</td>
<td>20027</td>
<td>36069</td>
<td>110859</td>
<td>14542</td>
<td>99234</td>
<td>52376</td>
</tr>
<tr>
<td>23</td>
<td>16023</td>
<td>30299</td>
<td>96701</td>
<td>12610</td>
<td>87396</td>
<td>45247</td>
</tr>
<tr>
<td>24</td>
<td>12896</td>
<td>25432</td>
<td>84335</td>
<td>10993</td>
<td>75787</td>
<td>39617</td>
</tr>
<tr>
<td>25</td>
<td>10634</td>
<td>22152</td>
<td>75887</td>
<td>9831</td>
<td>68227</td>
<td>34368</td>
</tr>
<tr>
<td>26</td>
<td>8371</td>
<td>18285</td>
<td>64033</td>
<td>8398</td>
<td>57865</td>
<td>29263</td>
</tr>
<tr>
<td>27</td>
<td>6814</td>
<td>15681</td>
<td>55379</td>
<td>7424</td>
<td>50375</td>
<td>25209</td>
</tr>
<tr>
<td>28</td>
<td>5459</td>
<td>13318</td>
<td>47294</td>
<td>6510</td>
<td>43834</td>
<td>21382</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In this case, QP range for encoding the current frame, which is denoted as \( QR \), is updated by actual bits produced from the previous frames, is decided by
\[ QR_n = AVBit_{n+1} - \frac{AVBit_n - AVBit_{n+1}}{2}, \quad (0 \leq n \leq 50) \]  

(7)

where \( n \) is QP index, \( AVBit \) is total average bits of the QP of the sequences. According to the QR, the number of bits of the QP table is estimated, and QP value of the current frame or current MB is computed.

In the final stage, we update modeling parameters for the current frame QP. They consist of weighted combination of two values: 1) the number of bits produced from the previous frame; 2) the number of bits by scaling the average bits from the reference 20 frames. For frame-level rate control, the target number of bits for each frame is first determined adaptively according to the frame complexity. To estimate the current frame complexity, we use the modeling parameters above. In order to show improvement of the mismatch ratio between the target bits and actual achieved bits per frame, we define as follows:

\[ UBit_{QP} = a_1 \times RBit_{r-1} + a_2 \times SBit \]  

(8)

Where \( UBit_{QP} \) is estimated bits by QP of the previous frames, which is updated target bits to QP table. \( RBit_{r-1} \) is the number of bits produced from the previous frame, \( SBit \) is the average bits from the previous frames with the same QP value in the reference frames, \( a_1 \) and \( a_2 \) are weighting factors and experimental values. Figure 4 shows the procedure to update QP table.

The method to allocate the number of bits of the current frame is decided by

\[ TB = \frac{TargetBits}{FrameRate} \]

\[ BPF = \frac{TB_r + TB}{FrameRate} \]

\[ TB_r = \begin{cases} 0 & \text{if } 0 > \text{PreviousFrameRestBit} \\ TB_r & \text{otherwise} \end{cases} \]  

(9)

where \( TB \) is the number of target bits per frame, \( BPF \) is the number of estimated bits for the current frame, \( TB_r \) is available bits for the rest frame rate, \( TB \) is different bits between estimated bits from previous frame and actual produced bits.

4. Experimental Result

We have implemented our proposed rate control scheme by enhancing the JM12.1 test model software. As a reference for comparisons, the H.264/AVC rate control algorithm was selected (as is implemented on reference software JM12.1). We employed three test sequences of the QCIF 4:2:0 format (176×144 pixels). The frame rate is fixed at 30 fps, a total of 300 frames were coded, frames were not skipped, and search range is ±32. More results are reported in Table 2 and Table 3.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Target Bit Rate</th>
<th>PSNR(Y)</th>
<th>Encoding Time(Hz)</th>
<th>JM</th>
<th>Proposed</th>
<th>Gain</th>
<th>JM</th>
<th>Proposed</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Container</td>
<td>64K 37.56 37.57 0.01 110808 35058 31.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>128K 40.20 40.48 0.28 116889 36024 30.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256K 43.33 43.53 0.20 101818 35820 37.61</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>512K 47.05 47.26 0.21 95734 36593 38.51</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>768K 49.07 49.41 0.34 97427 36114 37.07</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1024K 50.51 50.93 0.41 98181 35848 36.51</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Foreman</td>
<td>64K 32.10 32.18 0.08 146845 36188 24.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>128K 35.70 35.80 0.10 139171 36200 26.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256K 39.27 39.38 0.11 131464 36429 27.71</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>384K 41.46 41.60 0.14 122323 35648 29.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>512K 43.10 43.23 0.13 124569 36518 29.32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>768K 45.54 45.83 0.29 120236 35950 29.90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mobile</td>
<td>1024K 47.48 47.76 0.28 120670 35847 29.71</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>64K 24.30 24.57 0.27 141140 35674 25.28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>128K 27.01 27.11 0.15 152766 36100 23.63</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256K 30.07 30.21 0.15 152224 35980 23.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>384K 32.14 32.35 0.21 152323 35980 23.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>512K 33.62 33.70 0.08 137201 35800 26.09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>768K 36.26 36.32 0.06 139427 36896 26.46</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2 compares the average PSNR values and average encoding time with the proposed and the JM12.1 algorithms.
For “Container”, “Foreman”, and “Mobile” sequence, we have improved the average PSNR values by up to 0.23 dB, 0.16 dB, and 0.13 dB respectively, but we have time saving of 64.67 %, 71.94%, and 74.94 % respectively. As shown in Fig. 6, in encoding time, our method is more stable to various bit rates.

Table 3 also shows that the average target bit error with the proposed and the JM12.1 algorithms. Our method reduces the target bit error for most sequences, can achieve accurate target bit rates.

Figure 5. (a)-(c) Comparison of PSNR performance for JM 12.1 and our proposed scheme.

Figure 6. (a)-(c) Comparison of encoding time for JM 12.1 and our proposed scheme.
In experimental result, we achieved an average PSNR gain of up to 0.17 dB over the entire test set. It was found that our method improves PSNR deviation well in most cases. Table 2 and Table 3 also show that our scheme can achieve accurate target bit rates, and show that the time saving of 71%. These results are all very desirable for various target bit rates or frame rates applications.

5. Conclusion

In this paper, we have presented an efficient real-time rate control scheme without skipping the frame to effectively allocate the number of bits for H.264/AVC video encoding. Our new and simple frame complexity measure is developed to enhance the existing MAD-based method and is applied to our bit allocation for real-time rate control. QP accuracy is very important to prevent the overflow or underflow to target channel of the low bandwidth. Therefore, we have presented a QP control scheme to adjust the computed QP mainly based on actual encoding results of previous coded frame.

As demonstrated in our experiments, in comparison to H.264/AVC rate control, our proposed algorithm achieves higher average PSNR with smoother visual quality. The actually produced bits by each frame are closer to the target bits.

ACKNOWLEDGEMENTS

This work was supported by Seoul R&BD Program (10544).

REFERENCES