QUANTIZER CONTROL FOR H.264/AVC STREAMING OVER NETWORKS

Minqiang Jiang, Xiaoquan Yi, and Nam Ling

Department of Computer Engineering
Santa Clara University
Santa Clara, California 95053, USA
Email: {mjiang, xyi, nling}@scu.edu

ABSTRACT

H.264/AVC is the newest international video coding standard, intended for a broad variety of applications. Similar to former video standards such as H.261, MPEG-1, MPEG-2, H.263, and MPEG-4, H.264/AVC is based on a block-based hybrid coding framework. The number of bits and distortion for each image block is controlled by that block’s quantization parameter. Rate control is used to select the quantization parameters so that the encoder produces bits under the channel bandwidth and buffer constraints while pursuing high video quality. This paper presents a novel quantizer controlling method to further adjust the computed quantization parameter (QP). Simulation results show that the H.264 coder, using the proposed method, can achieve lower relative control error and better visual quality and PSNR performance than the existing H.264 rate control method (JVT_G012).

Keywords: Rate control, Quantizer control, Video coding, H.264, AVC, video streaming.

1. INTRODUCTION

H.264/AVC is the newest international video coding standard, developed jointly by ITU-T as Recommendation H.264 and by ISO/IEC as International Standard 14496-10 (MPEG-4 part10) Advanced Video Coding (AVC) [1]. Its main target is to double the coding efficiency (which means halving the bit rate necessary for a given level of fidelity) in comparison to any other existing video coding standards for a broad variety of applications. Similar to former video standards such as H.261, MPEG-1, MPEG-2, H.263, and MPEG-4, H.264/AVC exploits the spatial, temporal and statistical redundancies in the source video based on a block-based hybrid coding framework, but it has introduced many sophisticated coding features and techniques, such as variable block-size motion compensation, multiple reference pictures, directional spatial prediction for intra coding, small block-size transform, in-the-loop de-blocking filtering, arithmetic entropy coding, etc. Once we decide the prediction mode and motion vector for each block in a frame, it is the quantization parameter (QP) that controls coding fidelity and the number of bits generated. However, since the level of redundancy varies from frame to frame, the number of bits per frame is variable. For constant bit-rate applications, a buffer must be placed between the video encoder and the transmission channel to smooth out the output bit streaming. The rate control scheme is then responsible for selecting quantization parameters to prevent the buffer from overflow or underflow.

The relationship between rate and distortion for texture coding has been given a considerable amount of attention. For example, in [2], Ding proposed a generic rate-quantizer model, which can be adapted according to changes in picture activity. Chiang and Zhang [3] proposed a quadratic rate-quantizer model to predict the number of actual bits produced when a certain quantization parameter is used. In [4] [5], it was shown how the model could be simultaneously applied to different coding contexts, such as frame level, object level, and macroblock (MB) level rate control algorithm. This model was also used in an adaptive rate control scheme [7] [8], which forms the basis for H.264 rate control recommendations [9]. However, it is observed that there exists relatively large mismatch between the number of target bits and the number of actual bits. When the buffer occupancy is high enough, the allocated target number of bits may be very small even less than zero. For dealing with such a problem, the common measure is to set a lower bound for frame targets [4] or texture bits [7] [10]. To do so, the encoder will usually produce more bits than the real target bits. This is one reason why there exists a large error between the actual bits and target bits. In addition, if the buffer occupancy remains above 50% of buffer size and the
cumulative difference between the actual bits and target bits is large enough, the buffer may be in danger of overflow. In contrast to this, the buffer may be in danger of underflow. To solve these problems, we propose a novel quantizer controlling method to further adjust the computed QP.

The organization of the rest of the paper is as follows. The next section briefly reviews the quadratic rate-quantizer model. Section 3 presents our proposed quantizer controlling method. Our experiment results are provided in Section 4. This paper concludes with Section 5.

2. QUADRATIC RATE-QUANTIZER MODEL

Let \( T_{\text{texture}} \) denotes the number of bits for texture coding, \( \text{MAD} \) the mean absolute difference of the texture values, which is an indication of encoding complexity, \( QP \) the quantization parameter of the frame, and \( X1 \) and \( X2 \) the first- and second-order model parameters. The quadratic rate-quantizer model (1) is used to solve the quantization parameter \( QP \) for the current frame.

\[
T_{\text{texture}} = \frac{X1 \times \text{MAD}}{QP} + \frac{X2 \times \text{MAD}}{QP^2}
\]

(1)

\( X1 \) and \( X2 \) are determined by using the linear regression analysis from the previous encoding results as follows:

\[
X1 = \frac{\sum_{i=1}^{n} \frac{QP \times A_i}{\text{MAD}_i} - X2 \times QP^{-1}}{n},
\]

(2)

\[
X2 = \frac{\sum_{i=1}^{n} \frac{A_i}{\text{MAD}_i} - (\sum_{i=1}^{n} QP^{-1}) (\sum_{i=1}^{n} \frac{QP \times A_i}{\text{MAD}_i})}{n \sum_{i=1}^{n} QP^{-2} - (\sum_{i=1}^{n} QP^{-1})^2},
\]

where \( n \) is the number of frames observed in the past, and \( A_i \) is the actual texture bits of the encoded \( i \)th frame. \( T_{\text{texture}} \) in (1) is given by

\[
T_{\text{texture}} = T - T_{\text{header}}
\]

(3)

where \( T \) denotes the bit budget for the current frame, and \( T_{\text{header}} \) can be estimated by the bits used for motion and header information of the previous frame. Once \( X1, X2, \) and \( T_{\text{texture}} \) are obtained, it is easy to solve the quadratic equation (1) for the quantization parameter of the current frame, denoted as \( QP_{cf} \). To maintain the smoothness of visual quality among successive frames, the computed \( QP_{cf} \) is limited to change within a range (e.g. 25% of the previous frame’s quantization parameter \( QP_{pf} \) in [6]). In H.264 rate control scheme [7] [9], \( QP_{cf} \) is adjusted by

\[
QP_{cf} = \min\{QP_{pf} + 2, \max\{QP_{pf} - 2, QP_{cf}\}\}
\]

(4)

However, this kind of adjustment is not enough in some cases. More detailed discussion is given in the next section.

3. QUANTIZER CONTROL SCHEME

From the above subsection, we obtain \( T_{\text{texture}} \) by subtracting \( T_{\text{header}} \) which can be estimated by the bits used for motion and header information of the previous frame, from the frame bit budget \( T \). It is clear that the \( T_{\text{texture}} \) may be less than zero. In [5], a lower bound, which equals the bit rate divided by 30, was imposed on \( T \) to prevent this case from occurring. In H.264 [7] [9], a lower bound for texture bits \( T_{\text{texture}} \) is set by

\[
T_{\text{texture}} = \max\{T_{\text{texture}}, R_i/(\text{MINVALUE} \times F_s)\}
\]

(5)

where \( \text{MINVALUE} \) is a constant and its typical value is 4, \( R_i \) is bit rate for the sequence, and \( F_s \) is frame rate of the source video. However, the computed \( QP_{cf} \) by using the lower bound will result in generating more bits than the number of target bits actually allocated to the current frame. It is necessary to further adjust the computed \( QP_{cf} \) so as to achieve its target bits more accurately. We adjust \( QP_{cf} \) simply by adding 1, i.e.,

\[
QP_{cf} = QP_{pf} + 1
\]

(6)

When the buffer occupancy is higher than 50% of buffer size, if the actual number of bits generated by coded frames remains larger than their corresponding target bits, more accurately, if the cumulative difference between the actual bits and target bits becomes larger and larger, this means that the actual bits after encoding cannot be decreased sufficiently to maintain a stable buffer status. The buffer is in danger of overflow. In some cases, overflow cannot even be avoided. Similarly, if the cumulative difference between the actual bits and target bits becomes smaller and smaller while the buffer level is lower than 30% of buffer size, the buffer may be in danger of underflow.

To solve these problems, the following method is proposed based on above analysis. We first define:

\[
C_n = \begin{cases} 
A_n/T_n, & \text{if } A_n > T_n \\
-T_n/A_n, & \text{if } T_n > A_n 
\end{cases}
\]

(7)
\[ H_1^2 \text{or} H_2^2 = \sum_{i=1}^{\Sigma} C_i \]  

(8)

where \( A_n \) and \( T_n \) are the actual bits and target bits of the \( n \)th frame in the past. \( H_1^1 \) is computed when the current buffer level remains above 50%; otherwise, it is set to be zero. \( H_2^2 \) is computed when the current buffer level remains below 30%; otherwise, it is set to be zero. The \( QP_{cf} \) for the current frame is further adjusted by

\[
Q P_{cf} = \begin{cases} 
Q P_{cf} + \Delta_h, H_1^1 > TH_1 \\
Q P_{cf} - \Delta_h, H_2^2 < TH_2
\end{cases}
\]  

(9)

where \( \Delta_h \) is a threshold with a typical value 1, \( TH_1 \) is the threshold of \( H_1^1 \) with a typical value 8, and \( TH_2 \) is the threshold of \( H_2^2 \) with a typical value -6 in our experiments.

4. EXPERIMENTAL RESULTS

Extensive experiments have been conducted to evaluate the performance of our proposed scheme. The reference is the algorithm described in [7]. All test sequences used are in QCIF 4:2:0 formats. Test platform is JM6.1 [10]. To measure rate control performance, we define relative control error as

\[
E_{rc} = \frac{A - T}{T} \times 100\%
\]  

(10)

where \( A \) and \( T \) are the actual and target bits for each video frame. We plot \( E_{rc} \) of both rate control algorithms for “Foreman” and “Carphone” in Figures 1 and 2, respectively. It can be seen that our proposed scheme yields a smaller control error.

5. CONCLUSION

The peak-signal-to-noise ratio (PSNR) of each frame in “Foreman” and “Carphone” are plotted in Figures 3 and 4, respectively. When frames were skipped, the respective previous encoded frames were used in the PSNR computation. It can be seen that with our proposed scheme the picture quality is improved. The results of other three test videos are listed in Table I. For some video frames in “Carphone” during high motion, the gain is even up to 2 dB. The picture quality improvement is due to applying our quantizer control scheme.
In this paper, we investigated the cases where the adjustment of computed QP is needed. The objective of such adjustment is to meet frame target bits budget more accurately. The straightforward strategy is to monitor the cumulative difference between the actual bits and the target bits. For those frames imposed by a lower bound for texture coding, the computed QP must also be adjusted. Simulation results show that the original rate control algorithm (JVT-G012), adding our quantizer controlling method, can achieve lower relative control error, as well as better visual quality and PSNR performance.

REFERENCES


**TABLE I – Experimental results of H.264 rate control**

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Ave. Control error</th>
<th>Ave. PSNR (dB)</th>
<th>Bit rate (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman(64k)</td>
<td>27.58%</td>
<td>12.68%</td>
<td>35.00</td>
</tr>
<tr>
<td>Carphone(48k)</td>
<td>48.66%</td>
<td>4.25%</td>
<td>35.10</td>
</tr>
<tr>
<td>Akiyo(32k)</td>
<td>9.45%</td>
<td>7.61%</td>
<td>41.81</td>
</tr>
<tr>
<td>News(24k)</td>
<td>18.59%</td>
<td>10.08%</td>
<td>32.64</td>
</tr>
<tr>
<td>Grandma(48k)</td>
<td>31.15%</td>
<td>6.92%</td>
<td>38.08</td>
</tr>
</tbody>
</table>