HEVC Rate Control Algorithm Based on Predictive Weight

Lu WANG, Guo-bao RU and Liang-cai GAN
School of Electronic Information, Wuhan University, Wuhan 430072, China

Keywords: HEVC (High Efficiency Video Coding), Rate control; LCU (Largest Coding Unit), Bit allocation, Parameter Update.

Abstract. Aiming at the shortcomings of HEVC (High Efficiency Video Coding) rate control algorithm using only the information of inter-coded units without using the information of the intra-coded units in the frame to allocate the target bit and update the model parameters, this article proposed a LCU (Largest Coding Unit) layer rate control algorithm based on predictive weight. The proposed algorithm makes full use of the information of both inter-coded units and intra-coded units in the frame, and utilizes a target bit allocation method based on the prediction weight and a model parameter updating method based on the prediction weight which can accurately achieve the target output bit rate, and improve video quality while ensuring the smooth stability of video quality. The experimental results show that in the premise of the same rate control error, BD-PSNR (Bjøntegaard delta peak signal-to-noise rate) of improved algorithm is improved by 0.059dB on average on Low-Delay configuration, and 0.040dB on average higher on Random-Access configuration, and the video quality PSNR (Peak Signal to Noise Ratio) of improved algorithm is less volatile.

Introduction

With the rapid development of intelligent devices and the Internet, video-related applications are becoming more and more common. However, the amount of video information is usually very large, while the capacity of the communication channel for data transmission is limited, so the video information needs to be compressed before transmission[1]. The latest generation of video coding standards HEVC (High Efficiency Video Coding) have been released by JCT-VC (Joint Collaborative Team on Video Coding) organization[2]. In the same video quality constraints, compared with the previous generation coding standard H.264 standard[3], HEVC (High Efficiency Video Coding) standard can save about 50% of the bit stream[4].

Under the limitation of the target rate, by selecting the best series of coding parameters, the rate control algorithm can make the output code rate after compression coding conform to the established target bit rate, and makes the encoded video quality as good as possible. At present, HEVC (High Efficiency Video Coding) mainly has JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) rate control algorithm[6] and JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) rate control algorithm[7]. Compared with JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) algorithm, JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm target bit error is smaller, the video quality is better, while the coding complexity is also significantly reduced. Although the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm has many advantages, it also has the following disadvantages:

(1) In the LCU (Largest Coding Unit) layer target bit allocation, the algorithm only uses the inter prediction information, and does not use the intra prediction information;

(2) When we update the LCU (Largest Coding Unit) layer model parameters, the algorithm only uses the coding information of the encoding unit itself to update it, without using the information of intra and inter prediction. Aiming at the shortcomings of JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, this paper presents an improved LCU (Largest Coding Unit) layer rate control algorithm.

The algorithm makes full use of the information of both inter-coded units and intra-coded units in the frame, and utilizes a target bit allocation method based on the prediction weight and a model
parameter updating method based on the prediction weight which can accurately achieve the target output bit rate, and improve video quality while ensuring the smooth stability of video quality.

The Intra-frame Prediction Model

Definition 1 Basic unit content characteristics

The symbol S represents the content characteristics of the basic unit, the specific definition is as follows in formula (1):

$$S = D \times R = \text{MAD} \times \text{bpp}_{\text{real}}$$

(1)

Where $D$ represents the distortion of the basic unit in the broad sense; $R$ represents the required bits for the basic unit coding in general; $\text{MAD}$ represents the Average Absolute Error of Pixels in Base Unit and the distortion of the basic unit; $\text{bpp}_{\text{real}}$ represents the number of bits per pixel actually required after the basic unit is encoded.

Intra-frame Prediction Method

For the content feature of the basic unit, as shown in Figure 1, the frame intra prediction model is set up. $CTU_0$ represents the basic unit to be encoded at present; $CTU_1 \sim CTU_6$ represents the basic unit next to $CTU_0$ and the encoder has completed the encoding of $CTU_1 \sim CTU_6$. If the content characteristic values of two adjacent basic units are approximately equal, they can be considered to be basically the same, that is, the rate distortion model parameter and the target bit number distribution are basically the same. The texture information in the current frame can be compared by comparing the difference in the value of the content characteristics of the base units in the adjacent and different directions, and the same unit of the same texture is assigned the same rate distortion model parameter and the target bit number allocation. When a basic unit is encoded, the encoder can get its distortion and the number of bits required. The encoder can be predicted using the content characteristics of the adjacent coded base units before encoding the next basic unit.

\[
\begin{array}{cccc}
CTU_1 & CTU_2 & CTU_3 & CTU_4 \\
CTU_5 & CTU_6 & CTU_0 & \\
\end{array}
\]

Figure 1. HEVC basic unit coding diagram.

For the current basic unit $CTU_0$, the intra prediction is performed by the following formula (2):

\[
\begin{align*}
S_{0,H} &= S_6 & \text{if} & & S_2 \approx S_3 \\
S_{0,V} &= S_3 & \text{if} & & S_2 \approx S_6 \\
S_{0,L} &= S_2 & \text{if} & & S_1 \approx S_6 \\
S_{0,R} &= S_4 & \text{if} & & S_3 \approx S_6 \\
\end{align*}
\]

(2)

Where $S_1, S_2, S_3, S_4$ and $S_6$ represents the characteristic values of the content in the related basic unit; $S_{0,H}, S_{0,V}, S_{0,L}$ and $S_{0,R}$ represents the $CTU_0$ characteristic value of the content in the horizontal direction prediction, the vertical direction, the upper left corner direction, and the upper right corner direction.

The bit rate control algorithm in reality, the condition on the right side of the formula (2) is
assumed, and their content characteristics are considered to be the same if the relative error of the content characteristic of the adjacent basic unit is less than 0.3, that is, the same model parameters. The same target bit is allocated, and the corresponding basic unit in the left side of the equation (2) is the prediction unit of the current basic unit, for example, and so on. At the same time, the encoder is calculated according to the prediction order in the formula (2). If there are two prediction units in the direction, the encoder will stop the calculation. And then using the model parameters or the number of target bits corresponding to the prediction units in the two directions to update the model parameters of the current basic unit or the number of target bits.

If there are only one prediction unit, The encoder will directly use the model parameter or the target bit number of the prediction unit to update the model parameter or the target bit number. If there are no prediction unit, the encoder won’t update the model parameters or the number of target bits.

**The Selection of Predictive Weights**

When it comes to intra-frame prediction and inter-frame prediction, the algorithm defines a prediction weight parameter in formula (3) and formula (4) to measure the accuracy of the corresponding prediction so that the encoder can adaptively select relatively better prediction mode before encoding.

\[
\omega_{\text{inter}} = \begin{cases} 
\frac{S_{\text{lastSameLevel}}}{S_{\text{currlCU}}} & \text{if } S_{\text{lastSameLevel}} \leq S_{\text{currlCU}} \\
\frac{2 \cdot S_{\text{currlCU}} - S_{\text{lastSameLevel}}}{S_{\text{currlCU}}} & \text{if } S_{\text{currlCU}} < S_{\text{lastSameLevel}} \text{ and } S_{\text{lastSameLevel}} \leq 2 \cdot S_{\text{currlCU}} \\
0 & \text{others}
\end{cases} \tag{3}
\]

Where \( \omega_{\text{inter}} \) represents the weights of inter prediction of the current basic unit; \( S_{\text{lastSameLevel}} \) represents the content characteristics of the cells at the same location as the current basic unit in the same frame and the adjacent frame of the current frame; \( S_{\text{currlCU}} \) represents the content of the current base unit.

\[
\omega_{\text{intra}} = \begin{cases} 
\frac{\omega_{\text{intra}_1} + \omega_{\text{intra}_2}}{2} & \text{if exist 2 units : } \omega_{\text{intra}_1} > 0.7 \\
\omega_{\text{intra}_1} & \text{if exist 1 units : } \omega_{\text{intra}_1} > 0.7 \\
0 & \text{others}
\end{cases} \tag{4}
\]

Where \( \omega_{\text{intra}_i} \) represents the intra prediction weight in each direction of the current basic unit, its value is calculated by the formula (3), then in the formula (3) \( S_{\text{lastSameLevel}} \) represents the content value of the unit adjacent to the current basic unit; \( \omega_{\text{intra}} \) represents the final intra prediction weight of the current base unit. It should be noted that the intra prediction weight is calculated after the current basic unit coding is completed.

**HEVC Rate Control Algorithm Based on Predictive Weight**

**Target Bit Allocation Based on Intra-Frame Prediction**

According to the intra-frame prediction, there is only one prediction unit, the algorithm specific target bit allocation method is as follows:
In equation (5), \( R_{\text{coded, Pic}} \) represents the actual number of bits required for all the basic units that have been encoded in the current frame; \( \omega_{\text{coded, i}} \) represents the predicted weight of the bit allocation of each basic unit in the current frame. In equation (6), \( \text{MAD}_i \) represents the average absolute error of the first basic unit in the current frame. In equation (7), \( N_{\text{pixels, i}} \) represents the total number of pixels in the \( i \) basic unit in the frame; \( \text{pred}_j \) represents the predicted value of the \( j \) pixel in the \( i \) basic unit, \( \text{org}_j \) represents the original value of the \( j \) pixel in the \( i \) base unit. In equation (8), \( T_{\text{CurrlCU, intra}} \) represents the final intra-prediction target bits.

**Target Bit Allocation Based on Predictive Weight**

This paper proposes a target bit allocation algorithm based on predictive weights. The algorithm combines the intra prediction and inter prediction algorithm, and the encoder can adaptively select the best prediction mode according to the size of the prediction weight. First, the algorithm finds the corresponding prediction unit according to the intra prediction method. Then, based on the intra prediction weight and the inter prediction weight of the prediction unit, the weight of the predicted basic unit to be encoded after the prediction is obtained. If there are two prediction units, the weight of the current base unit is calculated as follows:

\[
\omega_{\text{CurrlCU, intra, pred}} = \frac{\omega_{\text{intra, pred, 1}} + \omega_{\text{intra, pred, 2}}}{2}
\]

(9)

\[
\omega_{\text{CurrlCU, inter, pred}} = \frac{\omega_{\text{inter, pred, 1}} + \omega_{\text{inter, pred, 2}}}{2}
\]

(10)

In formula (9), \( \omega_{\text{CurrlCU, intra, pred}} \) represents the predicted intra prediction weight of the current basic unit; \( \omega_{\text{intra, pred, 1}} \) and \( \omega_{\text{intra, pred, 2}} \) represents actual intra prediction weight of the corresponding prediction unit. In formula (10), \( \omega_{\text{CurrlCU, inter, pred}} \) represents the predicted inter prediction weight of the current basic unit; \( \omega_{\text{inter, pred, 1}} \) and \( \omega_{\text{inter, pred, 2}} \) represents actual inter prediction weight of the corresponding prediction unit.

Finally, according to the predicted weight of the current base unit, the encoder decides which prediction method to use:

\[
T_{\text{CurrlCU}} = \begin{cases} 
T_{\text{CurrlCU, inter}} & \text{if } \omega_{\text{CurrlCU, inter, pred}} > \omega_{\text{CurrlCU, intra, pred}} \\
T_{\text{CurrlCU, intra}} & \text{if } \omega_{\text{CurrlCU, intra, pred}} \leq \omega_{\text{CurrlCU, intra, pred}}
\end{cases}
\]

(11)

Where \( T_{\text{CurrlCU}} \) represents the final target number of bits of the base unit; \( T_{\text{CurrlCU, inter}} \), \( T_{\text{CurrlCU, intra}} \) represents the inter-prediction target bits and the final intra-prediction target bits. \( \omega_{\text{CurrlCU, intra, pred}} \), \( \omega_{\text{CurrlCU, inter, pred}} \).
\( \omega_{\text{Curr\_Intra\_pred}} \) represents the predicted inter prediction weight and the predicted intra prediction weight of the current basic unit.

\[
T_{\text{CurrLCU\_Inter}} = \frac{T_{\text{CurrPic}} - R_{\text{header}} - R_{\text{coded\_Pic}}}{\sum_{i \in \text{AllNotCodedLCUInPic}} \omega_i} \cdot \omega_{\text{CurrLCU}}
\] (12)

In equation (12), \( T_{\text{CurrLCU\_Inter}} \) represents the inter-prediction target bits, \( R_{\text{header}} \) represents the estimated number of head bits required for the current frame, its value is estimated and based on the number of head bits actually required for the encoded frame that is closest to the current frame and belonging to the same class; \( R_{\text{coded\_Pic}} \) represents the actual number of bits required for all the basic units that have been encoded in the current frame; \( \omega_{\text{CurrLCU}} \) represents the weight of the bit allocation of the inter-frame prediction of the current basic unit; \( \omega_i \) represents the bit allocation of the inter-frame prediction of each basic unit in the current frame.

**Model Parameter Updating Based on Inter-frame Prediction**

It is assumed that the basic units of the same position in the adjacent frame of the same class have the same rate distortion curve as the current base unit in the current frame. Thus, the model parameters can be calculated from the coordinates of two points on the rate distortion curve:

\[
\begin{align*}
\lambda_1 &= \alpha \cdot \text{bpp}_1 \beta \\
\lambda_2 &= \alpha \cdot \text{bpp}_2 \beta
\end{align*}
\] (13)

In formula (13), \( \lambda_1 \), \( \text{bpp}_1 \) represents the Lagrangian multiplier used in the current basic unit encoding and the number of bits per pixel actually required to complete the encoding; \( \lambda_2 \), \( \text{bpp}_2 \) represents the Lagrangian multiplier used in the basic unit encoding at the same position in the adjacent frame of the same class and the number of bits per pixel actually required to complete the encoding. To solve the equation of the formula (13), we can get the following formula for updating the model parameters:

\[
\begin{align*}
\beta &= \frac{\ln \lambda_1 - \ln \lambda_2}{\ln \text{bpp}_1 - \ln \text{bpp}_2} \\
\alpha &= \exp(\ln \lambda_1 - \beta \cdot \ln \text{bpp}_1)
\end{align*}
\] (14)

Equation (14) is a model parameter update algorithm based on inter prediction. The following special cases need to be handled separately:

1. if \( \text{bpp}_1 = \text{bpp}_2 \), the value of \( \beta \) cannot be updated by Equation (14), then we take \( \beta = -0.36 \), the value of \( \alpha \) is still updated by Equation (14);
2. if \( \text{bpp}_1 = \text{bpp}_2 \) and \( \lambda_1 = \lambda_2 \), it means that the contents of the two basic units are exactly the same. And the value of \( \beta \) cannot be updated by Equation (14), then we take \( \beta = -0.36 \), the value of \( \alpha \) is still updated by Equation (14);
3. Because the distortion curve is a convex function, the model must satisfy \( \beta < 0 \). When calculated according to equation (14), if \( \beta \geq 0 \), the value of \( \beta \) cannot be updated by Equation (14), then we take \( \beta = -0.36 \), the value of \( \alpha \) is still updated by Equation (14). Finally, the updated model parameters are limited to a reasonable range of values, the range of the value of \( \beta \) is \([-3.0, -0.36]\), the range of the value of \( \alpha \) is \([1.0 \times 10^{-20}, 1.0 \times 10^{8}]\).

**Model Parameter Updating Based on Inter-frame Prediction**

If the motion of the original video is so severe, the correlation between adjacent frames will be
weakened, thus affecting the accuracy of model parameter updates based on inter prediction. Therefore, in order to improve the accuracy of model parameter prediction, this paper proposes the method of intra prediction and uses the mean value of model parameters to update the current base unit model parameters.

\[ \alpha_{\text{intra}} = \sqrt{\alpha_1 \cdot \alpha_2} \]  
\[ \beta_{\text{intra}} = \frac{\beta_1 + \beta_2}{2} \]  

In the formulas (15) and (16), \( \alpha_{\text{intra}} \) and \( \beta_{\text{intra}} \) represent the model parameters of the current basic unit based on intra prediction; \( \alpha_1 \) and \( \beta_1 \) represent predictive unit model parameters; \( \alpha_2 \) and \( \beta_2 \) represent another predictive unit model parameters.

**Model Parameter Updating Based on Predictive Weight**

The model parameter updates algorithm based on predictive weight combines the intra prediction and inter prediction algorithm, and the encoder can adaptively select the best prediction mode according to the size of the prediction weight.

\[
\begin{cases}
\alpha = \alpha_{\text{intra}}, \beta = \beta_{\text{intra}} & \text{if } \omega_{\text{Curr}_-\text{Inter}_-\text{pred}} > \omega_{\text{Curr}_-\text{Intra}_-\text{pred}} \\
\alpha = \alpha_{\text{inter}}, \beta = \beta_{\text{inter}} & \text{if } \omega_{\text{Curr}_-\text{Inter}_-\text{pred}} \leq \omega_{\text{Curr}_-\text{Intra}_-\text{pred}}
\end{cases}
\]  

In equation (17), \( \alpha \) and \( \beta \) represent the final model parameters of the base unit before encoding; \( \alpha_{\text{inter}} \) and \( \beta_{\text{inter}} \) represent model parameters based on inter prediction; \( \alpha_{\text{intra}} \) and \( \beta_{\text{intra}} \) represent model parameters based on inter prediction; \( \omega_{\text{Curr}_-\text{Inter}_-\text{pred}} \) and \( \omega_{\text{Curr}_-\text{Intra}_-\text{pred}} \) represent inter prediction weights and intra prediction weights.

**Experimental Results and Analysis**

In the experiment, HEVC (High Efficiency Video Coding) official recommended bit rate is used as the target bit rate of the rate control[10]. We add the HEVC (High Efficiency Video Coding) rate control algorithm based on predictive weight into the reference test model HM-10.0[8], which replaces the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, and then performs the coding test with the HEVC (High Efficiency Video Coding) official recommendation test video sequences[9].

**Video Quality after Encoding**

Table 1 and Table 2 respectively compare the video quality PSNR (Peak Signal to Noise Ratio) of the JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) algorithm, the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, and the proposed algorithm on Low-Delay configuration and on Random-Access configuration. According to the experimental data, on Low-Delay configuration, the video quality of the proposed algorithm is higher than the video quality of JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) by 1.340dB. Compared with JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, the PSNR (Peak Signal to Noise Ratio) of the proposed algorithm is improved by 0.263dB. On Random-Access configuration, the video quality of the proposed algorithm is 2.371dB which is higher than the video quality of JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) algorithm. Compared with JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, the PSNR (Peak Signal to Noise Ratio) of the proposed algorithm is improved by 0.136dB.
Rate-distortion Performance Comparison

In this paper, BD-PSNR (Bjøntegaard delta peak signal-to-noise rate) and BD-Rate \[^{[11]}\] (Bjøntegaard delta rate) are used to characterize the rate-distortion performance of the algorithm. Table 3 and Table 4 respectively compare the rate-distortion performance of the JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) algorithm, the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, and the proposed algorithm in this paper on Low-Delay configuration and on the Random-Access configuration. According to the experimental data, on Low-Delay configuration, the BD-PSNR (Bjøntegaard delta peak signal-to-noise rate) is improved by 0.784 dB on average compared with that of JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) algorithm and the BD-PSNR is improved by 0.059 dB compared with the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm. On the Random-Access configuration, the BD-PSNR (Bjøntegaard delta peak signal-to-noise rate) is improved by 0.040 dB compared with that of JCTVC-H0213 (Joint Collaborative Team on Video Coding-H0213) algorithm and the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm.

Table 1. Video quality on Low-Delay configuration (unit: dB).

<table>
<thead>
<tr>
<th>Sequence name</th>
<th>Target bit rate</th>
<th>PSNR JCTVC-H0213</th>
<th>JCTVC-K0103</th>
<th>Proposed</th>
<th>Proposed VS JCTVC-H0213</th>
<th>Proposed VS JCTVC-K0103</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Kimono</strong> (1920x1080)</td>
<td>6000</td>
<td>41.614</td>
<td>41.709</td>
<td>41.714</td>
<td>+0.100</td>
<td>+0.005</td>
</tr>
<tr>
<td></td>
<td>4000</td>
<td>40.558</td>
<td>40.775</td>
<td>40.786</td>
<td>+0.228</td>
<td>+0.011</td>
</tr>
<tr>
<td></td>
<td>1600</td>
<td>37.273</td>
<td>38.064</td>
<td>38.102</td>
<td>+0.829</td>
<td>+0.038</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>35.270</td>
<td>36.251</td>
<td>36.353</td>
<td>+1.083</td>
<td>+0.102</td>
</tr>
<tr>
<td><strong>PartyScene</strong> (832x480)</td>
<td>2000</td>
<td>31.127</td>
<td>32.111</td>
<td>32.187</td>
<td>+1.060</td>
<td>+0.076</td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>29.365</td>
<td>30.094</td>
<td>30.213</td>
<td>+0.848</td>
<td>+0.119</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>26.865</td>
<td>26.852</td>
<td>27.067</td>
<td>+0.202</td>
<td>+0.215</td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>26.198</td>
<td>25.917</td>
<td>26.180</td>
<td>-0.018</td>
<td>+0.263</td>
</tr>
<tr>
<td><strong>BasketballPass</strong> (416x240)</td>
<td>1500</td>
<td>39.991</td>
<td>40.148</td>
<td>40.245</td>
<td>+0.254</td>
<td>+0.097</td>
</tr>
<tr>
<td></td>
<td>850</td>
<td>36.879</td>
<td>36.992</td>
<td>37.088</td>
<td>+0.191</td>
<td>+0.096</td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>33.272</td>
<td>33.155</td>
<td>33.266</td>
<td>-0.006</td>
<td>+0.111</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>31.600</td>
<td>31.437</td>
<td>31.531</td>
<td>-0.069</td>
<td>+0.094</td>
</tr>
<tr>
<td><strong>Johnny</strong> (1280x720)</td>
<td>4000</td>
<td>43.274</td>
<td>43.818</td>
<td>43.834</td>
<td>+0.560</td>
<td>+0.016</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>41.812</td>
<td>43.125</td>
<td>43.152</td>
<td>+1.340</td>
<td>+0.027</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>41.157</td>
<td>42.238</td>
<td>42.296</td>
<td>+1.139</td>
<td>+0.058</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>39.982</td>
<td>41.164</td>
<td>41.224</td>
<td>+1.242</td>
<td>+0.060</td>
</tr>
</tbody>
</table>
Table 2. Video quality on Random-Access configuration (unit: dB).

<table>
<thead>
<tr>
<th>Sequence name</th>
<th>Target bit rate</th>
<th>JCTVC-H0213 PSNR</th>
<th>JCTVC-K0103 PSNR</th>
<th>Proposed PSNR</th>
<th>Proposed VS JCTVC-H0213</th>
<th>Proposed VS JCTVC-K0103</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cactus (1920x1080)</td>
<td>10000</td>
<td>37.154</td>
<td>37.686</td>
<td>37.689</td>
<td>+0.535</td>
<td>+0.003</td>
</tr>
<tr>
<td></td>
<td>7000</td>
<td>35.954</td>
<td>37.124</td>
<td>37.125</td>
<td>+1.171</td>
<td>+0.001</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>32.688</td>
<td>35.024</td>
<td>35.059</td>
<td>+2.371</td>
<td>+0.035</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>31.624</td>
<td>33.708</td>
<td>33.743</td>
<td>+2.119</td>
<td>+0.035</td>
</tr>
<tr>
<td>BQMall (832x480)</td>
<td>2000</td>
<td>35.869</td>
<td>37.935</td>
<td>37.977</td>
<td>+2.108</td>
<td>+0.042</td>
</tr>
<tr>
<td></td>
<td>1200</td>
<td>34.135</td>
<td>35.969</td>
<td>36.038</td>
<td>+1.903</td>
<td>+0.069</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>31.228</td>
<td>32.344</td>
<td>32.438</td>
<td>+1.210</td>
<td>+0.094</td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>30.247</td>
<td>31.173</td>
<td>31.309</td>
<td>+1.062</td>
<td>+0.136</td>
</tr>
<tr>
<td>RaceHorses (416x240)</td>
<td>1500</td>
<td>38.869</td>
<td>40.502</td>
<td>40.521</td>
<td>+1.652</td>
<td>+0.019</td>
</tr>
<tr>
<td></td>
<td>850</td>
<td>35.603</td>
<td>37.437</td>
<td>37.472</td>
<td>+1.869</td>
<td>+0.035</td>
</tr>
<tr>
<td></td>
<td>384</td>
<td>32.155</td>
<td>33.363</td>
<td>33.430</td>
<td>+1.275</td>
<td>+0.067</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>30.583</td>
<td>31.529</td>
<td>31.573</td>
<td>+0.990</td>
<td>+0.044</td>
</tr>
<tr>
<td>FourPeople (1280x720)</td>
<td>4000</td>
<td>42.830</td>
<td>43.077</td>
<td>43.085</td>
<td>+0.255</td>
<td>+0.008</td>
</tr>
<tr>
<td></td>
<td>2000</td>
<td>41.403</td>
<td>41.639</td>
<td>41.655</td>
<td>+0.252</td>
<td>+0.016</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>39.129</td>
<td>39.199</td>
<td>39.233</td>
<td>+0.104</td>
<td>+0.034</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>36.019</td>
<td>36.116</td>
<td>36.219</td>
<td>+0.200</td>
<td>+0.103</td>
</tr>
</tbody>
</table>

Table 3. Rate-distortion performance on Low-Delay Configuration.

<table>
<thead>
<tr>
<th>Category</th>
<th>Rate-distortion performance</th>
<th>Proposed VS JCTVC-H0213</th>
<th>Proposed VS JCTVC-K0103</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BD-Rate</td>
<td>BD-PSNR</td>
<td>BD-Rate</td>
</tr>
<tr>
<td>Class B</td>
<td>-28.78%</td>
<td>0.948</td>
<td>-1.97%</td>
</tr>
<tr>
<td>Class C</td>
<td>-13.17%</td>
<td>0.544</td>
<td>-2.21%</td>
</tr>
<tr>
<td>Class D</td>
<td>-12.34%</td>
<td>0.533</td>
<td>-1.25%</td>
</tr>
<tr>
<td>Class E</td>
<td>-45.92%</td>
<td>1.109</td>
<td>-2.52%</td>
</tr>
<tr>
<td>Average</td>
<td>-25.05%</td>
<td>0.784</td>
<td>-1.99%</td>
</tr>
</tbody>
</table>

Table 4. Rate-distortion performance on Random-Access Configuration.

<table>
<thead>
<tr>
<th>Category</th>
<th>Rate-distortion performance</th>
<th>Proposed VS JCTVC-H0213</th>
<th>Proposed VS JCTVC-K0103</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BD-Rate</td>
<td>BD-PSNR</td>
<td>BD-Rate</td>
</tr>
<tr>
<td>Class B</td>
<td>-35.32%</td>
<td>1.396</td>
<td>-0.77%</td>
</tr>
<tr>
<td>Class C</td>
<td>-41.39%</td>
<td>1.991</td>
<td>-1.94%</td>
</tr>
<tr>
<td>Class D</td>
<td>-35.64%</td>
<td>1.810</td>
<td>-0.95%</td>
</tr>
<tr>
<td>Class E</td>
<td>-3.83%</td>
<td>0.138</td>
<td>-0.50%</td>
</tr>
<tr>
<td>Average</td>
<td>-29.05%</td>
<td>1.334</td>
<td>-1.04%</td>
</tr>
</tbody>
</table>

Conclusion

Aiming at the shortcoming of JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm, this paper presents a LCU (Largest Coding Unit) layer rate control algorithm based on predictive weight. The improved algorithm makes full use of both the information of the intra units in the frame and the information of the inter units in the frame, and proposes a target bit allocation method based on the prediction weight and a model parameter updating method based on the prediction weight which can accurately achieve the target output bit rate, and improve video quality while ensuring the smooth stability of video quality. The experimental results show that BD-PSNR (Bjøntegaard delta peak signal-to-noise rate) of improved algorithm is 0.059dB higher than that of the JCTVC-K0103 (Joint Collaborative Team on Video Coding-K0103) algorithm on Low-Delay configuration, and 0.040dB higher on Random-Access configuration and the video quality PSNR (Peak Signal to Noise Ratio) of improved algorithm is less volatile.

52
Acknowledgement

This research was financially supported by National Science Fund subsidized project (No.61671333, No.61072041).

References


