The SLDSRC Rate Control Scheme for H.264

Jianguo Jiang[1,2], Wenju Zhang[1], and Man Xuan[1]

[1] School of Computer & Information, Hefei University of Technology, Hefei 230009, China
[2] Engineering Research Center of Safety Critical Industrial Measurement and Control Technology, Ministry of Education, Hefei 230009, China
jianguoj@163.com, wena_tt@yahoo.cn, eel.xuan@gmail.com

Abstract. A novel slice-layer double-step rate control (SLDSRC) scheme for H.264 is proposed. It not only resolves the problem of inter-dependency between rate-distortion (R-D) optimization (RDO) and rate control (RC), but also improves control accuracy by introducing double-step mechanism, new source rate prediction model, header-bit prediction method. The new rate-quantization (R-Q) model distributes bit rate more reasonable; the novel header-bit prediction method satisfies the requirement of high accuracy at low bit rate. Experimental results show the proposed algorithm heightens the control precision, improves the PSNR and reduces fluctuation of output bit rate, compared to RC algorithm in JVT-H017.

Keywords: Double-step RC, R-Q model, SAQD, control accuracy, PSNR.

1 Introduction

RC, one of the important video coding technologies, has been playing an important role in the video transmission, storage and the hardware design. It has been extensively studied in many standards (MPEG-2, MPEG-4, H.263, H.264) [1-6]. With the existence of chicken and egg dilemma [5], it is very complex to achieve RC for H.264. JVT reference software JM adopts the RC algorithm proposed in JVT-H017 [6]. Based on it, many improved algorithms have been proposed. But most of them do not change the holistic structure of the RC algorithm for H.264 although associate the complexity of the image itself to optimize bit allocation [7-9] for overcoming the weakness that the complexity of video image isn't taken into account in JVT-H017, and there are still some limitations.

2 SLDSRC Algorithm

2.1 Double-Step Encoder Mechanism

The SLDSRC scheme is divided into the preparation step and the encoding step, denoted as p_step and e_step respectively in Fig.1. Each frame is performed through the above two steps.

S. Boll et al. (Eds.): MMM 2010, LNCS 5916, pp. 683–688, 2010.

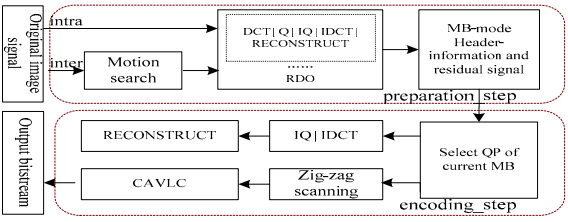

Fig. 1. The framework of SLDSRC

p_step: To encode a new frame, update QP_0 according to QP distribution for slices in a frame and the change trend of image complexity between adjacent frames, and after motion search, perform RDO for all MBs in the current frame for obtaining the header information and residual signal. This step makes preparation for e_step. QP and processing unit in this step are QP_0 and a frame, respectively.

e_step: Search the best QP for each slice in the QP discrete set $[QP_0-2,\ QP_0+2]$, recorded as QP_c, and then encode the residual signal of each slice obtained in p_step. QP and processing unit in this step are QP_c and a slice, respectively.

Generally, the decrease of the coding gain is not much even though QP_0 and QP_c are different as long as the difference is restricted to a small range, as $|QP_c - QP_0| \le \delta, \delta \le 2$. We observe almost identical results for numerous test sequences and show only "foreman" as a representative one in table 1. Experiment results demonstrate that PSNR just decrease about 0.12dB if $|QP_c - QP_0| = 2$, and the decrease is negligible when $|QP_c - QP_0| = 1$.

It should be noted that if a slice is inter-coded, the residual signal for QP_0 is simply re-quantized using QP_c; if it is intra-coded, the residual signal should be updated since QP_c may be different from QP_0 and its adjacent reference units are no longer the reconstructed units for QP_0. For such a case, update residual signal following the same intra-mode determined by the RDO for QP_0.

Table 1. A compare of PSNR for $QP_c - QP_0 \in [-2, +2]$

QPc-QP0 / Rate(kbps)	-2	-1	0	1	2
48	31.61	31.70	31.74	31.68	31.68
64	32.01	32.98	33.00	32.98	32.94
96	34.66	34.75	34.77	34.72	34.71
150	37.65	37.74	37.78	37.75	37.75
250	40.26	40.41	40.44	40.38	40.35
500	44.11	44.26	44.30	44.23	44.22

2.2 The R-Q Model Based On SAQD

The quadratic Laplace-distribution-based R-Q model [5] has been widely used, but this kind of ρ-domain prediction model is less accurate than the q-domain model, and the inaccuracy is mainly due to the roughness of residual-signal complexity denotation, such as MAD-denotation. It only reflects the time-domain residual difference, but can not reflect the state of actual coding bit stream. In frequency-domain [10], in

terms of the statistical characteristic of DCT coefficients [11], we develop the sum of absolute quantization distortion SAQD and exploit a better R-Q model.

SAQD is defined as formula (1):

$$\mathrm{SAQD} = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} d_{i,j}, d_{i,j} = \begin{cases} 0, & |X_{i,j}| > \varphi \cdot Qstep \\ |X_{i,j}|, & |X_{i,j}| \le \varphi \cdot Qstep \end{cases} . \quad (1)$$

where $X_{i,j}$ is the DCT coefficient for the position (i, j) before quantization, Qstep, N and φ are the quantization stepsize, MB-height, and the threshold constant, respectively.

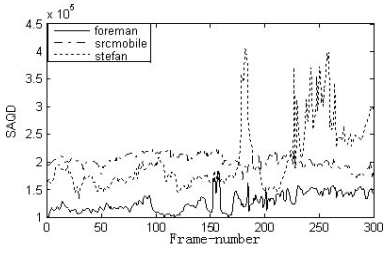

Fig. 2. SAQD variation for three sequences

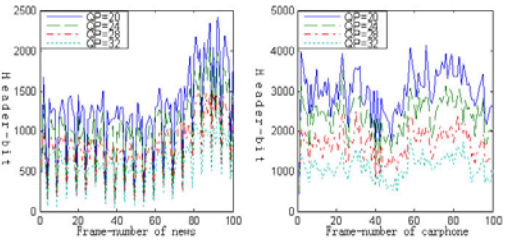

Fig. 3. Variation of header bits with QP

It's well known that the smaller the residual DCT coefficient is, the greater the QP is, the greater the error between before and after quantization will be, the more detailed information will be lost, resulting in worse image distortion as well as bigger SAQD. Contrarily, more detailed information will be reserved, the smaller the distortion will be and the smaller the SAQD will be. The SAQD variation for three sequences with the QP 28 and the format CIF is shown as Fig.2. We can see the SAQD for srcmobile is bigger than foreman. That's because the complexity of the former image is greater than the latter. Meanwhile, the SAQD stability of the above two sequences are higher than stefan, which precisely reflects the characteristics of intense image movement and large complexity-change of stefan. Therefore, SAQD can be used to denote image complexity accurately.

Due to higher accuracy of SAQD denotation as image complexity, we update the quadratic R-Q model [5] as formula (2):

$$B_s - B_{head} = \left(\frac{X_1}{\mathrm{Qstep}^2} + \frac{X_2}{Qstep}\right) \cdot SAQD . \quad (2)$$

where B_s, B_{head} are all the bits allocated to coding blocks and the corresponding header bits, respectively. Qstep is quantization stepsize, X_1, X_2 are model parameters, updated by linear regression technique after encoding each slice, which can be referred to JVT-H017 [6].

2.3 Header-Bit Prediction

In order to predict header bits, JVT-H017 treats the average header bits for the encoded units as that of the current encoding unit. It is simple, but not effective. The paper presents an accurate and effective method to predict header bits.

In the RDO process, the encoder determines the encoding mode for every MB by minimizing Lagrange cost function, shown as formula (3):

$$J_{MODE}(S_k, I_k \mid QP, \lambda_{MODE}) = D_{REC}(S_k, I_k \mid QP) + \lambda_{MODE} \cdot R_{REC}(S_k, I_k \mid QP). \quad (3)$$

$$\lambda_{MODE} = 0.85 \times 2^{(QP-12)/3}. \quad (4)$$

where I_k is the encoding mode of MB S_k, and R_{REC} and D_{REC} are the bit rate and distortion of encoded block, respectively.

The larger QP becomes, the larger λ_{MODE} will become, and the larger the importance of R_{REC} in the Lagrange cost function will get, thus more attention will be paid to R_{REC} while the smaller the importance of D_{REC} will fall. For such a case, I_k becomes simple, such as rough division mode, small MV, and therefore the header bits will decrease. Experimental results demonstrate that header bits decrease as QP increases as Fig.3.

Since QP_0 in p_step and QP_c in e_step may be different, the header bits generated in p_step may not be equal to the actual header bits generated in e_step. However, header bits decrease as QP increases, and $QP_c (QP_c \in [QP_0 - 2, QP_0 + 2])$ is equal to QP_0 or almost evenly fluctuates around QP_0 (as shown in experimental results), so the actual header bits in e_step equals to or almost evenly fluctuates around the actually generated header bits, and the number of the fluctuating bits can be counteracted one another. Based on the above, a concise and accurate method is developed for predicting header bits as formulate (5):

$$B_{head\_e,i} = B_{head\_p,i}. \quad (5)$$

where $B_{head\_e,i}$ is the header bits of the current frame, i-th frame, and $B_{head\_p,i}$ is actually generated header bits in p_step.

3 Experiment Results and Analysis

The presented SLDSRC algorithm was implemented on JM10.0 platform for H.264 baseline-profile encoder under constant bit rate constraint. In order to brighten the advantages of the new models and methods in SLDSRC algorithm, we initialize GOP-layer QP and distribute frame-layer bit rate as JVT-H017. Several QCIF sequences are encoded with IPPP format, regarding a frame in intra-frame coding as a slice and a MB in inter-frame coding as a slice. φ in equation (1) is endowed with an experiential value 1.0, X_1, X_2 in equation (2) are initialized with 1.0, 0.0, respectively. $slidwd_{\max}$ is set to experiential value 20.

In table 2, it is very clear that the bit rate using our algorithm is more approximate to the signal channel bandwidth, and the PSNR is raised. We know from Fig.4 that the bit rate fluctuation between two successive frames by our algorithm is smaller. Those prove that our algorithm not only makes a reasonable use of signal channel bandwidth, but also obtain higher control accuracy for a single frame, therefore avoids the video buffer underflow and overflow. A compare of the predicted header bits and the

Table 2. A compare of two algorithms in accuracy and PSNR

Test Sequences (QCIF)	Target Bits (kbps)	Inital QP	Control Accuracy (kbps)	Rate Offset (kbps)	PSNR (dB)
			JVT-H017 Proposed	JVT-H017 Proposed	JVT-H017 Proposed
foreman	48	32	48.10 48.00	0.10 0.00	31.74 32.00(+0.28)
	64	30	64.06 64.00	0.06 0.00	33.00 33.28(+0.28)
	96	26	96.06 96.00	0.06 0.00	34.77 34.98(+0.21)
carphone	48	32	48.03 48.01	0.03 0.01	32.59 32.98(+0.39)
	64	30	64.03 64.00	0.03 0.00	33.81 34.12(+0.31)
	96	28	96.01 96.00	0.01 0.00	35.63 35.90(+0.27)
Hall monitor	48	26	48.06 48.01	0.06 0.01	38.51 38.83(+0.32)
	64	24	60.06 60.01	0.06 0.01	39.60 39.90(+0.30)
	96	23	96.02 90.00	0.02 0.00	40.84 41.09(+0.25)
Mother daughter	48	32	48.04 48.00	0.04 0.00	38.47 38.80(+0.33)
	64	26	64.08 64.01	0.08 0.01	39.98 40.28(+0.30)
	96	23	96.11 96.02	0.11 0.02	41.84 42.11(+0.27)
news	48	32	48.09 48.01	0.09 0.01	35.28 35.65(+0.37)
	64	26	64.03 63.99	0.03 0.01	37.28 37.68(+0.40)
	96	23	96.09 95.98	0.09 0.02	39.77 40.02(+0.25)
salesman	48	32	48.11 48.02	0.11 0.02	36.27 36.48(+0.21)
	64	26	64.09 64.00	0.09 0.00	38.21 38.45(+0.24)
	96	23	96.17 96.02	0.17 0.02	40.46 40.74(+0.28)

actual header bits between two algorithms is shown as Fig.5. It is obvious that the number of predicted bits by our algorithm is closer to the actual header bits than JVT-H017, which reveals our algorithm predicts header bits more accurately.

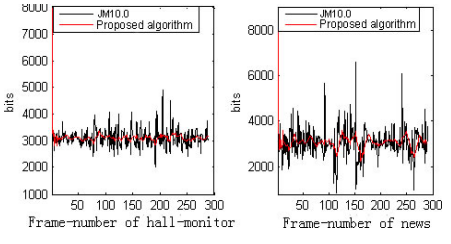

Fig. 4. Compare of bit rate fluctuation

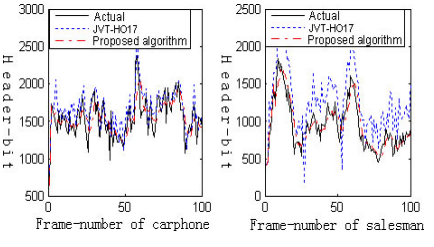

Fig. 5. Compare of header bits

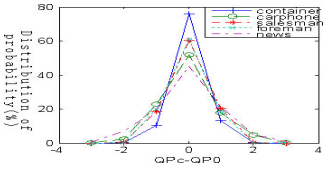

Fig. 6. Probability distribution of $QP_c - QP_0$

Fig.6 is the $QP_c - QP_0$ statistical probability distribution with QP_c ranging from $QP_0 - 3$ to $QP_0 + 3$ .We can conclude that the probability of $|QP_c - QP_0| = 3$ is very small, so we set QP_c among the discrete set $[QP_0\text{-}2, QP_0+2]$ in our algorithm. It validates the reasonability and feasibility of double-step algorithm.

4 Conclusion

The proposed SLDSRC algorithm adopts double-step coding mechanism to resolves the chicken and egg dilemma radically and introduces new source-bit prediction model, header-bit prediction method and R-Q model to improve control precision and image quality. In addition, the double-step coding style increases coding complexity, however it can not affect coding rate. The algorithm just applied to H.264 baseline-profile encoder, future work will be extended to the main profile and extended profile.

References

1. Lee, H.J., Chiang, T.H., Zhang, Y.Q.: Scalable Rate Control for MPEG-4 Video. J. IEEE Transactions on Circuits and Systems for Video Technology 10(6), 878–894 (2000)
2. Vetro, A., Sun, H., Wang, Y.: MPEG-4 rate control for multiple video objects. J. IEEE Transactions on Circuits and Systems for Video Technology 9(1), 186–199 (1999)
3. MPEG-2 Test Model 5, Doc. ISO/IEC JTC1/SC29 WG11/93-400 (April 1993), http://www.mpeg.org/MPEG/MSSG/tm5
4. Li, Z.G., Xiao, L., Zhu, C., Feng, P.: A Novel Rate Control Scheme for Video Over the Internet. In: Proceedings of ICASSP, Florida, USA, pp. 2065–2068 (2002)
5. Li, Z., Pan, F., Pang, K.: Adaptive basic unit step rate control for JVT, JVT–G012. In: Joint Video Team of ISO/IEC and ITU 7th Meeting, Pattaya, Thailand (2003)
6. Ma, S., Li, Z., Wu, F.: Proposed draft of adaptive rate control, JVT–H017. In: Joint Video Team of ISO/IEC and ITU 8th Meeting, Geneva (2003)
7. Jiang, M., Yi, X., Ling, N.: Improved frame step rate control for H.264 using MAD ratio. In: IEEE International Symposium on Circuits and Systems, British Columbia, Canada, pp. 813–816 (2004)
8. Miyaji, S., Takishima, Y., Hatori, Y.: A novel rate control method for H.264 video coding. In: IEEE International Conference on Image Processing, Genoa, Italy, pp. 11-309–11-312 (2005)
9. Jiang, M., Yi, X., Ling, N.: One enhancing H.264 rate control by PSNR-based frame complexity stimation. In: International Conference on Consumer Electronics, pp. 231–232 (2005)
10. Kim, Y.K., He, Z., Sanjit, K.M.: Low-delay rate control for DCT video coding via p-set source modeling. J. IEEE Trans. Circuits and Systems for Video Technology 11(8), 92–940 (2001)
11. Malvar, H.S., Hallapuro, A., Karczewicz, M., et al.: Low-Complexity Transform and Quantization in H.264/AVC. J. IEEE Trans. on Circuits and Systems for Video Technology 13(7), 598–603 (2003)