EFFECTIVE ERROR CONCEALMENT ALGORITHM BY BOUNDARY INFORMATION FOR H.264 VIDEO DECODER

Guang-Tung Jian, Mei-Juan Chen and Ming-Chieh Chi

Department of Electrical Engineering, National Dong Hwa University, Taiwan

ABSTRACT

H.264 is a new video coding standard which contains some novel coding functions and provides more coding efficiency than previous standards. However, video streams are sensitive to channel errors that often degrade the visual quality of the decoded sequence, not only corrupt the current frame but also propagate to subsequent frames. In order to solve this problem, the error concealment in the decoder is very important and useful, neither bit rate nor delay will increase during the transmission. In this paper, we propose an error concealment method by utilizing the temporal correlation of the lost macroblock and its neighboring macroblocks to reconstruct the damaged data. Simulation results show that both of the subjective and objective performances of the proposed techniques are superior to conventional temporal concealment methods for H.264 standard.

1. INTRODUCTION

One of main differences between H.264 and previous coding standards is that the motion estimation scheme has been greatly changed. In H.264, a 16x16 macroblock(MB) can be divided into different block types for motion estimation [1][2]. Fig. 1 shows the different block modes that are adopted in H.264. In video transmission, since compressed video streams are very sensitive to transmission errors, the loss of a packet usually introduces severe distortion to the reconstructed results. In temporal error concealment, the correlation between the current decoding frame and the previous reconstructed frames is exploited. The damaged image is replaced by the MB in the reference frame specified by the motion vector(MV) of the damaged MB. A decisive strategy with the temporal error concealment is that it requires the motion information and determines which MV is the optimal. The motion information of the corrupted MB may also be damaged and needs to be estimated. For the estimation of MVs, the following methods have been proposed.

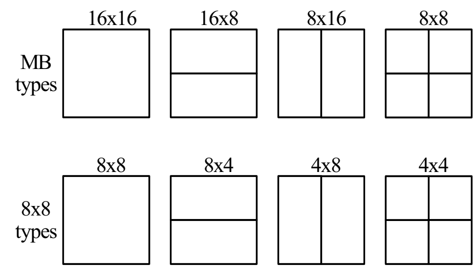


Fig. 1 Segmentation of the MB for motion compensation. Top: segmentation of MBs. Bottom: segmentation of 8x8 partitions.

The simplest way to determine the MV is utilizing the zero MV to reconstruct the lost MB, and it is well known as the temporal replacement (TR)[3]. Block matching algorithm (BMA) is often used to select an optimal MV to substitute for the lost one [4][5]. Chen *et al.* proposed the so-called refined boundary matching algorithm (RBMA) based on the boundary matching algorithm [6]. Wang *et al.* [7] have presented a modified BMA for the newest H.264 standard. In this technique, each candidate vector is used to conceal the damaged block. Zheng *et al.* [8] introduced an algorithm, which uses Lagrange interpolation formula, to constitute a polynomial that describes the MVs. In this paper, a temporal error concealment algorithm is proposed to take advantage of the coding characteristics in H.264 and recover the lost MB based on 4x4 blocks.

This paper is organized as follows. Section 2 describes the proposed algorithm in detail. Both objective and subjective performance comparisons are made and analyzed in section 3. Finally, section 4 draws the conclusion.

2. THE PROPOSED ERROR CONCEALMENT ALGORITHM FOR H.264

Fig. 2 shows the corresponding block positions of the lost MB and the neighboring MBs. The neighboring available MBs may be encoded by different block sizes as shown in Fig. 2, and the 4x4 blocks close to the lost MB are utilized to recover the lost data in the proposed algorithm. The lost 16x16 MB is recovered based on the 4x4 block in the proposed error concealment approach. In Fig. 2, $M_{(x,y)}$ denotes the locations of the 4x4 blocks in the lost MB. U_i and L_i indicate the nearest upper and lower 4x4 blocks of the upper and lower available MBs neighboring the lost MB, respectively. For simplicity, the illustrations and calculations mentioned below only consider about the recovery of the four top 4x4 erroneous blocks. The four bottom 4x4 blocks are concealed in the same strategy.

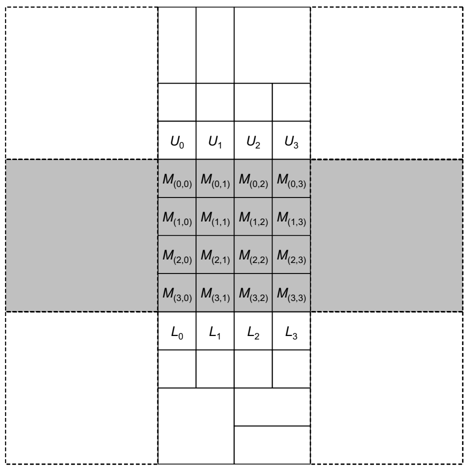


Fig. 2 Corresponding block position and its neighboring MBs.

- *Candidate Set of Motion Vectors*

In order to take advantage of the variable block sizes provided by H.264, the candidate MVs are obtained from the SAD calculations by the 4x1, 8x1 and 16x1 boundaries of the neighboring available MBs. Before deciding the candidate MVs, we first set the search range of the motion estimation. For the top four erroneous 4x4 blocks, $M_{(0,0)}$~$M_{(0,3)}$, (1) and (2) show the determinations in choosing the x and y values of the search range according to the neighboring available 4x4 blocks, U_0~U_3. $U_{i,x}$ and $U_{i,y}$ are the x and y components of MV of U_i, respectively. $D_{x,max}$, $D_{x,min}$, $D_{y,max}$ and $D_{y,min}$ are the search range restrictions for the SAD calculations in (3)~(9). Similarly, for the bottom four erroneous 4x4 blocks, $M_{(3,0)}$~$M_{(3,3)}$, the MVs of L_0~L_3 are utilized to determine the search range restrictions.

$$\begin{aligned} D_{x,max} &= \arg\max\{U_{0,x}, U_{1,x}, U_{2,x}, U_{3,x}\} \\ D_{x,min} &= \arg\min\{U_{0,x}, U_{1,x}, U_{2,x}, U_{3,x}\} \end{aligned} \quad (1)$$

$$\begin{aligned} D_{y,max} &= \arg\max\{U_{0,y}, U_{1,y}, U_{2,y}, U_{3,y}\} \\ D_{y,min} &= \arg\min\{U_{0,y}, U_{1,y}, U_{2,y}, U_{3,y}\} \end{aligned} \quad (2)$$

In Fig. 3, the 4x1 SAD calculations result in four candidate MVs. In addition, the other three candidate MVs are obtained in Fig. 4 and Fig. 5 in the same token. (3)~(6), (7)~(8) and (9) represent the SAD calculations of 4x1, 8x1 and 16x1 side pixels in the boundary, respectively. The candidate set of MVs consists of the seven motion vectors (MV_1 ~MV_7) obtained from (3)~(9). It should be noticed that dx and dy in (3)~(9) are restricted as ($D_{x,min}$,$D_{x,max}$) and ($D_{y,min}$,$D_{y,max}$) in (1) and (2), respectively.

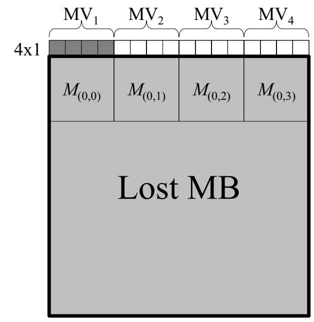


Fig. 3 The illustration of the four candidate vectors obtained by 4x1 SAD calculations.

$$SAD_1(dx,dy) = \sum_{i=0}^{3} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (3)$$

$$SAD_2(dx,dy) = \sum_{i=4}^{7} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (4)$$

$$SAD_3(dx,dy) = \sum_{i=8}^{11} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (5)$$

$$SAD_4(dx,dy) = \sum_{i=12}^{15} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (6)$$

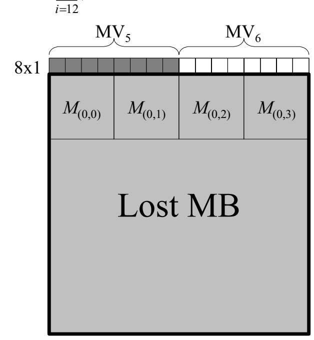


Fig. 4 The illustration of the two candidate vectors obtained by 8x1 SAD calculations.

$$SAD_5(dx,dy) = \sum_{i=0}^{7} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (7)$$

$$SAD_6(dx,dy) = \sum_{i=8}^{15} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (8)$$

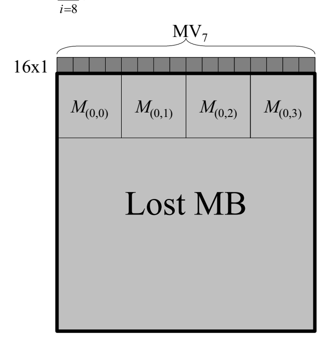


Fig. 5 The illustration of the candidate vector obtained by a 16x1 SAD calculation.

$$SAD_7(dx,dy) = \sum_{i=0}^{15} \left| P(x,y) - P'(x+i+dx, y+dy) \right| \quad (9)$$

- *Side Matching Algorithm*

Each candidate vector has a corresponding 4x4 candidate block in the reference frame for the concealment, and the side matching algorithm (SMA) will choose the best matching block to recover the top and bottom erroneous 4x4 blocks in the lost MB. As shown in Fig. 6, since the proposed method recovers the lost MB based on 4x4 blocks, the SMA is utilized to decide the MV from the MVs in the candidate set according to the 4x1 side pixels of the candidate block. The 4x1 side pixels of the candidate block in the reference frame and the 4x1 side pixels in the current frame are calculated by SMA, which selects the optimal MV to recover the corresponding lost 4x4 block. After the SMA calculations, the top and bottom eight 4x4 erroneous blocks will be concealed.

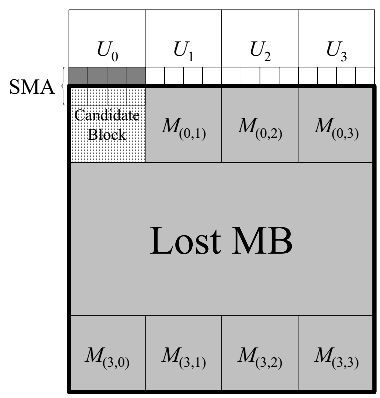


Fig. 6 The SMA calculation for the decision of the optimal candidate block.

- *The Recovery of The Middle 4x4 Blocks*

As mentioned above, the top and bottom eight 4x4 blocks are concealed according to the information of the neighboring upper and lower MBs. Then the middle erroneous eight 4x4 blocks will be filled by the concealed and neighboring 4x4 blocks as shown in Fig. 7. For example, when $M_{(1,0)}$ is in the process of concealment, the relationship between the two MVs of the concealed block $M_{(0,0)}$ and the neighboring block U_0 is utilized to decide the MV for $M_{(1,0)}$. If the two MVs of $M_{(0,0)}$ and U_0 are the same, we assume the MVs is smooth or with low activity, and $M_{(1,0)}$ is compensated by the same MV. Otherwise, when the two MVs are different, the region of the middle blocks may has high activities, the linear interpolation by $M_{(0,0)}$ and $M_{(3,0)}$ will be utilized to bring out the MV for $M_{(1,0)}$. The other middle erroneous blocks are compensated by the same rules. After the procedures of the proposed method, an erroneous MB can be concealed completely.

3. EXPERIMENTAL RESULTS

The H.264 reference software, Joint Video Team (JVT) Model (JM) v9.8[9], is utilized to simulate the different error concealment algorithms. Three QCIF sequences, including Foreman, Carphone and Football, are encoded for simulations at 15Hz frame rate for 100 frames.

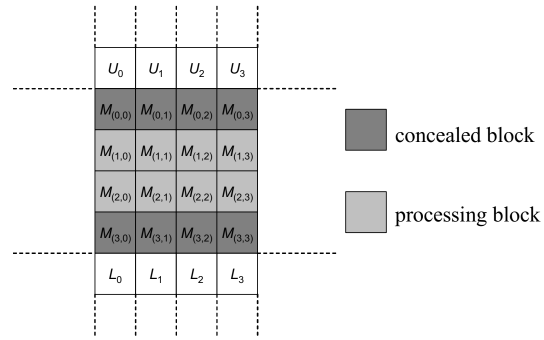


Fig. 7 The middle blocks and the concealed top and bottom blocks.

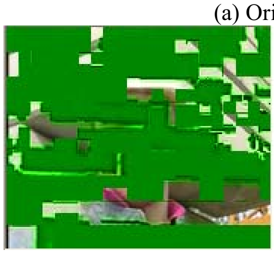
The GOP structure is IPPP… and no B frame is used. There are three different methods compared in the experiments. They are the temporal replacement (TR), the error concealment included in JM (JM) and the proposed approach. The packet loss rates (PLR) at 5%, 10% and 20% are tested to evaluate the performance in the experiments and the peak signal to noise ratio (PSNR) is used as the measurement of the objective performance. The target bitrates utilized to encode the sequences for transmissions are 64, 128 and 256Kbps.

Table Ⅰ demonstrates the objective comparison of the three concealment methods of different PLRs at different bitrates. Obviously, the proposed algorithm can obtain about 0.5 dB benefits in PSNR compared with TR and the reference implementation JM. Fig. 8 and Fig. 9 show the subjective quality comparisons of Foreman with PLR=10% and Carphone with PLR=20% at 128Kbps, respectively. In Fig. 8, although the erroneous frame has enormous error propagation, the proposed method could recover the frame with better subjective quality than other methods, especially in the background. The proposed method could provide better quality in the face of Carphone sequence as shown in Fig. 9.

4. CONCLUSION

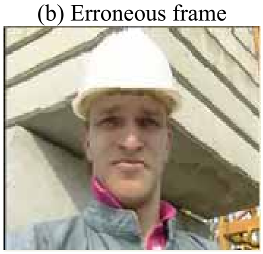
In H.264 standard, we have more information than previous video coding standards. This is very useful for the temporal error concealment. In this paper, we propose an effective temporal error concealment method which exploits the candidate MVs according to the different block sizes in H.264. And then, the spatial correlation SMA is utilized to find the best MV from the candidate set to recover the top and bottom erroneous blocks. For middle lost blocks, the relationship between the neighboring and the concealed 4x4 blocks decides the MV for concealment. Simulation results show that the proposed method can improve both visual quality and PSNR performance of the reconstructed video. Since the proposed algorithm fully utilizes the characteristics of motion estimation in H.264, it is easy to implement and has less time consumption.

(a) Original frame

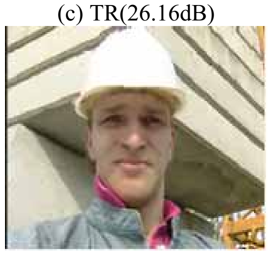


(b) Erroneous frame

(c) TR(26.16dB)



(d) JM(29.12dB)



(e) Proposed(30.08dB)

Fig. 8 Subjective quality comparison for sequence "Foreman" with PLR=10% at 128 Kbps (Frame #15).

(a) Original frame

(b) Erroneous frame

(c) TR(28.84dB)

(d) JM(29.69dB)

(e) Proposed(32.13dB)

Fig. 9 Subjective quality comparison for sequence "Carphone" with PLR=20% at 128 Kbps (Frame #9).

Table I
Average PSNR(dB) comparison for various error concealment methods

Video Sequence	Bitrate (Kbps)	Original PSNR	PSNR (dB)	Packet Loss Rate		
				5%	10%	20%
Foreman	64	35.62	TR	28.46	23.27	22.27
			JM	30.22	25.87	24.27
			PR	31.18	26.97	24.71
	128	38.72	TR	29.03	24.05	21.65
			JM	30.82	26.45	24.58
			PR	31.53	27.44	25.01
	256	42.10	TR	29.28	24.23	21.09
			JM	30.59	26.38	23.44
			PR	30.90	27.42	23.97
Carphone	64	37.90	TR	32.79	26.93	22.93
			JM	33.37	28.85	24.26
			PR	33.51	29.10	25.39
	128	41.59	TR	31.71	26.31	25.69
			JM	33.19	28.49	26.78
			PR	33.76	30.21	28.19
	256	45.08	TR	34.89	26.57	25.05
			JM	36.42	28.48	26.46
			PR	36.66	29.41	27.01
Football	64	23.31	TR	20.40	18.35	16.92
			JM	20.98	18.74	17.18
			PR	21.12	18.82	17.39
	128	27.07	TR	21.07	18.87	16.79
			JM	21.35	19.56	17.49
			PR	21.65	19.72	18.05
	256	30.66	TR	21.60	18.66	16.54
			JM	23.30	19.23	17.39
			PR	23.47	19.38	17.72

5. REFERENCES

[1] JVT G050r1, "Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)," May 2003.

[2] T. Wiegand and G. J. Sullivan, "Overview of the H.264/AC Video Coding Standard," *IEEE Transactions on Circuits and System for Video Technology*, vol.13, pp.560-576, July 2003.

[3] Aign S. and Fazel K., "Temporal and spatial error concealment technique for hierarchical MPEG-2 video codec," *IEEE International Conference on Communications*, Vol. 3, pp.1778-1783, June 1995.

[4] W. M. Lam, A. R. Reilbman and B. Liu, "Recovery of lost or erroneously received motion vectors," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. V417-V420, April 1993.

[5] S. Tsekeridou and I. Pitas, "MPEG-2 error concealment based on block-matching principles," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, pp. 646-658, June 2000.

[6] T. Chen, X. Zhang, and Y. Q. Shi, "Error concealment using refined boundary matching algorithm," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp.560-576, July 2003.

[7] Ye-Kui Wang, M. M. Hannuksela, V. Varsa, A. Hourunranta and A. Gabbouj, "The error concealment feature in the H.26L test model," *IEEE International Conference on Image Processing*, vol. 2, pp. 729-732, September 2002.

[8] Jinghong Zheng and Lap-Pui Chau, "A temporal error concealment algorithm for H.264 using Lagrange Interpolation," *IEEE International Symposium on Circuits and Systems*, vol. 2, pp.133-136, May 2004.

[9] JM 9.8. http://iphome.hhi.de/suehring/tml/