Error Resilience using a Reversible Data Embedding Technique in H.264/AVC

SHINFENG D. LIN, YU-LUNG SU and JHIH-YU HUANG

Department of Computer Science and Information Engineering
National Dong Hwa University, Hualien, TAIWAN 974

Abstract: H.264/AVC is a novel video codec standard that provides a better coding efficiency than other previous video coding standard. In this paper, we provide a new error resilience using a reversible data embedding technique based on H.264/AVC. The algorithm is stretchy that user can make a trade-off between the quality of recovery image and DCT coefficients' variation. The advantage of reversible data embedding is that if the transmission is error-free, the image quality in the encoder can be exactly recovered by H.264 in the decoder. Simulation results demonstrate that the proposed method makes the video coding sequence more robust than the existing techniques.

Keywords: error resilience, reversible data embedding, error concealment.

1. Introduction

H.264/AVC [1] is the latest international video coding standard of the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group. The new standard provides many new techniques to improve the coding efficiency of intra and inter frames. This technique increases the dependence of the neighboring frames. Each inter frame depends on its reference frame, and each intra block depends on its neighboring block. When one block has error, the macroblocks in the successive frames that reference the error macroblock will be affected. And the affected area may increase as the error propagates. It is known that intra frame is the first frame of a GOP (Group of Pictures), the accuracy of intra frame is therefore essential. When a block of intra frame has error, the error may propagate to B, P frame of this GOP. So error concealment and error resilient techniques are developed to solve this problem.

Error resilient methods [2-3] embed spatial and temporal information into image or video itself. Once an error is detected, the error resilient method can extract the embedded information and recover the error block. However, embedding the information for recovery may reduce the video quality. When no error occurs, the quality in the decoder is worse than the original coding of H.264. On the other hand, reversible data embedding, which is also called lossless data embedding, embeds invisible data into a digital image in a reversible manner. In this article, therefore, a new error resilient algorithm using reversible data embedding is proposed. The algorithm focuses on H.264 intra frame, and it affects the quality of all B and P frames in this GOP. If no errors occur, this algorithm will not degrade the quality since it is a lossless data embedding.

The organization of this article is as follows. In Section 2, the algorithm of reversible data embedding using difference expansion [4] and related works are briefly addressed. Section 3 presents the proposed error resilience for intra frame in H.264. Simulation results are demonstrated in Section 4. Finally, Section 5 draws conclusion.

2. Backgrounds and Related Works

The Difference Expansion (DE) algorithm [4] was proposed, it is one of the reversible data embedding technique that embeds invisible data into a digital image.

2.1 The Difference Expansion algorithm

The DE algorithm is a simple and efficient reversible data embedding for digital image. The formulas of the difference expansion algorithm are presented in the following. This algorithm includes bit *b* embedding and restoration.

For bit *b* embedding:

Assume we have two pixel values x and y, we would like to reversibly embed one bit *b*.

$$l := \left\lfloor \frac{x+y}{2} \right\rfloor, h := x - y. \quad (1)$$

$$h' = 2 \times h + b. \quad (2)$$

$$x' = l + \left\lfloor \frac{h'+1}{2} \right\rfloor, y' = l - \left\lfloor \frac{h'}{2} \right\rfloor. \quad (3)$$

where the symbol $\lfloor . \rfloor$ is the floor function meaning "the greatest integer less than or equal to".

For restoration:

From the embedded pair (*x'*, *y'*), we can extract the embedded bit *b* and restore the original pair (*x*, *y*).

$$l' := \left\lfloor \frac{x'+y'}{2} \right\rfloor, h'' := x' - y'. \quad (4)$$

$$b' = LSB(h''), h''' = \left\lfloor \frac{h''}{2} \right\rfloor. \quad (5)$$

$$x'' = l' + \left\lfloor \frac{h'''+1}{2} \right\rfloor, y'' = l' - \left\lfloor \frac{h'''}{2} \right\rfloor. \quad (6)$$

The DE algorithm can be easily understood by the following example. We would like to embed bit *b*(b=1) into the pair *x* = 5, *y* = 3.

Embedding:

$$l = \left\lfloor \frac{5+3}{2} \right\rfloor = 4, h = 5 - 3 = 2.$$

$$h' = 2 \times 2 + b = 5.$$

$$x' = 4 + \left\lfloor \frac{5+1}{2} \right\rfloor = 7.$$

$$y' = 4 - \left\lfloor \frac{5}{2} \right\rfloor = 2.$$

Restoration:

$$l' = \left\lfloor \frac{7+2}{2} \right\rfloor = 4, h'' = 7 - 2 = 5.$$

$$b' = LSB(5) = 1.$$

$$h''' = \left\lfloor \frac{h''}{2} \right\rfloor = \left\lfloor \frac{5}{2} \right\rfloor = 2.$$

$$x'' = 4 + \left\lfloor \frac{2+1}{2} \right\rfloor = 4 + 1 = 5.$$

$$y'' = 4 - \left\lfloor \frac{2}{2} \right\rfloor = 3.$$

Since the original pixel values x and y can be restored, the DE method may be applied many times. However the difference between (x, y) and (*x'*, *y'*) will then be increased.

2.2 Error resilient technique

There are two research topics in error resilience [5-6]. They are error resilient encoding and error concealment techniques. Some related research works of error resilience are briefly addressed below.

2.2.1 Error resilient encoding

In [7], the robust entropy coding, reversible variable length coding, was proposed to do the error resilience. Most of the error resilient techniques belong to the class of error resilient prediction [8-9]. The main ideas of error resilient prediction are information pre-processing and self embedding before encoding. In [10], layered coding with unequal error protection distributed the image/video into base layer and enhanced layer, and protected coding in base layer.

2.2.2 Error concealment

Recovery of texture information [11-12] utilized temporal or spatial information to conceal the error area. Recovery of coding modes and motion vectors [13-14] employed the correlation of spatial or temporal information, and recovered the motion vector or intra prediction mode.

3. The Proposed Error Resilience in H.264

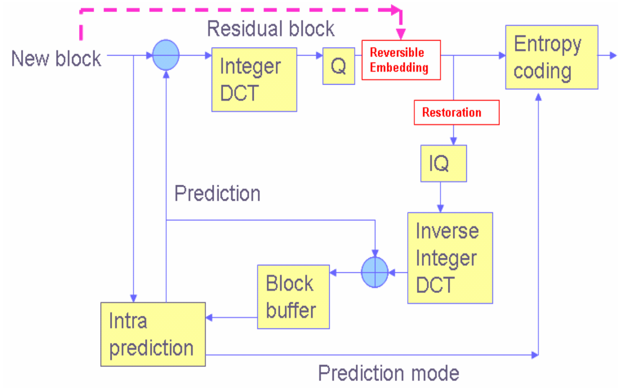

Figure 1 Embedding system in H.264 encoder.

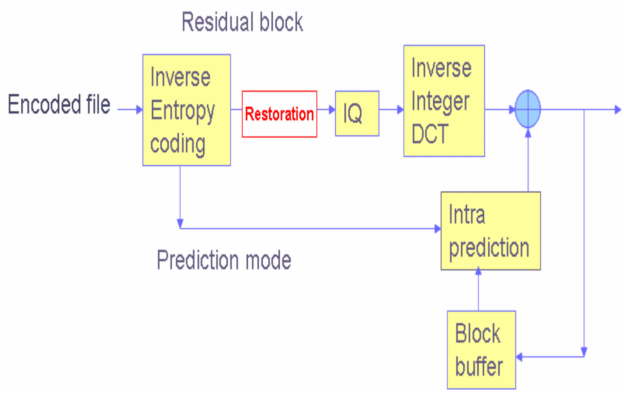

Figure 2 Embedding system in H.264 decoder.

In video coding, intra frame is more important than other (B or P) frames. H.264 is high compression and neighboring dependence. If errors occur, therefore, any change on intra frame will make error propagation effect on B or P frames. Most of error resilient techniques may solve this problem. However, they sacrificed the video quality for embedding image itself. If the transmission is error-free, the image quality can not be exactly recovered by H.264 in the decoder. To solve this problem, a technique combines error resilience and reversible data embedding is proposed. The proposed resilient algorithm employs a reversible data embedding technique in residual DCT coefficients of intra block.

3.1 Proposed scheme I

The embedding system is included in H.264 coding standard. As shown in Figures 1 and 2, the "Reversible embedding" is in H.264 encoder. In addition, the "Restoration" is in H.264 encoder and decoder.

We use a reversible data hiding technique to embed information into integer DCT coefficients. It is noted that a reversible embedding technique, Difference Expansion, hides one bit every two pixels. Therefore a 16x16 macroblock has capacity of 256/2=128 bits. Since DE method can restore the original value and extract the embedded bit, we may apply DE method many times to embed more bits in every two pixel values. However the distortion of two pair values, before and after embedding, will then be increased.

The important information for recovery is extracted from image itself, it includes two choices, pixel luminance and block DCT coefficients. The reason why we choose pixel value instead of DCT coefficients for recovery is because (1)Sign bit: DCT coefficient may be a negative value, so we need sign bit to record it. (2)Big range variation: The range of DCT coefficient value is bigger, so it is hard to use fixed length coding to embed. (3)Variable length coding: Because the range of DCT coefficient is bigger, we must use variable length coding, therefore embed additional symbol table for decoding. (4)High complexity: Using variable length coding will cause more encoder and decoder complexity. We, therefore, select pixel luminance to embed into DCT coefficients.

In the first proposed scheme, as shown in Fig. 3, we select 8x8 pixels to reduce the volume of bit-embedding. Note that four 4x4 sub-blocks of a macroblock are selected to construct an 8x8 pixels block for embedding. For a 16x16 block, its capacity of one time reversible embedding is 128 bits. Since the 8x8 pixels block has 8x8x8=512 bits to embed, this results in 4 times DE embedding. In order to avoid continual block loss, however, we must keep some offset between protected block and embedded block. Figures 3 and 4 illustrate the proposed reversible embedding in the encoder. In Fig. 4, the DCT coefficients are embedded and reversible 4 times.

Fig. 5 shows the restoration in the encoder and the decoder. In the restoration process, the bit stream can be retracted and reversed to have original pixels' luminance. The 16x16 protected block can be recovered using interpolation from a 8x8 block. And the embedded block can then resume its original quality. Using reversible embedding is appropriate, because no overflow or underflow problem occurs if data expansion is made on DCT coefficients. The information of protected block must be embedded into embedded block because if error occurs in this protected block, the information in embedded block can be used to repair this protected block.

The proposed scheme is effective except some shortages. These shortages are discussed as

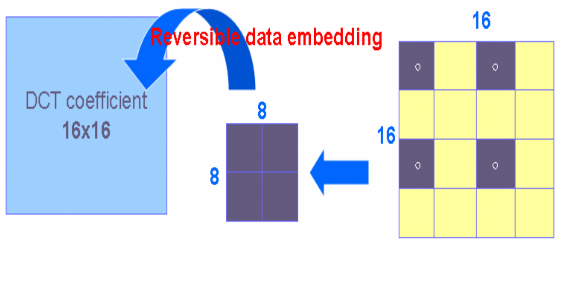

Figure 3 The proposed scheme I.

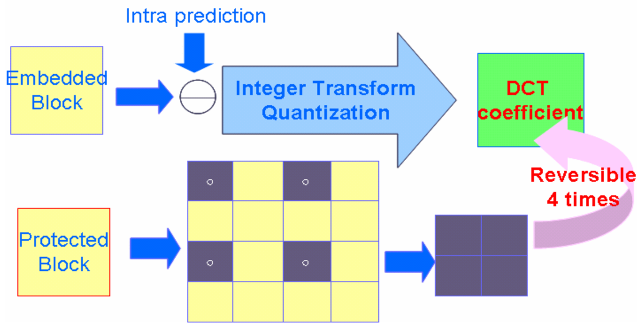

Figure 4 Reversible embedding in the encoder.

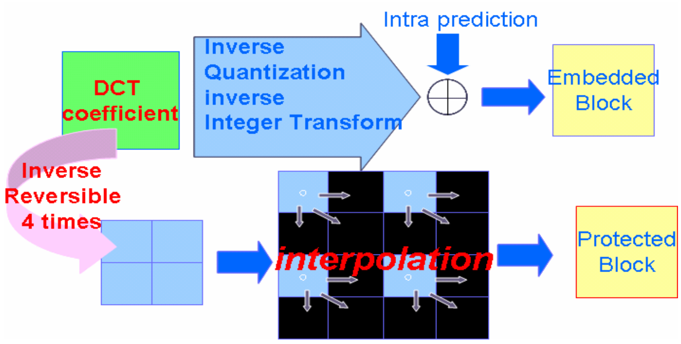

Figure 5 Restoration in the encoder and the decoder.

follows:
(1) Selected pixels for embedding are excessively concentrated on certain area. Using selected pixels to reconstruct the whole block may greatly degrade image quality.
(2) Reversible 4 times makes the difference of DCT coefficients larger. More reversible embedding may increase the burden on entropy coding.

The reconstruction problem can be improved by the proposed scheme II. In addition, entropy coding problem can be improved by the proposed scheme III.

3.2 Proposed scheme II

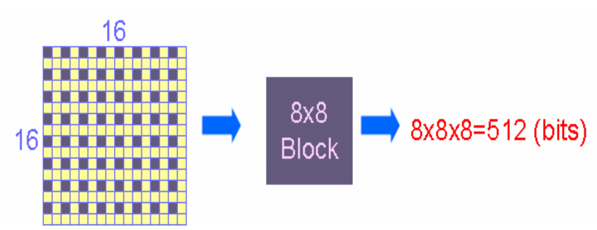

Figure 6 The proposed scheme II.

In order to improve reconstruction problem in scheme I, scheme II uses different method to select embedding bits. Scheme II takes 1 pixel out of every 2 pixel vertically and horizontally, as shown in Figure 6. When reconstructing the damaged block, scheme II provides better image quality than scheme I due to better interpolation. The number of pixels is still 8x8 block and the capacity is the same as that of proposed scheme I.

3.3 Proposed scheme III

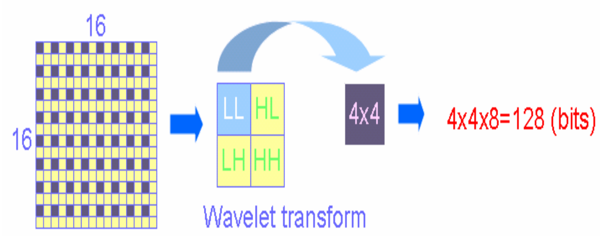

Figure 7 The proposed scheme III.

Fig. 7 illustrates the proposed scheme III. In order to avoid 4 times of DE method which makes the differences of DCT coefficients larger, scheme III reduces the number of embedded pixels. First, 8*8 block is selected according to scheme II. The block is then wavelet transformed and LL band is chosen for embedding. The number of pixels is now 4x4 block and the capacity is $\frac{1}{4}$ of that in previous proposed scheme I.

4. Simulation Results

In our simulation, we used different packet loss rate of Foreman, Carphone video sequences with 176x144 QCIF format and H.264 reference software JM9.5. The peak signal-to-noise ratio ($PSNR = 10 \times \log_{10}(\frac{255^2}{MSE}) dB$) is evaluated at different packet loss rate using the proposed schemes and other existing coding schemes. Comparisons among the proposed schemes at 15% loss rate are presented in Figures 8 and 9. Table 1 and Table 2 list the results of the proposed schemes with different loss rate. Table 3 is the comparison with other techniques. It is noticed that our results are at least 1-2 dB higher than those of the existing techniques. Based on the experiment results, it is concluded that the proposed error resilient technique outperforms the existing techniques.

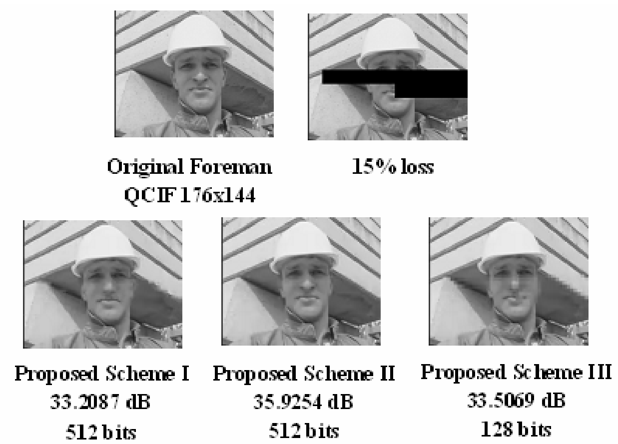

Figure 8 Foreman QCIF with 15% loss rate.

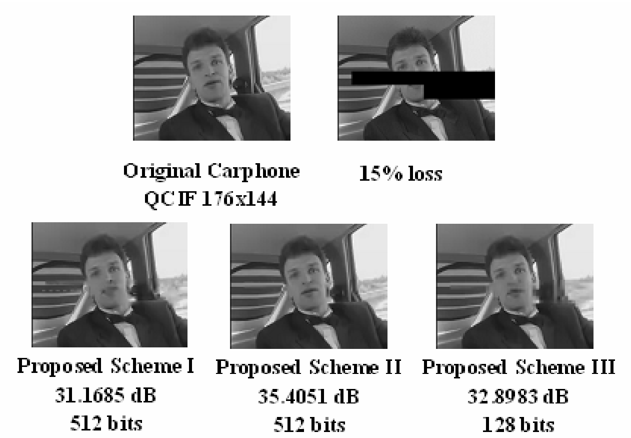

Figure 9 Carphone QCIF with 15% loss rate.

5. Conclusion

Information embedding causes the damage of the original image for traditional error resilient techniques. If no error occurs, the quality resulting from the traditional error resilient techniques is not the same as original image. Error resilience using reversible data embedding is, therefore, an important technique nowadays. Being reversible, the original digital content can be completely restored.

The main contribution of this work is the development of an embedding technique which combines reversible data hiding with H.264 video standard. This paper presents an effective algorithm for error resilience in H.264/AVC. The algorithm is flexible that user can make a trade-off between recovered image quality and DCT coefficients' variation. According to the simulation results, our proposed method makes a significant improvement over the previous error resilient methods. The advantage of reversible data embedding is that if the transmission is error-free, the image quality in the encoder can be exactly recovered by H.264 in the decoder.

Table 1 Foreman QCIF with different loss rate

Loss rate	Scheme I (512 bits)	Scheme II (512 bits)	Scheme III (128 bits)
0%	36.97	36.97	36.97
5%	35.1065	36.4823	35.3131
10%	34.0103	36.1912	34.3364
15%	33.2087	35.9254	33.5069
20%	32.8728	35.5782	32.9903
25%	32.6032	35.5141	32.7401
30%	32.2471	35.3542	32.4313

Table 2 Carphone QCIF with different loss rate

Loss rate	Scheme I (512 bits)	Scheme II (512 bits)	Scheme III (128 bits)
0%	38.341	38.341	38.341
5%	35.8949	37.259	35.5809
10%	32.7421	36.2776	34.1843
15%	31.1685	35.4051	32.8983
20%	30.3492	34.7706	31.5117
25%	29.8763	34.339	30.7624
30%	28.9361	33.7665	29.8521

Table 3 Comparison with other techniques

	Without data embedding			With data embedding			Proposed error resilience schemes		
Loss rate	Zero-S	JM7.3	BNM	DEVCS [16]	ERDE [17]	Kang and Leou [15]	Scheme I	Scheme II	Scheme III
0%	38.47	38.47	38.47	37.46	37.72	37.51	38.341	38.341	38.341
10%	8.95	30.45	31.79	30.53	33.35	35.45	32.7421	36.2776	34.1843
15%	8.87	29.53	30.89	29.62	32.63	34.27	31.1685	35.4051	32.8983
20%	8.6	27.95	29.72	27.94	31.49	33.2	30.3492	34.7706	31.5117

References:

[1] ITU-T Rec. H.264 / ISO/IEC 11496-10, “Advanced Video Coding,” *Final Committee Draft, Document JVTG050*, 2003.

[2] Jinghong Zheng and Lap-Pui Chau, “Error-Resilient Coding of H.264 Based on Periodic Macroblock,” *IEEE Trans. On broadcasting*, vol. 52, no. 2, June 2006.

[3] Chowdary B. Adsumilli, Mylène C. Q. Farias, Sanjit K. Mitra and Marco Carli, “A Robust Error Concealment Technique Using Data Hiding for Image and Video Transmission Over Lossy Channels,” *IEEE Trans. On circuits and systems for video technology*, vol. 15, no. 11, November 2005.

[4] Jun Tian, “Reversible Data Embedding Using a Difference Expansion,” *IEEE Trans. on circuits and system for video technology*, vol. 13, no. 8, August 2003.

[5] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, “Error resilient video coding techniques,” *IEEE Signal Processing Magazine*, vol. 17, no. 4, July 2000, pp. 61-82.

[6] Y. Wang and Q. F. Zhu, “Error control and concealment for video communication: a review,” *Proceedings of the IEEE*, vol. 86, no. 5, May 1998, pp. 974-997.

[7] D. W. Redmill and N. G. Kingsbury, “The EREC: an error-resilient technique for coding variable-length blocks of data,” *IEEE Trans. on Image Processing*, vol. 5, no. 4, April 1996, pp. 565-574.

[8] R. Zhang, S. L. Regunathan and K. Rose, “Video coding with optimal Inter/Intra-mode switching for packet loss resilience,” *IEEE J. on Selected Areas in Communications*, vol. 18, no. 6, June 2000, pp. 966-976.

[9] W. H. J. Chen and J. N. Hwang, “The CBERC: a content-based error-resilient coding technique for packet video communications,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11, no. 8, Aug. 2001, pp. 974-980.

[10] M. Gallant and F. Kossentini, “Rate-distortion optimized layered coding with unequal error protection for robust Internet video,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 11, no. 3, March 2001, pp. 357-372.

[11] Z. Wang, Y. Yu, and D. Zhang, “Best neighborhood matching: an information loss restoration technique for block-based image coding systems,” *IEEE Trans. on Image Processing*, vol. 7, no. 7, July 1998, pp. 1056-1061.

[12] X. Li and M. T. Orchard, “Novel sequential error-concealment techniques using orientation adaptive Interpolation,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 12, no. 10, Oct. 2002, pp. 857-864.

[13] M. J. Chen, L. G. Chen, and R. M. Weng, “Error concealment of lost motion vectors with overlapped motion compensation,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 7, no. 3, June 1997, pp. 560-563.

[14] J. Zhang, J. F. Arnold and M. R. Frater, “A cell-loss concealment technique for MPEG-2 coded video,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 10, no. 4, June 2000, pp. 659-665.

[15] Li-Wei Kang and Jin-Jang Leou, “An Error Resilient Coding Scheme For H.264 Video Transmission Based on Data Embedding,” *IEEE ICASSP 2004.*

[16] J. Song and K. J. R. Liu, “A data embedded video coding scheme for error-prone channels,” *IEEE Trans. On Multimedia*, vol. 3, no. 4, pp. 415-423, 2001.

[17] L. W. Kang and J. J. Leou, “A new error resilient coding scheme for H.263 video transmission,” *in Proc. of IEEE Pacific-Rim Conf. on Multimedia*, Hsinchu, Taiwan, December 2002, pp. 814-822.