EFFICIENT ERROR CONCEALMENT FOR THE WHOLE-FRAME LOSS BASED ON H.264/AVC

Bo Yan[1] *and Hamid Gharavi*[2]

[1]Department of Computer and Information Technology, Fudan University, China
[2]National Institute of Standards and Technology, Gaithersburg, MD, USA

ABSTRACT

For low bitrate video communications, each video frame usually fills the payload of a single network packet. In this situation, the loss of a packet may result in loosing the entire video frame. Currently, most existing error concealment algorithms can only deal with the loss of macroblocks and are not able to conceal the whole missing frame. In this paper, we have proposed a new hybrid motion vector extrapolation (HMVE) algorithm to recover the whole missing frame. The proposed algorithm is capable of estimating the missing motion vectors with much greater accuracy than other existing methods. Experimental results show that it is highly effective and significantly outperforms other existing frame recovery methods.

Index Terms— video error concealment, whole-frame losses, error-resilient video transmission, H.264/AVC

1. INTRODUCTION

H.264/AVC is the latest video compression standard, which achieves a significant reduction in bit rate compared to previous standards [1]. Due to the high compression ratio of H.264/AVC, it is common that an entire coded picture fits the packet size when transmitting low resolution sequences at low bitrates. Since packetization will lead to significant overhead, usually one coded frame data will be packetized into one RTP packet in order to save the bit rate. In such a scenario, the loss of an RTP packet results in the loss of a whole frame. Because of the temporal prediction, the loss of a frame can significantly affect the quality of subsequent frames in H.264/AVC.

In order to combat channel errors, frame concealment is an effective method to mask the effect of missing frames by creating subjectively acceptable images. So far, many frame concealment methods have been proposed [2–8]. These methods generally generate the MVs of the lost frame with the optical flow estimation [2–6], or the motion vector extrapolation (MVE) [7, 8]. The drawback is that the quality of the concealed frame is not satisfactory.

This work is supported in part by NSFC(Grant No.: 60703034), and in part by Shanghai Pujiang Program(Grant No.: 07PJ14017).

In this paper, we propose a new hybrid motion vector extrapolation (HMVE) algorithm to give more accurate estimation for the motion vectors (MVs) of the lost frames, than other conventional methods in order to conceal the lost frames. Experimental results show that the HMVE algorithm significantly outperforms other existing methods in concealing the lost frame.

The rest of this paper is organized as follows. In Section 2, we briefly introduce the existing methods. In Section 3, we propose our HMVE algorithm, which can improve the decoded video quality significantly after transmission over error-prone channels. Then we evaluate the proposed method by simulations and present the results in Section 4. Finally, in Section 5, we draw the conclusions.

2. CONVENTIONAL METHODS

In the past few years, some frame concealment techniques have been proposed to combat frame loss during video transmission [2–8]. Based on optical flow, Belfiore proposed a method to conceal the lost frames [3]. This method usually provides a relatively good quality, however its performance is not always better than the frame copy (FC). In addition, it's hard to determine the number of the reference frames while calculating the optical flow of each pixel.

MV extrapolation (MVE) is a simple but efficient way to achieve the MV of the lost frame [7]. In this method, the MVs of macroblocks (MBs) are extrapolated from the last decoded frame to the missing frame. This method is able to overcome the disadvantage of incorrect MB displacement, but the block (8×8) based MV is too rough to cause block artifacts.

Chen proposed a pixel-based MVE (PMVE) method to conceal the missing frame, which is able to get the MV by extending the MV extrapolation (MVE) method to the pixel level [8]. The pixels in the missing frame can be divided into two parts:

- For a pixel which is covered by at least one extrapolated MB, the MV is estimated by averaging the MVs of all overlapped MBs.
- For a pixel which is not covered by any of the extrapolated MBs, the MV is duplicated from the MV of the

same pixel in the previous frame [8].

If the estimated MV is $MV = (MV_x, MV_y)$, each missing pixel $p_m(x, y)$ can be recovered as follows:

$$p_m(x, y) = p_r(x + MV_x, y + MV_y) \quad (1)$$

where $p_r(x, y)$ refers to pixels in the previous frame [8].

This PMVE method provides a similar performance as block-based MVE does in little motions, but it is able to improve the performance greatly in large motion scenes.

3. THE PROPOSED HMVE ALGORITHM

Although the PMVE method provides a better performance, it does have a shortcoming. The MVs of the lost pixels, which are extrapolated from the MVs, may not be accurate. Some MVs are very likely to be extrapolated wrongly, especially in large motion scenes. This shortcoming will damage the accuracy of the MV for the pixel, thus the performance will be degraded. In order to overcome this problem we propose a hybrid MVE (HMVE) method based on PMVE, which uses not only the extrapolated MVs of the pixels, but also the extrapolated MVs of the blocks. This proposed algorithm is able to discard the wrongly extrapolated MVs in order to obtain the accurate MV.

In H.264/AVC, the smallest unit for motion estimation and compensation is a 4×4 block [1]. Thus in our proposed HMVE algorithm, we use a 4×4 block as the concealment unit as shown in Fig. 1.

As opposed to PMVE, HMVE divides the pixels of the missing frame into three parts:

- Part A: pixels that are covered by at least one extrapolated 4×4 block. For example, in Fig. 1 Part A includes pixels of $\{1, 2, 3, 4, 5, 6, 7, 9, 13, 14, 15\}$ in the concealed block$_1$.
- Part B: pixels that are not covered by any of the extrapolated 4×4 blocks. But the block which the pixel belongs to, has the overlapped area with the extrapolated block. For example, pixels of $\{8, 10, 11, 12, 16\}$ in the concealed block$_1$ belong to Part B.
- Part C: pixels that are not covered by any of the extrapolated 4×4 blocks. And the block which the pixel belongs to, doesn't overlap with the extrapolated block. For example, all the pixels in the concealed block$_2$ belong to Part C.

In order to discard the wrongly extrapolated MVs, HMVE will use a new scheme to estimate the MV of each pixel.

Firstly, two possible MVs of each block in the missing frame will be estimated by the MVE method. As shown in Fig. 1, the MV of the block in the missing frame is estimated according to the extrapolated blocks, which occupy the missing block. The number of pixels in the overlapped areas (as shown in Fig. 1) is used to obtain the weight for the estimation. Let EB_n^j denotes the extrapolated 4×4 block from the jth block in the reference frame to the missing frame n, $MV(EB_n^j)$ denotes the MV of EB_n^j, and B_n^i denotes the ith 4×4 block in frame n. Then the weight is given by [7]:

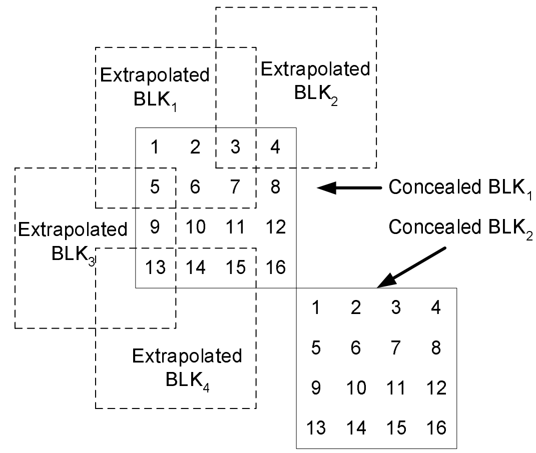

Fig. 1. Our proposed hybrid motion vector extrapolation.

$$w_n^{i,j} = \sum_{p \in B_n^i} f_j(p), \qquad i, j = 1, 2, \cdot \;\cdots, M \quad (2)$$

where

$$f_j(p) = \begin{cases} 1\ , & p \in EB_n^j \\ 0\ , & p \notin EB_n^j \end{cases} \quad (3)$$

M is the total number of blocks in a video frame.

Then two possible MVs of the missing block B_n^i are obtained, which are denoted by $MV_m(B_n^i)$ and $MV_a(B_n^i)$. $MV_m(B_n^i)$ is obtained by selecting the MV of the extrapolated block with the maximum weight $w_n^{i,j}$. $MV_a(B_n^i)$ is obtained by the weighted mean value of the MVs of all the overlapped extrapolated blocks. They are obtained as follows.

- $MV_m(B_n^i)$:

$$MV_m(B_n^i) = MV(EB_n^{j^*}) \quad (4)$$

where

$$j^* = \arg\max\{w_n^{i,j}\}, \qquad j = 1, 2, \cdot \;\cdots, M$$

- $MV_a(B_n^i)$:

$$MV_a(B_n^i) = \frac{\sum_{j=1}^{M} MV(EB_n^j) w_n^{i,j}}{\sum_{j=1}^{M} w_n^{i,j}} \quad (5)$$

If block B_n^i has not been overlapped by any extrapolated blocks, the MV of this block is null.

Secondly, an extrapolated MV set $MVS_p(P_n^{x,y})$ for each pixel is obtained after extrapolation. $P_n^{x,y}$ represents the pixel with the coordinate (x, y) in frame n. After extrapolation, pixels in Part A (as shown in Fig. 1) are covered by at least one extrapolated block, thus the $MVS_p(P_n^{x,y})$ of them includes the MVs of all overlapped extrapolated blocks. For pixels in Part B and C, the $MVS_p(P_n^{x,y})$ is null.

Then the new MV set $MVS(P_n^{x,y})$ will be constructed for different types of pixels as follows;

- For pixels in Part A:

$$MVS(P_n^{x,y}) = \{MV_m(B_n^i), MV_a(B_n^i), MVS_p(P_n^{x,y})\} \quad (6)$$

  where $P_n^{x,y} \in B_n^i$.

  In this MV set, some MVs may be obtained due to the incorrect extrapolation. In order to get the more accurate MV, the wrongly extrapolated MVs should be discarded. Let $MV(i)$ denote the ith component in $MVS(P_n^{x,y})$. Firstly, HMVE calculates the distances $Dis(i,j)$ between $MV(i)$ and other MVs, $MV(j)$, in $MVS(P_n^{x,y})$. Then $num(i)$ is calculated based on $Dis(i,j)$ which denotes the number of $Dis(i,j)$ values that are less than the predefined threshold T for $1 \leq j \leq N$ and $j \neq i$. Finally, only the $MV(i)$, which satisfies the condition of $num(i) \geq N-1$, will be considered as the true MV candidate. Other MVs will be discarded from $MVS(P_n^{x,y})$ as the wrongly extrapolated MVs. In this case, the accurate MV set is obtained.

- For pixels in Part B:

$$MVS(P_n^{x,y}) = \{MV_m(B_n^i), MV_a(B_n^i)\} \quad (7)$$

- For pixels in Part C:

$$MVS(P_n^{x,y}) = \{MV(P_{n-1}^{x,y})\} \quad (8)$$

  where $MV(P_{n-1}^{x,y})$ is the MV of the same pixel in the previous frame.

Finally, the MV of each pixel $MV = (MV_x, MV_y)$ is estimated by averaging the components of the MV set $MVS(P_n^{x,y})$. With help of the estimated MV, each missing pixel can be recovered as shown in (1).

4. SIMULATION RESULTS

To evaluate the proposed algorithm, experiments are implemented using the JM10.2 H.264/AVC codec for two QCIF video sequences, "Mobile" and "Bus". The frame rate is 30 frames/s and the period of I frame reset is 15. A constant QP of 22 is maintained for all frames. As for the packetization scheme, all the compressed video streams related to one frame are stuffed into one packet. This is a typical condition for streaming RTP/UDP video over a network in which all packets have the same priority. In this case, packet loss means frame loss.

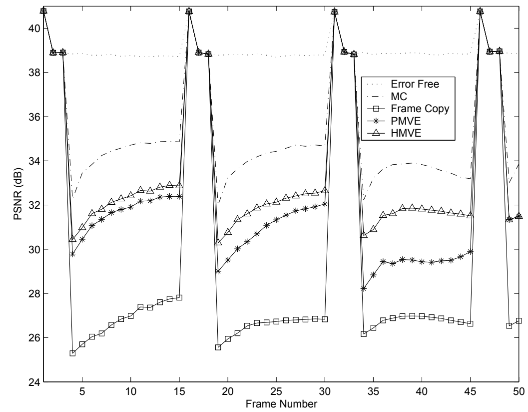

Fig. 2. "Mobile" sequence PSNR comparison vs. frame number.

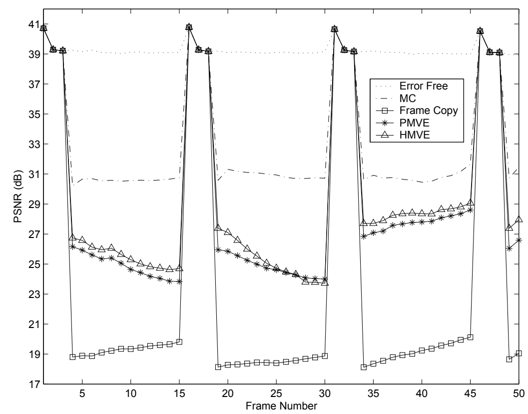

Fig. 3. "Bus" sequence PSNR comparison vs. frame number.

In this simulation, we compare performances of the proposed HMVE algorithm with PMVE, frame copy (FC) and motion compensation (MC). FC means copying the previous frame directly for concealment. MC means that the original MVs are correctly received, but the residual information is lost. Therefore, MC could be regarded as the "upper bound" while comparing the performances of temporal error concealment methods. In this simulation, a P-frame is dropped in each GOP. The dropped frame is then concealed by MC, FC, PMVE and HMVE respectively. Their corresponding PSNR values are calculated and compared. Fig. 2 and Fig. 3 show the simulation results with PSNR vs. frame number for different test sequences. Experimental results report that the proposed HMVE algorithm significantly outperforms FC and PMVE for different sequences.

For clearer comparison, Table 1 presents the average PSNR performances over the erroneous frames only, which are defined as frames corrupted by the frame losses. In this table, Gain1 and Gain2 are the gains that HMVE improves over FC and PMVE respectively. As shown in this table, the

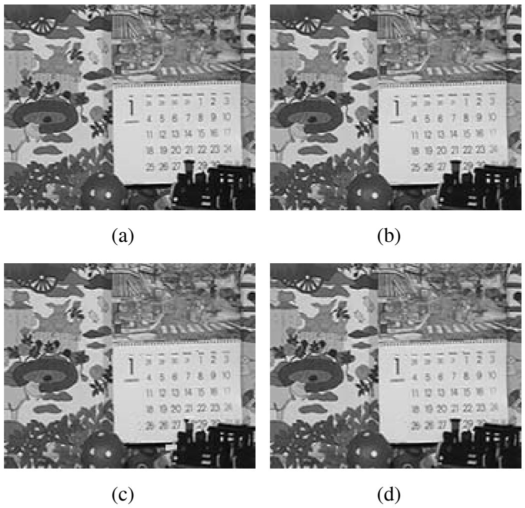

Fig. 4. Restored 34th frame of ”Mobile” sequence. (a)error-free frame (38.91 dB); (b)FC (26.16 dB); (c)PMVE (28.22 dB); (d)HMVE (30.62 dB. Gains over FC and PMVE are 4.46 dB and 2.4 dB respectively).

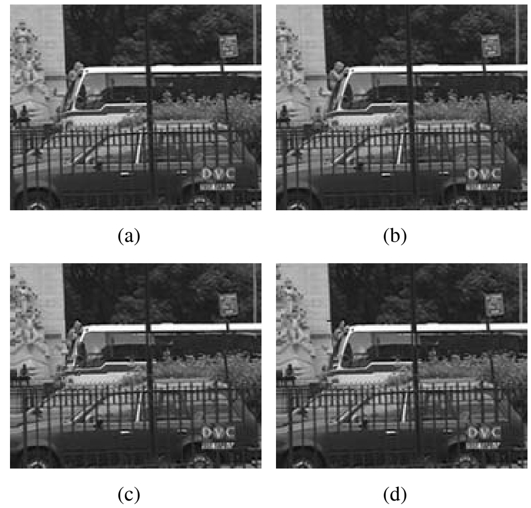

Fig. 5. Restored 19th frame of ”Bus”sequence. (a)error-free frame (39.19 dB); (b)FC (18.13 dB); (c)PMVE (25.95 dB); (d)HMVE (27.38 dB. Gains over FC and PMVE are 9.25 dB and 1.43 dB respectively).

Table 1. Comparison of the Average PSNR Performance Over Erroneous Frames Only.

Sequence	PSNR (dB)			Gain (dB)	
	FC	PMVE	HMVE	Gain1	Gain2
Mobile	25.89	29.59	30.67	4.79	1.09
Bus	18.43	26.25	27.30	8.88	1.06

proposed HMVE algorithm yields higher PSNR performance than PMVE and FC, and is able to provide up to 8.88 dB and 1.09 dB better PSNR performances than FC and PMVE respectively.

For subjective evaluation, Fig. 4 and Fig. 5 show the results of one frame extracted from different sequences, where (a) is the error-free frame, (b) to (d) are the images reconstructed using FC, PMVE and the proposed HMVE respectively. Due to discarding the wrongly extrapolated MV, the proposed HMVE is able to capture local motions more accurately than PMVE and leads to a concealed frame with less block artifacts. In Fig. 4 and Fig. 5, it can be seen that the image quality is consistent with the PSNR measurement and the visual improvements are observed to be more significant than the PSNR improvements. In these two figures, (d) is perceptually superior to (b) and (c), especially around the edges of the image objects.

5. CONCLUSIONS

A new frame concealment algorithm, namely HMVE, is proposed to combat frame loss during video transmission over error-prone networks. As opposed to conventional methods, HMVE constructs a new MV set and discards the wrongly extrapolated MV. Then the MV of each pixel in the lost frame is estimated more accurately than with other conventional methods. It is capable of significantly improving the video quality that has been corrupted by transmission errors.

6. REFERENCES

[1] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, “Draft ITU-T recommendation and Final Draft International Standard of Joing Video Specification (ITU-T Rec. H.264 - ISO/IEC 14496/10 AVC),” *IEEE Transactions on Image Processing*, May 2003.

[2] Z. Wu and J.M. Boyce, “An error concealment scheme for entire frame losses based on H.264/AVC,” in *Proc. of IEEE ICASSP’06*, May 2006.

[3] S. Belfiore, M. Grangetto, E. Magli, and G. Olmo, “Concealment of whole-frame losses for wireless low bit-rate video based on multiframe optical flow estimation,” *IEEE Transactions on Multimedia*, vol. 7, no. 2, pp. 316 – 329, Apr. 2005.

[4] P. Baccichet, D. Bagni, A. Chimienti, L. Pezzoni, and F.S. Rovati, “Frame concealment for H.264/AVC decoders,” *IEEE Transactions on Consumer Electronics*, vol. 51, no. 1, pp. 227–233, Feb. 2005.

[5] Z. Zhou and S. Xie, “Error concealment based on robust optical flow,” in *Proc. of International Conference on Communications, Circuits and Systems*, May 2005, vol. 1, pp. 547 – 550.

[6] C.M. Huang, K.C. Yang, and J.S. Wang, “Error resilience supporting bi-directional frame recovery for video streaming,” in *Proc. of IEEE ICIP’04*, Oct. 2004, vol. 1, pp. 537–540.

[7] Qiang Peng, Tianwu Yang, and Changqian Zhu, “Block-based temporal error concealment for video packet using motion vector extrapolation,” in *Proc. of IEEE International Conference on Communications, Circuits and Systems and West Sino Expositions*, June 2002, vol. 1, pp. 10–14.

[8] Y. Chen, K. Yu, J. Li, and S. Li, “An error concealment algorithm for entire frame loss in video transmission,” in *Proc. of IEEE Picture Coding Symposium (PCS’04)*, 2004.