

MULTIFRAME ERROR CONCEALMENT FOR WHOLE-FRAME LOSS IN H.264/AVC

Yi Liu, Jiajun Bu*, Chun Chen, Linjian Mo, and Kewei He

College of Computer Science, Zhejiang University, Hangzhou, P. R. China
bjj@zju.edu.cn

ABSTRACT

In this paper, we propose a novel error concealment algorithm for the whole-frame loss based on multiframe. In concealing procedure, besides improve the quality of the concealed lost frames, the proposed algorithm also guarantees the quality of the succeeding frames that reference the lost frame in order to control the propagation of errors. Macroblocks in the lost frame are concealed in an appropriate order which is determined by exploiting the information from partly decoded succeeding frames. Experimental results demonstrate that the proposed algorithm not only outperforms previous algorithms in both PSNR evaluation and visual quality of the concealed lost frames, but also minimizes the error propagation.

Index Terms—Error concealment, multiframe video coding, error propagation, whole-frame loss, video transmission

1. INTRODUCTION

In order to combat the errors caused by packet losses during video transmission, lots of techniques have been developed. Error concealment (EC) is one of them, which recovers lost areas by postprocessing based on characteristics of image and video signals.

In low-bitrate video transmission, one data packet usually carries a whole frame. Since the compress size of one video frame can be very small, partitioning one frame into several slices or packeting one frame into several data packets will introduce a significant bits and transmission overhead. Consequently, the loss of a packet directly results in a whole-frame loss. On the other hand, in high-bitrate applications, burst packet losses caused by traffic congestion, also lead to whole-frame losses. Therefore, whole-frame loss EC algorithms are needed for such scenarios.

Furthermore, prediction-based video coding techniques are very sensitive to packet losses, because the error in the lost area not only degrades the quality of reconstructed frames, but also propagates to succeeding frames. Thus, an EC procedure should not only minimize the immediate impact of packet loss on the current frame, but should also minimize the propagation of errors to succeeding frames.

In this paper, we proposed a novel EC algorithm for whole-frame loss in H.264/AVC based on multiframe. The

multiframe video coding is supported by H.264/AVC standard. It offers a high coding efficiency as well as benefits error controlling in video transmission, especially for online encoding applications.

The main idea of the proposed algorithm is that both the quality of the lost frame and its succeeding frames are guaranteed, in order to control propagation of errors. Macroblocks (MBs) in the lost frame are concealed in an appropriate order which is determined by exploiting the information from partly decoded succeeding frames. Extensive experiments have demonstrated that the proposed algorithm not only outperforms previous algorithms on objective and subjective quality of concealed lost frames, but also minimizes the error propagation.

After discussing related work in section 2, we describe the proposed algorithm in section 3. Experimental results and conclusion are presented in section 4 and 5 respectively.

2. RELEATED WORK

Many EC techniques have been developed for error control in the past decade [1-3]. These techniques which based on temporal or spatial interpolation are capable of recovering the lost area while part of the MBs in one frame or slice is lost. However, when a whole-frame loss is encountered at the decoder, their effects are limited.

S. Belfore et al. proposed an EC algorithm for the whole-frame loss based on multiframe optical flow estimation (CAP) [4]. The algorithm exploits motion vector (MV) information of last received frames and estimates the motion of the lost frame. Then last received frames are projects onto the lost one. One weakness of CAP is that the complexity is too high for realtime implementation, due to large numbers of filter operations in the pixel domain.

P. Baccichet et al. proposed a block based whole frame EC algorithm (CAB), which significantly decreases the complexity of CAP in tradeoff with a little PSNR loss [5]. However, both CAP and CAB only consider the quality of the recovered lost frame, yet without analyzing the error propagation in succeeding frames.

In [6], Y. Lee et al. analyzed the error propagation in succeeding frames and proposed two multiframe EC algorithms. The first one is based on spatial interpolation. It estimates the smoothness of the recovered area and blocks that reference the lost area in succeeding frames.

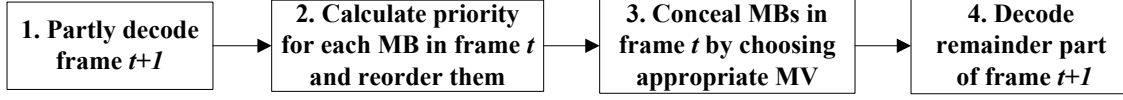


Fig. 1. Main process of proposed error concealment algorithm when frame t has lost

The second one conceals the lost MBs by recovering the MV based on temporal redundancy. These two algorithms both need lots of information from the spatial neighboring of the lost area, which is few or even unavailable in the whole-frame loss scenario. Furthermore, most aforementioned algorithms conceal the lost area in scan order, which should be modified for EC algorithms based on multiframe in order to increase available guidance information for concealing.

3. OUR PROPOSAL

We assume that the current received P frame is F_{t+1} , and the last correctly decoded P frame is F_{t-1} , then decoder can detect that P frame F_t has lost.

In this paper, we use multiple frames $F_{t+1} \dots F_{t+L}$ to conceal F_t , where L denotes the number of succeeding frames of F_t . F_t is in the range from 0 to N_{ref} which represents the maximum number of reference frames used in decoding. When L is set to 0, the EC algorithm becomes single frame based. When L is larger than 1, extra frame buffer is needed at the decoder, and F_t has to be concealed after more frames such as F_{t+2} , F_{t+3} had been received. That leads to the delay presentation of F_t , which may be intolerable for realtime applications. Thus, we set L to 1 in our experiments. Further discussion for L is presented at the end of this section.

Fig. 1 depicts the main process of the proposed algorithm. In the first step, we decode those 4×4 blocks in F_{t+1} whose reference frame precedes F_t , such as F_{t-1} , F_{t-2} , etc.

In the second step, in order to get a decoding order of MBs in F_t where adequate information from succeeding frames could be used, we calculate a priority value for each MB in F_t . First we define several symbols here:

mb_i : a MB in F_t , i is a integer between 0 and numb of MBs in one frame.

b_j : a 4×4 block in F_{t+1} , j is a integer between 0 and number of 4×4 blocks in one frame.

P_i : priority of mb_i , initialized to 0.

NB_j : number of decoded neighbors of b_j .

For each undecoded b_j in F_{t+1} , we get the reference area of b_j in F_t . Notice that the reference frame of undecoded blocks in F_{t+1} must be F_t unless it belongs a MB that encoded in INTRA mode. The reference area is rounded to integer pixel position in F_t . Priority of MBs in F_t that is covered by the reference area is increased by the covering pixel number multiplies NB_j . Fig. 2 illustrates an example of step 2. Two 4×4 blocks b_0 , b_1 and their neighbors in F_{t+1} are shown in the right part where

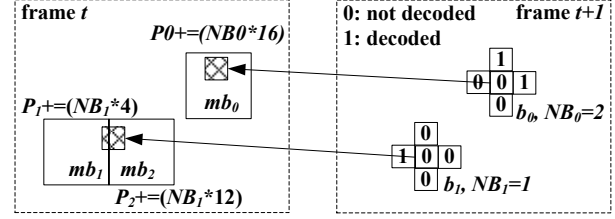


Fig. 2. Priority calculation for each MB in step 2.

undecoded blocks are marked with 1. Shadowing blocks in the left part are their reference areas. Specifically, the reference area of b_0 is all in mb_0 , while reference area of b_1 covers 4 pixels in mb_1 and 12 pixels in mb_2 . Thus P_0 , P_1 , P_2 are increased by $NB_0 * 16$, $NB_1 * 4$ and $NB_1 * 12$ respectively.

Then, MBs in F_t are reordered by the priority. They will be concealed in descent order in step 3. Let MVG_i denote the motion vector group (MVG) of mb_i . MVG is a group of MVs for a MB according to how the MB is partitioned. For example, it may contain two MVs for one MB with two 16×8 partitions, or four MVs for one MB with four 8×8 partitions.

For each mb_i , a set of candidate MVGs are tested to decode mb_i . The MVG set MVC_i for mb_i includes:

- MVG of the spatially corresponding MB in F_{t-1} , F_{t+1} ;
- average of aforementioned two MVGs;
- MVG of decoded neighboring MB in F_t .
- zero motion vector.

Assume that now $mv_{i,k}$ in MVC_i is chosen as the MVG of mb_i . Then mb_i will be partitioned into several partitions. The number of these partitions $N_{part,i,k}$ is equal to the number of MVs in $mv_{i,k}$. For example, in Fig. 3 (a), mb_0 is partitioned into two 16×8 parts. Each part has a single MV. After concealing mb_i using $mv_{i,k}$, all undecoded 4×4 blocks in F_{t+1} that become decodable will be decoded. As shown in Fig. 3(a), blocks in F_{t+1} whose reference area is A, B and C will become decodable after mb_0 in F_t had been concealed. Let $N_{block,i}$ denote the number of such blocks. Then we estimate the boundary effects in F_t and F_{t+1} respectively:

$$BV_{i,k}^t = \sum_{part}^{N_{part,i,k}} \sum_{dir} bv_{part,dir}^t, dir \in \{LEFT, RIGHT, TOP, BOTTOM\} \quad (1)$$

$$BV_{i,k}^{t+1} = \sum_{block}^{N_{block,i}} \sum_{dir} bv_{block,dir}^{t+1}, dir \in \{LEFT, RIGHT, TOP, BOTTOM\} \quad (2)$$

where dir represents four neighboring directions. And for a $m \times n$ size block, bv_{LEFT}^t and bv_{LEFT}^{t+1} are defined as:

$$bv_{LEFT}^t = \begin{cases} \sum_{y=0}^{m-1} (I_{y,0} - I_{y,-1})^2, & \text{if left block is in another available and decoded MB;} \\ 1 - \sum_{y=0}^{m-1} (I_{y,0} - I_{y,-1})^2, & \text{if left block is in the same MB;} \\ 0, & \text{otherwise;} \end{cases} \quad (3)$$

$$bv_{LEFT}^{t+1} = \begin{cases} \sum_{y=0}^m (I_{y,0} - I_{y,-1})^2, & \text{if left block is decoded} \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $I_{y,x}$ is the intensity value for each pixel, as shown in Fig. 3 (b). bv_{dir}^t and bv_{dir}^{t+1} in other directions are defined in a similar way. We will not present them here to save space.

When all candidates in MVC_i had been tested for mb_i , we get a variable BV which represents the total boundary effects for each MVG. And the MVG with minimum BV is chosen for mb_i :

$$BV_{i,k} = \frac{BV_{i,k}^t}{\min\{BV_i^t\}} + \frac{BV_{i,k}^{t+1}}{\min\{BV_i^{t+1}\}} \quad (5)$$

$$MVG_i = \arg \min_{mvg_{i,k}} \{BV_{i,k}\} \quad (6)$$

Specially, if there are several MVGs whose BV value is similar to the minimum BV , we will choose the one who has the minimum BV^{t+1} from them for mb_i . Since BV^{t+1} is estimated by more accurate information from correctly decoded pixels in succeeding frames. Equation (6) becomes:

$$MVG_i = \arg \min_{mvg_{i,k}} \{BV_{i,k}^{t+1} \mid \frac{BV_{i,k}}{\min\{BV_i\}} < T_{range}\} \quad (7)$$

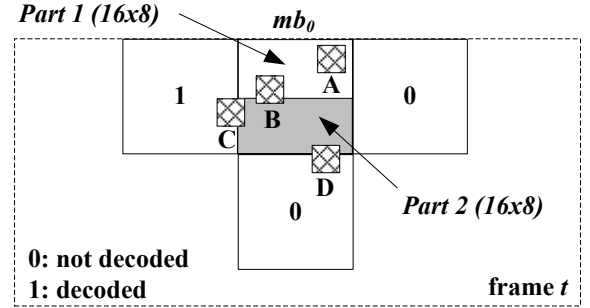
where T_{range} is a threshold that represents the range of similarity. Through experiments, we find that 1.1 is a suitable value for T_{range} .

Finally, in step 4, remainder 4x4 blocks that belong to MB encoded in INTRA mode are decoded in scan order.

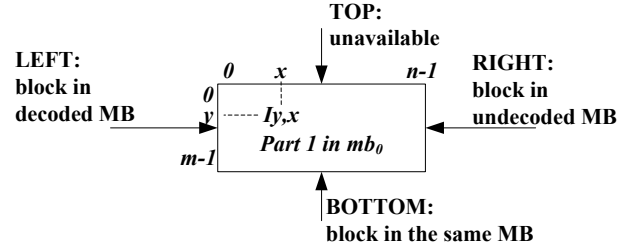
Further discussion of L

The proposed algorithm can be easily extended to support larger L by adding more frame buffers at decoder, and estimating boundary effects for more partly decoded succeeding frames. The error propagation could be further minimized with larger L , due to more available information could be utilized in succeeding frames. However, the extra memory needed and the delay introduced to presentation of lost frames cannot be neglected.

An alternative approach to increase useful information for decoder with constrained memory and realtime requirement is utilizing error resilience at the encoder. Assume that current encoding frame is frame t , the encoder can guide some blocks of frame t to reference frame $t-2$ more. It will benefit the proposed EC algorithm at decoder while frame $t-1$ is lost. How to choose appropriate blocks to



(a) A top row MB mb_0 and its neighbors MBs in frame t .



(b) Enlargement of part 1 of mb_0 .

Fig. 3. Illustration of concealing MB in step 3.

adjust the reference frame in order to maintain the coding efficiency can be a piece of future work.

4. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed EC algorithm, we applied the proposed algorithm to a H.264 video codec (JM 10.2 [7]) and did plenty of experiments on the well-known test sequences. All sequences are in a QCIF size and 15 fps. The bitrate is set to 64 kbps and 128 kbps, which are typical wireless video transmission scenarios [8]. Moreover, GOPs are employed and GOP size is set to 30.

First we compare the PSNR of concealed lost frames and their succeeding frames. Note that COPY stands for the EC algorithm that copies the last frame to the lost frame. Fig. 4 depicts the comparison when frame 257 in FOREMAN and frame 46 in TABLETENNIS have lost. The proposed algorithm not only improves the objective quality of the lost frame, but also obtains a faster recovery from packet losses.

For subjective evaluation, frame 67 in FOREMAN and frame 37 in TABLETENNIS concealed by CAP and the proposed algorithm are shown in Fig. 5. Image (c) and (d) that are recovered by CAP possess severe distortions in the face area and table area respectively. Obviously, Image (e) and (f) that are recovered by the proposed algorithm provide much better visual quality.

The average PSNRs of concealed frames under different packet loss rate are reported in Table 1. It can be seen that the proposed algorithm outperforms CAP and COPY under all conditions, particularly when decoding the sequence with complicate motion and high loss rate.

Table 1. Average PSNR under different packet loss rate

Sequence	Bitrate (kbps)	PLR = 5%			PLR = 10%			PLR = 20%		
		COPY	CAP	Proposed	COPY	CAP	Proposed	COPY	CAP	Proposed
Foreman	64	31.19	31.96	32.72	28.77	30.15	31.91	24.41	27.50	30.01
	128	33.45	34.39	35.91	30.04	31.67	33.74	24.41	27.58	30.35
News	64	37.79	37.48	38.26	35.73	35.76	36.58	31.61	32.51	33.53
	128	39.80	39.23	40.53	37.12	37.18	38.32	30.50	32.12	33.53
TableTennis	64	26.52	27.38	27.77	24.89	25.87	26.69	22.18	23.15	24.03
	128	27.77	28.78	29.55	25.70	26.33	27.31	22.30	23.42	24.64
Coastguard	64	25.70	25.98	26.22	24.56	24.67	25.16	22.16	23.68	23.76
	128	27.45	27.73	28.37	25.96	26.21	26.84	22.53	24.32	24.95

5. CONCLUSION

In this paper, a novel multiframe EC algorithm for whole-frame loss has been proposed. The algorithm exploits information from partly decoded succeeding frames, to determine an appropriate order for concealing MBs in lost frame, in order to improve the performance of concealment as well as combat the error propagation. Future work includes jointly work on the encoder and decoder, and designing efficient error resilient algorithms based on multiframe in order to further improve the proposed EC algorithm .

6. REFERENCES

[1] Y. Wang, Q. F. Zhu, "Error control and concealment for video communication: a review", *Proc. IEEE*, VOL. 86, No. 5, pp. 974-997. 1998.

[2] L. Su, Y. Zhang, W. Gao, Q. Huang, Y. Lu, "Improved error concealment algorithms based on H.264/AVC non-normative decoder", *ICME-IEEE Int. Conf. Multimedia Expo*, 2004

[3] D. Kim, S. Yang, J. Jeong, "A new temporal error concealment method for H.264 using adaptive block sizes", *ICIP-IEEE, Int. Conf. Image Processing*, 2005.

[4] S. Belfore M. Grangetto, E. Magli, G. Olmo, "An error concealment algorithm for streaming video", *ICIP-IEEE Int. conf. Image Processing*, 2003.

[5] P. Baccichet, D. Bagni, A. Chimienti, L. Pezzoni and F. S. Rovati, "Frame concealment for H.264/AVC decoders", *IEEE Trans. Consumer Electronics*, pp. 227-233, 2005.

[6] Y. Lee, R. Mersereau, "Multiframe error concealment for MPEG-Coded video delivery over error-prone networks", *IEEE Trans, Image Processing*, VOL. 11, NO. 11, 2002

[7] JM 10.2 , <http://iphome.hhi.de/suehring/tml/download/>

[8] "Common Test Conditions for RTP/IP over 3GPP/3GPP2—Software and Amendments", in VCEG(SG16/Q6), XIV Meeting, Santa Barbara, CA, Sep. 2001. ITU-T VCEG-M77.

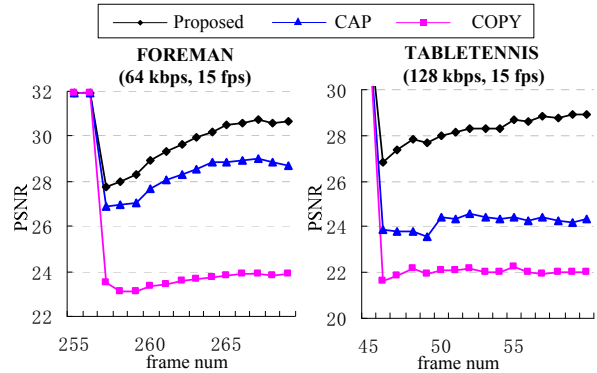


Fig. 4. PSNR of the lost frame and its succeeding frames.

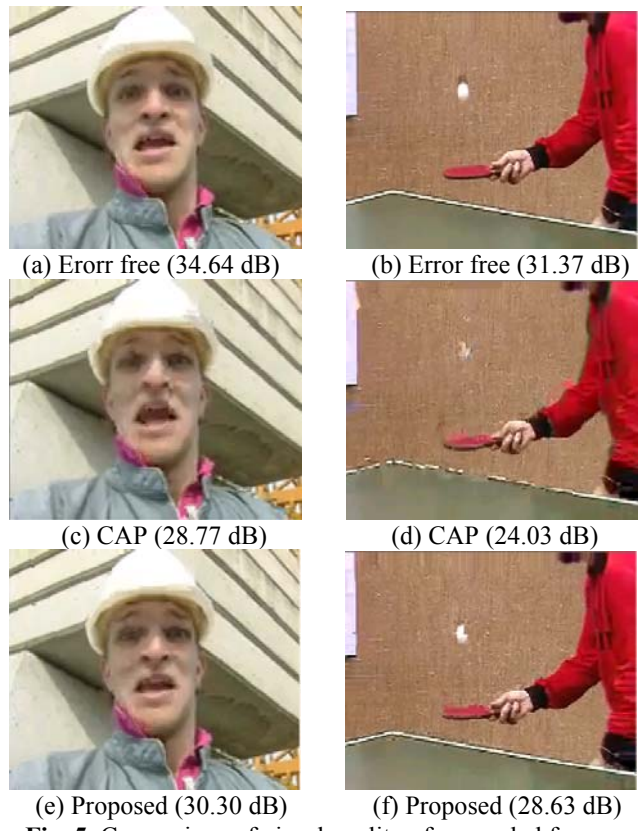


Fig. 5. Comparison of visual quality of concealed frames.