

# Image Concealment based on FEC

Aniruddha D.Shelotkar  
TIT, Bhopal

Vishwajit K.Barbudhe  
TIT, Bhopal

## ABSTRACT

Any error-concealment technique can be applied at the decoder, it is necessary first to find out whether and where a transmission error has occurred. Error detection can be performed at the transport coder/decoder and at the image decoder. One way to perform error detection at the transport coder is by adding header information containing a sequence number. At the transport decoder, the sequence number can be used for packet-loss detection. Another method at the transport level is to use a forward error correction (FEC). There are also methods that can exploit the characteristics of natural image signals where the differences of pixel values between two neighboring lines are used for detecting transmission errors. When the difference is greater than a threshold, the current image segment is declared to be damaged. When VLC is used in the source code, any damage to a single bit can cause resynchronization, resulting in the subsequent bits being undecodable. However, this property can be used as a means to detect transmission errors. In addition, the syntax embedded in the bit stream can also be used for error detection.

**Index Terms**— Forward Error Concealment, interpolation, joint source channel coding, robust image transmission, image system model.

## 1. INTRODUCTION

In this paper, we proposed whole frame loss error concealment algorithm is described in detail. As mentioned before, the motion copy method is a simple and effective whole frame loss error concealment method. However, the motion copy method is based on the assumption that the motion vector difference between consecutive frames is slight [2]. Based on this assumption, the motion copy method directly uses the motion vectors from the previous frame to conceal the lost frame. Nevertheless, there is no refinement operation performed for the motion vectors obtained by the motion copy algorithm. Based on this observation, the main

idea of our proposed whole frame loss [8] error concealment algorithm is to further refine the

Motion vectors derived by the motion copy method and consequently improve the quality of concealed frames. It is well-known that there exists a strong temporal relationship between continuous frames. In other words, we can use the motion vector information of preceding frames to provide some clues about motion behavior for the current lost frame. As a result, the motion vectors of several previous frames are taken into consideration in our proposed algorithm [3]. Fig.1 shows the flowchart of our proposed algorithm.

First, the incoming frame is checked to identify whether the incoming frame is an erroneous frame or not. If the incoming frame is erroneous, the proposed whole frame error concealment algorithm is applied. For the erroneous frame, it will be firstly divided into  $4 \times 4$  block size. The motion vector normalization process is used to normalize the motion vectors. Once the motion vectors have been normalized, the motion vector differences between any successive frame pairs will be calculated and the calculated motion vector differences are used to obtain the motion vector difference refinement area. Afterwards, the motion vector differences within the refinement area are used to form the next motion vector difference refinement area.

This motion vector difference refinement area selection process will be recursively performed until the motion vector difference refinement area becomes unchanged between consecutive iterations. The motion vector differences within the finally selected motion vector difference refinement area are counted to compute the number of occurrences. The motion vector differences in  $x$  and  $y$  directions with maximum frequency of occurrences will be selected as the final refined motion vectors, respectively. Finally, the best motion vector for error concealment is derived by adding the refined motion vectors to the motion vectors obtained by the motion copy method. The detail process of our proposed algorithm will be explained as follow

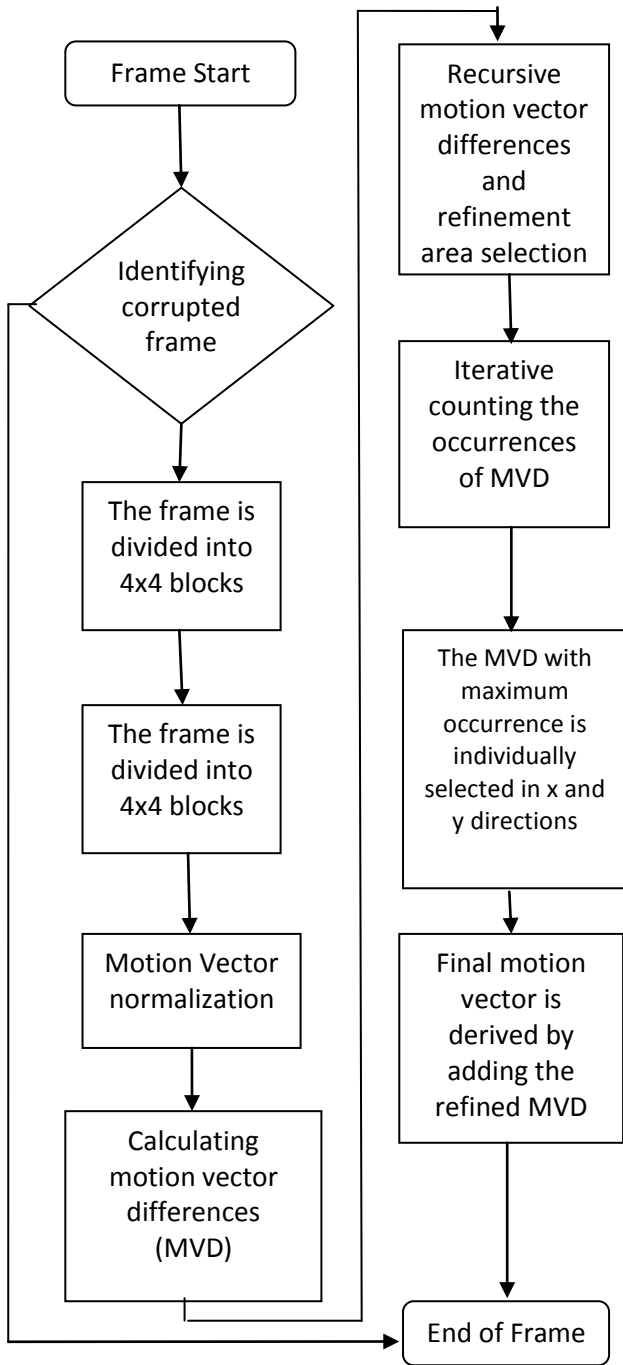


Fig.1 Flowchart of the proposed algorithm

## 2. ALGORITHM

The chosen method to be implemented is an image error concealment technique using FEC. It consists of one-dimensional boundary matching technique, directional weighting interpolation, and median filter. Most of the significant edges can be efficiently recovered with the edge-oriented interpolation. First we employ 1-D block boundary matching between the neighboring blocks to find the edge direction for the lost block. In the process, the location of missing blocks can be found from forward error correction or using the image detection methods. Then the recovered pixel

is interpolated along the edge direction based on the estimated result.

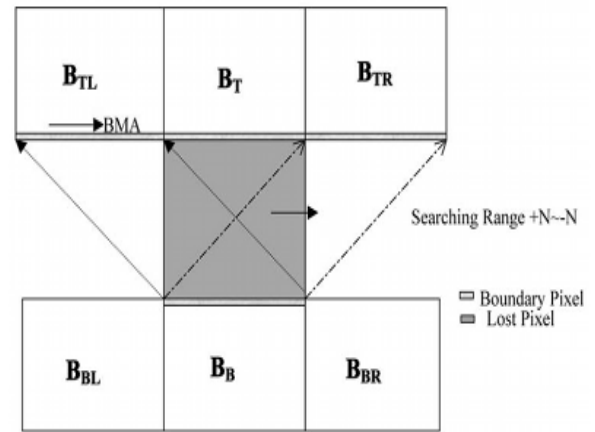


Fig 1 Bottom to Top Approach

Two boundary matching are performed: one from bottom to top of the block, and the other from top to the bottom of the missing block. If the block size used is  $N \times N$ , the 1-D boundary matching is performed between two rows of available pixels: one of dimension  $N$  that belong to the bottom (top) neighboring block  $B_B$  ( $B_T$ ), and the other of dimension  $3 \times N$  belonging to the top, top-left and top-right (bottom, bottom-left, bottom-right) neighboring blocks  $B_T$ ,  $B_{TL}$ ,  $B_{TR}$  ( $B_B$ ,  $B_{BL}$ ,  $B_{BR}$ ). So  $2 \times N$  mean absolute difference (MAD) are

Computed between the bottom (top) row of available  $N$  pixels, and the top (bottom) row of  $3N$  pixels as expressed by:

$$MAD(Mx) = \sum_{i=0}^{n-1} |f_{0,i}^{BB} - f_{N-1,i+Mx}^{BTL,BT,BTR}|$$

Where  $Mx$  is a search vector that is from  $-N$  to  $N$ . Then we can obtain the best match (BMA) corresponding to the minimum MAD value as  $BMA = \min_{Mx} MAD(Mx)$  with  $Mx$  from  $-N$  to  $N$ .

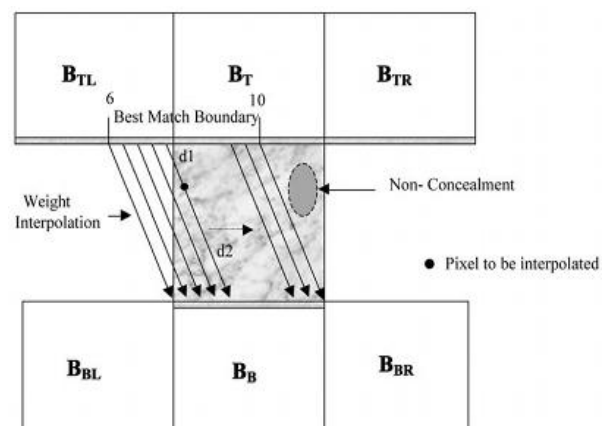
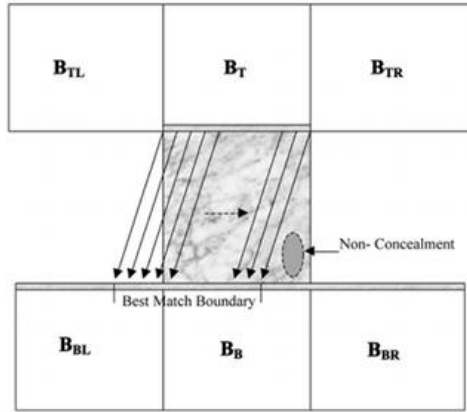


Fig 2 Interpolation



**Fig 3 Top to Bottom Approach**

So in case of a "bottom-to-top" BMA after comparing 2N MADs, we can find the best vector that matches the block  $B_b$  and the blocks  $B_{TL}, B_T, B_{TR}$  in boundary. The vector can give a direction to the edge for the lost block. If the estimated result BMA value is less than one threshold, this implies that there exists a significant edge. In this case, the lost pixels are interpolated along the direction of the best vector. The pixels are interpolated weighting them with the distance between the two boundaries. So if the pixel has to be interpolated with the two selected pixels  $p_1$  and  $p_2$ , the interpolation can be computed using:

$$f_{m1,n1}^1 = \frac{f_{N-1,i}^{B_{TL},B_T,B_{TR}} \cdot d_2 + f_{0,k}^{B_B} \cdot d_1}{d_1 + d_2}$$

Where  $d_1$  and  $d_2$  are the distances between the interpolated pixel to the best matching boundary and the bottom block one, respectively. If the location of the interpolated pixel is closed to the bottom block, the weighting of the boundary pixel of block  $B_b$  is increased since  $d_1$  becomes larger. It can be seen that is a simple linear interpolation. Next, using the same procedure above, the "top-to-bottom" BMA is calculated and then used to interpolate the pixels between the top block  $B_T$  and the three blocks  $B_{BL}, B_B, B_{BR}$ . Finally, the lost pixel is recovered from the merging of the results of the two interpolations. If the interpolated pixel is overlapped the results are averaged. Not all the pixels can be recovered after the two-direction interpolations. The nonlinear median filter is used to interpolate the residual not recovered pixels to avoid blurring the images.

### 3. CONTENT ESTIMATION AND BLOCK CLASSIFICATION

As introduced in the first section, one EC method may perform better than the others depending on the content of the video sequence. Therefore, it is necessary to select a suitable EC method to conceal the EB based on the local edge characteristics. However, the real values of pixels in the EB are not available when conceals the EB. In order to conceal each EB with a suitable EC method, the content in the EB should be estimated according to the characteristics in the survived neighboring blocks. In this work, the content of each EB is estimated firstly and then it is classified into one of the three categories. Where the three categories are defined as Follows:

**Uniform Block:** the gray level of EB may be constant or nearly so. I.e., there is no obvious edge in the block.

**Edge block:** the block locates on the boundary of two or more parts with different gray level. Because the size of block is not large, there are few edges passing through the block and the direction of each edge, in general, is with no or little change.

**Texture block:** both gray level and edge direction varies significantly in the block, so the edge magnitudes of many directions are very strong. .

## 4. PSNR EVALUATION AND RESULT:

### Evaluation Criteria

Performance criteria include reconstructed image quality, computation complex, etc. Objective image quality criteria, such as SNR and PSNR, are widely used because they are easy to calculate. Subjective criteria are much more applicable, since most images will be judged by human vision system at last. However, since it is very difficult to do large-scale subjective image quality evaluation, we will use objective evaluation primarily. PSNR is

$$PSNR = 10 \times \log_{10} \left[ \frac{255 \times 255}{\sum_{x=1}^W \sum_{y=1}^H [p(x,y) - p'(x,y)]^2} \right]$$

Where  $p(x, y)$  is the original gray value of pixel  $(x, y)$  and  $p'(x, y)$  is the gray value of the reconstructed pixel. Computation complexity is an important criterion for some applications. For example, the computation power of a desktop computer is much higher than a cell phone or PDA. Hence, for applications on desktop computer, some moderate complex algorithms are applicable. However, for cell phone, less computation complexity is much more desirable than high reconstructed image quality.

### Widely accepted objective criteria of image quality:

For the evaluation of the algorithm performance we choose the PSNR metric. We compared the interpolation here proposed with a simple linear interpolation, on various uncompressed videos and Images. We applied the interpolations over a damaged version of the video, where several slices are missed for the entire duration, i. e. in all the frames the same slides of blocks have been damaged. Obviously the result highly depends by the video contents: motion, textures and so on. We can see that the proposed algorithm can gain or loose few dB decimals with respect to a linear interpolation. On the following graphs we just compare the PSNR calculated and Total time required.



**Fig a. Missing block & reconstructed image.**

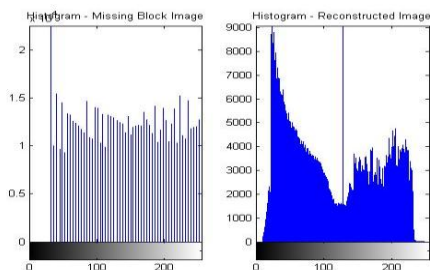


Fig a. Histogram of Missing block image & Reconstructed image.

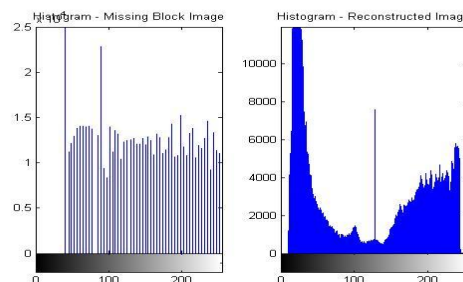


Fig c. Histogram of Missing block image & Reconstructed image.

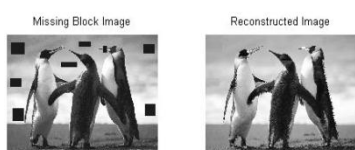


Fig b. Missing block image & reconstructed image.

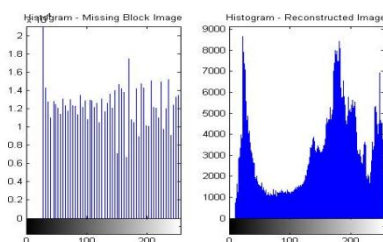


Fig b. Histogram of Missing block image & Reconstructed image.



Fig c. Missing block image & reconstructed image.

SR NO.	IMAGE	PSNR	TIME RQ
1	Fig a	11.90db	146.67s
2	Fig b	7.63 db	139.39s
3	Fig c	11.22db	340.86s

**Comparison with different Algorithm:**

Image	PSNR	DCT	Harris on	Inpainting	FEC
checkerboard	R	48.46	9.39	34.29	8.12
jdice		12.78	14.74	10.70	3.56
D3		11.37	11.26	11.20	12.1
grains		7.71	8.56	8.85	11.7
bark		12.13	12.77	12.27	15.9
bullseye		36.01	11.19	12.93	5.81
e					

**5. CONCLUSIONS**

In this thesis different error concealment methods in the image error concealment and the temporal domains have been implemented as functions written in the Matlab language. These functions have been added to the decoder Matlab source code. Each method is more or less efficient than the other according to the structure of the image.

The first implementation started with the image error concealment. The first step done in this field is based on the interpolation of the pixel values within the damaged macro block from the pixels within the surrounding area, the distance between the concealed pixels where the background is constant. Here one advantage of the image error concealment is the low complexity compared with enhanced error concealment in the temporal domain. The error concealment in the temporal domain is based on the copying algorithm. In this error concealment technique the frames stored in the

decoded frame buffer are used to conceal the missing part within the affected frame. This algorithm just replaces damaged macro block by the corresponding macro block in this frame. For this purpose copy-paste function has been integrated to get the index of the frame within the decoded picture buffer, which has the maximal correlation with the affected frame. Minimizing MSE is very well understood from the mathematical point of view and it can be concluded that suitable metric for analyzing visual quality is PSNR. With regard to implementation complexity, a method is proposed by analyzing the number of operations for every error concealment method. The number of operations gives useful information for deciding which error concealment method is appropriate in terms of complexity. Most of the discussed methods can also be implemented in another way, depending on the required ratio between memory and computational power. Finally how exactly error concealment is implemented is described. Some of the commands required for running encoder and decoder configuration files, are explained. For performing the comparison of the original and concealed Image sequence PSNR Metric are used. Although PSNR has been criticized precisely for not correlating well with perceived quality measurement.

## 6. REFERENCES

- [1] J. W. Park and S. U. Lee, "Recovery of corrupted image data based on the NURBS interpolation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 1003–1008, Oct. 1999.
- [2] X. Lee, Y. Zhang, and A. Leon-Garcia, "Information loss recovery for block-based image coding techniques—A fuzzy logic approach," *IEEE Trans. Image Processing*, vol. 4, pp. 259–273, Mar. 1995.
- [3] J.-W. Suh and Y.-S. Ho, "Error concealment based on directional interpolation," *IEEE Trans. Consumer Electron.*, vol. 43, pp. 295–302, Aug. 1997.
- [4] Z. Wang, Y. Yu, and D. Zhang, "Best neighborhood matching: An information loss restoration technique for block-based image coding systems," *IEEE Trans. Image Processing*, vol. 7, pp. 1056–1061, Jul. 1998.
- [5] W.-M. Lam and A. Reibman, "An error concealment algorithm for images Subject to channel errors," *IEEE Trans. Image Processing*, vol. 4, pp. 533–542, May 1995.
- [6] W. Zhu, Y. Wang, and Q. Zhu, "Second-order derivative-based smoothness measure for error concealment in DCT-based codec's," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 713–718, Oct. 1998.
- [7] K. Jung, J. Chang, and C. Lee, "Error concealment technique using projection data for block-based image coding," in *Proc. SPIE Conf. Visual Communication and Image Processing*, vol. 2308, 1994, pp. 1466–1476.
- [8] H. Sun and W. Kwok, "Concealment of damaged block transform coded images using projections onto convex sets," *IEEE Trans. Image Process.*, vol. 4, no. 4, pp. 470–477, Apr. 1995.
- [9] G. S. Yu, M. M. K. Liu, and M. W. Marcellin, "POCS-based error concealment for packet video using multiframe overlap information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 4, pp. 422–434, Aug. 1998.
- [10] A. Bilgin, Z. Wu, and M. W. Marcellin, "Decompression of corrupt Jpeg2000 codestreams," in *Proc. IEEE Data Compression Conf.*, Snowbird, UT, Mar. 2003, pp. 123–132.
- [11] I. Moccagatta, S. Soudagar, J. Liang, and H. Chen, "Error-resilient coding in JPEG-2000 and MPEG-4," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 899–914, Jun. 2000.
- [12] A. Natu and D. Taubman, "Unequal error protection of JPEG2000 codestream in wireless channels," in *Proc. IEEE Global Telecommunications Conf.*, vol. 1, Taipei, Taiwan, R.O.C., Nov., 2002, pp. 534 –5348.
- [13] D. Nicholson, C. Lamy, C. Poulliat, and X. Naturel, "Result of core experiments on header error protection (JPWL C01)," ISO/IEC JTC 1/SC 29/WG 1 N 2935, May 2003.
- [14] JPEG 2000 Part 1—Core Coding System.