Block Loss Recovery Using Fractal Extrapolation for Fractal Coded Images

Yun Ho Noh
Display Product Research Lab., LG Electronics Inc.
quixote@lge.co.kr, Kumi, Korea

Sang Hyun Kim
DSP Department, Medison R&D Center
ksh@medison.co.kr, Seoul, Korea

Nam Chul Kim
Dept. of Electronic Engineering, Kyungpook National University
nckim@ee.kyungpook.ac.kr, Taegu, Korea

Abstract

The effect of block loss due to the cell loss in the ATM transmission is more serious in fractal coded images than in DCT coded images. It is the reason that in fractal coded images the effect of block loss is not confined to each lost block itself but is propagated to the range blocks other than the lost blocks. A new algorithm is presented for recovering the blocks lost in the transmission of the images coded by Jacquin's fractal coding. The key technique of the proposed BLRA (block loss recovery algorithm) is a fractal extrapolation that estimates the lost pixels by using the contractive mapping parameters of the neighboring range blocks which satisfy the connectivity to a lost block. The proposed BLRA is applied to the lost blocks in the iteration of decoding. Some experimental results show the proposed BLRA yields excellent performance in PSNR as well as subjective quality.

1. Introduction

It has already been accepted that asynchronous transfer mode (ATM) is the target communication protocol for future broadband integrated-services digital networks (BISDN) which support a variety of applications including the transmission of high-resolution images and videos. In spite of many advantages offered by ATM protocol, cell loss problem remains as the main obstacle to image and video transmission [1]. Cell loss recovery therefore is one of the most important aspects in the design of a future image and video codec.

In the past few years, various approaches [2]-[4] have been proposed to combat the block loss problem caused by lost cells in block coded images. Most of them utilize linear interpolation to estimate the DCT coefficients or the pixel intensities in a lost block from the information of its neighboring blocks. Furthermore, the recovery algorithms are oriented mainly to block coded images such as DCT coded image, in which each block is coded independently, but not to fractal coded images.

Fractal coding has been extensively used in many applications including image and video coding because of the attraction that the fractal technique utilizes the self-similarity in an image itself. In Jacquin's method [5], an original image first is partitioned into non-overlapping range blocks called as parent block. Under a certain condition, a parent block may be split into up to four non-overlapping child blocks. Each range block is classified based on its geometric feature into three classes - shade, midrange, and edge block. For each range block, the full search is then made for the best contractive mapping such that minimizes the distance between the range block and one of contractive mapping of larger domain blocks in the image itself. The best contractive mapping parameters are coded in different way according to each class. After encoding, the coded parameters with the information on block type and class are transmitted. At the receiver, the image is reconstructed from any initial image by the iteration of the contractive mapping.

In DCT coded images, the effect of block loss is confined only to each lost block because of no inter-block processing. However, the effect of block loss in fractal coded images is not confined to each lost block itself but is propagated to the range blocks mapped from the lost blocks. So it causes degradation of image quality more serious than in DCT coded images.

In this paper, a new algorithm using fractal extrapolation is presented for recovering the blocks lost in the ATM transmission of the images coded by Jacquin's fractal coding technique. In the algorithm, it is assumed that the information on block type and class is transmitted with higher priority and is secured against loss because such an information is indispensable for reconstruction. The key technique of the proposed BLRA (block loss recovery algorithm) is a fractal extrapolation that estimates the lost pixels by using the contractive mapping parameters of the neighboring range blocks which satisfy the connectivity to a lost block. If there is no such a neighboring block, Salama's linear interpolation [2] is applied to it. For preventing error propagation, the proposed BLRA is performed on the lost blocks during the iteration of decoding. Thus, the proposed algorithm is not a post-processing but iterative processing in the decoding.

2. Fractal Coding and ATM Transmission

In Jacquin's fractal coding [5], two-level partition and block classification are introduced to obtain greater coding efficiency. An original image first is partitioned into non-overlapping range blocks called as parent block. If the fractal approximation of a parent block is not satisfactory, the parent block is split into up to four non-overlapping child blocks. Based on the block classification used by Ramamurthi and Gersho [6], each parent or child range block is classified into three classes - shade, midrange, and edge block. The contractive mappings available for each class are differently chosen. The detailed coding procedure according to each class is shown as follows.

For a shade block in which the intensity variation is very small, the average gray level of the range block R_i only is coded as the contractive mapping parameter g_i, that is,

$$g_i = quant[\overline{R}_i] \tag{1}$$

where $\overline{R}_i$ denotes the average gray value of the range block and $quant[\cdot]$ means uniform quantization. For a midrange block in which the intensity variation is comparatively large, the range block R_i is matched with the available contractive mappings of larger domain block in the image itself, as shown in Fig. 1. The domain blocks D_j are first contracted as small as the range block by the spatial contraction of

$$S(D_j) = \frac{1}{4}\{D_j(k,l) + D_j(k+1,l) + D_j(k,l+1) + D_j(k+1,l+1)\}, \tag{2}$$

where $k, l \in \{0,1,\ldots,2B-1\}$. All of the available contractive mappings next are performed by the compositions of a contrast scaling and a luminance shift:

$$\tau_i(D_j) = \alpha_i[S(D_j)] + \Delta g_i, \tag{3}$$

where the contrast scaling α_i is selected as one of $\{0.55, 0.7, 0.85, 1.0\}$, and the luminance shift Δg_i is obtained by

$$\Delta g_i = quant(\overline{R}_i - \alpha_i \overline{D}_j), \tag{4}$$

where D_j is the average gray level of the domain block. The full search is then made for the best contractive mapping that minimizes the distortion between the range block and $\tau_i(D_j)$.

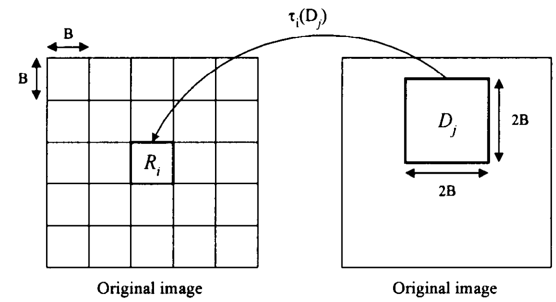


Fig. 1. Procedure of contractive mapping.

For an edge block in which the intensity variation is very large, all of the available contractive mappings are performed by the compositions of a contrast scaling, a luminance shift, and an isometry:

$$\tau_i(D_j) = I_i[\alpha_i(S(D_j)) + \Delta g_i], \tag{5}$$

where the isometry transform I_i means one of four rotations and four reflections, and the contrast scaling α_i is selected as one of $\{0.45, 0.6, 0.8, 1.0\}$.

After coding each range block differently according to its class, the contractive mapping parameters are transmitted to the receiver. At the receiver, an original image is reconstructed from an arbitrary initial image by the iteration of the contractive mapping.

In order to secure the transmission of image and video through packet network, typically ATM network, the two-layer transmission that gives different levels of priority has been proposed [1]. In two-layer transmission, the information that does not cause disastrous effect is transmitted with lower priority although the information is lost by the defect of network or queuing delay in switch. In addition, the information of higher importance that causes fatal degradation on the reconstructed image quality is transmitted with higher priority.

Although the information of a block is lost, we can recover the lost block if we know whether it is coded as a parent or child and what its class is. Since these two kinds of information are indispensable for reconstruction, we divide Jacquin's fractal code into two layers of header code and main code suitable for lossy network. The header code contains the information on block type and class. The main code contains the contractive mapping parameters. The header code of higher importance is entitled to higher priority, and the main code of lower importance is entitled to lower priority and then both are transmitted.

3. Block Loss Recovery of Fractal Coded Images

In Jacquin's fractal coding, each range block may be coded as only a parent block or only four child blocks, and may be coded as a parent block and one to three child blocks. An arbitrary parent or child block can be lost in the transmission of fractal coded bit stream. Especially in case that a parent block is lost, the loss pattern of the lost parent block depends on the coding pattern of child blocks in the lost parent block.

The proposed BLRA makes use of the correlation between a parent block and its child block or between the lost parent or child block and its neighboring blocks. It also considers the characteristics of block class and edge direction, and performs the different procedure according to each class. In case that a parent block is lost, the BLRA checks whether or not the lost parent block has its child blocks that satisfy the connectivity to the lost parts. If so, it recovers the lost parts by fractal extrapolating them from the child blocks. Otherwise, if there are the neighboring parent blocks that satisfy the connectivity, it recovers them by fractal extrapolating them from the neighbors. If there is no such a neighbor, we recover the lost parts by linear interpolating them from the four neighbors. Salama's method [2] is chosen as a linear interpolation technique.

In case that a child block is lost, it checks whether or not there is a parent block of the lost child block. If so, it recovers the lost child block by using the mapping parameters of its parent block. Otherwise, it recovers the lost child block by fractal extrapolating the lost block from a child block that satisfies the connectivity among its eight neighboring blocks. If there is no such a child block in the neighboring child blocks, it recovers the lost block by linear interpolating the block from its four neighboring blocks. The detailed block loss recovery according to each class is the same as in case of a lost parent block having no child block.

Error propagation due to mapping of some range blocks from the lost blocks also is prevented by performing the proposed recovery algorithm in the iteration. Thus, the proposed method is not post-processing but iterative processing in the decoding process. The fractal extrapolation used in the proposed BLRA is expressed by

$$\hat{\tau}_i(D_j) = \alpha_{i'} \, I_{i'} \, (S(D_{(j')'})) + \Delta g_{i'} , \qquad (6)$$

where $\alpha_{i'}$, $I_{i'}$, and $\Delta g_{i'}$ denote the contrast scaling, isometry, luminance shift of a neighboring range block which satisfies the connectivity to the lost parts, and $D_{(j')'}$ denotes the corresponding neighbor of the domain block $D_{j'}$.

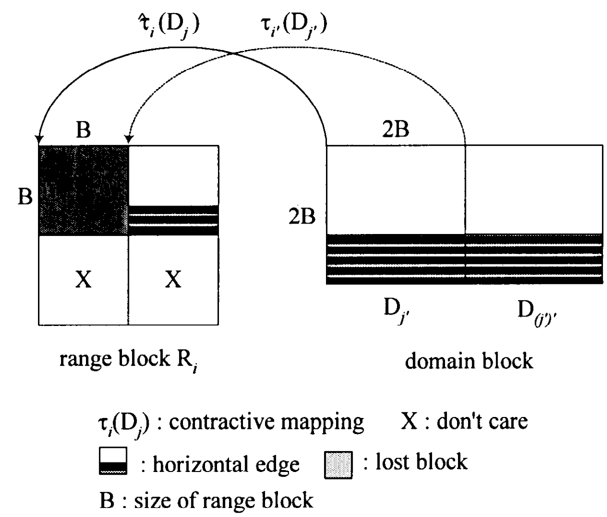


Fig. 2. Example of fractal extrapolation.

As shown in Fig. 2, fractal extrapolation $\hat{\tau}_i(D_j)$ is the same as the contractive mapping for the neighbor range block except that the position of the domain block of its range block is shifted. That is, if the lost block cuts off horizontal edge, we recover the lost block by using the fractal mapping parameters of the horizontal neighbor range block that satisfy the connectivity to a lost block. If the lost block cuts off vertical edge, we recover it by using those of the vertical neighbor range block, and if the lost block cuts off diagonal edge, we recover it by using those of the diagonal neighbor range blocks. The equation (6) above is utilized when a lost block is edge block. In case of a lost block being midrange block, the isometry I_i is excluded. Next, we describe the different BLRA procedure according to each class.

3. 1. Block Loss Recovery for Shade Parent Block

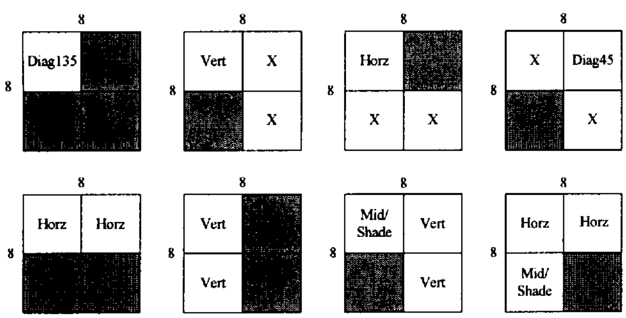
In case of a shade block being lost, we recover the lost shade block as follows. If a lost parent block has child blocks of shade, we recover the lost parts by fractal extrapolating them from the shade blocks. Otherwise, we recover the lost parts by fractal extrapolating them from the shade blocks among their eight neighboring blocks. In both of the cases, we average them when the shade blocks are more than two. If there is no shade block in the eight neighboring blocks, we recover the lost parts by linear interpolating them from their four neighboring blocks. This case however rarely happens.

3. 2. Block Loss Recovery for Midrange Parent Block

If a lost parent block has a child block of midrange, we recover the lost parts by fractal extrapolating them from the child block. Otherwise, we recover the lost parts by fractal extrapolating them from a midrange block, which satisfies the connectivity to the lost parts, among the eight neighboring blocks. We here decide that the neighboring midrange blocks satisfy the connectivity when there are two or more neighboring midrange blocks in which any two are arranged horizontally, vertically, and diagonally. In case that there is no midrange block in the eight neighboring blocks, we recover the lost parts by linear interpolating them from the four neighboring blocks. This case however rarely happens.

3. 3. Block Loss Recovery for Edge Parent Block

If a lost parent block has a child block of edge that satisfies the connectivity to the lost parts, we recover the lost parts by fractal extrapolating them from the child block. Otherwise, we recover the lost parts by fractal extrapolating them from an edge block, which satisfies the connectivity to the lost parts, among its eight neighboring blocks. In case that there is no such an edge block in the neighboring blocks, we recover the lost parts by linear interpolating them from the neighboring blocks.



Diag135 : reverse diagonal edge, Diag45 : diagonal edge,
Horz : horizontal edge, Vert : vertical edge,
X : don't care, ■ : lost block.

Fig. 3. Loss pattern of a lost parent block having edge child blocks.

In case that an edge child block in which the edges are expected to be cut off by the lost parts, as shown in the upper four figures of Fig. 3, we recover the lost parts by fractal extrapolating them from the edge block. In case that edge connectivity is satisfied even though a part of parent block is lost, as shown in the lower four figures of Fig. 3, we recover the lost parts by fractal extrapolating them from a neighboring parent block of shade or midrange rather than edge. It is the reason that if we recover the lost parts by fractal extrapolating them from a child or parent block of edge, the edge will be blurred.

In case of a lost block without child block, we recover the lost parent block by fractal extrapolating it from an edge block that satisfies the connectivity to the lost parent block among the eight neighboring blocks. We here decide that the neighboring edge blocks satisfy the connectivity when there are two or more neighboring edge blocks in which any two are arranged horizontally, vertically, and diagonally, as shown in Fig. 4. If there is no such an edge block in the eight neighboring blocks, we recover it by linear interpolating it from its four neighboring blocks. This case however rarely happens.

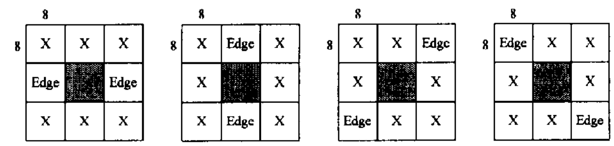
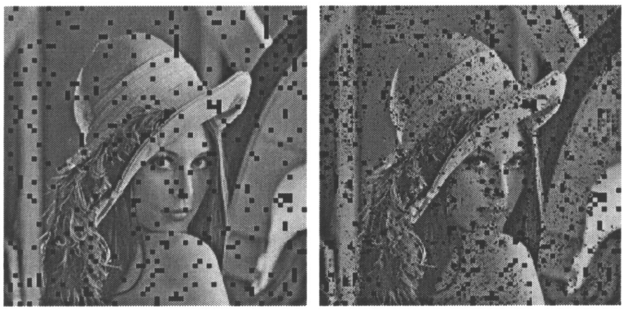


Fig. 4. Neighbor pattern of a lost parent block having no child block.

4. Experimental Results and Discussion

We have performed various experiments to evaluate the performance of the proposed BLRA. Test image used in simulation is Lena image of 512×512 pixels and 256 gray levels. Fig. 5 shows a DCT-coded image and a fractal-coded image in which the block loss ratio (BLR) due to the ATM cell loss is given as $BLR = 10^{-1}$. We see in this figure that the effect of the block loss is confined only to each lost block in DCT coded images but is propagated to the range blocks other than the lost blocks.

Fig. 6(a) shows the result of recovering the fractal coded image of $BLR = 10^{-1}$ by Salama's linear interpolation as a post-processing after decoding it. The BLRA using the linear interpolation does not recover the lost blocks well because it does not consider the error propagation due to mapping from the lost blocks during the iteration. Shown in Fig. 6(b) is the result of recovering the fractal coded image by the proposed BLRA, in which the degradation from block loss is nearly removed. Table 1 shows the PSNR results of the linear interpolation and the proposed BLRA during the iteration. We see that the proposed BLRA is superior in PSNR to the linear interpolation. Especially the proposed one has better performance in edges than the linear interpolation. Fig. 7 shows the enlargements of the original Lena and the images recovered by the linear interpolation and the proposed method. We see that the proposed one is also superior in the subjective quality to the linear interpolation. Fig. 8 shows the PSNR curve of the linear interpolation and the proposed BLRA according to BLR.

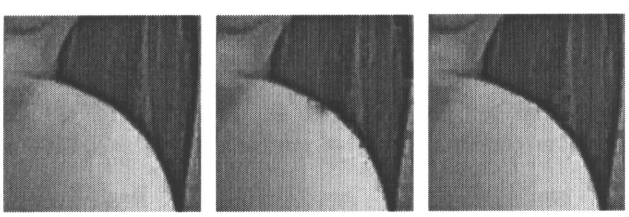


(a) (b)

Fig. 5. Lossy images of BLR(block loss ratio) = 10^{-1}: (a) DCT coded image; (b) Fractal coded image.

(a) (b)

Fig. 6. Results of recovering Fig. 1(b): (a) Linear Interpolation as a post-processing; (b) the proposed BLRA.



(a) (b) (c)

Fig. 7. Enlargements of recovered images: (a) Original Lena; (b) linear interpolation; (c) the proposed BLRA.

Table 1. Results of block loss recovery obtained by the linear interpolation and the proposed BLRA during the iteration.

Class	Lossy Image	Linear Interpolation	Proposed Method
Edge	16.43 dB	32.17 dB	32.43 dB
Midrange	19.44 dB	33.57 dB	33.62 dB
Shade	23.16 dB	33.44 dB	33.60 dB
Total	14.24 dB	31.91 dB	32.37 dB

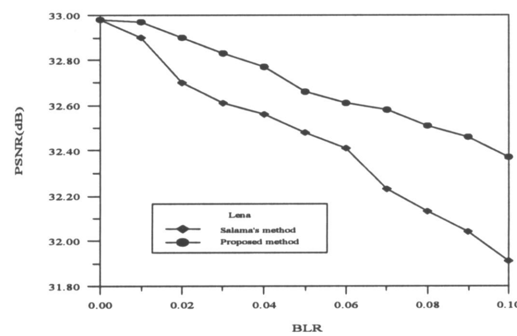


Fig. 8. PSNR performance according to BLR.

5. References

[1] W. Verbiest, L. Pinnoo, and B. Voeten, " The impact of the ATM concepts on video coding," IEEE J. Select. Areas Commun., vol. 6, pp. 1623 – 1632, Dec. 1988.

[2] P. Salama, N. B. Shroff, E. J. Coyle, and E. J. Delp, "Error concealment technique for encoded video streams," Proc. ICIP'95, pp. 9-12, Washington D.C., 1995.

[3] Y. Wang and Q. Zhu, "Maximally smooth image recovery in transform coding," IEEE Trans. Commun., vol. 41, no. 10, pp. 1544-1551, Oct. 1993.

[4] X. Lee, Y. Zhang, and A. Leon-Garcia, "Information loss recovery for block-based image coding techniques - A fussy logic approach," IEEE Trans. Image Processing, vol. 4, no. 3, pp. 256-273, Mar. 1995.

[5] A. E. Jacquin, "Fractal image coding: A review," Proc. IEEE, vol. 81, no. 10, Oct. 1993.

[6] B. Ramamurthi and A. Gersho, "Classified vector quantization of images," IEEE Trans. Commun., vol. 34, Nov. 1986.