A Novel Method Based on Sampling Theory to Recover Block Losses for JPEG Compressed Images

Siamak Talebi and Farokh Marvasti

Department of Electronic Engineering
King's College London
University of London
Strand, London WC2R 2LS,UK
E-mail: farokh.marvasti@kcl.ac.uk

ABSTRACT

A new method to recover *(8x8)* block losses for JPEG compressed images is proposed, which can be used in wireless UMTS and Asynchronous Transfer Mode (ATM) networks. The problem of image reconstruction with *(8x8)* block losses is transformed to a system of linear equations using two-dimensional trigonometric polynomials. The simulation results show the feasibility of this method. The sensitivity analysis of the proposed method shows that this method is robust against additive noise.

1. INTRODUCTION

In the JPEG system, Huffman coding is used to code the DCT coefficients in order to decrease the bit rate. The DC coefficient is treated separately from the AC coefficients and is differentially coded [1]. During transmission of the JPEG-coded images in wireless UMTS or ATM networks, some packets are lost. This loss results in the loss of consecutive blocks in the images. Since the DC coefficients of JPEG coded blocks are differentially encoded, the loss of packets affects the corresponding blocks packed into that packet and also affects all the DC coefficients of the blocks packed in the succeeding packets. In other words, the error in the DC coefficients propagates through out all the blocks succeeding the lost blocks. Such error propagation makes error recovery very difficult. Many schemes have been proposed to recover the missing pixels from the remaining pixels of the image. Some of these schemes implement iterative and non-linear techniques in the recovery process [2 - 7]. But none of them is able to recover the missing pixels when many (8×8) blocks are lost. Also most of the previous methods need much computational effort. In this paper a new technique is proposed to recover the (8×8) blocks of losses in an erasure channel by over-sampling the original image before packetization. This technique is based on the linear system of equations using 2-D trigonometric polynomials. The paper is organised as follows: In section II, a new algorithm for burst error recovery of an image is presented. In section III, simulation results and the sensitivity of the simulated technique to additive noise are presented. Finally, this paper is concluded in section IV.

2. THE PROPOSED TECHNIQUE

In this section, we propose a novel method to recover the (8×8) block losses for compressed images. It is based on the observation that the recovery of pixels can be formulated as a set of linear system of equations using 2-D trigonometric polynomials. For the sake of clarity, square images are considered, but the technique can be also applied to rectangular images.

0-7803-6293-4/00/$10.00©2000 IEEE

Let us assume that a 2-D signal such as an image is sampled at the Nyquist rate yielding a discrete signal $x_{org}(i,k)$, $i,k=0,..,N-1$. The image is partitioned into (8×8) blocks. Then we use a new 2-D transform such that the kernel of the transform is equal to $\frac{4}{N \cdot M} c_u c_v \cos\left[\frac{(2n+1)uq_1\pi}{2N}\right] \cdot \cos\left[\frac{(2m+1)vq_2\pi}{2M}\right]$ where $u=0,..,N-1$, $v=0,..,M-1$ and the coefficients q_1 and q_2 are positive prime integers with respect to N and M, respectively. For the sake of clarity, we shall call this transform the *Sorted Discrete Cosine Transform* (SDCT). Now each block is transformed by a 2-D (8×8) SDCT. Then we add an (8×8) block of zeros at the end of each block. The inverse 2-D (16×8) SDCT transform of each (16×8) block will lead to an over-sampled version of the original signal $x_{over}(i,k)$ $i=0,..,2N-1$ and $k=0,..,N-1$ with $2N \times N$ samples. This (16, 8) code is capable of correcting a block of (8×8) from a block of (16×8).

During transmission of this JPEG-coded over-sampled image in a wireless UMTS or ATM network, some packets are lost. This loss results in the loss of consecutive (*8×8*) blocks in the images.

At the receiver, after reconstruction of the JPEG image, the missing part of the over-sampled image in each (*16×8*) block can be one of the upper or lower block of (*8×8*). The block loss is denoted by $e(n_i,m_i) = x_{over}(n_i,m_i)$, where (n_i,m_i) $i=1,..,64$ specifies the positions of the lost pixels of the block. The value of $e(n,m)$ for any position except (n_i,m_i) is zero.

For the $e(n,m)$, the SDCT is given by

$$E(u,v) = \frac{4}{16\times 8} c_u c_v \sum_{n=0}^{15}\sum_{m=0}^{7} x(n,m)\cdot\cos\left[\frac{(2n+1)uq_1\pi}{2\times 16}\right]\cdot\cos\left[\frac{(2m+1)vq_1\pi}{2\times 8}\right] \tag{1}$$

$$u=0,1,...,15, \quad v=0,1,....,7, \qquad c_t = \begin{cases} \frac{1}{\sqrt{2}}, & t=0 \\ 1, & t\neq 0. \end{cases}$$

We change to vector notion and will treat the 16×8 array $e(n,m)$ also as a vector by staking the columns of $e(n,m)$. In this form $E(u,v)$ is described by

$$E(u_l,v_l) = \frac{4}{16\times 8} c_{u_l} c_{v_l} \sum_{i=0}^{127} e(n_i,m_i)\cos\left[\frac{(2n_i+1)u_l q_1\pi}{2\times 16}\right]\cdot\cos\left[\frac{(2m_i+1)v_l q_2\pi}{2\times 8}\right]$$

$$u=0,1,.....,15 \quad n=0,1,.....,15$$
$$v=0,1,.....,7 \quad m=0,1,.....,7 \tag{2}$$

where $l=0,1,..,127$. In the matrix form, we have

$$\boldsymbol{E}=\boldsymbol{C}\cdot\boldsymbol{e}, \tag{3}$$

where $\boldsymbol{E}$ is 128×1, $\boldsymbol{C}$ is 128×128 and $\boldsymbol{e}$ is 128×1. The received *(16×8)* block, $d(n,m)$, can be thought of as the original over-sampled *(16×8)* block, $x_{over}(n,m)$, minus the missing *(16×8)* block $e(n,m)$. The error block, $e(n_i,m_i)$, is the difference between the corrupted and the original over-sampled block. Hence, it is equal to the values of the missing *(16×8)* block for $(n,m)=(n_i,m_i)$ and is equal to zero otherwise. The corresponding relationship in the DCT domain is

$$E(u_l,v_l)=x_{over}(u_l,v_l)-D(u_l,v_l), \quad l=0,1,\ldots,127. \tag{4}$$

From (4), considering that $X_{over}(u_j,v_j)=0$, for $j=64,\ldots,127$, we have

$$E(u_j,v_j)=-D(u_j,v_j). \tag{5}$$

Let $\boldsymbol{Q}$ be the 64×64 principal submatrix of $\boldsymbol{C}$ from (3), obtained by deleting from $\boldsymbol{C}$ all rows and columns except those with indices i and j (such that (n_i,m_i) denote the position of the pixel losses, and (u_j,v_j) denote the known values of $E(u_j,v_j)$). The result is

$$E_{64\times 1}(u_j,v_j)=Q_{64\times 64}\cdot e_{64\times 1}(n_i m_i). \tag{6}$$

Let us assume for the moment that $\boldsymbol{Q}$ is invertible; the question of Invertibility of $\boldsymbol{Q}$ will be discussed in the next section. Therefore, from (5), we have

$$e_{64\times 1}(n_i,m_i)=Q_{64\times 64}^{-1}\cdot E_{64\times 1}(u_j v_j). \tag{7}$$

By sorting the values of $e_{64\times 1}(n_i,m_i)$to the *(8×8)* matrix form, the block losses will be obtained.

2.1 Invertibility of the Q matrix

Initially we choose $q_1=q_2=1$ so that we may discuss the Invertibility of $\boldsymbol{Q}$, and later determine more suitable values for q_1 and q_2. For this choice the SDCT is identical to DCT. As it was mentioned earlier, the goal is to recover (8×8) block losses for the JPEG images or any other system that uses (8×8) blocks for compression during transmission in wireless UMTS or ATM network. Our method is able to recover one of the (8×8) blocks from every two consecutive lines in the same column. Therefore, there are only two $\boldsymbol{Q}_{64\times 64}$ matrices based on the location of the block loss. The eigenvalues of these matrices are shown in Fig. 1. These eigenvalues confirm that these matrixes are invertible. In order to reduce computational complexity, we can compute the inverse of these two Q matrices and store them once in the memory of the system.

The difference between the pixel of the original image and the reconstructed compressed image can be regarded as additive noise. Therefore our proposed method should be robust against additive noise. To achieve this condition and in order to reduce the dynamic range of the eigenvalues of the two Q matrices, we use suitable values for q_1 and q_2 (the idea comes from [2]). Our simulation results show that the proper values for q_1 and q_2 are equal to 7 and 1, respectively. In this situation the eigenvalues are shown in Fig. 2 and we have a stable method.

3. SIMULATION RESULT

A *(256x256)* image is used for the simulation of the algorithm, Fig. 3-(a). The image is partitioned into (8×8) blocks and each block is transformed by 2-D (8×8) SDCT. We add an (8×8) block of zeros at the end of each the SDCT coefficients of the (8×8) block. The image is transformed using SDFT. By taking an inverse 2-D (16×8) SDFT, the over-sampled image is produced. The reconstructed over-sampled image of JPEG does not appear to have similarity with the original image. According to the algorithm, the number of (8×8) block losses is limited to 1024 in this case. Then, 324 blocks are erased randomly from the image as shown in Fig. 3-(b). The JPEG image after reconstruction is shown in Figs. 3-(c). The PSNR for this simulation is equal to*36.96dB*.

For our proposed method, the sensitivity of the algorithm to additive noise is simulated as follows: A white random noise of uniform distribution with an approximate amplitude of *1/20* of the transmitted image *(SNR=25.14dB)* is added to the JPEG over-sampled image. The SNR after the recovery of the image is equal to 24.04dB. The result for this case is shown in Fig. 3-(d). The SNR values show that the new method is robust against additive noise. This can be attributed to the low dynamic range of the eigenvalues of the Q matrices.

4. CONCLUSION

We have shown that the new algorithm has three advantages. Firstly, it is ideal to recover the missing (8×8) consecutive block losses for the JPEG compressed images. Secondly, in terms of complexity, it is simpler than other techniques; thirdly, it is very robust in correcting block losses with respect to additive noise. But the disadvantage of this method is that it cannot recover all the (8×8) block losses in the consecutive line and the same columns.

5. REFERENCES

[1] K. R. Rao and J. J. Hawang, "*Techniques & Standards for Image, Video & Audio Coding.*" Prentice-Hall, Inc., 1996.

[2] S. Talebi and F. Marvasti, "A Novel Method for Burst Error Recovery of Images," *Proceeding of the International Workshop on Sampling Theory and Applications (SAMPTA' 99)*, Leon, Norway, pp. 41-47, August 1999.

[3] T. Strohmer, "Computationally Attractive Reconstruction of Bandlimited Images from Irregular Samples," *IEEE Trans. on Image Processing*, vol. 6, no. 4, April 1997.

[4] F. A. Marvasti, C. Liu, and G. Adams, "Analysis and recovery of multidimensional signals from irregular samples using non-linear and iterative techniques," *signal processing,* vol. 36, pp. 13-30, 1994

[5] K. D. Sauer and J. P. Allebach, "Iterative reconstruction of band-limited images from nonuniformly spaced samoles," *IEEE Trans. Circuits Syst.*, vol. 34, no. 12, pp. 1497-1506, 1987.

[6] H. G. Feichtinger and T. Strohmer, "Fast iterative reconstruction of band-limited images from irregular sampling values," in *Proc. Conf. CAIP, Computer Analysis of Images and Patterns,* D. Chetvertikov and W. Kropatsch, Eds. Budapest, 1993, pp. 82-91.

[7] H. Feichtinger and T. Strohmer, "Recovery of missing segments and lines in images," *Opt. Eng.: Issue Digital Image Recov. Synth*, vol. 33, no. 10, pp. 3283-3289, 1994.

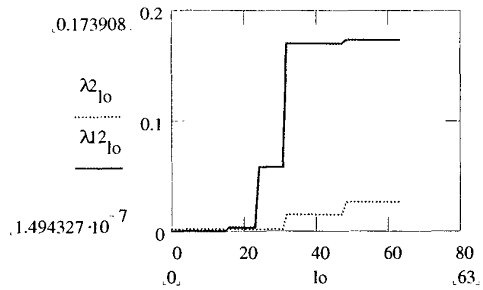

Fig. 1. The eigenvalues of the two Q matrices (for $q_1=q_2=1$).

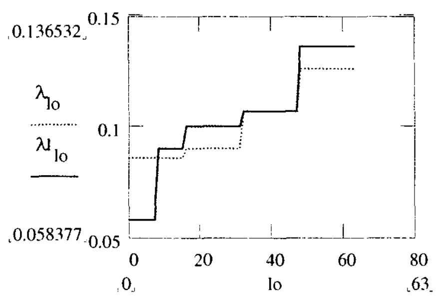

Fig. 2. The eigenvalues of the two Q matrices (for $q_1=7$, $q_2=1$).

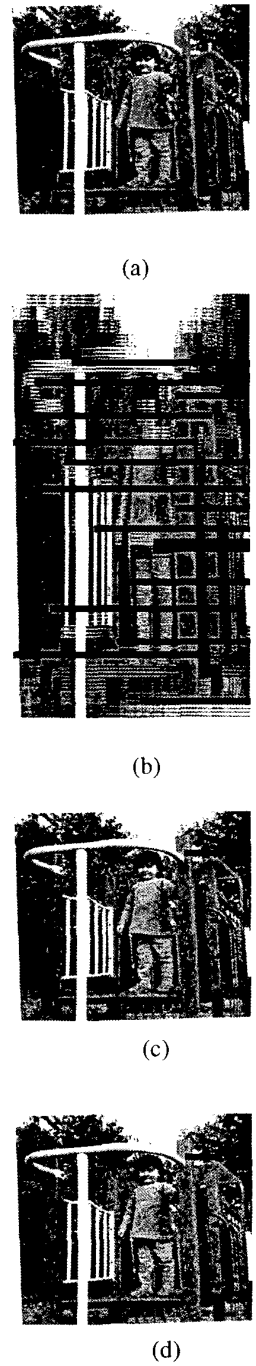

Fig. 3. The result of the simulation: (a) The original image; (b) The over-sampled image with blocks loses; (c) the image after reconstruction; (d) the image after reconstruction with additive noise (the black blocks in the (c) and (d) denote the (16×8) block losses and they cannot be reconstructed).