ELSEVIER

Signal Processing: *Image Communication* 14 (1999) 473–492

SIGNAL PROCESSING:
IMAGE
COMMUNICATION

Error concealment algorithms for compressed video

Min-Cheol Hong[a], Harald Schwab[b], Lisimachos P. Kondi[a], Aggelos K. Katsaggelos[a,*]

[a] *Northwestern University, Department of Electrical and Computer Engineering, Evanston, IL 60208, USA*
[b] *University of Vienna, Department of Mathematics, Vienna, Austria*

Abstract

In this paper we propose an error concealment algorithm for compressed video sequences. For packetization and transmission, a two layer ATM is utilized so that the location of information loss is easily detected. The coded image can be degraded due to channel error, network congestion, and switching system problems. Seriously degraded images may therefore result due to information loss represented by DCT coefficients and motion vectors, and due to the interdependency of information in predictive coding. In order to solve the error concealment problem of intra frames, two spatially adaptive algorithms are introduced; an iterative and a recursive one. We analyze the necessity of an oriented high pass operator we introduce, and the requirement of changing the initial condition in iterative regularized recovery algorithm. Also, the convergence of iteration is analyzed. In recursive interpolation algorithm, the edge direction of the missing areas is estimated from the neighbors, and estimated edge direction is utilized for steering the direction of interpolation. For recovery of the lost motion vectors, an overlapped region matching algorithm is introduced. Several experimental results are presented. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Error concealment; Two layer ATM; Regularization; Oriented high pass operator; Overlapped region matching

1. Introduction

Error concealment is intended to ameliorate the effects of channel impairment, such as bit-errors in noisy channels or cell loss in packet networks, by utilizing a priori information about typical images in conjunction with available image redundancy to provide an acceptable rendition of affected image regions. Still images (JPEG encoded) are processed on a frame by frame basis [11]. The frame structure of H.261, H.263 and MPEG is composed of intra/inter frames and intra/inter/predictive frames [3,10,18]. The intra frames of compressed video sequences are basic frames on which the generation of inter frames is based. Therefore, information loss in intra frames may lead to adverse effects on succeeding inter frames. Such effects are accumulated consecutively until the next intra frame is processed, resulting in significant error propagation. The missing information of inter frames is motion vectors and predictive discrete cosine transform (DCT) coefficients. Therefore, the recovery of DCT

* Corresponding author. E-mail: aggk@ece.nwu.edu

0923-5965/99/$ – see front matter © 1999 Elsevier Science B.V. All rights reserved.
PII: S0923-5965(98)00061-7

coefficients of intra frames and motion vectors of inter frames is important under the assumption that critical overhead information is protected and received at decoder.

Error concealment methods can be grouped into: (1) selective recovery of lost information [7,25], (2) reordering of compressed video data so that it is more robust to channel errors [23], (3) error detection and correction of bit streams [16], and (4) concealment of the error effects at the post-processor [8,22,26]. Each group of method has its advantages and disadvantages. In the first group, there are several approaches, such as limiting error propagation, replenishment, and resynchronization. However, these approaches are quite expensive computationally. The second group, with methods such as interleaving and scrambling of compressed information, requires the modification of encoder and decoder. In addition, if the channel is not noise free, it needs extra computation to recover the missing information. The third group of methods requires extra overhead to protect the compressed information. On the other hand, recovery at the post-processor does not require any extra overhead and system modifications. Therefore, it is a desirable approach if post-processing is allowed.

The two steps in solving an error concealment problem are the detection of and the recovery of the errors. The asynchronous transfer mode (ATM) can be utilized to detect the location of missing blocks [6,20,24]. For the second step, various post-processing approaches have been reported in the literature. The use of smoothing constraints between lost data and neighbors was addressed in [26] to estimate the missing DCT coefficients. A projection onto convex sets (POCS) approach was proposed in [22]. Edge continuity and smoothness between the recovered block and its neighbors were enforced in discrete Fourier transform (DFT) domain. Salama et al. [19] proposed a Bayesian approach to error concealment. The original image is modeled as a Huber Markov Random Field (HMRF), using the Gibbs–Markov equality, where a maximum a posteriori (MAP) estimator was used to spatially interpolate missing macro blocks.

In this paper, we propose two methods to recover lost DCT coefficients and one method to recover lost motion vectors generated by an H.263 video codec. A two-layered ATM is used for packetization and transmission of the compressed video stream. The two methods used for the recovery of intra frames are an iterative regularized recovery algorithm and a spatially adaptive recursive interpolation algorithm. The knowledge that the missing block is correlated with its neighbors is used to estimate lost DCT coefficients, and to predict the direction of interpolation for the missing blocks. The lost motion vectors for inter frames are estimated using an overlapped region matching approach. The proposed algorithms lead to signal to noise ratio (SNR) improvement and containment of error propagation.

This paper is organized as follows. The two layer H.263 video codec is described in Section 2. The ATM cell structure of intra and inter frames is also shown. In Section 3 we propose an iterative regularized error concealment algorithm. An oriented high pass operator introduced and convergence of the algorithm is established. The adaptive interpolation algorithm is introduced in Section 4 and the overlapped motion vector estimation algorithm in Section 5. Finally experimental results are shown in Section 6, and conclusions are presented in Section 7.

2. Two layer ATM codec

A two layer ATM method is used in this work to treat and transmit the overhead and block information separately. Various results in the literature have used ATM for video codec [2,6,20]. A basic advantage of ATM is that it can detect the location of missing blocks. This is achieved by using a byte for storing the cell number in each packet. This technique can also be used with other packet network such as the internet. Our error concealment aglorithms are also applicable to the networks. A layer H.263 video encoder is shown in Fig. 1(a). The critical overhead information of picture, group of blocks (GOB), and macro block (MB) levels is transmitted through the base layer channel. The second layer channel is utilized for transmission of the DCT coefficients and motion vectors. The compressed information is packetized and transmitted via the base and the second layers. Fig. 1(b) shows

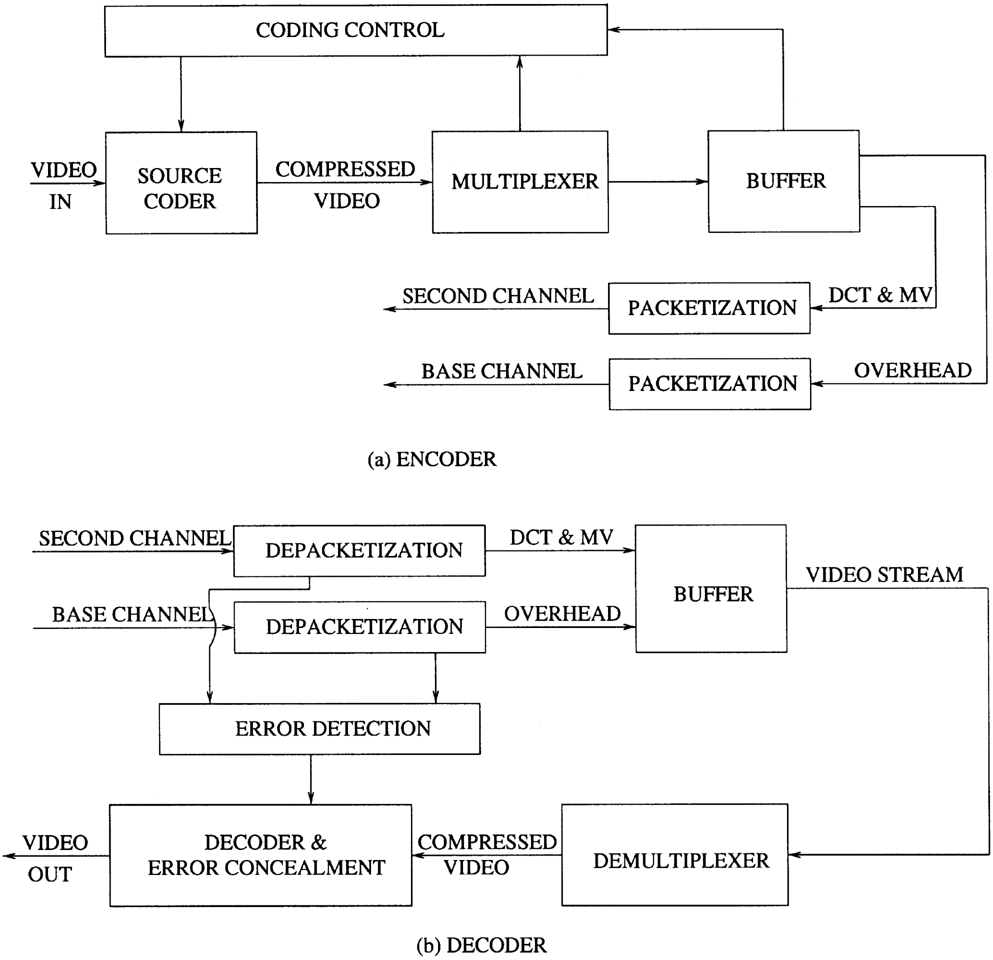

Fig. 1. Two layer H.263 (a) encoder and (b) decoder.

the block diagram of the decoder, where the decoder perform the inverse procedure of the encoder. The transmitted information is depacketized and cell errors are detected. The detected location of missing blocks is passed to the post processing unit (PPU) for error recovery. The above coder and decoder are compatible to existing H.263 codec, and therefore there is no need to modify existing systems for implementing the proposed error concealment algorithm.

In reported results in the literature, the motion vectors are assumed to be received without error, as they are packetized into the cells of the base layer. However, in order to better protect the base information, the motion vectors are assigned to the second layer. The recovery of lost motion vector is therefore addressed in the paper.

For transmission of video streams, fixed length cells of 48 bytes have been utilized. We use a method similar to the one in [6] for cell packetization for the transmission of the base layer data. 47 bytes are used for overhead information and one byte for the cell number. However, the second layer structure is different. Fig. 2(a) shows the cell structure of intra frames. 1 byte is assigned for cell number, MB address, and end-of-cell (EOC), respectively. Since 1 MB is composed of six 8×8 pixel blocks (4 for luminance and 2 for chrominance),

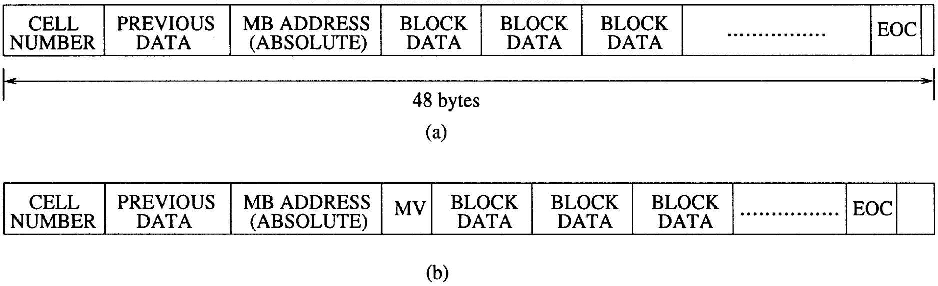

Fig. 2. Cell structure of (a) intra frames and (b) inter frames.

one of these six blocks might be assigned to two cells. If therefore the second cell is lost, the data of this block assigned to the previous cell also needs be discarded. In order to avoid this, a code word which recognizes the end of a block of a macro block is required. The 8 bit code word (1000 0000), not used in H.263, is used for the EOC. The data following the EOC is ignored. According to this structure, the beginning of a cell should be the starting information of a block. Fig. 2(b) shows the cell structure of inter frames.

3. Adaptive regularized DCT coefficients recovery

3.1. Problem formulation

When an image x of size $U \times V$ is compressed by an $M \times M$ dimensional block DCT, the original image and its DCT coefficients, X, are related by

$$X = Bx, \quad \text{and} \quad x = B^{\mathrm{T}}X, \tag{1}$$

where X and x are lexicographically ordered, and of size $UV \times 1$, and B is the $UV \times UV$ block circulant DCT matrix. By the unitary property of DCT, $BB^{\mathrm{T}} = I$. The DCT coefficients are quantized at the source coder to reduce the bit rate, and the quantized DCT coefficients are represented by

$$\hat{X} = Q[X], \tag{2}$$

where $Q[\cdot]$ denotes the quantization operator, and $\hat{X}$ represents the quantized version of the vector X. Some or all of the quantized DCT coefficients can be lost due to channel or system errors. Then the quantized DCT coefficients at the decoder can be described by

$$\hat{X} = \sum_{n \in \mathscr{R}} \hat{X}_n + \sum_{n \in \mathscr{L}} \hat{X}_n, \tag{3}$$

where $\mathscr{R}$ and $\mathscr{L}$ denote the set of received and lost coefficients, respectively, and $\hat{X}_n$ is the nth element of the vector $\hat{X}$. Then the problem at hand is to estimate the lost DCT coefficients from the received DCT coefficients. Eq. (3) can be rewritten as

$$\hat{X} = (I_{\mathrm{r}} + I_{\mathrm{l}})\hat{X} = I_{\mathrm{r}}\hat{X} + I_{\mathrm{l}}\hat{X}, \tag{4}$$

where I_{r}, I_{l} are diagonal $UV \times UV$ matrices that define the location of the received and lost DCT coefficients, respectively. The diagonal element of I_{r} and I_{l} are either 1 or 0, according to whether or not the coefficient is received. Since the roles of I_{r} and I_{l} are important to estimate the lost DCT coefficients, let us consider the properties. They are idempotent and orthogonal. That is,

$$\begin{aligned}
&I_{\mathrm{l}} = I - I_{\mathrm{r}},\\
&I_{\mathrm{r}} \cdot I_{\mathrm{r}} = I_{\mathrm{r}}, \qquad I_{\mathrm{l}} \cdot I_{\mathrm{l}} = I_{\mathrm{l}},\\
&I_{\mathrm{r}} = I_{\mathrm{r}}^{t}, \quad I_{\mathrm{l}} = I_{\mathrm{l}}^{t},\\
&I_{\mathrm{r}} \cdot I_{\mathrm{l}} = I_{\mathrm{l}} \cdot I_{\mathrm{r}} = 0.
\end{aligned} \tag{5}$$

Multiplying both sides of Eq. (4) with B^{T}, transforms it to the spatial domain. That is,

$$B^{\mathrm{T}}\hat{X} = B^{\mathrm{T}}I_{\mathrm{r}}\hat{X} + B^{\mathrm{T}}I_{\mathrm{l}}\hat{X}, \tag{6}$$

where $B^{\mathrm{T}}I_{\mathrm{r}}X$ is the observed data after decoding. Eq. (6) is rewritten as

$$x_{\mathrm{r}} = \hat{x} - B^{\mathrm{T}}I_{\mathrm{l}}\hat{X} = \hat{x} - x_{\mathrm{l}}, \tag{7}$$

where x_r, $\hat{x}$, x_l are the observed, reconstructed without channel errors, and lost in the channel, images, respectively.

3.2. Functional minimization

The problem at hand is the recovery of the lost coefficients or the estimation of $\hat{x}$ using the available information x_r, according to Eq. (7). Using a regularization approach [13], towards the solution of this problem, a solution is obtained by minimizing the following spatially adaptive functional:

$$M(\alpha, \hat{x}) = \|x_r - \hat{x}\|^2_{W_1} + \alpha\|C\hat{x}\|^2_{W_2}, \tag{8}$$

where α is the regularization parameter and C is the regularization operator. The choice of the diagonal weighting matrices W_1 and W_2 can be justified in various ways. For example, the noise visibility function [1] used in [13] was defined as a function of the local variance. We also use the local variance in this work to define W_1 and W_2. The local mean $m_{\hat{x}}$ and local variance $\sigma^2_{\hat{x}}$ are defined by

$$m_{\hat{x}}(i,j) = \frac{1}{(2P+1)(2Q+1)} \sum_{m=i-P}^{i+P} \sum_{n=j-Q}^{j+Q} \hat{x}(m,n), \tag{9}$$

$$\sigma^2_{\hat{x}} = \frac{1}{(2P+1)(2Q+1)} \sum_{m=i-P}^{i+P} \sum_{n=j-Q}^{j+Q} [\hat{x}(m,n) - m_{\hat{x}}(i,j)]^2, \tag{10}$$

where $(2P+1)(2Q+1)$ represents the extent of the two dimensional analysis window which is symmetric about the point (i,j). Then the $((i-1)\cdot U + j, (i-1)\cdot U + j)$ element of the diagonal matrix W_1 is defined by

$$W_1((i-1)\cdot U + j, (i-1)\cdot U + j) = \frac{\sigma^2_{\hat{x}}(i,j)}{\max_{i,j}\sigma^2_{\hat{x}}(i,j)} \quad \text{for } 1 \leqslant i \leqslant U,\ 1 \leqslant j \leqslant V. \tag{11}$$

The diagonal matrix W_2 is defined as $W_2 = I - W_1$. From Eq. (11), the entries of W_2 take values between 0 and 1. They are close to 0 at the edges, while 1 at the flat regions.

When all DCT coefficients of a block are lost (block burst), a considerable discontinuity is created between this burst and its neighborhood. The α trim mean filter is used in replacing the intensity values of a burst block by a constant value. Consider Fig. 3. An $M \times M$ burst block is shown and a black band of width one surrounding it. The intensity values of this band are ordered and denoted by $x_{\text{dec}}(i)$. If j is the index of their median, then

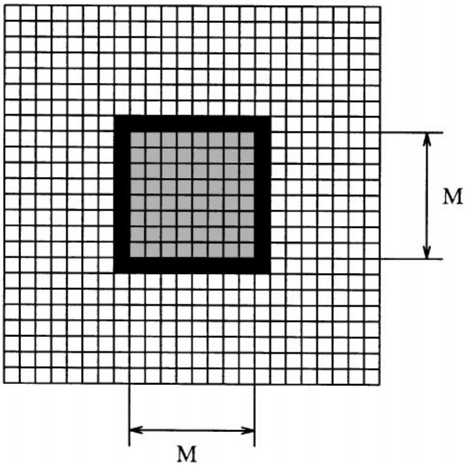

Fig. 3. Example of neighboring pixels.

$$E_{\alpha(\text{trim})}[x] = \frac{1}{2\cdot\alpha_{\text{trim}} + 1} \sum_{i=j-\alpha_{\text{trim}}}^{j+\alpha_{\text{trim}}} x_{\text{dec}}(i), \tag{12}$$

where $2\alpha_{\text{trim}}$ represents the values used for obtaining the mean. Clearly for $\alpha_{\text{trim}} = 0$, the median filter results, while for $\alpha_{\text{trim}} = 2(M+1)$, the mean filter results. Therefore, for a burst block, the received data x_r are represented by $E_{\alpha(\text{trim})}[x]$ from Eq. (12).

The functional $\|Cx\|$ is critical in the regularization approach followed in this work. It incorporates the prior information of a smooth original image into the recovery problem. The 2D Laplacian operator has been widely and successfully used in image restoration problems [9,12–14]. In the problem at hand, it does not seem to provide satisfactory results, primary at the boundary of two blocks, one with no lost coefficients and one which has lost all its coefficients in the channel.

We therefore propose the oriented high pass operator, shown in Fig. 4. An $M \times M$ damaged block is divided into 8 regions, up, down, left, right, and 4 diagonal regions. For each region, the high pass filtered value of $x(i,j)$ is given by

$$C[x(i,j)] = \begin{cases} -3x(i,j) + x(i-1,j+1) + x(i,j+1) + x(i+1,j+1) & \text{for } (i,j) \in \mathrm{R}1, \\ -3x(i,j) + x(i,j+1) + x(i+1,j+1) + x(i+1,j) & \text{for } (i,j) \in \mathrm{R}2, \\ -3x(i,j) + x(i+1,j+1) + x(i+1,j) + x(i+1,j-1) & \text{for } (i,j) \in \mathrm{R}3, \\ -3x(i,j) + x(i+1,j) + x(i+1,j-1) + x(i,j-1) & \text{for } (i,j) \in \mathrm{R}4, \\ -3x(i,j) + x(i-1,j-1) + x(i,j-1) + x(i+1,j-1) & \text{for } (i,j) \in \mathrm{R}5, \\ -3x(i,j) + x(i-1,j) + x(i-1,j-1) + x(i,j-1) & \text{for } (i,j) \in \mathrm{R}6, \\ -3x(i,j) + x(i-1,j+1) + x(i-1,j) + x(i-1,j-1) & \text{for } (i,j) \in \mathrm{R}7, \\ -3x(i,j) + x(i,j+1) + x(i-1,j+1) + x(i-1,j) & \text{for } (i,j) \in \mathrm{R}8. \end{cases} \tag{13}$$

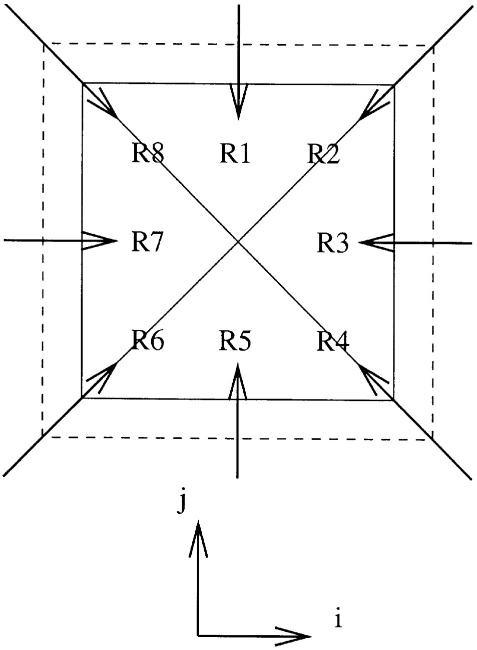

Fig. 4. Oriented high pass filter.

The first term on the right hand side of Eq. (8) expresses the fidelity to the data while the second term represents the requirement that the solution is smooth. The regularization parameter α controls the trade-off between these two requirements. For $\alpha = 0$, the solution of Eq. (8) is the observed data, x_r. On the other hand, for $\alpha = \infty$, the solution is an image of constant intensity (over-smoothed). Using Eq. (7), Eq. (8) can be written as

$$M(\alpha, \hat{X}) = \|B^\mathrm{T} I_\mathrm{l} \hat{X}\|^2_{W_1} + \alpha \|C(x_\mathrm{r} + B^\mathrm{T} I_\mathrm{l} \hat{X})\|^2_{W_2}. \tag{14}$$

A necessary condition for $M(\alpha, \hat{X})$ to have a minimum is that its gradient is equal to zero. That is,

$$\begin{aligned} \frac{\partial M(\alpha, \hat{X})}{\partial \hat{X}} &= 2 I_\mathrm{l}^\mathrm{T} B W_1^\mathrm{T} W_1 B^\mathrm{T} I_\mathrm{l} \hat{X} \\ &\quad + 2\alpha I_\mathrm{l}^\mathrm{T} B C^\mathrm{T} W_2^\mathrm{T} W_2 C(x_\mathrm{r} + B^\mathrm{T} I_\mathrm{l} \hat{X}) \\ &= 0. \end{aligned} \tag{15}$$

Let us define $W_A = W_1^\mathrm{T} W_1$ and $W_B = W_2^\mathrm{T} W_2$. Due to the properties of B and I_l and I_r, Eq. (15) can be rewritten as

$$\begin{aligned} &(I_\mathrm{l} B W_A B^\mathrm{T} I_\mathrm{l} + \alpha I_\mathrm{l} B C^\mathrm{T} W_B C B^\mathrm{T} I_\mathrm{l}) I_\mathrm{l} \hat{X} \\ &\quad = -\alpha I_\mathrm{l} B C^\mathrm{T} W_B C x_\mathrm{r} \end{aligned} \tag{16}$$

or

$$K(\alpha) I_\mathrm{l} \hat{X} = -\alpha I_\mathrm{l} B C^\mathrm{T} W_B C x_\mathrm{r}, \tag{17}$$

where $K(\alpha) = (I_\mathrm{l} B W_A B^\mathrm{T} I_\mathrm{l} + \alpha I_\mathrm{l} B C^\mathrm{T} W_B C B^\mathrm{T} I_\mathrm{l})$. Replacing $I_\mathrm{l} \hat{X}$ from Eq. (7) into Eq. (16) we obtain

$$K(\alpha) B \hat{x} = (K(\alpha) B - \alpha I_\mathrm{l} B C^\mathrm{T} W_B C) x_\mathrm{r} \tag{18}$$

or

$$K(\alpha) \hat{X} = (K(\alpha) B - \alpha I_\mathrm{l} B C^\mathrm{T} W_B C) x_\mathrm{r}. \tag{19}$$

$K(\alpha)$ as well as $K(\alpha)B$ are nonsingular matrices. Therefore, Eq. (18) can be solved directly. However, due to the large size of the matrices an iterative solution of Eq. (18) is proposed in Section 3.3. According to Eq. (17), if $C = I$ and $x_\mathrm{r} = 0$ (no information is received for the whole block), no lost information can be recovered. However, for $C \neq I$, even if $x_r = 0$ for the whole block, the received information in the neighboring blocks will be used in recovering the lost information in the current block.

With the proposed regularization approach, the choice of the regularization parameter is critical, and depends upon the amount of available prior information. When the noise variance and high frequency energy are given, the set theoretic approach can be used to determine the regularization parameter [13,14]. If only one of the two pieces of information is available, the Constrained Least Squares (CLS) approach [9] can be used. When no prior information is available, the regularization parameter can be defined iteratively using the partially restored image [12].

Following the set theoretic regularization approach, the regularization parameter is chosen as

$$\alpha = \left(\frac{\|B^T I_l \hat{X}\|^2_{W_1}}{\|C\hat{x}\|^2_{W_2}} \right), \tag{20}$$

where C is normalized, i.e, $C^T C = 1$, and using Parseval's theorem, α can be rewritten as

$$\alpha = \frac{\|B^T I_l \hat{X}\|^2_{W_1}}{\|\hat{X}\|^2_{W_2}} \leqslant \frac{\|I_l \hat{X}\|^2_{W_1}}{\|I_l \hat{X}\|^2_{W_2} + \|I_r \hat{X}\|^2_{W_2}}, \tag{21}$$

where $\|\hat{x}\|^2 = \|I_l \hat{X}\|^2 + \|I_r \hat{X}\|^2$, since $\hat{X} = (I_r + I_l)\hat{X}$ and $I_r \cdot I_l = 0$. Clearly from Eq. (21), $0 \leqslant \alpha \leqslant 1$. If no coefficients are lost in the channel, $\alpha = 0$, while if all coefficients are lost, $\alpha = 1$.

In estimating $\|I_l \hat{X}\|^2$ to be used in the calculation of α in Eq. (21), the eight neighboring blocks of a degraded blocks are used. Let $X_{(m,n)}(k,l)$ denote the (k,l)th coefficient of (m,n)th block. If $X_{(m,n)}(k,l)$ is missing, then we use

$$\hat{X}^2_{(m,n)}(k,l) = \frac{1}{S} \sum_i \sum_j \hat{X}^2_{(i,j)}(k,l), \ (i,j) \neq (m,n), \tag{22}$$

$$\|(I_l \hat{X})_{(m,n)}\|^2 = \sum_k \sum_l \hat{X}^2_{(m,n)}(k,l), \tag{23}$$

$$\|I_l \hat{X}\|^2 = \sum_m \sum_n \|(I_l \hat{X})_{(m,n)}\|^2, \tag{24}$$

where S denotes the number of errorless neighboring blocks, and (i,j) represents the location of errorless neighboring blocks.

3.3. Iterative solution and convergence analysis

From Eq. (18), the recovery algorithm becomes in the DCT domain

$$\hat{X}_0 = \beta B x_r, \tag{25}$$

$$\hat{X}_{k+1} = \beta[K(\alpha)B - \alpha I_l B C^T W_B C]x_r + [I - \beta K(\alpha)]\hat{X}_k, \tag{26}$$

where β is the relaxation parameter to ensure convergence of the algorithm. Multiplying both sides of the equation above by B^T we obtain the iteration in the spatial domain, that is,

$$\hat{x}_0 = \beta x_r, \tag{27}$$

$$\hat{x}_{k+1} = \beta B^T[K(\alpha)B - \alpha I_l B C^T W_B C]x_r + [I - \beta B^T K(\alpha) B]\hat{x}_k. \tag{28}$$

Generally, in dealing with iterative algorithms the convergence as well as the rate of convergence are very important issues. The contraction mapping theorem serves as basis for establishing convergence of iterative algorithms. Let us rewrite Eq. (25) as $\hat{X}_{k+1} = G\hat{X}_k$. According to this theorem, iteration (25) converges to a unique fixed vector, X^* such that $G\hat{X}^* = \hat{X}^*$, for any initial condition if G is a contraction. It means that for any two vectors $\hat{X}_1$ and $\hat{X}_2$ in the domain of G the following relation holds:

$$\|G\hat{X}_1 - G\hat{X}_2\| \leqslant \eta \|\hat{X}_1 - \hat{X}_2\|, \tag{29}$$

where η is strictly less than 1. Since the variation of W_A and W_B between two consecutive iterative processes are close to 0, the sufficient condition for convergence of Eq. (29) is

$$\|I - \beta K(\alpha)\| < 1, \tag{30}$$

or, equivalently,

$$0 < \beta < \frac{2}{\max_i \lambda_i(K(\alpha))}, \tag{31}$$

where $\lambda(A)_i$ is the ith eigenvalue of A.

4. An adaptive recursive interpolation algorithm

4.1. Basic idea of the algorithm

Several interpolation methods for error concealment of compressed intra frames have been reported in the literature [6,20]. Most of them use bilinear interpolation. They work well in areas of low spatial activity. However, such algorithms fail to obtain satisfactory results in high spatial activity regions. In this section we propose an interpolation algorithm which is capable of handling edges passing through the missing areas. In other words, the direction of interpolation depends on the direction of the edge, in the neighborhood of the missing data. This technique is similar to the Voronoi tessellations [5,21]. Clearly, the notation of neighborhood is tied to the metric d, which is used. In this algorithm the metric d is defined with the use of the maximum norm, so that the distance between two points, $p_1 = (i_1, j_1)$ and $p_2 = (i_2, j_2)$ is given by

$$d(p_1, p_2) = \max(|i_1 - i_2|, |j_1 - j_2|). \tag{32}$$

This defines different orders of neighbors depending on the value of the distance function d. In the proposed algorithm, only neighbors of order one ($d(p_1, p_2) = 1$) and two ($d(p_1, p_2) = 2$) are used for the interpolation of missing data. As is known the convolution operation extends the support of a signal and therefore distributes the available information to the neighbors. This distribution of information to neighbors of up to order two can be done by convolving the data with a function $h(i,j)$ with

$$\mathrm{support}(h) \subset [-2,2] \times [-2,2], \tag{33}$$

where × is the Cartesian product. In Eq. (33), $h = 1$ for those neighbors used for interpolation, and $h = 0$ otherwise. Which pixels will be used for interpolation is determined by the edges around the missing data. For example, if an edge is passing through a missing area in the north-south direction, then only nonzero neighbors in the same direction are used, that is, neighbors with relative coordinates $\{(0,-1), (0,-2), (0,1), (0,2)\}$ (see Fig. 5). The way the function $h(i,j)$ is created is explained in Section 4.3. Again, let x be an image of size $U \times V$. The two dimensional convolution of x and h is denoted by the operator A_h, that is,

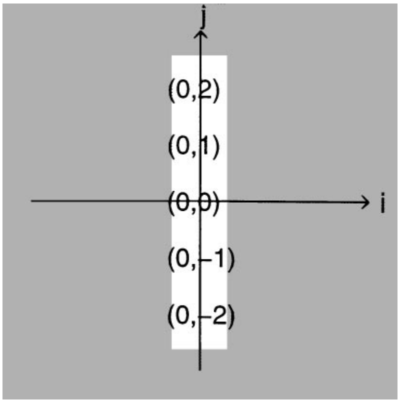

Fig. 5. Example of neighbors with relative coordinates.

$$A_h(x) = x{*}{*}h, \tag{34}$$

where ** represents the two dimensional convolution. Since $\mathrm{support}(h) \neq \emptyset$,

$$\mathrm{support}(x) \subset \mathrm{support}(A_h(x)). \tag{35}$$

In other words, information from pixels surrounding missing data is distributed inside the missing areas of x by using the operator A_h. Since a pixel can have a number of nonzero neighbors of order $\leqslant 2$ the distributed information has to be weighted by the inverse number of the neighbors which are used. For example, if four neighbors are used for the interpolation of a certain point (i_0, j_0), the result of the convolution $A_h(x)(i_0, j_0)$ must be weighted by $\frac{1}{4}$. It is clear that the neighbors which belong to the set of missing data must not be counted. These weights are derived as follows. First an indicator function u is defined:

$$\begin{aligned} &u(i,j) = 1 \text{ if data are available for the location } (i,j),\\ &u(i,j) = 0 \text{ otherwise,} \end{aligned} \tag{36}$$

where $(i,j) \in (1,\dots,U) \times (1,\dots,V)$. Then the operator A_h is applied to this indicator function. Since both functions u and h only have values equal to 1 or 0, $A_h(u)$ counts the number of nonzero neighbors, and therefore the weight for a certain point (i_0, j_0) is given by $(A_h(u)(i_0, j_0))^{-1}$. For those points where no nonzero neighbors of order $\leqslant 2$ are available or used, the weights are set equal to zero. As a result, a weighted version of the operator A_h is defined by

$$\tilde{A}_h(x)(i,j) = \frac{A_h(x)(i,j)}{A_h(u)(i,j)} \quad \text{for } A_h(u)(i,j) \neq 0, \tag{37}$$

$$\tilde{A}_h(x)(i,j) = 0 \quad \text{for } A_h(u)(i,j) = 0.$$

4.2. Description of the algorithm

As described in Section 3.1, the decoded image without channel errors $\hat{x}$ should be estimated using the received data x_r. Then, the indicator function becomes

$$u(i,j) = \begin{cases} 1 & \text{if } (i,j)\text{th pixel is received,} \\ 0 & \text{otherwise.} \end{cases} \tag{38}$$

The set of missing data (gaps) is defined as

$$\mathscr{G} = \{(i,j) | u(i,j) = 0\}. \tag{39}$$

The problem at hand is to recover an image using h which is determined on the basis of edges in the near neighborhood. The first step of the algorithm is to find a sufficiently large rectangle around one gap or a set of adjoining gaps. Let $\mathscr{S}$ denote such a rectangle of size $U_1 \times V_1$ $(U_1 \leqslant U, V_1 \leqslant V)$, where the left upper corner of the rectangle is at the location (i_1, j_1). Then, the gaps inside $\mathscr{S}$ are described by

$$\tilde{\mathscr{G}} = \mathscr{S} \cap \mathscr{G}. \tag{40}$$

Using the existing edges in this rectangle, a function h is determined, as explained in Section 4.3. Then, the interpolation procedure proceeds as follows: First, define the initial condition as

$$x^0(i,j) = x_r(i + i_1 - 1, j + j_1 - 1), \tag{41}$$

where $(i,j) \in \{1, \ldots, U_1\} \times \{1, \ldots, V_1\}$, which is the restriction of x_r on the rectangle $\mathscr{S}$. The gaps at first step become

$$\mathscr{G}^0 = \tilde{\mathscr{G}}. \tag{42}$$

To estimate the missing data at the boundary of the gap, we apply the operator $\tilde{A}_h$ to x^0, i.e.,

$$y^0 = \tilde{A}_h(x^0). \tag{43}$$

As mentioned before, we get

$$\mathtt{support}(x^0) \subset \mathtt{support}(y^0), \tag{44}$$

which means that y^0 also contains values $\neq 0$ inside the gap $\mathscr{G}^0$. We update the existing data with these new data by

$$x^1(i,j) = \begin{cases} y^0(i,j) & \text{for } (i,j) \in \mathscr{G}^0, \\ x^0(i,j) & \text{for } (i,j) \notin \mathscr{G}^0, \end{cases} \tag{45}$$

where $(i,j) \in \{1, \ldots, U_1\} \times \{1, \ldots, V_1\}$. The recursive algorithm continues by

$$y^{l+1} = \tilde{A}_h(x^{l+1}), \tag{46}$$

$l = 0,1,2, \ldots L$. A finite sequence of contracting gaps is obtained by

$$\mathscr{G}^0 \supset \mathscr{G}^1 \supset \cdots \supset \mathscr{G}^l \supset \cdots \supset \mathscr{G}^L = \emptyset, \tag{47}$$

where $\mathscr{G}^l = \{(i,j) | x^l(i,j) = 0\}$. In each step, the indicator function u is updated.

After filling the gap in this rectangle, we are looking for the next rectangle and continue the restoration until all gaps in x_r are filled.

4.3. Choice of h and $\mathscr{S}$

As pointed out, the missing data should be determined by the neighbors of order $\leqslant 2$. Therefore, the algorithm uses h with

$$\mathtt{support}(h) \subset [-2,2] \times [-2,2]. \tag{48}$$

If there are edges passing through the missing areas, we want to use only their neighbors which have the same direction to the edges. It means that the elements of the function h should describe the orientation of the edges around the missing areas. In order to define the edge direction, the eight directions which are rotated by 22.5°, like N-S, NNE-SSW, NE-SW, NEE-SWW, E-W are defined (some examples are shown in Figs. 6 and 7).

Every direction defines a different function $h_{t=\{1,2,\ldots,8\}}$, and the degree of the degradation of the edges around the missing area is measured by convolving the data with these functions h_t. The eight l_2-norms measuring the degradations of

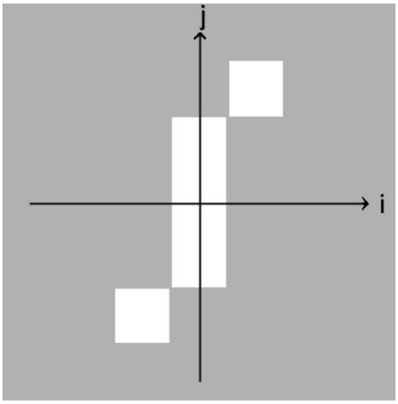

Fig. 6. NNE-SSW edge direction.

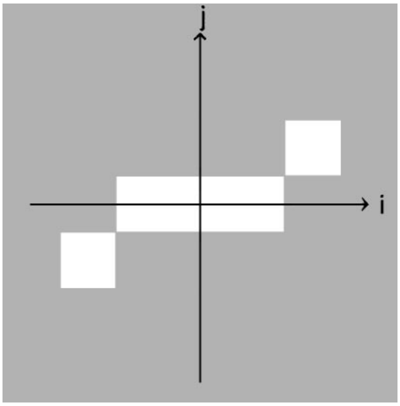

Fig. 7. NEE-SWW edge direction.

each convolution process ($t = \{1, 2, \dots, 8\}$) are defined as

$$n_t = \left(\sum_{(i,j)\in\mathscr{B}} (x_{\mathrm{r}}(i,j) - \tilde{A}_{h_t}(x_{\mathrm{r}})(i,j))^2 \right)^{1/2}, \tag{49}$$

where $\mathscr{B}$ is the set resulting from $\mathscr{S}$ by removing a band around the boundary of $\mathscr{S}$ (due to the use of circulant convolution) and a band around the area of missing data.

Since the h_t with the same orientation with the edges in the neighborhood of the missing data causes a smaller degradation, a criterion for choosing the directions by using a threshold $\tau > 1$ is defined as

$$U = \{s \in \{1, \dots, 8\} | n_s \leqslant \tau * \min_{t=\{1,2,\dots,8\}}(n_t)\}. \tag{50}$$

The experiments have shown that $\tau = 1.7$ yields the best results. Then, the weights for each of the eight directions are given by

$$w_t = \begin{cases} 1 & \text{for } t \in U, \\ 0 & \text{for } t \notin U. \end{cases} \tag{51}$$

The function h is the combination of these directions, that is,

$$h(i,j) = \mathrm{sign}\left(\sum_{t=1}^{8} w_t h_t(i,j) \right). \tag{52}$$

5. Motion vector estimation

Inter frames are reconstructed using the motion vectors and the DCT coefficients of the prediction error. Therefore, the loss of the motion vectors seriously degrades the decoded image. This degradation propagates to the subsequent inter frames until an intra frame is encountered. The reconstruction of the lth inter frame takes the form

$$\hat{x}(i,j;l) = \hat{x}(i + Vx, j + Vy; l - 1) + \hat{x}_p(i,j;l), \tag{53}$$

where (Vx, Vy) represents the motion vector for the (i,j)th pixel and $\hat{x}_p(i,j;l)$ denotes the prediction error.

The encoding mechanism of motion vectors in H.263 consists of taking the difference between the current motion vector and the median vector of the neighboring macro blocks. Therefore, the loss of a macro block motion vector propagates to the remaining macro blocks in the frame. In Eq. (53), Vx and Vy are determined by

$$\begin{aligned} Vx &= Dx + Px, \\ Vy &= Dy + Py, \end{aligned} \tag{54}$$

where Px and Py are the median values of the three neighboring macro blocks, and Dx and Dy are the transmitted differential vectors.

Several motion vector recovery algorithms have appeared in the literature [4,17,19,20,25]. In [20,25], the lost motion vector is replaced by the average of motion vectors of the neighboring macro blocks. A Bayesian approach is utilized in [19]. In [17], a boundary matching (BM) approach

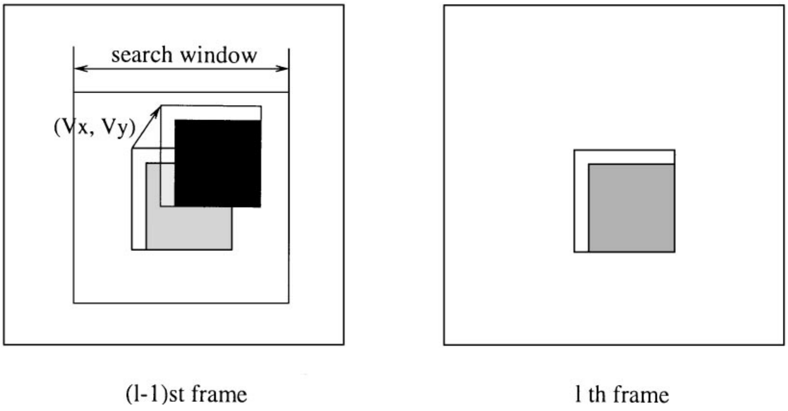

Fig. 8. Motion vector estimation.

is proposed to replace the lost motion vector by a motion vector which minimizes a cost function. In [4], side matching was used to estimate the lost motion vector.

In this paper, we propose an overlapped region approach for the recovery of lost motion vectors. This approach also appears in [15]. This approach has the advantage that edge or signal continuity between the missing block and its neighbors can be preserved. The motion vector is re-estimated without requiring any differential information from the neighboring blocks. Fig. 8 depicts the idea behind the algorithm. Two frames are shown in it, the current lth frame on the right and the previous $(l-1)$th frame on the left. The gray region represents a missing block in the frame, and the band above and to the left of it the neighboring pixels to be used for the estimation of the lost motion vectors. Such a band is used since only blocks to the left and above the current block have been decoded. The $(l-1)$st decoded frame is utilized as a reference frame, and the motion vector for the lost macro block is determined as

$$(Vx,Vy) = \arg\left\{ \min_{(m,n)\in S_{mv}} \sum_i \sum_j |\hat{x}(i,j;l) - \hat{x}(i-m,j-n;l-1)| \right\}, \tag{55}$$

where (i,j) represents the pixels inside the band to the left and above of the missing block, S_{mv} denotes the search region in the previous frame, and $|\cdot|$ denotes the absolute value. An example of a match is shown in the left part of Fig. 8.

Since region matching assumes that the displacement within the region is homogeneous, support of this region is critical. A large support decreases the reliability of the estimated motion vectors, because the correlation between the missing block and the region around it used for matching is reduced. On the other hand, a small support region might be matched to various locations in the previous frame. On the basis of our experiments, 4–8 rows and columns of neighboring pixels result in good matching results.

6. Experimental results

A number of experiments have been performed with the proposed algorithms. Two 176×144 QCIF ('mother and daughter' and 'foreman') sequences were used, and the criterion for evaluating the performance is peak to signal to noise ratio (PSNR). When an 8 bit $U\times V$ image is used, PSNR is defined by

$$\mathrm{PSNR} = 10\log\frac{255^2(UV)}{\|x-\tilde{x}\|^2}, \tag{56}$$

where x and $\tilde{x}$ denote the original and the restored images, respectively. The criterion $\|x_{k+1}-x_k\|^2/\|x_k\|^2 \leqslant 10^{-6}$ was used for terminating the iteration algorithm proposed in Section 3.

Table 1
Total cell number and cell loss number for encoded frames (mother and daughter sequence)

	Total cell no.	Cell loss no.
1st fr.	47	3
8nd fr.	16	2
10th fr.	17	0
13th fr.	27	1
18th fr.	28	2
22th fr.	19	1
25th fr.	19	0
28th fr.	22	1

Table 2
Total cell number and cell loss number for encoded frames (foreman sequence)

	Total cell no.	Cell loss no.
1st fr.	88	5
13nd fr.	21	2
16th fr.	23	1
20th fr.	18	0
23th fr.	20	1
27th fr.	24	1

Since the spatial activity of the ‘mother and daughter’ sequence is lower than that of the ‘foreman’ sequence, the encoded frame number of the first sequence is smaller than that of the second sequence, as shown in Tables 1 and 2. Also, the number of cells used to packetize the first sequence is relatively small. Therefore, one cell loss of the first sequence leads to a relatively large loss of macro blocks. When the bit rate is 64 kbits/s, the average macro block number packetized into a cell is 2 for intra frames and 4.5 for inter frames of the ‘mother and daughter’ sequence, while 1 for intra frames and 4.7 for inter frames of the ‘foreman’ sequence. Tables 1 and 2 show the encoded frame numbers and packetized cell numbers. Typical switches operate at the Mbps range. Therefore, the traffic generated by the source we are investigating (Kbps range) is a small percentage of the total traffic. In other words, the average length of time between the packets generated by my source, is considerably greater than the average length of time the buffer is full for a typical switch. Because of this, although the loss at the switch is bursty, the loss my packets are experiencing is not bursty, but rather uniformly distributed. In this experiments, 5% cell loss is generated by a random number function, as was also used in [6].

6.1. Recovery of intra frames

Figs. 9 and 10 show the decoded and degraded intra frames of the mother and daughter sequence. When the lost cells include background or simple edges, the restored image leads to satisfactory results. Figs. 11 and 12 show the restored images

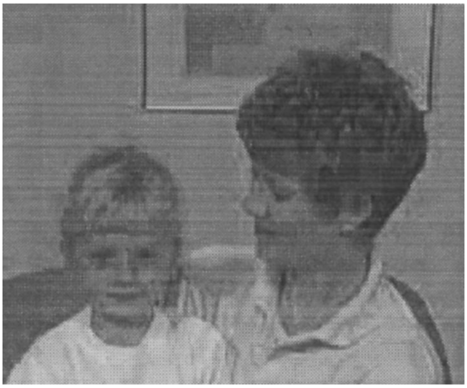

Fig. 9. Decoded 1st frame of mother and daughter sequence (intra frame), PSNR = 33.55 dB.

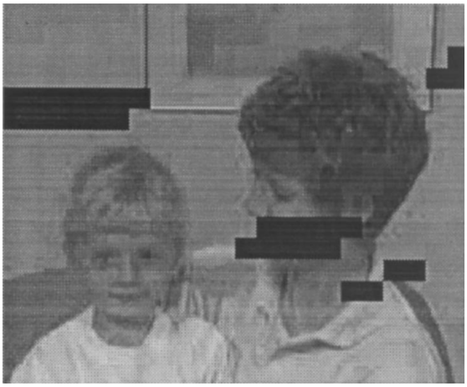

Fig. 10. Missing block location of 1st frame of mother and daughter sequences (intra frame).

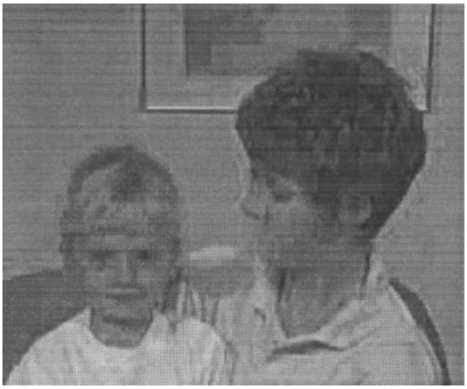

Fig. 11. Restored 1st frame of mother and daughter sequences (intra frame), proposed method in Section 3, PSNR = 31.71 dB.

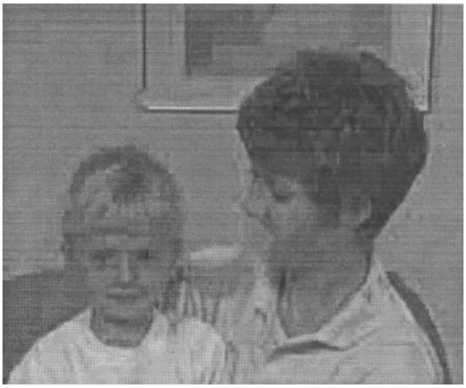

Fig. 12. Restored 1st frame of mother and daughter sequences (intra frame), proposed method in Section 4, PSNR = 31.60 dB.

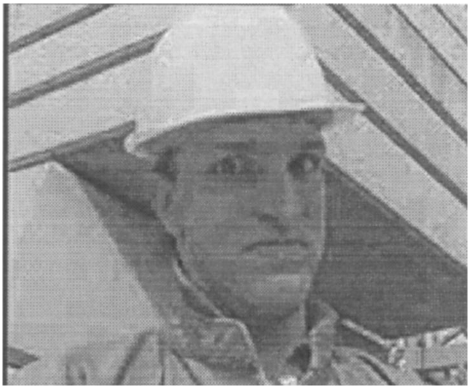

Fig. 13. Decoded 1st frame of foreman sequences (intra frame), PSNR = 32.44 dB.

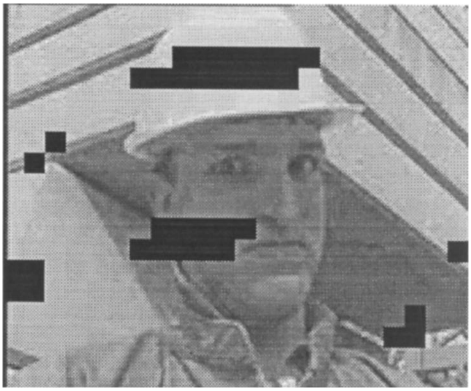

Fig. 14. Missing block location of 1st frame of foreman sequences (intra frame).

proposed in Sections 3 and 4, respectively. However, if the lost cells contain an isolated edge or occlusions, the neighbors do not support reliable information to recover the missing areas, resulting in failure of obtaining the satisfactory results. The failure of recovery propagates the bad effects to the next inter frames until the next intra frame is encoded.

The proposed algorithm in Section 3 leads to satisfactory results, when the edge direction of the missing block is the same as the proposed oriented high pass smoothing direction. On the other hand, the proposed algorithm in Section 4 leads to satisfactory results, when the neighbors can support reliable interpolation direction.

Figs. 13 and 14 show the decoded and degraded 1st frames of the foreman sequence, respectively. The restored images proposed in Sections 3 and 4 are shown in Figs. 15 and 16. The recovered image by the iterative regularization approach leads to a blurred version in the areas where the edge direction of the missing blocks is quite different from the smoothing direction of the proposed high pass operator. On the other hand, the recursive interpolation succeeds in extracting the interpolation direction of the missing blocks from its neighbors, because the edge of neighbors passes through the missing blocks.

The normalized iteration step error for the algorithm in Section 3 is shown in Fig. 17 as a function

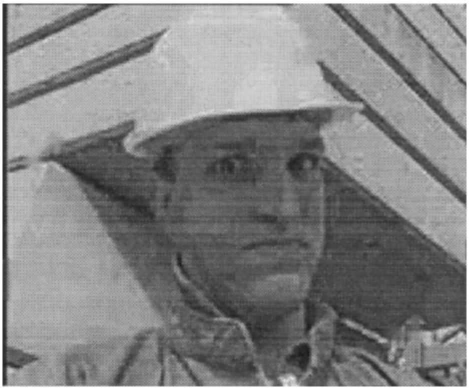

Fig. 15. Restored 1st frame of foreman sequences (intra frame), proposed method in Section 3, PSNR = 30.03 dB.

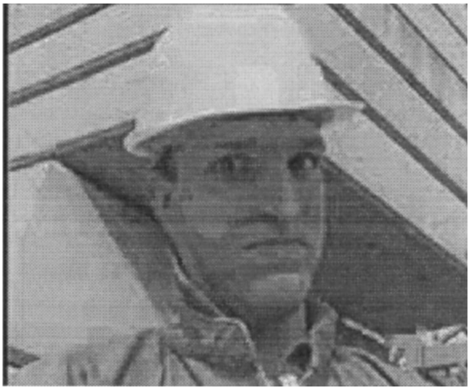

Fig. 16. Restored 1st frame of foreman sequences (intra frame), proposed method in Section 4, PSNR = 31.00 dB.

of the iteration number for the two experiments described in this section. Since the termination criterion $\|x_{k+1} - x_k\|^2/\|x_k\|^2 \leqslant 10^{-6}$ is used, the algorithm converged after 9 iterations for the 'foreman' frame and 10 iterations for the 'mother and daughter' frame.

6.2. Recovery of motion vectors

We tested the motion vector recovery algorithm proposed in Section 5 and compared it with the approaches of finding the average and the median of the neighboring motion vectors of the missing

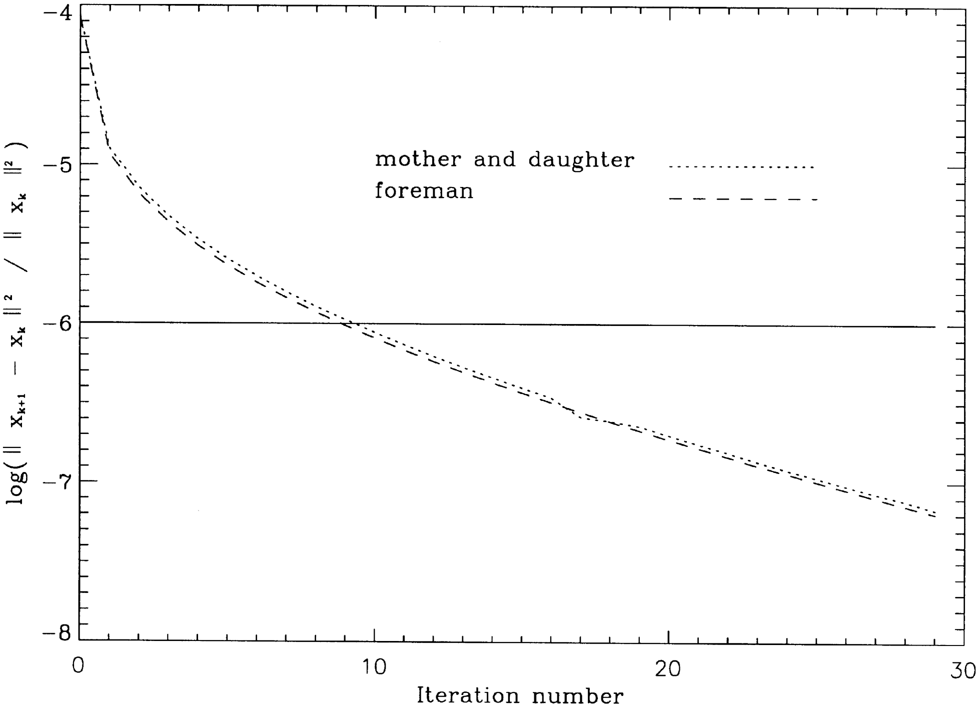

Fig. 17. Normalized iteration step error versus iteration number.

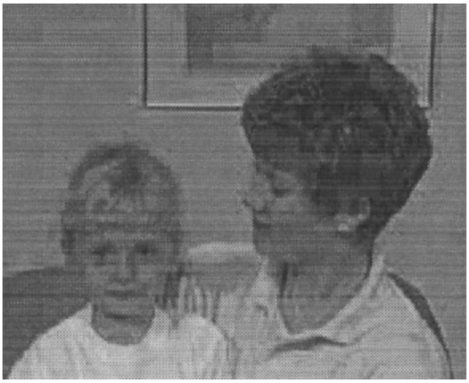

Fig. 18. Decoded 8th frame of mother and daughter sequences (inter frame), PSNR = 33.55 dB.

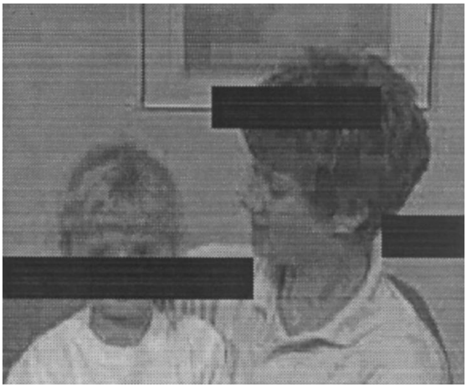

Fig. 19. Missing macro block location of 8th frame of mother and daughter sequences (inter frame).

macro block. The vector average approach used the motion vectors on the left and above the missing macro block, and the vector median approach utilized the motion vector on the left and the two motion vectors above the missing macro block. Fig. 18 shows the decoded 8th frame of the first sequence, and the black of Fig. 19 represents the location of the missing motion vectors. The reconstructed images in Figs. 20–22 show the results by the average, median and proposed motion vector recovery algorithms, respectively, when the missing blocks of intra frames are restored by the algorithm in Section 3. When the motion vectors are lost in the areas of no motion, the result of the proposed

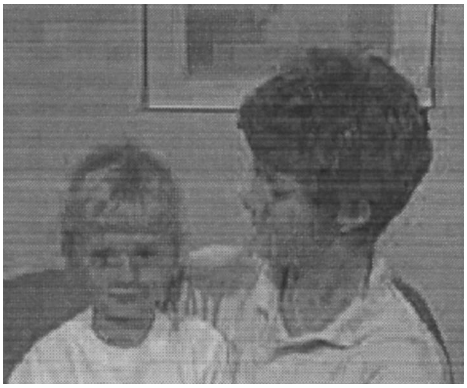

Fig. 20. Restored 8th frame of mother and daughter sequences (inter frame), average motion vector of neighbors, PSNR = 30.44.

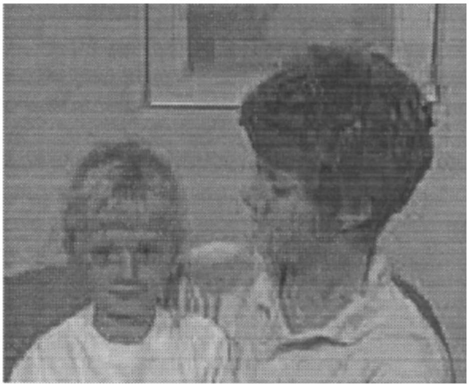

Fig. 21. Restored 8th frame of mother and daughter sequences (inter frame), median motion vector of neighbors, PSNR = 29.35 dB.

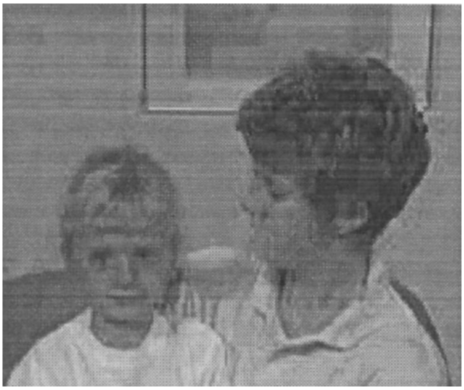

Fig. 22. Restored 8th frame of mother and daughter sequences (inter frame), proposed algorithm, PSNR = 30.77 dB.

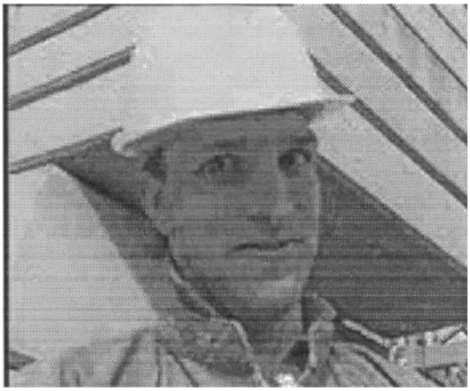

Fig. 23. Decoded 14th frame of foreman sequence (inter frame), PSNR = 31.26.

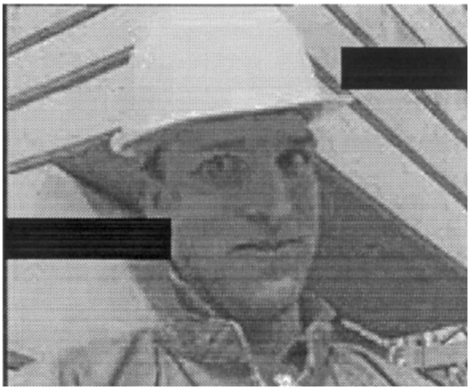

Fig. 24. Missing macro block location of 14th frame of foreman sequence (inter frame).

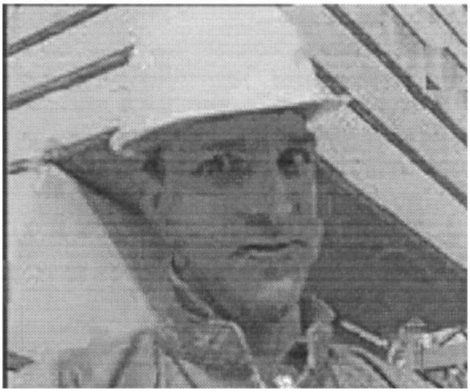

Fig. 25. Restored 14th frame of foreman sequences (inter frame), average motion vector of neighbors, PSNR = 27.13 dB.

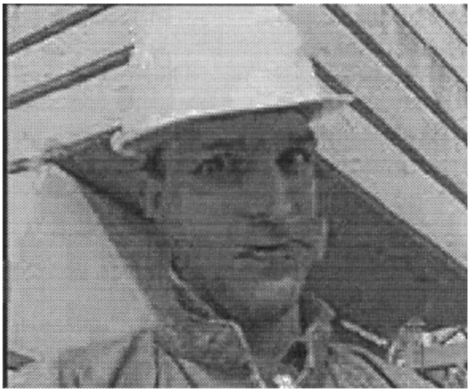

Fig. 26. Restored 14th frame of foreman sequences (inter frame), median motion vector of neighbors, PSNR = 28.24 dB.

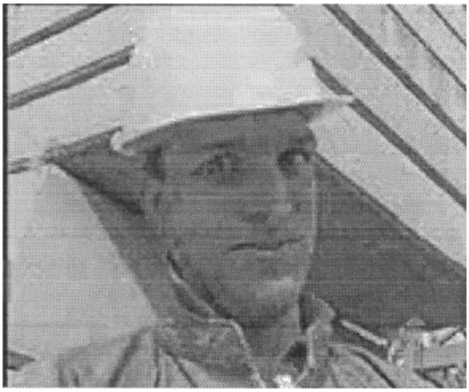

Fig. 27. Restored 14th frame of foreman sequences (inter frame), proposed algorithm, PSNR = 29.10.

algorithm is similar to the ones obtained by the other two approaches. Also, the errors caused by motion vector estimation are not propagated, since the median vector filter can correct the (vector mean and vector median) errors. However, when the lost cell includes the motion vectors of moving areas, the two approaches (vector mean and vector median) result in the image discontinuities. On the other hand, the proposed algorithm leads to satisfactory results, since the motion recovery algorithm is obtained using neighborhood information. The blurring of the 'lip' comes from the failure to recover this information in the intra frame.

The improved performance of the proposed motion vector recovery algorithm is demonstrated with foreman sequence. Figs. 23 and 24 show the

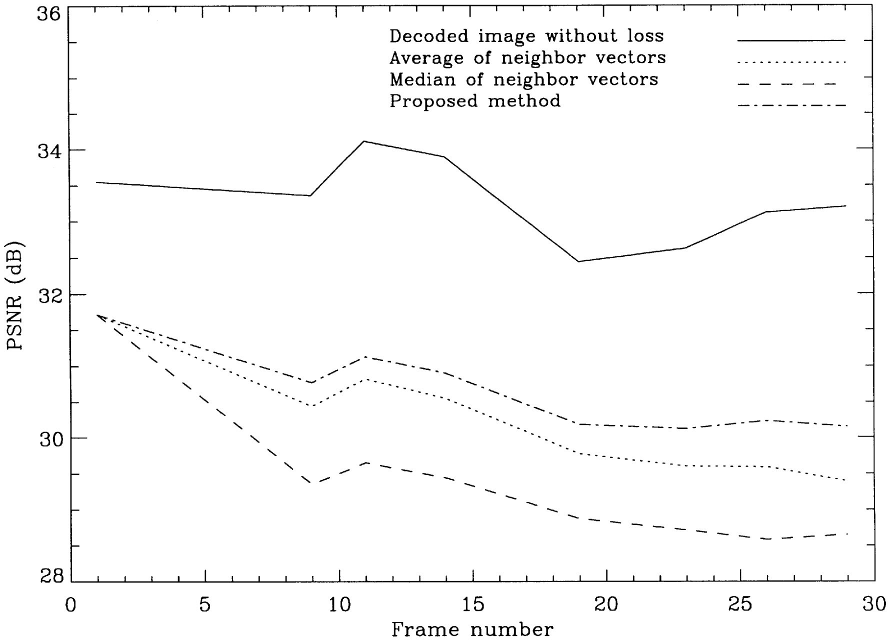

Fig. 28. PSNR comparison (mother and daughter sequence, intra frame recovery: proposed method in Section 3).

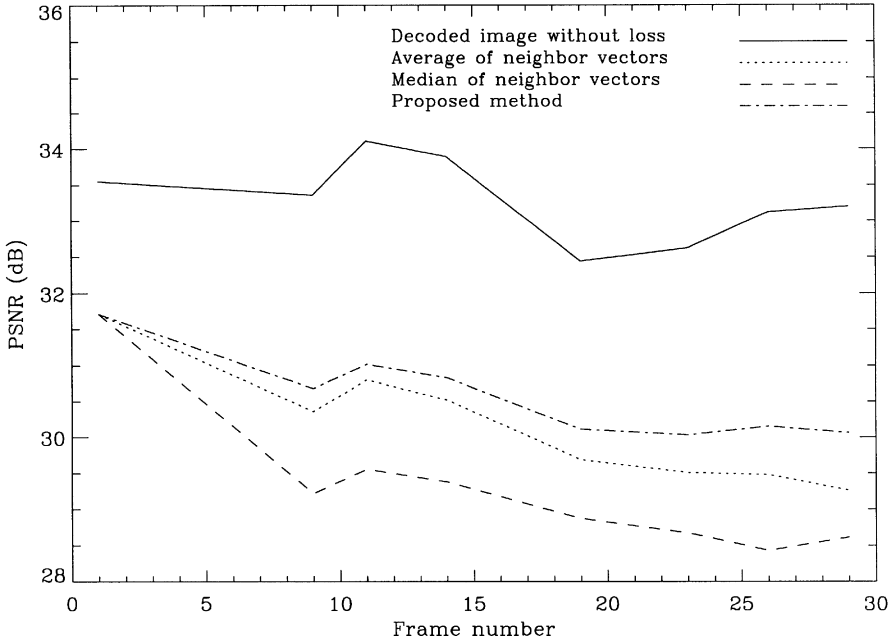

Fig. 29. PSNR comparison (mother and daughter sequence, intra frame recovery: proposed method in Section 4).

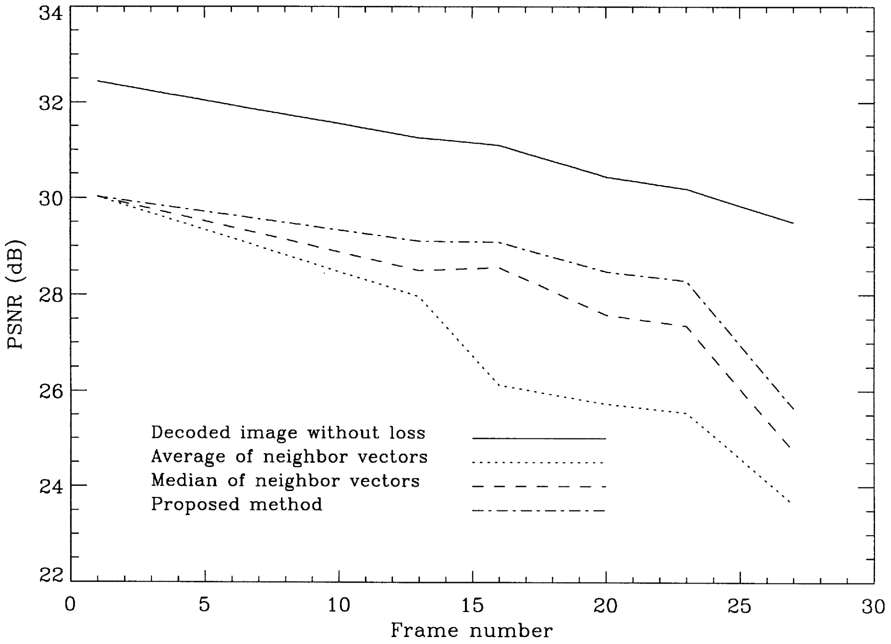

Fig. 30. PSNR comparison (foreman sequence, intra frame recovery: proposed method in Section 3).

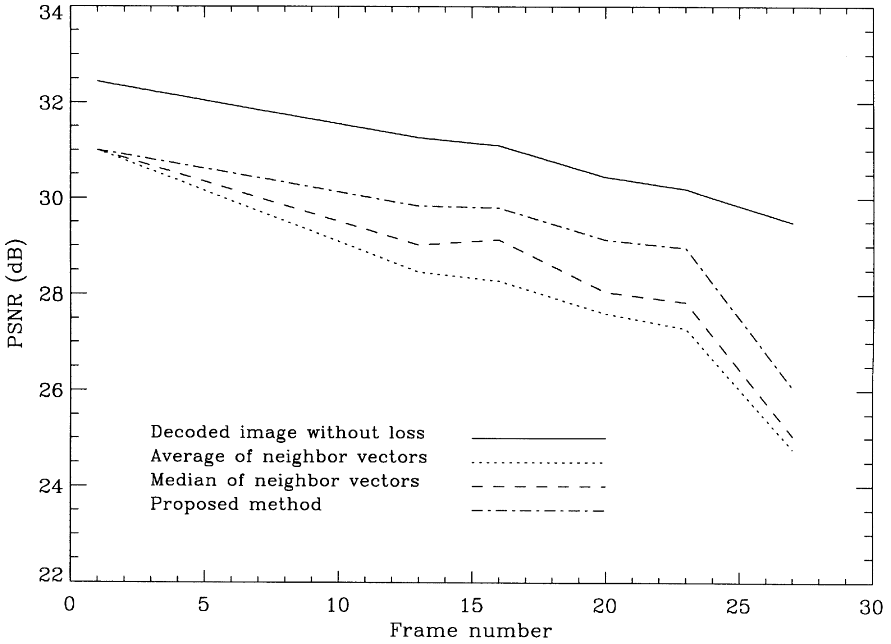

Fig. 31. PSNR comparison (foreman sequence, intra frame recovery: proposed method in Section 4).

14th decoded and the motion vector loss images, respectively. The motion vectors are lost in the moving areas of the frame. The upper right missing area has relatively small motion and the motion vectors of the neighboring macro blocks are similar to the displacement of the lost areas. The restored results of the average approach fail to estimate the correct motion vector, and there exists a slight error propagation due to the motion vector errors. The reconstructed image by the average approach is shown in Fig. 25. On the other hand, the images recovered by the median and the proposed algorithms shown in Figs. 26 and 27 recover the missing information successfully. In addition, the successful estimation of the lost motion vectors can avoid the error propagation caused by the motion vector errors. The performance of the proposed algorithm is relatively far superior at half of the frame. As shown in Figs. 25–27, the two other approaches result in edge discontinuities, while the proposed algorithm leads to satisfactory results.

Figs. 28–31 show the PSNR comparison for the two sequences, when the missing blocks for intra frames is restored by the iterative regularization approach in Section 3 and the recursive interpolation approach in Section 4, when 5% cell loss is generated. From these results, it is clear that the proposed motion vector estimation algorithm outperforms consistently the other approaches it is compared against.

7. Conclusions

This paper introduces an error concealment algorithm using a two layer ATM codec. In order to packetize and transmit the DCT coefficients and the motion vectors, fixed length cells are utilized. Two approaches for recovering the errors in intra frames are proposed. An iterative regularization approach which employs a high pass operator to incorporate the smoothness into the solution. A recursive interpolation approach is also introduced, which estimates the edge direction from the neighbors of the missing area, and establishes the weights of the filter depending on the estimated edge direction. The performance of the two approaches depends on the correlation between the missing block and its neighbors. For recovery of motion vectors in inter frames, an overlapped motion estimator is proposed. In order to keep the continuity between the missing macro block and its neighbors, the neighboring pixels are utilized to determine the lost motion vector. The proposed approaches are tested experimentally, leading to very satisfactory results.

References

[1] G.L. Anderson, A.N. Netravali, Image restoration based on a subjective criterion, IEEE Trans. System Man Cybernet. SMC-6 (December 1976) 845–853.

[2] S.L. Blake, S.A. Rajala, F. Gong, T.L. Mitchell, Efficient techniques for two-layer coding of video sequences, in: IEEE Proc. Internat. Conf. on Image Processing, Vol. 3, November 1994, pp. 253–257.

[3] CCITT Draft Revision of Recommendation H.261, Video codec for audiovisual services at $p \times 64$ kbits/s, June 1989.

[4] M.J. Chen, L.-G. Chen, R.-M. Weng, Error concealment of lost motion vectors with overlapped motion compensation, IEEE Trans. Circuits Systems Video Technol. 7 (3) (June 1997) 560–564.

[5] H. Feichtinger, K. Grochenig, Wavelets: Mathematics and Application, CRC Press, Boca Raton, FL, 1994.

[6] M. Ghanbari, V. Seferidis, Cell-loss concealment in atm video codecs, IEEE Trans. Circuits Systems Video Technol. 3 (3) (June 1993) 238–247.

[7] P. Haskell, D. Messerschmitt, Resynchronization of motion compensated video affected by atm cell loss, in: IEEE Proc. Internat. Conf. on Acoust. Speech and Signal Processing, March 1992, pp. 545–548.

[8] M.-C. Hong, A.K. Katsaggelos, An iterative regularized error concealment algorithm, SPIE Proc. 2952 (October 1996) 82–91.

[9] B.R. Hunt, The application of constraint least squares estimation to image restoration by digital computer, IEEE Trans. Computers 22 (9) (September 1973) 805–812.

[10] ITU-T, Draft ITU-T Recommendation H.263, December 1995.

[11] Joint Photographic Expert Group, JPEG Technical Specification, Revision 8, August 1990.

[12] M.G. Kang, A.K. Katsaggelos, General choice of the regularization functional in regularized image restoration, IEEE Trans. Image Processing 4 (5) (May 1995) 594–602.

[13] A.K. Katsaggelos, Iterative image restoration algorithms, Opt. Eng. 28 (7) (July 1989) 735–748.

[14] A.K. Katsaggelos, J. Biemond, R.W. Schafer, R.M. Mersereau, A regularized iterative image restoration algorithm, IEEE Trans. Signal Process. 39 (4) (April 1991) 914–929.

[15] A.K. Katsaggelos, F. Ishtiaq, L.P. Kondi, M.-C. Hong, M. Banham, J. Brailean, Error resilience and concealment in video coding, in: Proc. European Signal Processing Conf., Rhodes, Greece, 1998, pp. 221–228.

[16] W. Lam, A.R. Reibman, An error concealment algorithm for images subject to channel errors, IEEE Trans. Image Processing 4 (5) (May 1995) 535–542.

[17] W. Lam, A.R. Reibman, B. Liu, Recovery of lost or erroneously received motion vectors, in: IEEE Proc. Internat. Conf. on Acoustics Speech and Signal Processing, Vol. 5, 1993, pp. 417–420.

[18] Motion Picture Experts Group, MPEG video simulation model 3 (SM3), July 1990.

[19] P. Salama, N. Shroff, E.J. Delp, A bayesian approach to error concealment in encoded video streams, in: IEEE Proc. Internat. Conf. on Image Processing, 1996, pp. 49–52.

[20] P. Salama, N.B. Shroff, E.J. Coyle, E. Delp, Error concealment techniques for encoded video streams, in: IEEE Proc. Internat. Conf. on Image Processing, 1995, pp. 9–12.

[21] T. Strohmer, Computationally attractive reconstruction of band limited images from irregular samples, IEEE Trans. Image Processing 6 (4) (April 1997) 540–548.

[22] H. Sun, W. Kwok, Concealment of damaged block transform coded images using projection onto convex sets, IEEE Trans. Image Processing 4 (4) (April 1995) 470–477.

[23] A.S. Tom, C. Lung, F. Chu, Packet video for cell-loss protection using deinterleaving and scrambling, in: IEEE Proc. Internat. Conf. on Acoustics Speech and Signal Processing, May 1991, pp. 2857–2860.

[24] W. Verbiest, L. Pinnoo, B. Voeten, The impact of the atm concept on video coding, IEEE J. Selected Areas Commun. 6 (9) (December 1988) 1623–1632.

[25] M. Wada, Selective recovery of video packet loss using error concealment, IEEE J. Selective Areas Commun. 7 (5) (June 1988) 807–814.

[26] Q. Zhu, Y. Wang, L. Shaw, Coding and cell-loss recovery in dct-based packet video, IEEE Trans. Circuits Systems Video Technol. 3 (3) (June 1993) 248–258.