Optimization of Packetization Masks for Image Coding Based on an Objective Cost Function for Desired Packet Spreading

Joost Rombaut, Aleksandra Pižurica, *Member, IEEE,* and Wilfried Philips, *Member, IEEE,*

***Abstract*—In image communication over lossy packet networks (e.g., cell phone communication) packet loss errors lead to damaged images. Damaged images can be repaired with passive error concealment methods, which use neighboring coefficient or pixel values to estimate the missing ones. Neighboring image data should thus be spread over different packets. This paper presents a novel robust packetization method for the transmission of image content in lossy packet networks. We first define novel criteria for a good packetization. Based on these properties we propose a cost function for packetization masks. We then use stochastic optimization to calculate optimal packetization masks. We test our packetization technique on both wavelet coding and DCT coding. Compared to other packetization techniques we are able to achieve the same or better mean quality of the reconstructed images but with less fluctuation in quality, which is important for the viewer experience. In this way, we significantly increase the worst case quality, especially for high packet loss rates. This leads to visually more pleasing images in case of a passive reconstruction.**

***Index Terms*—Dispersive packetization, set partitioning, error concealment, robust image coding.**

I. Introduction

THIS paper deals with the transmission of image data over low-bandwidth, error-prone channels, such as wireless cell phone communication. In these networks, bit-errors can be introduced or even worse, complete blocks of information (data packets) can get lost. This leads to quality degradations. When the error rate is too high, the receiver may not even be able to decode the correctly received information because of synchronization problems and other errors. Error protection mechanisms include adding redundant information, such as in error control coding or forward error correction [1]. In some cases, the receiver can request the retransmission of a lost or erroneous packet from the sender. This technique is known as *active* error concealment. A good overview of different techniques is given in [1]. Active error concealment is impractical in time-critical applications such as real-time communication. In applications such as broadcasting there is no return channel and thus retransmission cannot be applied either.

J. Rombaut, A. Pižurica and W. Philips are with the Department for Telecommunications and Information Processing (UGent-TELIN-IPI-IBBT), Ghent University, Sint-Pietersnieuwstraat 41, 9000 Ghent, Belgium. E-mail: Joost.Rombaut@telin.ugent.be, Tel: +32-9-264-95-30, Fax: +32-9-264-42-95, Web: http://telin.ugent.be/∼jorombau/.

Corresponding author: J. Rombaut.

A. Pižurica is a postdoctoral research fellow of the FWO, Flanders, Belgium.

Passive or post-processing error concealment exploits the property that pixel values are highly correlated in space and time [2], and that lost information can therefore be estimated by using neighboring information. Although compression techniques remove as much redundancy as possible, some redundancy is still present in the compressed data. This redundancy can be used to recover lost information. Examples of passive recovery techniques include [1], [3], [4] where the problem is the reconstruction of damaged images and video after the loss of some Discrete Cosine Transformation (DCT) blocks. Reconstruction of wavelet coefficients of wavelet coded images and intraframe coded video is treated in [5]–[8], and passive recovery of lost motion vectors is addressed, e.g., in [9].

All these techniques interpolate lost elements (such as pixels, wavelet coefficients, ...) from their correctly received spatio-temporal neighbors. An implicit assumption is that the probability of a simultaneous loss of two or more neighboring elements is low. This is accomplished by the packetization, i.e., by spreading neighboring elements as much as possible over many packets. For passive error concealment with a packetization scheme, a packet is received either completely without any error, or it is considered as lost. So, small errors like bit errors (bit switching, bit loss or bit insertion) in a packet should be detected and repaired at the network level. If the bit errors cannot be repaired at the network level, the damaged packet should be flagged as lost, and it should be repaired by passive reconstruction.

Different approaches to packetization are presented in the literature. A simple technique is the splitting of a frame into slices, as depicted in Fig. 1 (a). *Slicing* does not help for the reconstruction of lost elements in the middle of a slice: if a slice gets lost, interpolation of the inner elements is almost impossible. However, slicing prevents errors from propagating all over the image as the slices are coded independent from each other. This technique is used in H.263+ [10], MPEG1, MPEG4, ... The main disadvantage is that if a packet (i.e., a slice) gets lost, the neighbors of each lost element (pixel, motion vector, wavelet coefficient, ...) are also lost. Reconstruction by using neighboring elements is therefore impossible. Better packetization techniques were introduced in [9], [11]. In [9], parity based slicing is proposed, where the elements of each slice are distributed such that four neighboring elements are stored in four different packets. Therefore if only one packet is lost, all four neighbors are available for reconstruction. This technique is illustrated in Fig. 1 (b). This parity based slicing mechanism allows better reconstruction of lost elements at the

cost of a small decrease in compression efficiency as shown in [9]. Even better spreading of the elements among the different packets is achieved with the technique of [12]. This technique is based on an algorithm for partitioning the $\mathbb{Z}^2$ Lattice [11], which maximizes the minimum spatial distance between two elements of the same packet (i.e. it maximizes the minimum *intra-partition* distance (i.p.d.)). An example is shown in Fig. 1 (c).

In this paper we propose a new packetization method. We explain our method using the concept of packetization masks. A packetization mask (examples are given in Fig. 1) is a lattice of which the elements are indices identifying the packets in which the corresponding pixels of an image, or the coefficients of a subband, are stored. In essence, our approach is based on the maximization of the intra-partition distance as in [11], but — in contrast to [11] — we simultaneously aim at spreading the neighbors of the elements of a packet *equally* over all other packets. Our reasoning is that each packet should ideally carry the same amount of information pertaining to the interpolation of the elements of every other packet. In this way, the number of available neighbors is always the same, *even in the case of multiple packet loss, and so the worst case image quality is maximized*. The maximization of the number of available neighbors is essential for all passive error concealment techniques. The more available neighbors, the better the reconstruction quality of a lost coefficient. Not only the reconstruction quality but also the *computation time* depends on the number of available neighbors. For example, the concealment method of [8] converges faster to its optimal reconstruction value when more neighbors are available. A dispersive packetization technique should therefore aim to maximize the number of available neighbors of a lost coefficient.

The paper is organized as follows. In Section II, we briefly introduce the problem of passive error concealment for which packetization is essential. We describe the reconstruction approach that we will apply to wavelet coded images in Section II-A and the reconstruction of images with lost DCT-blocks in Section II-B. In Section III, we define the rules for good packetization, and we present a novel algorithm for the optimization of packetization masks based on stochastic sampling. Numerical comparison of our packetization with the existing representative techniques is given in Section IV. The results demonstrate that our packetization masks yield the highest quality of the reconstructed frames. *On average*, the new masks perform equally well as the masks of [12], but the *fluctuation* of the quality is minimized. We conclude this work in Section V.

II. Reconstruction algorithm

In this section, we illustrate passive error concealment of images in case of packet loss to point out the need for good packetization and we describe the passive error concealment methods that will be used in our experiments in Section IV. We discuss two examples: reconstruction of wavelet coefficients and the reconstruction of pixels after the loss of DCT-coefficients.

A. Reconstruction of wavelet coefficients

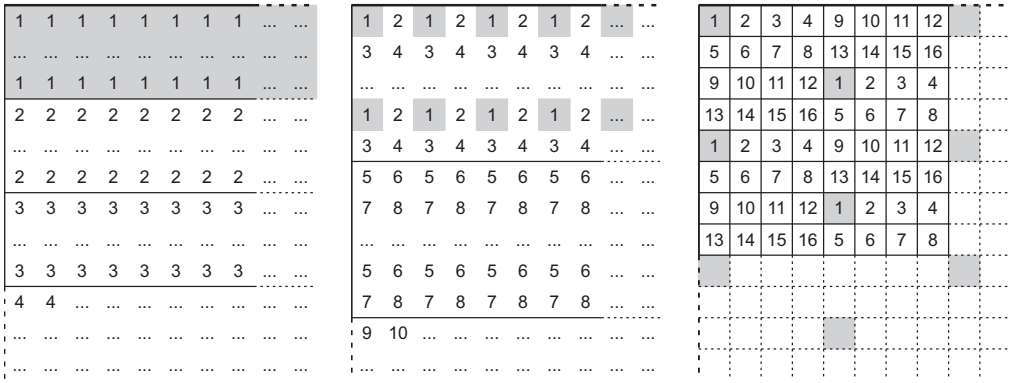
Image compression based on wavelet transformation [13]–[15] has some advantages over compression based on the Discrete Cosine Transformation (DCT). It often yields a higher compression factor for the same image quality, and wavelet compression is more scalable: even at high compression rates, there is less block distortion than with compression based on DCT. As in [7], we use the bi-orthogonal wavelet transform [16]–[18]. We use the following notation: LL_ℓ denotes the low pass subband (the scaling coefficients) at the decomposition level ℓ; the wavelet coefficients are organized into the subbands LH_i, HL_i and HH_i, which denote respectively horizontal, vertical and diagonal details at the decomposition level i. A detailed overview of wavelet decompositions is given in [16]–[19].

In image coding, LL_ℓ, and LH_i, HL_i, HH_i, $i = 1 \dots \ell$, are quantized and entropy coded to take advantage of similarities between subbands (e.g., zerotree coding [13], JPEG2000 [15]). If a wavelet coded image is sent over a network, the compressed coefficients are typically stored into different packets. For example, if an image with a resolution of 256×256 pixels is encoded at a bitrate of 1 bit per pixel, then its coefficients should be spread over at least 15 packets (of 546 bytes on average) to be suitable for Internet transmission without fragmentation, as the maximum transfer unit[1] where fragmentation is guaranteed to not occur, is a packet of 576 bytes. In Fig. 2, the packetization techniques of Fig. 1 are adapted for the packetization of (trees of) wavelet coefficients.

If a packet gets lost during transmission, in techniques without error reconstruction the missing coefficients are typically replaced by zeros and hence the damaged image has black holes due to the lost low frequency coefficients [5]–[8]. In this paper we will adopt a slightly different approach: all missing coefficients are replaced by zero, except for all missing LL_ℓ coefficients. These are replaced by the value which corresponds to a pixel value of 128. This very easy and straightforward technique should not be considered as an error concealment technique, but it is a fairer comparison than replacing all missing coefficients with zeros. An example of an image damaged by packet loss is shown in Fig. 3 (b).

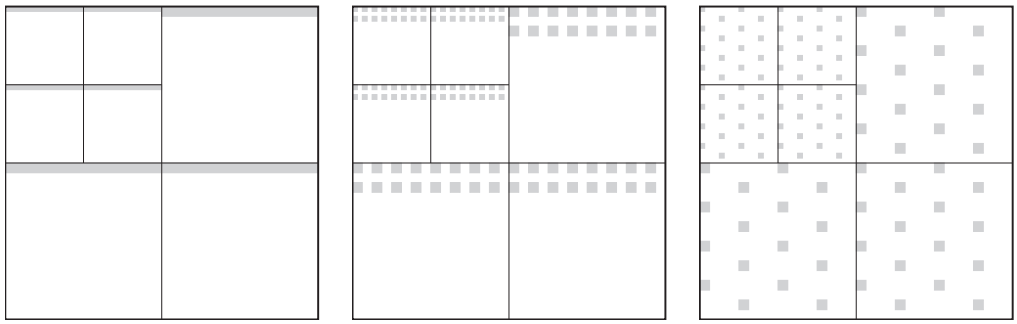
As an example, we illustrate the reconstruction approach of [7]. This approach estimates low frequency coefficients using the interpolation kernel from Fig. 3 (c). This interpolation kernel assigns the biggest weights to the closest neighbors as they are likely to be most similar in value to the missing coefficient. For high loss-rates a two pass interpolation can be used: lost coefficients without correctly received neighbors are repaired in the second pass, after their neighbors have been repaired in the first pass. The loss of the high frequency coefficients has less impact on the visual quality than the loss of low frequency coefficients, except near edges, where the coefficients are typically large. For the LH_i and HL_i subbands a simple one-dimensional linear interpolation gives

[1]Each host in the Internet must be prepared to accept datagrams of up to 576 bytes [20]. Therefore, packets bigger than 576 bytes may be fragmented if there is a router along the transmission path which cannot cope with these bigger packets. If any of these fragments is lost, the whole packet will be lost, as the other fragments are useless on their own.



(a) (b) (c)

Figure 1. Packetization techniques in the literature. Each technique is depicted by a lattice which shows the numbers of the packets in which the elements are stored: elements with the same number belong to the same packet. *One element* can be a tree of wavelet coefficients, a block of DCT-coefficients, a motion vector, ... as detailed in Section II. **(a)** *Slicing* [10]: a lattice is divided into slices. The elements of the same slice are stored in the same packet. **(b)** *Parity based slicing* [9]: the elements of the same slice are divided into four subgroups. **(c)** *Maximization of the minimum intra-partition distance* [11]: elements in the same packet are physically located as far apart from each other as possible.



(a) (b) (c)

Figure 2. Packetization techniques of Fig. 1 applied to wavelet coding. For each technique the coefficients of one packet are marked in gray. Each packet contains a number of families of wavelet coefficients. **(a)** *Slicing*. **(b)** *Parity based slicing*. **(c)** *Maximization of the minimum intra-partition distance*.

(a) (b) (c) (d)

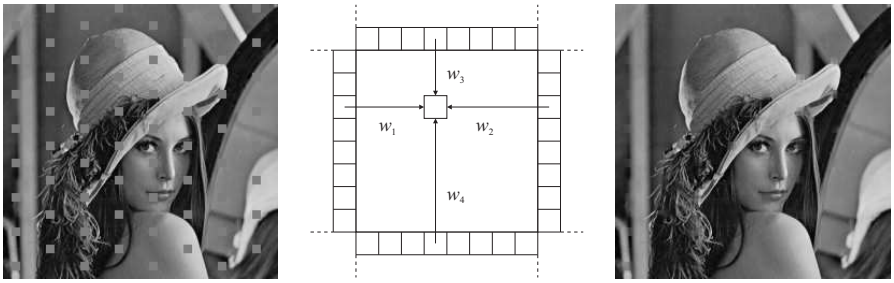
Figure 3. **(a)** The original *Lena*-image. **(b)** Lena after loss of 1 out of 16 packets (6.25% of the coefficients) with the packetization technique of [12] (PSNR=25.51dB). **(c)** The interpolation kernel for the interpolation of LL_ℓ coefficients according to the reconstruction technique of [7]. **(d)** Reconstruction of the lost coefficients of (b) by using the technique of [7] (PSNR=31.14dB).

satisfactory results [5]–[7]. In the HH_i-subbands, the loss of coefficients is rarely noticeable, and thus coefficients in these subbands are usually not repaired, but are set to zero. The reconstruction result of Fig. 3 (b) is shown in Fig. 3 (d).

B. Reconstruction of DCT-coefficients

In DCT compression (e.g., JPEG [21]), the image is first divided into 8×8 pixel blocks. To each of these blocks, the DCT transform is applied and the resulting coefficients are compressed. If the image is sent over a network, several 8×8 blocks are grouped and sent as one packet. If a packet gets lost, the damaged image will have "holes" of 8×8 pixels. Typically, lost pixels are set to zero, which results in "black holes". As mentioned before, we set lost pixels to the value 128, which results in "gray holes" which are less noticeable than "black holes". An example is shown in Fig. 4 (a). In contrast to wavelet reconstruction, error concealment techniques for DCT-compressed data operate in the pixel domain rather than the transform domain. In this case, image pixels from neighboring blocks are used to reconstruct the pixels in lost blocks.

As an example, we illustrate the reconstruction method of [3]. In this approach, each pixel from a lost block is



(a) (b) (c)

Figure 4. **(a)** Lena image after loss of 1 out of 16 packets (6.25% of the DCT-blocks) (PSNR=24.65dB). **(b)** Interpolation scheme for a pixel of a lost DCT block for the algorithm of [3]. **(c)** Reconstruction of (a) by using the technique of [3] (PSNR=35.65dB).

interpolated as depicted in Fig. 4 (b), by using the border pixels from the neighboring blocks. The interpolation weights $w_1, \ldots, w_4$ are inversely proportional to the distance between the lost pixel and the corresponding border pixel. Fig. 4 (c) illustrates a result of this method. Other techniques exist [4], [22], [23] where, for example, the mean value interpolation is replaced by a median value interpolation, or a two pass scheme is employed instead of a one pass scheme. All of these techniques have in common that a good interpolation depends on the correct reception of as many neighboring elements as possible. The subject of this paper is to design good packetization masks which ensure a high number of correctly received neighbors. In the next section, we discuss this topic in detail.

8	13	14	15	16	5	6	7	8	13
12	1	2	3	4	9	10	11	12	1
16	5	6	7	8	13	14	15	16	5
4	9	10	11	12	1	2	3	4	9
8	13	14	15	16	5	6	7	8	13
12	1	2	3	4	9	10	11	12	1
16	5	6	7	8	13	14	15	16	5
4	9	10	11	12	1	2	3	4	9
8	13	14	15	16	5	6	7	8	13
12	1	2	3	4	9	10	11	12	1

Figure 5. 8×8 mask (+ extra neighbors) calculated with the algorithm of [11] for $P = 16$. Periodicity vectors are drawn on the mask.

III. Packetization

In this section, we first discuss some known requirements for good packetization and then we define new desired properties that improve the quality of packetization. We define a new cost function based on these desired properties, and we propose a novel optimization method for minimizing the proposed cost function. As an introduction, we describe a state of the art packetization algorithm [11], which is a starting point in our development.

A. Preliminaries

We adopt the notation of [11], [12]. A signal f (in our case an image) is defined on a discrete domain D. A partition $\mathscr{P}$ of D is a set $\{S_0, S_1, \ldots, S_{P-1}\}$ of P non-empty, non-overlapping subsets S_i of D, which cover D. Note that by definition $\bigcup_{i=0}^{P-1} S_i = D$, and that $S_i \cap S_j = \emptyset, i \neq j$. In the following, the set S_i corresponds to the data of the ith packet

The algorithm of [11] optimizes the packetization, i.e., the choice of $\mathscr{P}$, by maximizing the *overall* minimum intra-partition distance (i.p.d.). The minimum intra-partition distance of packet S_i, is defined as:

$$d_{\min}(S_i) = \min_{\substack{s,t \in S_i \\ s \neq t}} d(s,t)$$

where $s = (x_s, y_s)$ and $t = (x_t, y_t)$ are elements of the packet S_i and $d(s,t)$ is the (Euclidean) distance between s and t: $d(s,t) = \sqrt{(x_s - x_t)^2 + (y_s - y_t)^2}$. The *overall* minimum i.p.d. is

$$d_{\min}(\mathscr{P}) = \min_{\substack{i=0,1,\ldots,P-1 \\ S_i \in \mathscr{P}}} d_{\min}(S_i).$$

The optimal partition $\mathscr{P}^*$ in the minimax sense is the partition which maximizes the minimal i.p.d.:

$$\mathscr{P}^* = \arg\max_{\mathscr{P} \in \mathbb{P}} d_{\min}(\mathscr{P})$$

where $\mathbb{P}$ is the set of all possible partitions.

For computational reasons, the 2D minimax optimization algorithm of [11] treats the partitioning of the $\mathbb{Z}^2$ lattice as a sphere packing problem imposing a periodicity constraint on the solution. An example of lattice partitioning produced by the algorithm of [11] for 16 packets ($P = 16$) is given in Fig. 5. It has $d_{\min}(S_i) = 4$. This example is also shown in Fig. 1 (c), and its application for wavelets in Fig. 2 (c).

A limitation of the algorithm of [11] is its restriction to periodical solutions. This periodicity constraint may hinder good concealment in case *two (or more)* packets get lost where each element of the first packet lies in the vicinity of an element of the second packet. This can be understood from the example in Fig. 5: all elements of packet 1 have neighbors coming from the same packets: 2, 5, 12 and 13. If one or more of these "neighboring" packets are also lost, the concealment becomes difficult for *all* elements of packet 1. We wish to overcome this problem by removing the periodicity constraint and by *equally spreading the information needed for repairing all the elements of a packet over all other packets*. This enables

		δ_3		
	δ_2	δ_1	δ_2	
δ_3	δ_1	y	δ_1	δ_3
	δ_2	δ_1	δ_2	
		δ_3		

Figure 6. *Set of Elements needed for Reconstruction* (SER) of an LL_ℓ-coefficient. The nth order elements are marked as $\delta_n (n = 1, 2, 3)$.

better error concealment because more neighbors are available for the reconstruction of lost elements.

In the remainder of this section we focus on the packetization of wavelet coefficients (of which the reconstruction is treated in II-A), but the theory presented here is widely applicable, as we show in the results section.

B. Desired properties for good packetization

Let $\delta(y)$ denote the *Set of Elements needed for Reconstruction* (SER) of an element y. This SER contains all the elements which are used to repair y if it is lost, according to a given restoration strategy. The SER of an LL_ℓ-coefficient, according to the restoration strategy of Section II-A, is given in Fig. 3 (c). It is important to notice that the closer the *Elements needed for Reconstruction* (ERs) are to y, the more important they are for the reconstruction of the lost center element y, as closer ERs are typically assigned bigger interpolation weights (Fig. 3 (c)). We distinguish first, second and third order ERs, at a distance of 1, $\sqrt{2}$ and 2 respectively. We denote the nth order SER as δ_n (Fig. 6) and the number of elements in δ_n by $\#\delta_n$.

We formulate the desired properties for good packetization as follows:

- *Large enough intra-packet distance (i.p.d.).* This property has already been used by other authors [11], [12], where it is called the *intra-partition distance requirement*. The minimal distance between two elements of the same packet, i.e., the intra-packet distance, should be large enough (i.e., not in each other's SER) so that lost elements interfere as little as possible with each other's reconstruction.
- *Maximal spreading of an SER over packets.* No two ERs of a given element y should be in the same packet if possible. In this way, we minimize the number of lost ERs for each element. If this property is not fulfilled, the loss of the packet containing these ERs makes the recovery of y very difficult. This situation is not so bad if these ERs are of high order (i.e., more distant).
- *Maximal spreading of the SERs of all elements of a packet over all other packets. All* ERs of *all* elements of a packet should be equally spread over all other packets. This is equivalent to requiring that the number of elements of a packet S_i in the SERs of the elements of another packet S_j is as close to constant as possible. In this way, in case of the loss of 2 or more packets, the number of coefficients with many lost ERs is minimized. For example, this spreading property is *not* fulfilled in Fig. 5: if packets 1, 2 and 5 are lost, every coefficient of packet 1 misses two first order ERs which may hinder good reconstruction. Note that our second desired property is fulfilled here.
  As a low order ER is more important than a high order ER, we make this third desired property even stronger by defining this spreading property for each nth order SER: *all* nth order ERs of *all* elements of a packet should be equally spread over all other packets. Thus, all packets S_i $(i \neq j)$ should contain (as closely as possible) the same number of elements that belong to the nth-order SERs of the elements of packet S_j. We define this number of elements $N_{n,j}$ as

$$N_{n,j} = \frac{\#S_j \#\delta_n}{P-1} \tag{1}$$

  where $\#S_j$ is the number of elements in the packet S_j (which we assume the same for all packets), $\#\delta_n$ is the number of elements in the nth order SER, and P is the total number of packets.

The first desired property (large enough intra-packet distance) is relevant for the loss of one or more packets, as this property describes the desired relationship between the elements of a packet. The second and third desired properties are relevant for the loss of two or more packets, as these properties describe the desired relationship between the elements of two packets. The packetization technique of [12] is based on the first property, but it does not take into account the other desired properties defined here.

C. The proposed cost function for a packetization mask

Now we define a cost function that measures how well a packetization complies with the previously defined desired packetization properties. Our cost function for the partition $\mathscr{P}$ is a weighted combination of three terms:

$$Q(\mathscr{P}) = \alpha Q_1(\mathscr{P}) + \beta Q_2(\mathscr{P}) + \gamma Q_3(\mathscr{P}), \tag{2}$$

where the penalties $Q_i(\mathscr{P})$, $i = 1, 2, 3$, correspond to the previously defined desired properties and the weighting factors α, β and γ reflect the importance attached to the different desired properties. This importance is based on a probabilistic model for the packet loss.

$Q_1(\mathscr{P})$ is the average penalty of all packets according to the first desired property. If there are x packets lost, the penalty $Q_1(\mathscr{P})$ should be counted x times. Of course, since we don't know how many packets will be lost for each image, we estimate the number of lost packets based on a probabilistic model for the packet loss. The weight α should be equal to the expected number of lost packets:

$$\alpha = \sum_{i=0}^{P} i \ \texttt{Prob}(N_{lost} = i), \tag{3}$$

where N_{lost} is the number of lost packets for an image and thus $\texttt{Prob}(N_{lost} = i)$ is the probability that i packets are lost. From this equation, it is clear that if the expected number of lost packets is equal to zero (i.e., no packet loss), the

packetization doesn't make a difference for the first desired property. Indeed, if all packets are correctly received, there is no reconstruction necessary, and it doesn't matter in which packets the coefficients are stored.

$Q_2(\mathscr{P})$ is the average penalty of all couples of packets according to the second desired property. If there are x packets lost, the penalty $Q_2(\mathscr{P})$ should be counted for all the couples of lost packets, i.e., $x(x-1)$ times[2]. The weight β should be equal to the expected number of ordered couples of lost packets:

$$\beta = \sum_{i=0}^{P} i(i-1)\ \texttt{Prob}(N_{lost} = i) \tag{4}$$

From this equation, it is clear that if there is never more than one packet lost for each encoded image, the packetization doesn't make a difference for the second desired property (there is never a loss of a couple of packets).

$Q_3(\mathscr{P})$ is the average penalty of all couples of packets according to the third desired property. As for the penalty $Q_2(\mathscr{P})$, the penalty $Q_3(\mathscr{P})$ should be counted for all the couples of lost packets. The weight γ is equal to the weight β.

$$\gamma = \sum_{i=0}^{P} i(i-1)\ \texttt{Prob}(N_{lost} = i) \tag{5}$$

Our first penalty $Q_1(\mathscr{P})$ penalizes small intra-packet distances according to the first desired property. We use the following notation: $s_{i,k}^{\mathscr{P}}$ is the position (i.e., the Euclidean coordinates) of the k^{th} element of the i^{th} packet (i.e., S_i) of partition $\mathscr{P}$. $d(s,t)$ is defined as the modulo distance between the elements s and t because in this way, we take into account that masks will be periodically extended: $d(s,t) = \sqrt{((x_s - x_t) \mod M)^2 + ((y_s - y_t) \mod N)^2}$, where M and N are the number of rows and columns respectively[3], and "mod" is the modulo operator. So, the distance between s and t is the smallest distance between s and t after periodic extension of the mask. The average penalty per packet for not satisfying the first property is:

$$Q_1(\mathscr{P}) = \frac{1}{P} \sum_{i=0}^{P-1} \sum_{k=0}^{\#S_i^{\mathscr{P}}-1} \sum_{\substack{l=0 \\ l \neq k}}^{\#S_i^{\mathscr{P}}-1} C(d(s_{i,k}^{\mathscr{P}}, s_{i,l}^{\mathscr{P}})) \tag{6}$$

where $C(d)$ is the cost attached to two elements separated by a distance d being in the same packet, as defined in Table I. For these costs we use the absolute values of the interpolation weights from the interpolation mask in Fig. 3 (c) as they are a measure of the importance of an ER at a distance d.

According to the second property (*maximal spreading of an SER over packets*) no two ERs of the same SER should be in the same packet in the ideal case. Therefore, if two or more ERs from the same SER are stored in one packet, we assign a cost to that and we group these costs in Q_2. This cost is proportional to the number of ERs and their order: lower order ERs receive a higher cost as these lower order ERs are more important. In particular, we define the average penalty for not satisfying the second property per couple of packets as:

$$Q_2(\mathscr{P}) = \frac{1}{P(P-1)} \sum_{i=0}^{P-1} \sum_{\substack{j=0 \\ i \neq j}}^{P-1} \sum_{k=0}^{\#S_i^{\mathscr{P}}-1} \left[\sum_{l=0}^{\#S_j^{\mathscr{P}}-1} C\left(d\left(s_{i,k}^{\mathscr{P}}, s_{j,l}^{\mathscr{P}}\right)\right) \cdot \mathbb{1}\left(\#\right. \right. \tag{7}$$

In the expression (7) $\mathbb{1}(expr)$ evaluates to 1 if $expr$ is true, otherwise it evaluates to 0. As defined before, $C(d)$ is the cost attached to two elements at a distance d away from each other (Table I). d_N is the distance at which the highest order SER is located; in our case $N = 3$, $d_N = 2$.

The right hand side of equation (7) sums products of two factors. The second factor is a condition which determines whether or not a cost is counted: if two or more elements from the same packet S_j appear in the SER of an element from another packet S_i, then the second factor evaluates to one and a cost (defined by the first factor) is added to $Q_2(\mathscr{P})$. Otherwise (i.e., if only one or no elements from the packet S_j appear in the SER of an element from another packet S_i), the second factor evaluates to zero and no cost is added. The first factor is the actual cost, proportional to the distances between the element of S_i and its ERs stored in S_j, as defined in Table I.

Table I
COST FUNCTION $C(d)$ FOR THE nTH ORDER SER, AT A DISTANCE d_n, IN CASE OF THE CONCEALMENT TECHNIQUE OF SECTION II-A. THE HIGHEST ORDER SER HAS $n = 3$.

n	d_n	$C(d_n)$
1	1	32
2	$\sqrt{2}$	10
3	2	4
>3	> 2	0

Our third penalty component expresses the third desired property (*maximal spreading of the SERs of all elements of a packet over the other packets*) from Section III-B. We count how many elements of each packet lie in the nth order SER of the elements of other packets, and we compare this number to the desired number of elements as given by (1). The absolute difference between the actual value and the desired value is then multiplied with the weight factor $C(d_n)$ of this nth order SER, as given in Table I. For the cost factor $C(d)$, we have again chosen the absolute values of interpolation weights as their magnitude is an indication for the importance of the nth order SER in the interpolation. The average penalty for not satisfying the third property per couple of packets is:

$$Q_3(\mathscr{P}) = \frac{1}{P(P-1)} \sum_{i=0}^{P-1} \sum_{\substack{j=0 \\ i \neq j}}^{P-1} \sum_{n=1}^{N} \left(C(d_n) \left| \sum_{k=0}^{\#S_i^{\mathscr{P}}-1} \sum_{l=0}^{\#S_j^{\mathscr{P}}-1} \mathbb{1}\left(d\left(s_{i,k}^{\mathscr{P}}, s\right.\right.\right. \right. \tag{8}$$

The right hand side of this equation sums the product of two factors. The second factor calculates how much the number of elements from each packet in the n^{th} order SER of the

[2]Note that we count ordered couples, as we want to measure the influence from packet S_i on packet S_j and vice versa.

[3]Note that M and N (which define the periodicity) should be chosen large enough such that (1) could still be fulfilled as good as possible. This is explained in Section III-D.

elements of another packet differs from the desired number $N_{n,j}$ as given by (1). The first factor is the actual cost $C(d_n)$ for the n^{th} order SER, as given in Table I.

D. The proposed algorithm for minimizing the cost function

Finding the optimal packetization for any given mask size is not trivial due to the large number of candidate solutions. For example, an $M \times N$ mask of which the elements have to be divided into P packets has $\frac{(MN)!}{((MN/P)!)^P(P!)}$ candidate partitions. For ease of calculation we have assumed that MN is a multiple of P, and that we have equally sized packets: each packet consists of MN/P elements. An exhaustive search is impossible. For example, a 256×256 lattice of which the elements have to be divided into 16 packets has 2.57e78867 candidate solutions.

A possible (suboptimal) simplification is to search for the optimal solution for a submask. This is a smaller mask with fewer elements, and hence with a reduced computation time. By periodically extending this solution, we obtain the desired mask size. Note that we must choose a mask size which is big enough so that cost $Q_3(\mathscr{P})$ can be as small as possible. In the case of Fig. 5 the size of the submask is 4×4 which is too small: the first order SER δ_1 of all elements of each packet are spread over 4 packets only (e.g., δ_1 of packet 1 is spread over packets 2, 5, 12 and 13). Any combined loss of two or more of these packets will affect the SER of *all* elements of these packets. If the SER of packet 1 was spread over more (ideally all) packets, the number of elements with one or more lost ERs would be minimized, which would lead to a higher PSNR and visual quality.

According to (1) each packet in the submask should contain $N_{1,0} = 4 \cdot 1/(16-1) = 0.2667$ elements of δ_1 of packet 1, which is of course infeasible: 4 packets contain 1 element of δ_1 of packet 1, and the other 11 packets none. By increasing the size of the submask to, e.g., 8×8 the third property can be fulfilled much better. In this case, according to (1) each packet in the 8×8 submask should contain $N_{1,0} = 4 \cdot 4/(16-1) = 1.0667$ elements of δ_1, which is still infeasible, but now fourteen packets can contain one element and the fifteenth packet contains 2 elements. Due to the better spreading (i.e., a closer approximation of $N_{1,0}$), this results in a smaller cost $Q_3(\mathscr{P})$. So the larger the size of the submask, the bigger the period, so the better the third property can be fulfilled.

By using submasks the number of possible candidates is drastically decreased, but it is still too large to allow an exhaustive search. For example, the optimal solution for an 8×8 sublattice for 16 packets should be chosen from 5.01e53 candidate solutions. Therefore we use stochastic optimization to find the optimal sublattice with high probability. In particular, we use Metropolis sampling [2], [24] within a simulated annealing framework. The Metropolis algorithm, modified for use in our packetization problem, is outlined in Fig. 7. We start with a randomly generated mask. A candidate mask (`new_mask`) is generated by switching two labels from the previous configuration. If `new_mask` lowers the penalty, it is accepted and the new mask becomes the best mask. Otherwise the candidate mask is accepted with a certain probability which

```
mask:=init_mask;
while(equilibrium_not_reached) {
  new_mask:=generate_new_mask(mask);
  delta_penalty:=penalty(new_mask)-penalty(mask);
  threshold:=exp(-delta_penalty/Tk);
  if(rand(1)<threshold) {
    mask:=new_mask;
    if(threshold>1) {
      best_mask:=new_mask;
    }
  }
}
return [mask, best_mask];
```

Figure 7. Metropolis sampling at temperature $\mathtt{T}_k$.

```
Tk:=init_T;
mask:=init_mask;
while(equilibrium_not_reached) {
  [mask,best_mask]:=Metropolis(mask,best_mask,Tk)
  Tk:=decrease(Tk);
}
return best_mask;
```

Figure 8. The simulated annealing algorithm.

depends on the sampling temperature $\mathtt{T}_k$, which is defined by the simulated annealing framework [2], [25], [26] (Fig. 8). The stop criterion `equilibrium_not_reached` is defined as follows: Metropolis should stop after $10 \times M \times N$ accepted masks, where $M \times N$ is the size of the submask. If this criterion is not reached after $100 \times M \times N$ tries, the algorithm should also stop.

A stochastic optimization algorithm reaches the global optimum with a high probability. A disadvantage is the large number of required iterations. This makes it impossible to generate masks on the fly in a client. However, this is not a problem as masks can be calculated in advance and stored in each client's memory. Important parameters are the mask size, the number of packets, and the distribution of the packet loss. In practice masks can be generated for a selected range of values of these parameters. As the probabilistic model for the packet loss, we propose the binomial distribution [27], [28] with an either low (about 5%) or high (between 10% and 25%) average packet loss rate. We propose the binomial distribution as this distribution is the most general: it models the loss of packets as independent and identically distributed (iid) events, and is therefore the easiest to use in the optimization of the packetization mask. This general iid binomial model is a good assumption if the real packet loss model is unknown. After generation, masks should be stored in the memory of all clients for subsequent use. An $M \times N$ submask with P packets uses only $M \times N \times \lceil \log_2 P \rceil$ bits, e.g., an 8×8 submask with 16 packets uses 32 bytes. If a client does not have the necessary mask in its memory (e.g., because of memory restrictions) it can be communicated in advance without much overhead.

IV. Results

We compare our packetization method to the representative methods from the literature [9]–[11]. We tested both the reconstruction of lost wavelet coefficients and lost DCT-blocks, using the reconstruction methods from Section II-A and II-B respectively.

1	2	3	4	1	2	3	4
5	6	7	8	5	6	7	8
9	10	11	12	9	10	11	12
13	14	15	16	13	14	15	16
1	2	3	4	1	2	3	4
5	6	7	8	5	6	7	8
9	10	11	12	9	10	11	12
13	14	15	16	13	14	15	16

Figure 10. The initial mask used for the proposed stochastic sampling algorithm.

A. *Influence of packetization on the reconstruction of unquantized wavelet coefficients*

In this section, we compare the quality of error-concealed wavelet coded images for different packetization schemes. Each image has 256×256 pixels, organized in 16 packets. We consider the slicing technique as given in [10], parity based slicing [9], and the packetization based on the maximization of the minimum i.p.d. [11], all introduced in Section I. For the proposed technique we have calculated two masks by the algorithm given in Section III-D. In the stochastic sampling, we used, as the probabilistic model for the packet loss, the binomial distribution with an average packet loss rate of 12.5%, i.e., on average 2 out of 16 packets get lost. For communication in the Internet with the User Datagram Protocol (i.e., without packet retransmission techniques), packet loss rates are typically in the range of 2% to 10% [1]. However, due to error bursts, the instantaneous packet loss rate can be higher than the average loss rate. Therefore, in the optimization of our packetization masks, we have chosen a loss rate of 12.5%, as this gives good results for both low and high loss rates, as shown further on in this Section.

One generated mask consists of a periodically extended 8×8 submask (Fig. 9 (a)), and the other one consists of a periodically extended 16×16 submask (Fig. 9 (c)). The initial 8×8 mask used in the metropolis algorithm is given in Fig. 10. The initial 16×16 mask is a periodical extension of this initial 8×8 mask. We use these two submasks to show the influence of the size of the submask on the quality of the reconstructed images. The bigger the submask, the easier the desired properties can be fulfilled, so the better the quality of the reconstructed images.

In Table II, the penalties (as defined in Section III-C) for each packetization technique are given for both the binomial packet loss model (B(16,0.125)) used in our stochastic sampling mask optimization and for two other packet loss models. For all models, the best mask according to the proposed criterion is the 16×16 mask produced by our algorithm (Fig. 9 (c)). For the packet loss model B(16,0.125), this mask is indeed optimal. For the other models, we can generate even better masks with lower penalties. Visual results later in this paper demonstrate that the values of the penalties for the different masks are consistent with the quality of the reconstructed images using the corresponding masks.

We used three test images in our experiments: *Lena*, *Goldhill* and *Barbara*. The images are wavelet transformed, and the coefficients are packed with different packetization techniques. In this test, the wavelet coefficients are not quantized and entropy coded, so there is no loss of quality due to compression. Initially, we do not use quantization in order to be able to study packet loss artifacts separately from quantization artifacts. The results of a similar test *with* quantization are given in Section IV-B.

Table II
MASK PENALTIES FOR WAVELET CODED DATA, WHERE THE PACKET LOSS MODEL IS THE BINOMIAL DISTRIBUTION B(16,π) FOR LOSS PROBABILITY π OF 6.25%, 12.50%, AND 25.00%. OUR MASKS ARE THOSE FROM FIG. 9 (A) (8×8) AND FIG. 9 (C) (16×16) OPTIMIZED FOR AVERAGE LOSS RATE OF 12.5%.

	Penalties		
Packetization Technique	B(16,0.0625) α=1,β=0.9375	B(16,0.125) α=2,β=3.75	B(16,0.25) α=4,β=15
Slicing [10]	8939	19886	47803
PB slicing [9]	2660	8851	31820
Maximum i.p.d. [11]	1220	4480	17920
Our method 8×8	388	1555	6222
Our method 16×16	259	1035	4132

We have simulated the loss of every possible combination of p packets out of a total of 16 for the three images, for $p = 1, \ldots, 5$. For every combination of packet loss and for each packetization method, we have repaired the images by using the same reconstruction method, which is the interpolation technique of [7] introduced in Section II-A. The results for the three test images are given in Table III. It also lists the average quality in case of no concealment. In this case lost coefficients are replaced by zero, except for the LL_ℓ coefficients. These are replaced by the value which corresponds to a pixel value of 128.

The results in Table III demonstrate that for the loss of one packet out of 16 (6.25% of the coefficients; $p = 1$), all packetization techniques perform similarly, except the slicing technique which performs much worse. The reason for this is that for each lost coefficient all the ERs are correctly received, except in the slicing technique. All the other techniques satisfy our first packetization property, i.e., have large (enough) intra-packet distances. This is also visible in the theoretically calculated penalty in Table II: the penalty of the slicing technique is much higher than those of the other techniques.

For the loss of two and more packets we notice that our method outperforms all the others in terms of the standard deviation of the quality of the reconstructed images and in terms of the worst case quality of the reconstructed images. For the other tested techniques, there is a stronger variation in reconstruction quality depending on which packets are lost, and a lower worst case reconstruction quality[4]. As expected, our 16×16 mask has a higher performance than our 8×8 mask. For a larger submask, periodicity have less of an impact, and our third packetization property (*maximal spreading of the SERs of all elements of a packet over the other packets*) can be better fulfilled. On the other hand, the calculation of a 16×16 submask takes about 12 times more computation time than the calculation of an 8×8 submask[5].

[4]The worst case is the combination of lost packets which yields the lowest reconstruction quality.

[5]The calculation of an 8×8 submask with 16 packets with a Matlab implementation on an AMD 3400+ lasts about 25 minutes.

6	16	15	13	2	1	11	2
4	7	11	10	5	16	3	12
1	14	12	3	8	9	13	10
9	15	5	13	4	11	14	4
12	8	11	6	15	7	5	3
7	2	16	4	2	10	8	6
9	14	6	5	9	14	16	10
1	8	12	1	3	7	13	15

(a)

15	14	16	11	3	13	10	9
4	1	8	6	2	15	5	6
13	7	11	14	9	11	4	8
16	15	13	10	3	1	12	7
2	8	12	7	2	5	14	4
11	5	10	6	16	9	8	10
12	15	1	13	5	12	3	16
6	3	4	9	7	14	2	1

(b)

2	10	5	15	13	5	14	4	10	15	7	4	6	10	5	13
14	13	7	14	9	7	3	1	5	4	8	13	5	2	15	1
9	12	4	6	8	10	12	6	13	16	2	12	8	11	7	8
5	11	16	15	13	4	5	2	14	6	11	1	14	9	6	2
1	10	7	12	9	3	15	1	7	12	3	13	10	3	5	12
4	9	2	14	1	2	13	16	9	15	5	8	11	15	8	14
3	6	16	5	11	7	3	10	11	4	7	16	12	2	9	11
2	7	8	12	6	16	1	15	6	1	2	13	4	10	16	12
16	14	11	15	9	2	11	13	14	9	15	8	1	6	8	4
1	7	13	3	10	8	4	10	5	4	7	3	5	13	3	9
10	12	1	16	7	5	6	12	16	11	10	14	16	11	4	5
15	6	14	15	1	2	3	14	9	7	1	8	15	10	14	8
2	4	8	3	11	4	15	10	2	6	3	12	11	6	13	9
3	16	12	9	5	14	7	12	4	15	16	13	2	4	7	10
6	7	13	10	1	2	3	1	13	11	5	9	8	3	11	8
15	9	11	16	8	6	16	9	6	14	3	12	1	16	14	12

(c)

12	9	16	12	3	13	16	3	7	5	14	9	11	16	8	13
1	3	13	11	6	9	1	12	6	12	13	7	1	3	6	15
16	8	15	7	10	14	11	9	2	11	10	15	8	9	10	11
5	9	4	5	3	12	13	7	15	5	1	4	11	5	1	2
14	12	2	14	6	8	4	1	3	13	6	7	2	13	3	7
1	11	7	13	9	7	10	5	12	16	14	9	12	10	15	16
6	4	16	15	16	3	2	8	7	6	4	5	15	1	9	13
5	3	2	12	10	5	6	16	11	10	13	11	8	6	4	14
12	15	14	8	4	13	15	1	14	12	1	9	2	13	11	16
4	11	6	5	7	14	11	2	3	8	15	6	5	1	8	10
3	14	2	15	12	16	6	4	5	9	4	12	16	7	2	9
6	16	7	4	13	10	1	8	13	1	2	13	9	10	12	15
4	8	10	3	11	5	12	7	6	11	15	8	11	3	7	14
5	7	15	14	9	8	3	10	14	3	9	14	12	6	8	10
16	11	3	4	16	10	1	13	4	16	2	1	4	9	15	2
4	10	2	8	5	2	14	8	1	15	10	6	2	5	14	7

(d)

Figure 9. **(a)** and **(c)** 8×8 and 16×16 packetization masks respectively, optimized for the reconstruction of lost wavelet coefficients. **(b)** and **(d)** 8×8 and 16×16 packetization masks respectively, optimized for the reconstruction of lost DCT coefficients. In the stochastic sampling, we used, for all masks, as the probabilistic model for the packet loss, the binomial distribution with an average packet loss rate of 12.5% ($\alpha = 2, \beta = 3.75$).

Table III

COMPARISON (PSNR IN DB) OF THE PROPOSED PACKETIZATION METHOD AND EXISTING METHODS FOR THE LOSS OF p PACKETS ($p = 1 \ldots 5$) FOR THE RECONSTRUCTION AFTER LOSS OF UNQUANTIZED WAVELET COEFFICIENTS.

	Lena			Goldhill			Barbara		
$p = 1$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	24.67			26.02			26.31		
Slicing [10]	29.30	0.90	28.04	30.38	1.72	25.94	28.88	1.16	26.50
PB slicing [9]	30.60	0.72	29.50	33.99	1.81	31.29	30.44	0.32	29.96
Maximum i.p.d. [11]	30.59	0.60	29.62	33.65	0.29	33.18	30.44	0.30	30.06
Our method 8×8	30.57	0.48	29.91	33.65	0.35	32.93	30.44	0.33	29.69
Our method 16×16	30.58	0.54	29.72	33.66	0.41	33.06	30.46	0.54	29.51
$p = 2$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	21.63			22.99			23.28		
Slicing [10]	26.03	0.84	22.95	27.00	1.44	19.54	25.63	0.88	23.16
PB slicing [9]	27.47	0.75	24.73	30.60	1.29	26.02	27.34	0.45	24.99
Maximum i.p.d. [11]	27.42	0.51	25.93	30.48	0.33	29.41	27.32	0.31	26.08
Our method 8×8	27.41	0.35	26.35	30.48	0.27	29.76	27.31	0.25	26.54
Our method 16×16	27.41	0.37	26.55	30.48	0.30	29.72	27.32	0.37	26.46
$p = 3$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	19.86			21.22			21.51		
Slicing [10]	23.99	0.88	19.96	24.93	1.40	19.09	23.66	0.75	21.14
PB slicing [9]	25.58	0.75	23.54	28.55	1.10	24.79	25.47	0.48	23.73
Maximum i.p.d. [11]	25.50	0.50	23.93	28.55	0.36	27.28	25.43	0.33	23.94
Our method 8×8	25.48	0.32	24.38	28.54	0.24	27.76	25.42	0.23	24.58
Our method 16×16	25.48	0.31	24.67	28.54	0.26	27.69	25.42	0.31	24.45
$p = 4$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	18.60			19.97			20.26		
Slicing [10]	22.44	0.91	18.14	23.36	1.39	18.41	22.21	0.67	20.01
PB slicing [9]	24.18	0.75	19.11	27.03	1.03	20.88	24.10	0.50	20.32
Maximum i.p.d. [11]	24.06	0.51	22.16	27.10	0.39	25.42	24.03	0.35	22.44
Our method 8×8	24.04	0.32	22.52	27.08	0.23	26.23	24.02	0.23	23.10
Our method 16×16	24.04	0.28	23.10	27.08	0.24	26.22	24.02	0.28	23.14
$p = 5$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	17.63			18.99			19.29		
Slicing [10]	21.15	0.92	17.54	22.07	1.36	17.49	21.05	0.60	19.02
PB slicing [9]	23.03	0.78	18.63	25.78	1.02	20.57	22.99	0.52	19.77
Maximum i.p.d. [11]	22.88	0.53	20.92	25.90	0.40	24.34	22.89	0.37	21.36
Our method 8×8	22.86	0.33	21.45	25.88	0.23	24.88	22.87	0.23	21.94
Our method 16×16	22.85	0.27	21.72	25.88	0.24	25.07	22.87	0.26	21.96

We also illustrate our results with some image material. Fig. 11 shows the damaged and the reconstructed *Goldhill*-images with the minimal quality, i.e., the worst combination of five lost packets for each packetization scheme. These results are also numerically presented in Table III for $p = 5$. For the *slicing* technique [10], if five slices get lost, it is impossible to reconstruct the lost inner coefficients of adjacent lost slices. The *parity based slicing* [9] gives better results, but if four adjacent slices get lost, the situation is the same as with *normal* slicing, and there is no advantage due to interleaving. The minimax method [11] gives better results. Even with five packets out of sixteen lost, it rarely happens that a lost coefficient has no correctly received ERs. Therefore lost coefficients can almost always be interpolated, especially in this example where there is a two pass interpolation: if a coefficient cannot be repaired because none of its first order ERs are available in the first pass, then it can still be repaired in the second pass, after some of the ERs have been repaired. However, due to the unequal spreading of the coefficients, there is the same repetitive pattern of lost coefficients throughout the whole image, as explained in Fig. 5, which may result in undesirable situations of lost elements with lost SERs. The highest worst case PSNR value is obtained with the proposed new packetization technique. By spreading the coefficients more equally, the repetitive pattern is broken. On average there are more ERs available than with the other packetization techniques. This gives a better reconstruction with a higher PSNR value. To compare the technique of [11] with our technique, Fig. 12 presents details of Fig. 11 (h), (i) and (j). Fig. 12 (a) clearly shows the black spots on the white house. Due to a better spreading of the coefficients and hence a better reconstruction, these spots are less visible in Fig. 12 (b) and (c).

B. Influence of packetization on the reconstruction of quantized wavelet coefficients

Now we repeat the experiment from Section IV-A but on quantized wavelet coefficients. We used the same test images *Lena*, *Goldhill* and *Barbara*. Their wavelet coefficients were quantized to a quality of respectively 33.64 dB, 31.66 dB and 36.13 dB. For state of the art wavelet image coders (JPEG2000 [15]) this corresponds to 0.66, 0.88 and 1.74 bits per pixel. For compression with packetization, the bitrate will be higher as the correlation between coefficients in the same packet will be lower due to the spreading of the coefficients. This is discussed further on in this Section.

The results for the three test images are given in Table IV. If we compare these results to Table III, we notice that the quality of the reconstructed images is always lower in this quantized case. Relative comparison between the methods remain the same as in Section IV-A. For $p = 1$, all packetization techniques perform well, except for the slicing technique. For the loss of two and more packets we notice that our method outperforms all the others in terms of the standard deviation and in terms of the minimum quality of the reconstructed images. In terms of the average quality of the reconstructed images, [9], [11], and the proposed technique are all comparable.

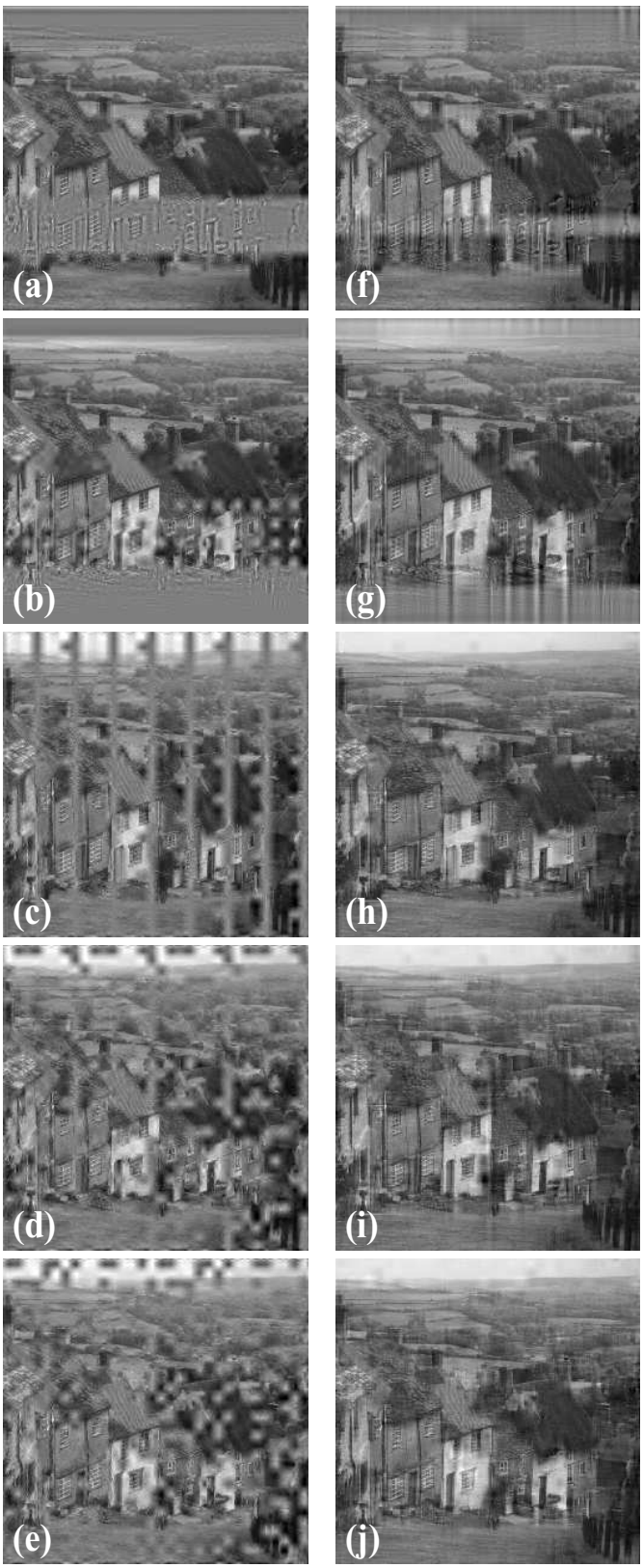


Figure 11. Wavelet coded *Goldhill*-image after the loss of 5 packets (31.25% of the coefficients). Left: images after packet loss. Right: reconstructed images. Top to bottom: slicing [10], parity based slicing [9], maximization of the minimum i.p.d. [11] and our technique for a 8×8 mask and for a 16×16 mask. The PSNR values of the images in the right column are respectively: 17.49 dB, 20.57 dB, 24.34 dB, 24.88 dB and 25.07 dB.

Figure 12. Enlarged details of the images of respectively Fig. 11 (h), (i) and (j).

Table IV
COMPARISON (PSNR IN DB) OF THE PROPOSED PACKETIZATION METHOD AND EXISTING METHODS FOR THE LOSS OF p PACKETS ($p = 1 \ldots 5$) FOR THE RECONSTRUCTION AFTER LOSS OF QUANTIZED WAVELET COEFFICIENTS.

	Lena			Goldhill			Barbara		
$p = 1$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	24.17			25.02			25.90		
Slicing [10]	28.03	0.65	27.10	28.02	1.08	24.86	28.20	1.01	26.10
PB slicing [9]	29.00	0.50	28.24	29.78	0.70	28.61	29.56	0.26	29.18
Maximum i.p.d. [11]	29.00	0.42	28.31	29.73	0.13	29.51	29.55	0.24	29.26
Our method 8×8	28.99	0.34	28.50	29.73	0.15	29.42	29.56	0.27	28.93
Our method 16×16	28.99	0.38	28.40	29.73	0.17	29.48	29.57	0.43	28.80
$p = 2$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	21.39			22.50			23.09		
Slicing [10]	25.45	0.73	22.68	25.84	1.17	19.29	25.35	0.83	22.99
PB slicing [9]	26.72	0.63	24.35	28.34	0.77	25.15	26.97	0.41	24.80
Maximum i.p.d. [11]	26.68	0.43	25.39	28.32	0.20	27.63	26.95	0.28	25.82
Our method 8×8	26.67	0.30	25.73	28.32	0.17	27.85	26.95	0.23	26.23
Our method 16×16	26.67	0.31	25.94	28.32	0.19	27.84	26.95	0.34	26.16
$p = 3$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	19.71			20.91			21.39		
Slicing [10]	23.66	0.81	19.83	24.22	1.24	18.88	23.51	0.73	21.06
PB slicing [9]	25.14	0.67	23.29	27.14	0.80	24.13	25.28	0.45	23.64
Maximum i.p.d. [11]	25.08	0.45	23.63	27.16	0.26	26.21	25.25	0.32	23.83
Our method 8×8	25.07	0.30	24.04	27.16	0.18	26.58	25.24	0.22	24.43
Our method 16×16	25.06	0.28	24.33	27.16	0.20	26.53	25.24	0.29	24.32
$p = 4$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	18.50			19.75			20.18		
Slicing [10]	22.23	0.87	18.07	22.90	1.29	18.24	22.13	0.66	19.96
PB slicing [9]	23.91	0.70	19.04	26.08	0.83	20.62	24.00	0.48	20.27
Maximum i.p.d. [11]	23.81	0.48	22.06	26.17	0.31	24.70	23.94	0.34	22.36
Our method 8×8	23.80	0.31	22.31	26.16	0.19	25.43	23.93	0.22	23.02
Our method 16×16	23.79	0.27	22.91	26.16	0.20	25.45	23.93	0.27	23.08
$p = 5$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	17.56			18.84			19.23		
Slicing [10]	21.02	0.89	17.49	21.76	1.29	17.37	21.00	0.60	18.99
PB slicing [9]	22.87	0.74	18.58	25.13	0.88	20.35	22.95	0.51	19.75
Maximum i.p.d. [11]	22.73	0.51	20.80	25.27	0.34	23.83	22.85	0.37	21.32
Our method 8×8	22.71	0.32	21.32	25.25	0.20	24.37	22.84	0.22	21.92
Our method 16×16	22.71	0.26	21.62	25.25	0.21	24.55	22.84	0.25	21.95

A disadvantage of the proposed packetization technique is that it has worse coding performance due to the interleaving of wavelet trees. Compression schemes typically achieve a high compression ratio by exploiting redundancy between neighboring (inter or intra subband) wavelet coefficients. By spreading neighboring coefficients over different packets (and by coding each packet independently), the redundancy between coefficients in the same packet is significantly lowered. This results in a worse compression performance. For the coding of the different packets, we use a similar approach as in [29]: trees of coefficients of a packet are coded independently of trees in other packets. Due to this independent coding of the coefficient trees, the compression ratio will not be as high as in the case of Zerotree Coding [13] or Set Partitioning in Hierarchical Trees Coding (SPIHT) [14]. These problems are typical for robust packetization and also occur for parity based slicing [9], the technique of [12] and the compression method of [29]. In [9] the loss in coding performance is about 4% due to the parity based slicing packetization of the motion vectors (compared to the normal slicing). In [29] and [12], the quality of a wavelet coded image with an interleaving packetization scheme is compared to the quality of an image coded with

SPIHT coding where the packetization is nothing more than dividing the bitstream into packets of a fixed length. Both [29] and [12] report a difference in quality of circa 1 dB, in favor of the SPIHT coding. We may therefore conclude that our technique will also suffer from this loss of coding efficiency. On the other hand, our packetization technique assures a graceful degradation in case of packet loss. In case of the SPIHT coding, packet loss will have a devastating effect on the image quality, as the loss of one packet could make other packets also useless.

C. Influence of packetization on the reconstruction of unquantized DCT coefficients

In case of packet loss, complete DCT-blocks (of 8×8 pixels) are lost as all coefficients from a block are coded together. In this case, interpolation is based on border pixels of correctly received neighboring blocks, as explained in Section II-B. So, in this case only the first order SER is used for interpolation. In contrast to the two pass interpolation scheme for wavelet coefficients (used in the previous experiment), we now use a simple one pass interpolation scheme. If a lost DCT block has no correctly received ERs, interpolation is impossible, and gray holes will be visible in the reconstructed image.

For this experiment we also calculated two masks to show the influence of the size of the submasks. Bigger submasks take more calculation time, but can fulfill the desired properties much better. We interpolate lost DCT blocks from the first order neighboring blocks only. The generated 8×8 and 16×16 masks are given in Fig. 9 (b) and (d) respectively. The initial 8×8 mask used in the metropolis algorithm is again the mask from Fig. 10. The initial 16×16 mask is a periodical extension of this initial 8×8 mask. The packet loss was again modeled by the binomial distribution with an average packet loss rate of 12.5%, i.e., B(16,0.125) model. In Table V, the penalties (as defined in Section III-C) for each packetization technique are given for both the model B(16,0.125) used in our stochastic sampling mask optimization and for two other packet loss models. For all models, the best mask according to the proposed criterion is the 16×16 mask produced by our algorithm (Fig. 9 (d)). For the packet loss model B(16,0.125), this mask is indeed optimal. For the other models, we can generate even better masks with lower penalties. The quality measurement will be proven correct by the image reconstruction tests.

For this setup, we also simulated the loss of every possible combination of p packets out of a total of 16 packets ($p = 1, \ldots, 5$). Reconstruction is done by using the method of Section II-B. Results are given in Table VI for respectively the *Lena*, *Goldhill* and *Barbara*-images. The general conclusions for DCT coding are similar to those for wavelet coding. For $p = 1, 2, 3$ all packetization techniques perform equally well, except for slicing, which gives significantly worse results. For $p \geq 4$, in terms of the average PSNR, the minimax technique of [11] (maximization of the minimum i.p.d.), and our technique performs similarly and better than both slicing-based methods. Our method outperforms [11] and both slicing methods in terms of standard deviation and the higher minimal quality of the reconstructed images.

Table V
MASK PENALTIES FOR DCT CODED DATA, WHERE THE PACKET LOSS MODEL IS THE BINOMIAL DISTRIBUTION B(16,π) FOR LOSS PROBABILITY π OF 6.25%, 12.50%, AND 25.00%. OUR MASKS ARE THOSE FROM FIG. 9 (B) (8×8) AND FIG. 9 (D) (16×16) OPTIMIZED FOR AVERAGE LOSS RATE OF 12.5%.

	Penalties		
Packetization Technique	B(16,0.0625) α=1,β=0.9375	B(16,0.125) α=2,β=3.75	B(16,0.25) α=4,β=15
Slicing [10]	211.7	462.9	1083.7
PB slicing [9]	39.7	158.9	635.7
Maximum i.p.d. [11]	23.5	93.9	375.5
Our method 8×8	6.4	25.6	102.3
Our method 16×16	4.1	16.5	66.1

In Fig. 13 we compare the results of the tested packetization methods on the *Goldhill*-image, showing *minimal* quality for the loss of 5 packets (31.25% of the coefficients). The corresponding PSNR results are in Table VI for $p = 5$. The visual results show that our 16×16 mask yields the highest quality. To make the improvement over [11] better visible, we show in Fig. 14 enlarged details of images from Fig. 13 (h), (i) and (j). The detail in Fig. 14 (a) (the method of [11]) shows a lot of gray blocks. The reconstructed images which were encoded with our packetization technique (Fig. 14 (b) and (c)) clearly have less gray holes due to a better spreading.

D. Influence of packetization on the reconstruction of quantized DCT coefficients

In a fourth experiment we investigated the packetization of quantized DCT coefficients. For this experiment the test images *Lena*, *Goldhill* and *Barbara* were compressed with a JPEG coder [21] to a quality of respectively 27.71 dB, 33.82 dB and 28.77 dB, which corresponds to respectively 0.37, 1.90 and 1.23 bits per pixel. If the JPEG compression is modified with a packetization step, the bitrate will be higher as the correlation between DCT blocks in the same packet will be lower due to the spreading of the blocks. This is comparable to the compression/packetization of wavelet coefficients. On the other hand, our packetization assures a graceful degradation in case of packet loss. If the compressed JPEG image is transferred without any intelligent packetization (e.g., dividing the bitstream into packets of a fixed length), any packet loss or error will have a devastating effect on the image quality, as the loss of one packet could make other packets also useless.

The results for the three test images are given in Table VII. We notice that the quality of the reconstructed images is always lower than in the case where no compression is used (Table VI). Apart from that, we can draw the same conclusions as in Section IV-C. For low packet loss rates ($p = 1$), all packetization techniques perform well, except for the slicing technique. For the loss of two and more packets we notice that our method outperforms all the others in terms of the standard deviation and in terms of the minimum quality of the reconstructed images. In terms of the average quality of the reconstructed images, the packetization based on the maximization of the minimum i.p.d. [11] and the proposed technique are comparable, while parity based slicing [9] performs slightly worse.

Table VI
COMPARISON (PSNR IN DB) OF THE PROPOSED PACKETIZATION METHOD AND EXISTING METHODS FOR THE LOSS OF p PACKETS ($p = 1 \ldots 5$) FOR THE RECONSTRUCTION AFTER LOSS OF UNQUANTIZED DCT-COEFFICIENTS.

	Lena			Goldhill			Barbara		
$p = 1$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	24.64			26.00			26.29		
Slicing [10]	30.39	1.57	26.56	30.62	2.50	22.99	29.46	1.01	27.93
PB slicing [9]	34.30	1.37	32.88	35.99	2.32	33.98	32.64	0.51	31.96
Maximum i.p.d. [11]	34.20	0.88	32.58	35.53	0.40	35.08	32.63	0.33	32.04
Our method 8×8	34.19	0.80	32.66	35.54	0.50	34.48	32.63	0.33	32.00
Our method 16×16	34.17	0.71	32.80	35.55	0.60	34.36	32.64	0.48	31.58
$p = 2$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	21.62			22.98			23.27		
Slicing [10]	26.76	1.64	21.86	26.98	2.23	19.59	26.15	0.88	23.68
PB slicing [9]	31.01	1.05	26.28	32.48	1.47	28.15	29.45	0.56	27.39
Maximum i.p.d. [11]	30.93	0.75	28.73	32.31	0.44	30.83	29.45	0.37	28.45
Our method 8×8	30.91	0.58	29.67	32.31	0.39	31.22	29.44	0.26	28.91
Our method 16×16	30.90	0.53	29.76	32.31	0.44	31.12	29.44	0.35	28.32
$p = 3$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	19.85			21.21			21.51		
Slicing [10]	24.39	1.64	19.74	24.73	2.04	19.13	24.10	0.83	21.43
PB slicing [9]	28.91	1.29	22.45	30.30	1.46	24.13	27.45	0.74	23.75
Maximum i.p.d. [11]	28.88	0.76	26.11	30.31	0.50	28.13	27.48	0.43	26.16
Our method 8×8	28.84	0.51	26.71	30.29	0.35	29.10	27.46	0.24	26.69
Our method 16×16	28.84	0.47	27.36	30.29	0.39	29.13	27.47	0.30	26.47
$p = 4$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	18.60			19.96			20.26		
Slicing [10]	22.56	1.51	18.22	23.04	1.86	18.17	22.59	0.76	20.18
PB slicing [9]	27.17	1.59	18.22	28.52	1.67	19.10	25.89	0.90	20.32
Maximum i.p.d. [11]	27.27	0.82	23.66	28.76	0.60	25.89	25.98	0.50	24.05
Our method 8×8	27.22	0.50	25.14	28.73	0.37	26.90	25.95	0.24	25.06
Our method 16×16	27.21	0.45	25.57	28.73	0.37	27.54	25.95	0.28	24.91
$p = 5$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	17.63			18.99			19.29		
Slicing [10]	21.07	1.32	17.15	21.69	1.67	17.51	21.36	0.68	19.23
PB slicing [9]	25.58	1.81	17.87	26.89	1.84	18.96	24.54	1.00	19.88
Maximum i.p.d. [11]	25.87	0.90	22.12	27.41	0.72	23.69	24.70	0.55	22.67
Our method 8×8	25.80	0.52	23.77	27.37	0.43	25.39	24.67	0.26	23.73
Our method 16×16	25.80	0.45	24.17	27.36	0.38	25.83	24.67	0.26	23.72

Figure 14. Enlarged details of the images of respectively Fig. 13 (h), (i) and (j).

V. CONCLUSION

We have presented a novel packetization method for robust transmission of image content. We formulated three desired properties for good packetization in terms of spreading the Elements needed for Reconstruction (ERs). The essence of these properties is in maximizing the probability that the ERs of a lost element are correctly received, both by avoiding that ERs are stored in the same packet, and by spreading the information needed for repairing a lost packet equally among all other packets. Starting from the defined desired properties, we proposed a novel cost function and developed the corresponding stochastic sampling based optimization method.

We have tested our method for the packetization of wavelet coefficients as well as for blocks of DCT coefficients. Compared to other packetization techniques we are able to achieve the same or better mean quality of the reconstructed images while we also reduce the fluctuation of the quality. Furthermore, we significantly increase the quality of the worst case scenario, especially for high packet loss rates. Extensions of the proposed method to video and to color images/video will be topics of further research. In video communication

Table VII
COMPARISON (PSNR IN DB) OF THE PROPOSED PACKETIZATION METHOD AND EXISTING METHODS FOR THE LOSS OF p PACKETS ($p = 1 \ldots 5$) FOR THE RECONSTRUCTION AFTER LOSS OF QUANTIZED DCT-COEFFICIENTS.

	Lena			Goldhill			Barbara		
$p = 1$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	22.27			25.37			24.44		
Slicing [10]	24.47	0.41	23.23	28.87	1.89	22.65	26.13	0.48	25.33
PB slicing [9]	25.35	0.11	25.09	31.77	0.73	31.04	27.44	0.15	27.24
Maximum i.p.d. [11]	25.35	0.12	25.12	31.72	0.16	31.54	27.44	0.10	27.24
Our method 8×8	25.35	0.12	25.12	31.72	0.21	31.27	27.44	0.10	27.23
Our method 16×16	25.35	0.08	25.23	31.72	0.25	31.20	27.44	0.15	27.09
$p = 2$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	20.36			22.68			22.31		
Slicing [10]	23.22	0.77	20.46	26.18	1.94	19.44	24.33	0.59	22.58
PB slicing [9]	24.90	0.26	23.32	30.23	0.82	27.24	26.34	0.29	25.18
Maximum i.p.d. [11]	24.90	0.20	24.23	30.19	0.28	29.24	26.34	0.19	25.82
Our method 8×8	24.90	0.15	24.55	30.18	0.23	29.50	26.34	0.12	26.06
Our method 16×16	24.90	0.12	24.61	30.19	0.27	29.42	26.34	0.17	25.77
$p = 3$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	19.03			21.02			20.89		
Slicing [10]	22.04	0.95	18.92	24.26	1.87	18.99	22.92	0.62	20.76
PB slicing [9]	24.39	0.54	21.03	28.87	1.08	23.72	25.35	0.49	22.72
Maximum i.p.d. [11]	24.41	0.29	23.18	28.92	0.38	27.24	25.38	0.27	24.48
Our method 8×8	24.40	0.19	23.66	28.91	0.25	28.03	25.37	0.15	24.85
Our method 16×16	24.40	0.16	23.77	28.92	0.28	28.03	25.37	0.19	24.70
$p = 4$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	18.02			19.83			19.82		
Slicing [10]	20.93	0.99	17.69	22.74	1.74	18.07	21.76	0.61	19.75
PB slicing [9]	23.76	0.84	17.69	27.56	1.39	18.98	24.40	0.67	19.87
Maximum i.p.d. [11]	23.86	0.40	21.86	27.80	0.49	25.37	24.48	0.36	23.02
Our method 8×8	23.85	0.23	22.89	27.78	0.30	26.27	24.46	0.17	23.81
Our method 16×16	23.85	0.20	23.05	27.78	0.30	26.83	24.46	0.20	23.70
$p = 5$	mean	stdv	min	mean	stdv	min	mean	stdv	min
Without concealment	17.19			18.89			18.96		
Slicing [10]	19.90	0.95	16.75	21.48	1.60	17.44	20.76	0.58	18.91
PB slicing [9]	23.02	1.10	17.43	26.25	1.62	18.85	23.47	0.81	19.49
Maximum i.p.d. [11]	23.24	0.53	20.88	26.73	0.62	23.39	23.61	0.44	21.92
Our method 8×8	23.22	0.29	22.02	26.69	0.36	24.96	23.59	0.20	22.84
Our method 16×16	23.22	0.24	22.20	26.69	0.32	25.34	23.59	0.20	22.84

applications, the proposed technique can be used for the dispersive packetization of I-frames. The packetization masks for the motion vectors and for the P-frames can then be shifted versions of the masks for the I-frames as in [12]. For the extension to color images, a similar mask shifting technique can be used. However, if a decorrelated color space is used (such as YUV), it might be beneficial to use the same masks for all the color channels, which restricts transmission errors to as few places in the image as possible.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their comments and suggestions.

REFERENCES

[1] Y. Wang and Q. Zhu, "Error control and concealment for video communication: A review," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[2] S. Li, *Markov Random Field Modelling in Computer Vision*, T. L. Kunii, Ed. Tokyo: Springer-Verlag, 1995.

[3] A. Raman and M. Babu, "A low complexity error concealment scheme for MPEG-4 coded video sequences," in *Tenth annual Symposium on Multimedia Communications and Signal Processing*, November 23-24 2001.

[4] E. Asbun and E. Delp, "Real-time error concealment in compressed digital video streams," in *Proceedings of the Picture Coding Symposium 1999*, April 1999.

[5] S. Hemami and R. Gray, "Subband-coded image reconstruction for lossy packet networks," *IEEE Transactions on Image Processing*, vol. 6, no. 4, pp. 523–539, April 1997.

[6] I. Bajić, J. Woods, and A. Chaudry, "Robust transmission of packet video through dispersive packetization and error concealment," in *Proc. Packet Video Workshop (PV'2000)*, Cagliary, Sardinia, Italy, May 2000.

[7] J. Rombaut, A. Pižurica, and W. Philips, "Passive error concealment for wavelet coded images adapted to a directional image correlation," in *Proc. SPIE's Optics East*, vol. 6001, Boston, MA, USA, 24 October 2005, pp. 60 010M 1–10.

[8] I. Bajic, "Adaptive map error concealment for dispersively packetized wavelet-coded images," *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1226–1235, May 2006.

[9] M. Stoufs, J. Barbarien, F. Verdicchio, A. Munteanu, J. Cornelis, and P. Schelkens, "Error protection and concealment of motion vectors in MCTF-based video coding," in *Proceedings of SPIE's International Symposium on Optics East*, vol. 5607, Philadelphia, PA, USA, October 25-28 2004, pp. 71–80.

[10] S. Wenger and G. Knorr, "Error resilient support in H.263+," *IEEE. Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 867—877, November 1998.

[11] I. Bajić and J. Woods, "Maximum minimal distance partitioning of the $\mathbb{Z}^2$ lattice," *IEEE Transactions on information theory*, vol. 49, no. 4, pp. 981–992, April 2003.

[12] ——, "Domain-based multiple description coding of images and video," *IEEE Transactions on Image Processing*, vol. 12, no. 10, pp. 1211–1225, October 2003.

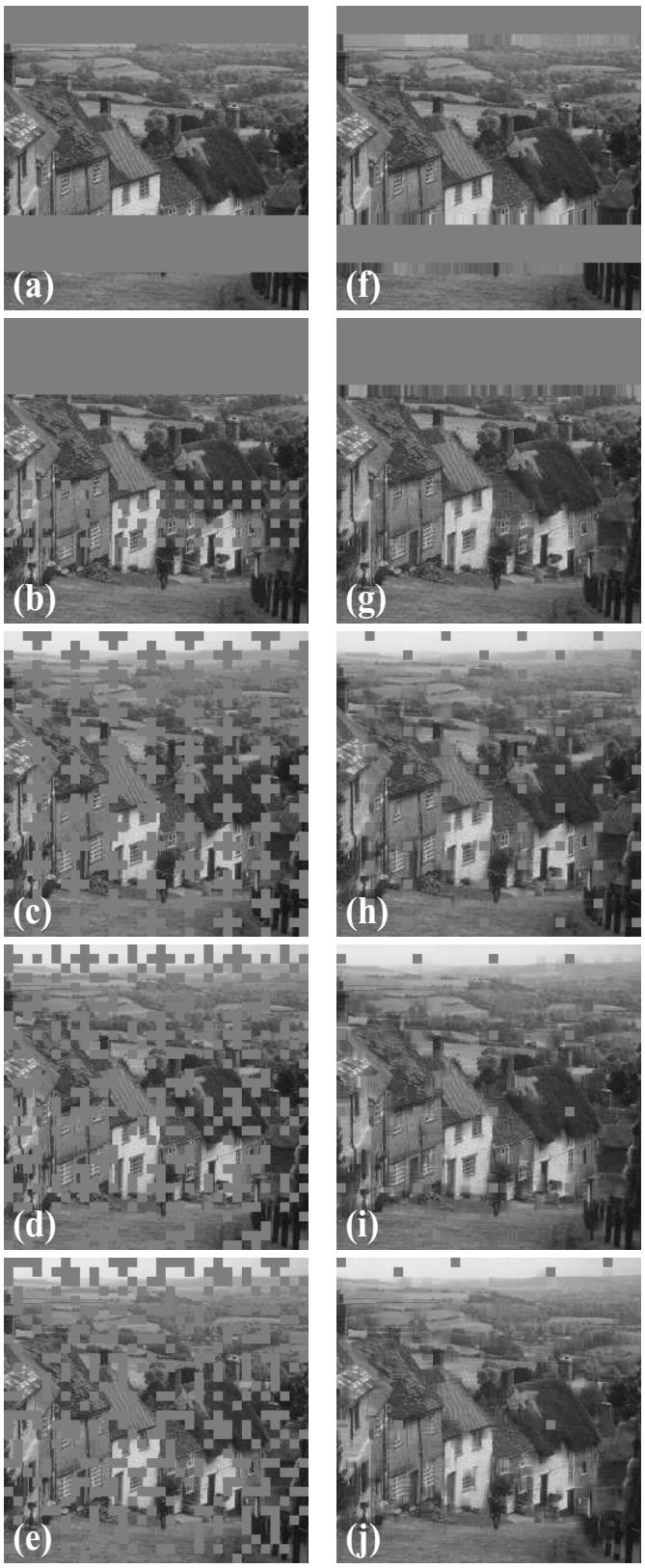


Figure 13. DCT coded *Goldhill*-image after the loss of 5 packets (31.25% of the DCT blocks). Left: images after packet loss. Right: reconstructed images. Top to bottom: slicing [10], parity based slicing [9], maximization of the minimum i.p.d. [11] and our technique for a 8×8 mask and for a 16×16 mask. The PSNR values of the images in the right column are respectively: 17.51 dB, 18.96 dB, 23.69 dB, 25.39 dB and 25.83 dB.

[13] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3445–3462, December 1993.

[14] A. Said and W. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243–250, June 1996.

[15] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. on Consumer Electronics*, vol. 46, no. 4, pp. 1103–1127, November 2000.

[16] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, July 1989.

[17] I. Daubechies, *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[18] S. Mallat, *A wavelet tour of signal processing*. Academic Press, 1998.

[19] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Prentice Hall, 2000.

[20] J. Postel, *Internet Protocol — RFC 791*. SRI Network Information Center, September 1981.

[21] G. Wallace, "The JPEG still picture compression standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[22] P. Salama, N. Shroff, and E. Delp, "A bayesian approach to error concealment in encoded video streams," *International Conference on Image Processing*, vol. 2, pp. 49–52, September 1996.

[23] Y. Zhang and K.-K. Ma, "Error concealment for video transmission with dual multiscale markov random field modeling," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 236–242, February 2003.

[24] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equations of state calculations by fast computational machine," *Journal of Chemical Physics*, vol. 21, pp. 1087–1091, 1953.

[25] V. Cerny, "Thermodynamical approach to the traveling saleman problem: An efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, pp. 41–51, 1985.

[26] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

[27] E. Collani and K. Dräger, *Binomial Distribution Handbook for Scientists and Engineers*. Boston: Birkhäuser Verlag, 2001.

[28] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill, 1965.

[29] J. Rogers and P. Cosman, "Robust wavelet zerotree image compression with fixed-length packetization," in *Proceedings of the Data Compression Conference*, March 1999, pp. 92–101.