

Joint Sparse Learning with Nonlocal and Local Image Priors for Image Error Concealment

Ali Akbari, *Member, IEEE*, Maria Trocan, *Senior Member, IEEE*, Saeid Sanei, *Senior Member, IEEE*, and Bertrand Granado

Abstract—Joint sparse representation (JSR) model has recently emerged as a powerful technique with wide variety of applications. In this paper, the JSR model is extended to error concealment (EC) application, being effective to recover the original image from its corrupted version. This model is based on jointly learning a dictionary pair and two mapping matrices that are trained offline from external training images. Given the trained dictionaries and mappings, the restoration is done by transferring the recovery problem into the sparse representation domain with respect to the trained dictionaries, which is further transformed into a common space using the respective mapping matrices. Then, the reconstructed image is obtained by back projection into the spatial domain. In order to improve the accuracy and stability of the proposed JSR-based EC algorithm and avoid unexpected artifacts, the local and non-local priors are seamlessly integrated into the JSR model. The non-local prior is based on the self-similarity within natural images and helps to find an accurate sparse representation by taking a weighted average of similar areas throughout the image. The local prior is based on learning the local structural regularity of the natural images and helps to regularize the sparse representation, exploiting the strong correlation in the small local areas within the image. Compared with the state-of-the-art EC algorithms, the results show that the proposed method has better reconstruction performance in terms of objective and subjective evaluations.

Index Terms—Robust image transmission, packet loss, error concealment, sparse representation, mapping learning.

I. INTRODUCTION

IMAGE and video transmission over error-prone channels, such as communication networks, always suffers from packet loss, leading to serious distortions in the received image/video. Over past decades, many error control techniques have been proposed. Generally, they can be classified into two categories: transmitter-based and receiver-based methods. The transmitter-based approaches retransmit the lost packets or send additional information from the transmitter to the receiver, trading the channel bandwidth for the increased error

robustness. Although these methods generally work well at the low packet loss rate (PLR), their performance is known to degrade as the PLR increases [1], [2]. Recently, a robust image transmission scheme has been proposed by Akbari *et al.* [1] and [3], by which a high quality image can be recovered, even at the high PLRs, at the expense of adding a simple random linear encoder at the transmitter side. However, the transmitter still needs to send extra information, degrading the efficiency of transmission through the low bandwidth connections.

In contrast to the transmitter-based methods, error concealment (EC), as a receiver-based method, is an alternative solution to mitigate the negative effects of the packet loss. The EC techniques recover the missing information without modifying the encoder or sending any additional information, leading to a better bandwidth efficiency [4]. These methods exploit the high spatial or temporal correlation existing among the lost areas and correctly received neighboring pixels to reconstruct a high quality image or video from its degraded version. Appropriate modeling of this intrinsic characteristic of the natural images plays an important role in the recovery performance of the EC algorithms which has been extensively studied over the past decades.

Sparse representation (SR) has been a powerful tool for image processing compression and solving inverse problems over the past decade [5]–[8]. In this technique, a signal can be represented by linear combination of a few atoms chosen from a pool called dictionary [9]. The used dictionary can be designed by some mathematical functions or learned from some training examples [10]. Recently, some SR-based EC methods have been developed and achieved desirable performance [11]–[13]. In these methods, the restoration is done by transferring the recovery problem into the SR domain. Then, the reconstructed image is obtained by back projection into the spatial domain. In other words, these methods improve the quality of corrupted images by uncovering the relationships between the corrupted and original image patches in the SR domain, where these relationships can be achieved more adaptively and accurately than those proposed for the spatial domain. Joint sparse representation (JSR) provides a powerful technique for learning the salient relationships between the data coming from different modalities [14]–[16] and can be efficiently evolved for the EC application [12], [13]. The main idea is to learn the relationships between the two modalities by training two dictionaries in a coupled manner, such that the sparse representations of the paired samples from the two modalities with respect to the trained dictionary pair are equal.

Manuscript received —, 2017; revised —, 2017; accepted —, 2017. Date of publication —, 2017; date of current version —, 2017. (Corresponding author: Ali Akbari).

A. Akbari is with the Institut Supérieur d'Electronique de Paris (ISEP), France and the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK (e-mail: ali.akbari@surrey.ac.uk).

M. Trocan is with the department of Signal, Images and Telecommunications (SITE), Institut Supérieur d'Electronique de Paris (ISEP), 28 rue Notre Dame des Champs, Paris, France (e-mail: maria.trocan@isep.fr).

S. Sanei is with the School of Science and Technology, Nottingham Trent University, Nottingham, NG11 8NS, UK (saeid.sanei@ntu.ac.uk).

B. Granado is with the Laboratoire d'Informatique de Paris 6 (LIP6), Pierre et Marie Curie University, Paris, France (e-mail: bertrand.granado@lip6.fr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Copyright 20xx IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending an email to pubs-permissions@ieee.org.

In this work, we build upon the recent success of the JSR techniques and propose a novel EC method. Different from [14]–[16] wherein the authors use the learned mappings and dictionaries for image synthesis and recognition, we reformulated the JSR-based dictionary-mapping algorithm in [16] for error concealment application. Throughout the paper, we describe the settings for learning the dictionaries and mappings. As in the JSR, two dictionaries corresponding to the original clean and corrupted image patches are trained in a coupled manner. It is assumed that the dictionaries of the original and corrupted patches are coupled to each other by identifying a mapping function. Based on this assumption, the objective is to learn the mapping function jointly with the dictionary pair. As the sparse representation of the original clean patch and the corrupted patch may not be well correlated, several coupling approaches are proposed in this paper to address this problem. In the recovery phase, the dictionary corresponding to the corrupted dataset is used to compute the sparse representation of the corrupted patch. This is then transformed into the common space using the respective mapping function in order to find a good estimation of the sparse representation of the original patch. Following this step, the dictionary corresponding to the original dataset is used to recover the original patch.

Apart from the JSR model to error recovery, some natural image priors can also be used as the regularization term to make the above joint encoding model more effective and robust for the EC application. In this paper, non-local self-similarity (NS) and local regularity (LR) priors are integrated into the proposed JSR model to further regularize the solution space. The NS model benefits from many repetitive image structures in the whole image and best preserves the edge sharpness and complex texture [17]–[21] in the concealed image. On the other hand, the LR model characterizes the local image structures, wherein each pixel is estimated via linear combination of its surrounding pixels within the support of the LR model [19], [20], [22], [23]. The way we mixed the local and non-local properties with the learned mappings and dictionaries is our main and second contribution in this paper. In contrast to the existing approaches that use these priors in the spatial domain, the novelty here is in the use of modeling the local and non-local priors in a transformed space. The reason is that it provides a more robust and accurate estimation of the sparse representation of the corrupted patch and the main features of the corrupted area can be recovered more accurately in comparison with those from spatial domain. Combining the JSR, LR, and NS models into a framework leads to a minimization problem. Designing a fast and efficient solution for this minimization problem is another contribution of this paper. The main contributions of this paper are therefore summarized as:

- We reformulated the JSR-based dictionary-mapping algorithm in [16] for the EC application.
- By combining the JSR model, NS and LR priors into a unified framework, a robust EC technique, namely joint sparse representation based-EC with the non-local and local regularization (JSR+NL) is proposed.

- A fast and effective algorithm for solving the designed minimization problem is proposed.
- A number of experiments are conducted to demonstrate the effectiveness of the proposed JSR+NL method in comparison with recent EC algorithms. Further, we analyze different parameter settings for our proposed approach, including the image patch size and the dictionary size.

A. Paper Organization

The paper is organized as follows. A review of the most related works is represented in Section II. Section III discusses the EC problem and the JSR concept. Further, a simple JSR-based EC algorithm is proposed in this section. Section IV details the proposed JSR+NL algorithm. Section V analyzes the influence of related parameters on the EC performance and then presents the experimental results on several test images. Finally, concluding remarks are made in Section VI.

B. Notation

For convenience, throughout this paper, image patch matrices whose columns are the vectorized image patches are denoted by boldface capital letters, *e.g.*, \mathbf{X} ; image patches are represented by boldface lowercase letters, *e.g.*, \mathbf{x} ; and finally scalars are denoted by italics, *e.g.*, x .

II. RELATED WORKS

In this section, we review the sparse representation literature as well as several recently proposed EC algorithms.

A. Sparse Representation

In recent years, sparse data representation has been extensively utilized in different applications by means of ℓ_0 -norm and ℓ_1 -norm minimization techniques [24]. Based on this modeling, an image patch is represented using a small number of basis functions chosen out of an over-complete dictionary [9]. The choice of over-complete dictionary plays an important role in the sparse representation modeling. One of the most flexible ways to obtain such an over-complete dictionary is by learning from a set of example image patches, which has been an active field of research over past decade [10], [25]–[27].

Recently, Yang *et al.* proposed a joint sparse representation modeling for image super resolution [28]. They assume that there exist two coupled over-complete dictionaries for two different modalities (low resolution and high resolution image patches), over which each paired samples of modalities have the same sparse representations. Based on this modeling, a coupled dictionary learning is proposed, in which the coupling is realized by enforcing the low and high resolution patches to share the same sparse feature space. This joint sparse representation modeling has been extensively used in other applications, including classification [29], cross-model matching [30], and multispectral image change detection [31]. In [15], Wang *et al.* assumed that the sparse representations of the two modalities are related to each other via a linear mapping function. This type of modeling relaxes the

strong assumption of the coupled dictionary learning algorithm in [28], and brings more flexibility to characterize the image structures. However, using a single linear mapping may not be good enough for the sparse representations of two modalities accurately [32]. In the field of cross-model matching, a new coupling model has been recently proposed by Huang *et al.* [16] which uses two linear mappings to project the sparse representations of the paired samples from the two different modalities into a common space to enable joint assessment of the two modalities.

B. Error Concealment

Many EC approaches have been developed in the literature and newer techniques are still emerging. These methods can be classified into three categories: interpolation-based, statistical-based and example learning-based methods.

The simple interpolation-based methods typically exploit the strong correlation among the neighboring blocks in the image [4], [33]–[36]. Although these methods are effective for real-time applications, in many cases, the quality of the concealed images is unsatisfactory in practice. Li *et al.* [37] propose an adaptive interpolation-based method, wherein the pixels in the missing blocks are successively recovered by exploiting a linear predictor on a small spatial window. This algorithm overcomes the blocking and blurring effects of the simple interpolation-based methods and exhibits the edge-preserving property. Different from the algorithm proposed in [37], Koloda *et al.* in [38] focus on seeking a distinct and robust linear predictor to select a linear combination of a few number of neighboring pixels for estimation of the lost pixels. Also, an adaptive linear predictor is proposed by Liu *et al.* in [39], in which the order of predictor is adaptively determined using a Bayesian information criterion.

The statistical-based methods usually assume that the missing pixels can be estimated by employing some fixed-function kernels or adaptive-structure kernels [40]–[43]. Koloda *et al.* use a minimum square error estimator in [41] to conceal the lost blocks, employing a probability density function. Zhai *et al.* in [43], obtain conditional expectation of the missing pixels given the surrounding available ones using a Bayesian framework. Markov random fields model is considered in [40] for filling the missing pixels. In [42], using a pair of Gaussian kernels and capturing the block-level similarity, the lost blocks are recovered as a weighted average of the neighboring pixels.

Example learning-based approaches are superior to both interpolation-based and statistical-based methods, since they are able to produce novel details that cannot be recovered by the aforementioned methods. Several example learning-based EC methods have been proposed in [11] and [44]. The method in [11] assumes that the sparse representation of the lost area in the image is the same as that of the known neighboring areas. However, this assumption is not satisfied when the lost area is located in the high textural region or the size of missing area becomes larger. Therefore, there is no guarantee for an accurate recovery of the missing regions. Motivated by the joint sparse representation model, the works in [12], [13] and [44] try to learn the relationships between

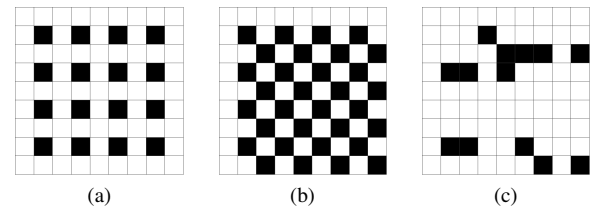


Fig. 1. Typical block loss patterns [39]; black blocks denote the corrupted blocks, (a) isolated loss and (b) consecutive loss. (c) random loss.

the patches in the corrupted and original images using two training datasets of the original and corrupted image patches. This relationship can be effectively found by transferring the original and corrupted patches into the sparse representation space over the trained dictionaries. The approach presented in this paper bares some similarities to the work in [44], [45], as we also learn a mapping function. However, the authors in [44], [45] train a mapping function between a lost block and its neighboring pixels by finding the similar patches in the input corrupted image in order to transfer the knowledge of available regions to the missing regions for the EC purpose. However, in the extreme, when the similar patches cannot be found in the image, it is impossible to learn a good mapper. The formulation investigated in our paper overcomes this issue by learning a mapper offline. Furthermore, the NS and LR priors, existing in the natural images, are exploited in a new way to improve the quality of concealed image.

III. JOINT SPARSE REPRESENTATION FOR EC

In this section, we firstly formulate the image EC problem to be considered in this paper. Next, based on the concept of joint sparse representation, a joint dictionary-mapping learning algorithm is introduced. Finally, the JSR model for the image EC is proposed.

A. Problem Formulation

In the existing image transmission systems, each image frame is divided into the non-overlapping blocks, which are separately encoded. In the packetization step, one or more encoded blocks are fed into one packet; therefore, while transmitted over an error-prone channel, the undesired packet erasure leads to the loss of an area of the image. Fig. 1 shows the different packet loss patterns occurred during the transmission over an error-prone channel.

Let \mathcal{L} denote a $B \times B$ lost block in the corrupted image and \mathcal{S} be the set of available pixels, called support area. Consider a corrupted patch of size $\sqrt{T} \times \sqrt{T}$, represented as a column vector $\mathbf{y} = [\mathbf{u}, \mathbf{v}]^T$, where $\mathbf{v} \in \mathbb{R}^P$ is a group of P unknown pixels in \mathcal{L} and $\mathbf{u} \in \mathbb{R}^{T-P}$ contains a set of adjacent and available pixels in \mathcal{S} , as shown in Fig. 2. The image EC problem asks: given a corrupted image patch \mathbf{y} , recover the original image patch \mathbf{x} using just the correctly received information. This problem is formulated as follows:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2, \quad (1)$$

where \mathbf{H} is a $T \times T$ diagonal matrix, whose diagonal entries are either 0 or 1. Here, value 0 refers to loss and 1 to correctly received corresponding pixel in the image. In this

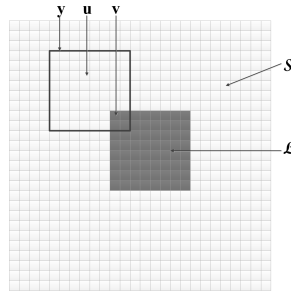


Fig. 2. Structure of the corrupted patch \mathbf{y} . Each square stands for one pixel. \mathcal{L} denotes the lost block and \mathcal{S} denotes the support area.

section, in order to solve this ill-posed problem, a joint sparse representation model is considered to regularize the solution space. The local and non-local priors are considered as another regularization term in the next section.

The corrupted and original patches belong to different visual observation subspaces \mathcal{X} and \mathcal{Y} , respectively. There is an intrinsic relationship, represented by mapping dunction \mathcal{F} , between these subspaces and if we could find this relationship, the error recovery can be achieved easily. One of the most flexible ways to discover the mapping function \mathcal{F} is by learning. In this way, the mapping function \mathcal{F} is learned from the training data offline. This mapping function is then utilized in the image reconstruction step [46], [47]. However, learning the mapping function \mathcal{F} in the spatial domain is difficult due to the existence of a complex structure among the training data. This complexity makes it hard to correlate the data of different modalities accurately. On the other hand, it is well-known that the image patches are sparse when represented by dictionaries [9]. Inspired by this fact, the mapping function \mathcal{F} can be found in the sparse representation domain more accurately. Therefore, a straightforward way to find the mapping function is to build two such dictionaries, one of which is responsible for the original patches in the observation subspace \mathcal{X} , whereas the other one is responsible for the corrupted patches in the observation subspace \mathcal{Y} . The JSR model [15], [16], [28] provides a powerful tool to learn these dictionaries jointly with the mapping function \mathcal{F} .

B. Joint Dictionary-Mapping Learning

In this section, we reformulate the JSR-based dictionary-mapping learning algorithm in [16], for EC. The main idea is to learn a dictionary pair for the corrupted and original patches in a coupled manner such that the sparse representations of the corresponding patches are maximally correlated in some transformed space.

To learn the dictionary pair, firstly, two training datasets \mathbf{X} and \mathbf{Y} are constructed as follows: A set of training images are corrupted by generating the isolated loss pattern (see Fig. 1). The lost blocks are recovered by a simple interpolation algorithm to generate an initial estimation for each corrupted image. The image patches of size $\sqrt{T} \times \sqrt{T}$, containing both known and corrupted pixels as shown in the Fig. 2, are extracted and considered as the set of corrupted training patches. The corresponding image patches at the same locations in the original images are recorded as the set of original training

patches. The mean intensity value of each patch is subtracted in order to improve the numerical stability of the dictionaries in representing the patch textures.

Let $\mathbf{X} = [\mathbf{x}_i]_{i=1}^N$ and $\mathbf{Y} = [\mathbf{y}_i]_{i=1}^N$ denote the two generated training sets, where the vectors $\mathbf{x}_i \in \mathbb{R}^T$ and $\mathbf{y}_i \in \mathbb{R}^T$ are the vector representations of the i -th original patch and corresponding corrupted patch of size $\sqrt{T} \times \sqrt{T}$, respectively. Suppose $\mathbf{D}_x \in \mathbb{R}^{T \times K}$ and $\mathbf{D}_y \in \mathbb{R}^{T \times K}$ denote the trained dictionaries for the sparse representation of the original and corrupted patches \mathbf{X} and \mathbf{Y} , respectively. These dictionaries are obtained by minimizing the following objective functions:

$$\begin{aligned} \arg \min_{\mathbf{A}_x, \mathbf{D}_x} & (\|\mathbf{X} - \mathbf{D}_x \mathbf{A}_x\|_2^2 + \lambda_x \|\mathbf{A}_x\|_1), \\ \arg \min_{\mathbf{A}_y, \mathbf{D}_y} & (\|\mathbf{Y} - \mathbf{D}_y \mathbf{A}_y\|_2^2 + \lambda_y \|\mathbf{A}_y\|_1), \end{aligned} \quad (2)$$

where $\mathbf{A}_x \in \mathbb{R}^{K \times N}$ and $\mathbf{A}_y \in \mathbb{R}^{K \times N}$ represent the corresponding sparse representation matrices. λ_x and λ_y are the regularization parameters and $\|\cdot\|_1$ denotes the ℓ_1 -norm¹.

It is assumed that the sparse representation matrices, \mathbf{A}_x and \mathbf{A}_y , are related to each other via a linear mapping matrix $\mathbf{M} \in \mathbb{R}^{K \times K}$. In [15], a mapping term, defined as:

$$\mathbf{E}_{\text{mapping}} = \|\mathbf{A}_x - \mathbf{M} \mathbf{A}_y\|_2^2, \quad (3)$$

is incorporated into the dictionary learning algorithm to find the two dictionaries, \mathbf{D}_x and \mathbf{D}_y , jointly with the mapping matrix \mathbf{M} . Although this coupled dictionary-mapping learning algorithm has been successfully applied to several tasks, such as image super resolution and image classification [15], as discussed in Section V, it might not guarantee to accurately match the corrupted and original patches for the EC application, especially when some important structures of the patches are corrupted. In other words, a more complex mapping matrix \mathbf{M} should be learned to achieve a more robust performance. This mapping matrix provides more freedom and flexibility to match the sparse coefficients of the original and corrupted patches. In the following, an approach for designing such a mapping function is described.

Recently, Huang *et al.* in [16] proposed a common space mapping approach for the cross-modal matching task, wherein a more efficient relationship is found for two different datasets \mathbf{X} and \mathbf{Y} . Inspired by this, we assume that the projections of the sparse representation matrices, \mathbf{A}_x and \mathbf{A}_y , into a K -dimensional common space using two mapping matrices, $\mathbf{M}_x \in \mathbb{R}^{K \times K}$ and $\mathbf{M}_y \in \mathbb{R}^{K \times K}$, are the same with high probability. In this case, a mapping term is defined as:

$$\mathbf{E}_{\text{mapping}} = \|\mathbf{M}_x \mathbf{A}_x - \mathbf{M}_y \mathbf{A}_y\|_2^2, \quad (4)$$

and incorporated into the dictionary learning algorithm (2), leading to the following minimization problem:

$$\begin{aligned} \arg \min_{\mathbf{D}_x, \mathbf{D}_y, \mathbf{A}_x, \mathbf{A}_y, \mathbf{M}_x, \mathbf{M}_y} & (\|\mathbf{X} - \mathbf{D}_x \mathbf{A}_x\|_2^2 + \|\mathbf{Y} - \mathbf{D}_y \mathbf{A}_y\|_2^2 \\ & + \gamma \|\mathbf{M}_x \mathbf{A}_x - \mathbf{M}_y \mathbf{A}_y\|_2^2 + \lambda_x \|\mathbf{A}_x\|_1 \\ & + \lambda_y \|\mathbf{A}_y\|_1 + \lambda_m (\|\mathbf{M}_x\|_2^2 + \|\mathbf{M}_y\|_2^2)), \end{aligned} \quad (5)$$

¹The ℓ_1 -norm is defined as follows: for $\mathbf{x} \in \mathbb{R}^N$, $\|\mathbf{x}\|_1 = \sum_{i=1}^N |x_i|$ where $|\cdot|$ stands for the absolute value operator.

where the first and second terms are fidelity terms associated with the data reconstruction error. These terms ensure that the data in two modalities are reconstructed accurately with minimum error. The third term denotes the mapping fidelity term to represent energy associated with the mapping error between the sparse representations of the corrupted and original training patches. Moreover, The fourth and fifth terms ensure that the representations of the data in two modalities \mathbf{X} and \mathbf{Y} with respect to dictionaries \mathbf{D}_x and \mathbf{D}_y are sparse. The coefficients γ , λ_x , and λ_y balance the image representation and sparsity, respectively. The ℓ_2 -norms on \mathbf{M}_x and \mathbf{M}_y impose additional constraints, regularized by λ_m , in order to provide numerical stability and avoid over-fitting.

The objective function (5) is convex with respect to each of \mathbf{D}_x , \mathbf{D}_y , \mathbf{A}_x , \mathbf{A}_y , \mathbf{M}_x , and \mathbf{M}_y when the others are fixed. An effective way to tackle the energy-minimization of (5) is proposed as follows: first, the dictionary pair, \mathbf{D}_x and \mathbf{D}_y , and the mapping matrices, \mathbf{M}_x and \mathbf{M}_y , are initialized as principal component analysis (PCA) basis and identity matrices, respectively. Then, three following steps are iterated until convergence: (1) updating the sparse coefficients, \mathbf{A}_x and \mathbf{A}_y , by fixing \mathbf{D}_x , \mathbf{D}_y , \mathbf{M}_x , and \mathbf{M}_y ; (2) updating the dictionary pair, \mathbf{D}_x and \mathbf{D}_y , by fixing \mathbf{M}_x , \mathbf{M}_y , \mathbf{A}_x , and \mathbf{A}_y ; (3) updating the mapping matrices, \mathbf{M}_x and \mathbf{M}_y , by fixing \mathbf{D}_x , \mathbf{D}_y , \mathbf{A}_x , and \mathbf{A}_y . The step of updating the mapping matrices is a ridge regression problem that is solved by the algorithm described in [16]. In this algorithm, a positive constant is added to the main diagonals of the mapping matrices. This small perturbation produces the diagonal mapping matrices and also guarantees that the obtained mapping matrices are invertible. This means that the solution always exists. More details can be found in [16].

C. The JSR based EC

Given the trained dictionaries, \mathbf{D}_x and \mathbf{D}_y , and the learned mappings \mathbf{M}_x and \mathbf{M}_y , the sparse representation of the corrupted patch \mathbf{y} (as shown in the Fig. 2), can be easily converted to the sparse representation of the original patch using the following optimization problem:

$$\arg \min_{\alpha_y, \alpha_x} (\|\mathbf{y} - \mathbf{D}_y \alpha_y\|_2^2 + \|\mathbf{x} - \mathbf{D}_x \alpha_x\|_2^2 + \gamma \|\mathbf{M}_x \alpha_x - \mathbf{M}_y \alpha_y\|_2^2 + \lambda_y \|\alpha_y\|_1 + \lambda_x \|\alpha_x\|_1). \quad (6)$$

This optimization problem is iteratively solved by alternately updating α_y and α_x using the two following problems:

$$\arg \min_{\alpha_y} (\|\mathbf{y} - \mathbf{D}_y \alpha_y\|_2^2 + \gamma \|\mathbf{M}_x \alpha_x - \mathbf{M}_y \alpha_y\|_2^2 + \lambda_y \|\alpha_y\|_1),$$

$$\arg \min_{\alpha_x} (\|\mathbf{x} - \mathbf{D}_x \alpha_x\|_2^2 + \gamma \|\mathbf{M}_x \alpha_x - \mathbf{M}_y \alpha_y\|_2^2 + \lambda_x \|\alpha_x\|_1). \quad (7)$$

Finally, the concealed patch is obtained via $\mathbf{x} = \mathbf{D}_x \alpha_x$. Instead of using this iterative algorithm here, a fast and yet effective way is proposed to obtain an approximated solution for the minimization problem (6).

Fig. 3 shows the pipeline of proposed JSR-based EC algorithm. First, an initial image is obtained using a simple interpolation algorithm [40]. Given a corrupted patch \mathbf{y} , its

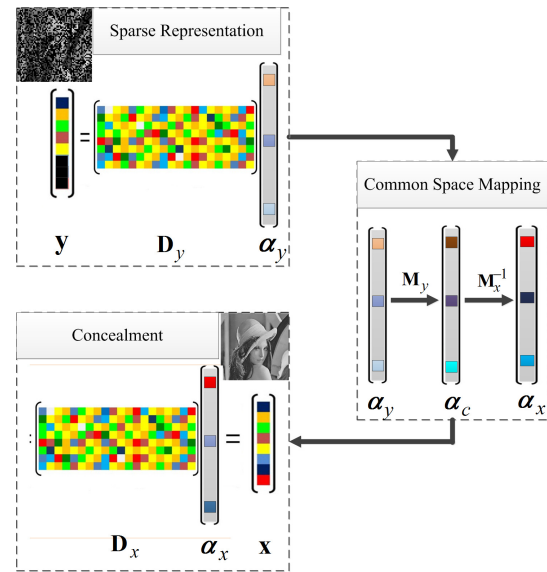


Fig. 3. Block diagram of the proposed joint sparse representation based image EC algorithm.

mean intensity value is subtracted, and then its sparse representation, α_y , with respect to the dictionary \mathbf{D}_y is obtained via the following minimization problem:

$$\arg \min_{\alpha_y} (\|\mathbf{y} - \mathbf{D}_y \alpha_y\|_2^2 + \lambda_s \|\alpha_y\|_p), \quad (8)$$

where $\|\alpha_y\|_p$ is the sparsity-inducing regularization term and λ_s denotes the regularization parameter that balances the tradeoff between the fidelity and sparsity terms. If $p = 1$, then $\lambda_s = \lambda_y$. It has been illustrated in [17], given a trained dictionary learned with ℓ_1 -norm, the sparse representation of the image patches with respect to this dictionary is in general of higher accuracy when using the ℓ_0 -norm. Therefore, we consider $p = 0$ in our implementations². The ℓ_0 -normalization problem (8) is efficiently solved by the well-known orthogonal matching pursuit (OMP) algorithm [48].

Once α_y is obtained by solving the minimization problem (8), it is projected into the K -dimensional common domain by $\alpha_c = \mathbf{M}_y \alpha_y$. It is assumed that the projection of the sparse representation of the original patch, α_x , into the common domain by $\mathbf{M}_x \alpha_x$, is also associated with the α_c , i.e. $\mathbf{M}_y \alpha_y \approx \mathbf{M}_x \alpha_x$. We next derive

$$\alpha_x = \mathbf{M}_x^{-1} \alpha_c = \mathbf{M}_x^{-1} \mathbf{M}_y \alpha_y. \quad (9)$$

For simplicity, we consider $\mathbf{P} = \mathbf{M}_x^{-1} \mathbf{M}_y$, then $\alpha_x = \mathbf{P} \alpha_y$. Further, the concealed patch is obtained via $\mathbf{x} = \mathbf{D}_x \alpha_x$. Following this, the mean intensity value of the patch is added back to the estimated patch. Finally, the unknown pixels, i.e. \mathbf{v} , are replaced by the corresponding pixels in the concealed patch \mathbf{x} (see Fig. 2).

D. Block Recovery Order

As it can be seen in Fig. 2, the corrupted patch \mathbf{y} contains P missing pixels of the lost block (In this paper, we consider $P = 4$.) So, the lost block cannot be recovered at one time. Instead,

²The $\|\cdot\|_0$ denotes the ℓ_0 -norm counting the nonzero elements

as in [38], we propose to recover the lost block sequentially. First, the lost block is partitioned into $\sqrt{P} \times \sqrt{P}$ blocks, as shown in Fig. 4. Then, the subblocks are recovered one by one based on a predetermined order. Consider the subblock that is located at the upper-left corner of the lost block in Fig. 4. In order to recover this subblock, the corresponding corrupted patch of size $\sqrt{T} \times \sqrt{T}$, consisting of this subblock and its available adjacent pixels, as shown in Fig. 2, is formed. Next, the patch is recovered using the procedure described at the previous section. Finally, the corrupted subblock is replaced with the corresponding pixels in the recovered patch. These steps are repeated for the next subblocks.

In order to improve the quality of the reconstructed block, two further considerations are employed: recovery order and overlap width. The reconstruction order, *i.e.* the order that subblocks are recovered, plays an important role to the performance of the proposed JSR-based EC algorithm. Since the accuracy of the recovered subblock depends on the reliability of its available adjacent pixels, the subblocks that have more reliable information at their support area are firstly recovered. This leads to a more precise estimation of the sparse coefficients via Eq. (8) and thus better reconstruction quality. The recovery order of an isolated lost block of size 8×8 pixels is shown in Fig. 4 where the subblocks illustrated by a brighter gray-level are recovered first. Using this order for recovering the subblocks also ensures that more reliable information at the support area of the next subblocks is already provided.

Addition to the recovery order, we also consider an overlap depth between the adjacent subblocks in order to avoid the blocking artifacts resulting from partitioning. Since the size of each subblock is 2×2 pixels, we set 1-pixel-width overlap depth between the adjacent subblocks. Next, the subblocks are recovered one by one and the value of recovered pixels located on the overlap area are averaged as their final values.

IV. JSR-BASED EC WITH NON-LOCAL AND LOCAL REGULARIZATION (JSR+NL)

Clearly, it is expected that the sparse representation vector α_x , obtained by Eq. (9) should be as near as to the true sparse representation vector α_x^t of the patch to be recovered. However, the JSR model, presented in the Section III-C, may not lead to recover the true sparse representation α_x^t due to the corrupted pixels in the input patch \mathbf{y} . On the other hand, there are many repetitive patterns and regular structures throughout the natural images. This NS model in combination with the LR prior existing in the natural images can be used as a regularization term to regularize the solution space of the minimization problem (8) and develop a much more accurate sparse representation model. The NS and LR models have been used in many applications, such as image compression [49], [50] and inverse problems [19]–[23], [51]. However, in these works, the NS and LR models are implemented in the spatial domain whereas here they are in the sparse representation domain. This regularization term is incorporated into Eq. (8) to develop a more effective EC algorithm, called JSR-based EC with the NS and LR models (JSR+NL).

The EC performance depends on the difference $\alpha_x - \alpha_x^t$. To faithfully reconstruct the original image, this difference

should be near to zero. By incorporating this difference into the minimization problem (8), we propose the following optimization problem:

$$\arg \min_{\alpha_y} (\|\mathbf{y} - \mathbf{D}_y \alpha_y\|_2^2 + \lambda_s \|\alpha_y\|_0 + \lambda_{nl} \|\mathbf{P} \alpha_y - \alpha_x^t\|_1), \quad (10)$$

where λ_{nl} represents the regularization parameter. This model enforces the sparse representation vector α_y to be estimated in a way that $\mathbf{P} \alpha_y$ is close to α_x^t . However, α_x^t is unknown and the ℓ_1 -norm in the objective function (10) cannot be directly measured. Nonetheless, if we find a good estimation of α_x^t , then we can develop a much more accurate sparse representation model.

Generally, α_x^t can be estimated in various ways. We propose to achieve a good estimation of α_x^t using the rich non-local and local redundancies existing in the input image. Then, a good estimation of α_x^t can be computed as the weighted average of the sparse representation associated with an estimation of the input patch from the non-local similar patches within the image and the sparse representation associated with an estimation of the input patch from the local neighboring pixels. This leads to the following optimization problem:

$$\arg \min_{\alpha_y} (\|\mathbf{y} - \mathbf{D}_y \alpha_y\|_2^2 + \lambda_s \|\alpha_y\|_0 + \lambda_{nl} \|\mathbf{P} \alpha_y - (a\beta_{ns} + b\beta_{lr})\|_1), \quad (11)$$

where β_{ns} is the sparse representation vector of the estimated patch obtained via linear combination of the non-local similar patches within the image and β_{lr} is the sparse representation vector of the estimated patch obtained from the local neighboring pixels. a and b are two constants balancing the contribution of the NS and LR models ($a+b=1$). In the following section, we discuss how the sparse representation vectors β_{ns} and β_{lr} are obtained.

For understanding how these estimations of α_x^t , obtained by β_{ns} and β_{lr} , can improve the quality, we should pay attention to the second regularization term in Eq. (10). This term regularizes the solution space of the first term in the minimization problem (10) more effectively than the first regularization term. Since, β_{ns} and β_{lr} capture the main features of the patches \mathbf{y}_{ns} and \mathbf{y}_{lr} , respectively, this regularization term adds an efficient constraint in order to improve the accuracy of $\mathbf{P} \alpha_y$.

A. Non-local Self-similarity for Regularization

The NS model is based on this fact that the higher level patterns, *e.g.* edges and texture, tend to repeat themselves within the image [17], [18], [51]. Based on this concept, an estimation of each input patch \mathbf{y} can be obtained via:

$$\mathbf{y}_{ns} = \sum_{i=1}^L w_i \mathbf{y}_i, \quad (12)$$

where \mathbf{y}_i is the i -th similar patch to the input patch \mathbf{y} and L denotes the number of similar patches within a large enough window of size $H \times H$ pixels. As can be seen in Fig. 5, the search window includes the area that is clean or already recovered. The weights w_i are calculated as a decreasing

function of the weighted Euclidean distance between patch \mathbf{y} and the i -th similar patch:

$$w_i = \frac{1}{\kappa} \exp(-\|\mathbf{y} - \mathbf{y}_i\|_2^2/h), \quad (13)$$

where h is a pre-determined scalar and κ denotes the normalization factor. The patches with a smaller Euclidean distance have larger weights in the average. The exponential function provides further control on the influence of similar patches on the estimation of the patch \mathbf{y} . The parameter h controls the slope of the exponential function and therefore the decay of the weights as a function of the Euclidean distances [52]. This weighting procedure emphasizes the patches which are closer to patch \mathbf{y} .

Given the estimated patch \mathbf{y}_{ns} , the sparse representation β_{ns} can be easily found by

$$\arg \min_{\beta_{ns}} (\|\mathbf{y}_{ns} - \mathbf{D}_x \beta_{ns}\|_2^2 + \lambda_{ns} \|\beta_{ns}\|_0), \quad (14)$$

The well-known OMP algorithm [48] is used to solve this minimization problem.

B. Local Structural Regularity for Regularization

The LR model is based on this assumption that a local area in the image is stationary, which states that there are meaningful local structures in the spatial domain of the natural image. Based on this property, one pixel can be predicted by weighted combination of its neighbors (called context of the LR model). Therefore, an estimation of each input patch \mathbf{y} can be obtained by estimating all its pixels sequentially.

The major challenge in the LR model is development of an accurate and fast predictive model, *i.e.* how to find the optimal weights in order to fully exploit the information contained in the context. In [50], [53], the weighting coefficients are estimated by a training procedure, where the training set is collected from the initially recovered image by searching the similar patch in the entire image. But, this procedure may not adapt the LR model to the image characteristics due to the poor quality of the initially recovered image and tends to produce the visual artifacts [22]. In this paper, we propose a learning procedure, wherein an external database of the high-quality training examples are used to obtain the weighting coefficients.

Let \mathbf{S} is a $N \times J$ matrix including N training vectors, extracted from a set of training images. Each row of \mathbf{S} consists of a $1 \times J$ vector that includes the pixels in the context of a target pixels z , as shown in Fig. 6. Let \mathbf{z} be a $N \times 1$ vector including the corresponding target pixels. Intuitively, N training vectors in the matrix \mathbf{S} are grouped into several clusters, and then a LR model is learned for each cluster. In this paper, the well-known K-means algorithm [54] is used for clustering. Let $\mathbf{S} = \{\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_C\}$, where $\mathbf{S}_k \in \mathbb{R}^{N_k \times J}$ represents the submatrix of N_k training vectors in the matrix \mathbf{S} which belongs to the cluster k and C is the total number of clusters. The weighting coefficients of the LR model of the k -th cluster, denoted by $J \times 1$ column vector \mathbf{a}_k , is obtained by solving the following least squares minimization problem:

$$\mathbf{a}_k = \arg \min_{\mathbf{a}} \|\mathbf{z}_k - \mathbf{S}_k \mathbf{a}\|_2^2, \quad (15)$$

where \mathbf{z}_k is a $N_k \times 1$ column vector that consists of the target pixels in \mathbf{z} corresponding to the training vectors in submatrix \mathbf{S}_k . A well-known closed-form solution for this minimization problem is $\mathbf{a}_k = (\mathbf{S}_k^T \mathbf{S}_k)^{-1} (\mathbf{S}_k^T \mathbf{z}_k)$ [50]. Note that \mathbf{a}_k includes the weighting coefficients of the LR model of k -th cluster.

Given an input patch \mathbf{y} and all C learned LR models, first, the k -th LR model is assigned to each pixel y_i of \mathbf{y} based on the minimum distance between its context, *i.e.* $\mathbf{s}_i \in \mathbb{R}^{1 \times J}$ as shown in Fig. 6, and the centroid of clusters $\{\mu_1, \mu_2, \dots, \mu_C\}$, *i.e.* $k = \arg \min_k \|\mathbf{s}_i - \mu_k\|$. Finally, the pixel y_i is updated via $y_i = \mathbf{s}_i \mathbf{a}_k$. All pixels in the input patch \mathbf{y} are sequentially estimated to obtain a prediction \mathbf{y}_{lr} for the input patch \mathbf{y} . Finally, the sparse representation vector β_{lr} is obtained by following the minimization function that is solved by the OMP algorithm [48]:

$$\arg \min_{\beta_{lr}} (\|\mathbf{y}_{lr} - \mathbf{D}_x \beta_{lr}\|_2^2 + \lambda_s \|\beta_{lr}\|_0). \quad (16)$$

The idea of using \mathbf{D}_x instead of \mathbf{D}_y in Eqs. (14) and (16) comes from the dictionary learning based image denoising algorithms [25]. As explained later, the obtained patches \mathbf{y}_{ns} and \mathbf{y}_{lr} are the noisy versions of the original image patch. Therefore, solving Eqs. (14) and (16) is a kind of image denoising and using \mathbf{D}_x provides better recovery quality.

C. Recovery Algorithm

The optimization problem (10) can be iteratively solved by two following separate steps.

Step1- Initialization: First, an initial sparse representation vector, denoted by $\alpha_y^{[0]}$, is obtained by the following optimization problem:

$$\alpha_y^{[0]} = \arg \min_{\alpha} (\|\mathbf{y} - \mathbf{D}_y \alpha\|_2^2 + \lambda_s \|\alpha\|_0). \quad (17)$$

This minimization problem is solved by the OMP algorithm [48]. Then, the initial estimation of \mathbf{x} , denoted by $\mathbf{x}^{[0]}$, is estimated as $\mathbf{x}^{[0]} = \mathbf{D}_x \mathbf{P} \alpha_y^{[0]}$. Based on the $\mathbf{x}^{[0]}$, initial non-local estimation of β_{ns} and local prediction of β_{lr} , *i.e.* $\beta_{ns}^{[0]}$ and $\beta_{lr}^{[0]}$ respectively, are obtained using Eqs. (14) and (16).

Step2- Enhancement: The accuracy of the sparse representation α_y is improved using an iterative process. At each iteration $l+1$, for fixed $\beta_{ns}^{[l]}$ and $\beta_{lr}^{[l]}$ obtained from previous iteration l , the updated sparse representation vector $\alpha_y^{[l+1]}$ is obtained via solving the following minimization problem:

$$\arg \min_{\alpha_y} (\|\mathbf{y} - \mathbf{D}_y \alpha_y\|_2^2 + \lambda_{nl} \|\mathbf{P} \alpha_y - a \beta_{ns}^{[l]} - b \beta_{lr}^{[l]}\|_1), \quad (18)$$

Then, Eqs. (14) and (16) are used to update β_{ns} and β_{lr} . Given $\alpha_y^{[l+1]}$, the concealed image patch is updated as $\mathbf{x}^{[l+1]} = \mathbf{D}_x \mathbf{P} \alpha_y^{[l+1]}$, which can be used to update the $\beta_{ns}^{[l+1]}$ and $\beta_{lr}^{[l+1]}$ using the minimizations in Eqs. (14) and (16), respectively. The improved $\beta_{ns}^{[l+1]}$ and $\beta_{lr}^{[l+1]}$ are then used to enhance the accuracy of α_y , and so on. The accuracy of the sparse representation α_y is gradually improved, which in turn improves the accuracy of α_x and thus the EC quality. Such a procedure is iterated until convergence. In the proposed algorithm, the recovery process can be stopped when $\|\mathbf{x}^{[l+1]} - \mathbf{x}^{[l]}\|_2 \leq \epsilon$.

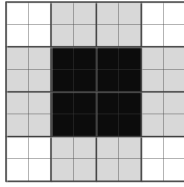


Fig. 4. Recovery order of an isolated lost 8×8 -pixels block. The pixels illustrated by a brighter gray-level are recovered first.

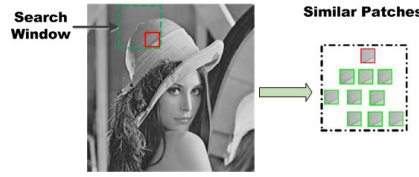


Fig. 5. Search window of the NS model. The search window includes the area that is clean or already recovered.

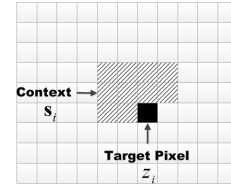


Fig. 6. Target pixel and its causal neighbors (context) of the LR model. Each square stands for one pixel.

Algorithm 1 JSR+NL for EC

- 1: **Input:** a corrupted image patch \mathbf{y} , \mathbf{D}_x , \mathbf{D}_y , \mathbf{M}_x , and \mathbf{M}_y ;
 - 2: **Output:** a concealed image patch \mathbf{x} ;
- Step1- Initialization**
- 3: **Initialize:** $\alpha_y^{[0]}$ by (17), $\mathbf{x}^{[0]} = \mathbf{D}_x \mathbf{P} \alpha_y^{[0]}$, $\beta_{ns}^{[0]}$ and $\beta_{lr}^{[0]}$ using (14) and (16), respectively;
- Step2- Enhancement**
- 4: **Repeat** on $l = 0, 1, \dots, L - 1$
 - 5: **Compute:** $\beta^{[l]} = a\beta_{ns}^{[l]} + b\beta_{lr}^{[l]}$;
 - 6: **Set:** $\alpha_y^{[l,0]} = \alpha_y^{[l-1]}$;
 - 7: **For** $j = 0, 1, \dots, J - 1$ **do**
 - 8: $\nu^{[l,j]} = \alpha_y^{[l,j]} + \mathbf{D}_y^T (\mathbf{y} - \mathbf{D}_y \alpha_y^{[l,j]}) / c$;
 - 9: $\alpha_y^{[l,j+1]} = \mathbf{P}^{-1} S (\mathbf{P} \nu^{[l,j]} - \beta^{[l]}) + \mathbf{P}^{-1} \beta^{[l]}$;
 - 10: **End for**
 - 11: **Update:**
 - 12: $\alpha_y^{[l+1]} = \alpha_y^{[l,J]}$;
 - 13: $\mathbf{x}^{[l+1]} = \mathbf{D}_x \mathbf{P} \alpha_y^{[l+1]}$;
 - 14: $\beta_{ns}^{[l+1]}$ and $\beta_{lr}^{[l+1]}$ using (14) and (16);
 - 15: **Until** $\|\mathbf{x}^{[l+1]} - \mathbf{x}^{[l]}\|_2 \leq \epsilon$.

A specific extension of the shrinkage algorithm [51], originally proposed in [55], is adapted to solve the convex minimization problem (18) iteratively. At the $j + 1 - th$ iteration, $\alpha_y^{[l,j+1]}$ is obtained by the following shrinkage operator:

$$\alpha_y^{[l,j+1]} = \mathbf{P}^{-1} S (\mathbf{P} \nu^{[l,j]} - \beta^{[l]}) + \mathbf{P}^{-1} \beta^{[l]}, \quad (19)$$

where $\nu^{[l,j]} = \alpha_y^{[l,j]} + \mathbf{D}_y^T (\mathbf{y} - \mathbf{D}_y \alpha_y^{[l,j]}) / c$, $\beta^{[l]} = a\beta_{ns}^{[l]} + b\beta_{lr}^{[l]}$, c is a constant, and $S(\cdot)$ is the soft-thresholding operator that accelerates the convergence of the recovery algorithm. At the first iteration, *i.e.* $j = 0$, $\alpha_y^{[l,0]}$ is set as $\alpha_y^{[l-1]}$ and at the end of last iteration $\alpha_y^{[l+1]}$ is updated as $\alpha_y^{[l,J]}$, where J is the number of iterations for solving (18). It should be noted that $\mathbf{P}^{-1} = \mathbf{M}_y^{-1} \mathbf{M}_x$. The complete details of the recovery algorithm are described in Algorithm 1.

At Step-1, an initial value for vector α is obtained and at Step-2, the accuracy of vector α is improved using an iterative process. If the accuracy of α at Step-1 is low, it leads to a low quality reconstructed image (at Step-1). Step-2 can compensate this quality loss with a larger number of iterations that might lead to a higher reconstruction time. So, the initial value of vector α influences the computational cost of the algorithm. In fact, the higher the initial reconstruction quality, the faster the recovery algorithm. The reason that we use ℓ_0 -norm at Step-1 is based on the observation illustrated by Mairal *et*

al. in [17]. Based on this work, given a trained dictionary learned with ℓ_0 -norm, the sparse representation of the image patches with respect to this dictionary is in general of higher accuracy when using the ℓ_1 -norm. Therefore, this initialization process via ℓ_0 -norm simplifies the subsequent work in Step-2 at a lower computational cost.

D. Discussion

As mentioned at the beginning of this section, the existing LR and NS models are implemented in the spatial domain. In these approaches, one can use the estimated patches, \mathbf{y}_{ns} and \mathbf{y}_{lr} , and simply improve the quality of the reconstructed image in the spatial domain instead of solving the minimization problem (10). However, it should be noted that the obtained patches, \mathbf{y}_{ns} and \mathbf{y}_{lr} , consist of a certain level of noise. When these noisy patches are used for recovery of the original patch in the spatial domain, the recovered patch looks unpleasant. In contrast, when the LR and NS models are implemented in the sparse representation domain, β_{ns} and β_{lr} , which are obtained by Eqs. (14) and (16), respectively, the noise is mitigated significantly. In other words, Eqs. (14) and (16) tend to extract the main features of the predicted patches \mathbf{y}_{ns} and \mathbf{y}_{lr} , respectively. Therefore, it leads to a more robust and accurate estimation of the sparse representation of the corrupted patch. This is one of the most important characteristics of the sparse representation techniques which has been used in many applications, especially image denoising.

V. EXPERIMENTAL STUDY

In this section, we briefly introduce the training datasets used by the joint dictionary-mapping learning algorithm described in Section III-B. Then, the influences of the related parameters on the EC performance are evaluated in order to select the appropriate values for these parameters. Finally, the performance of the proposed JSR and JSR+NL algorithms, presented in Sections III and IV respectively, are evaluated via a suite of simulations carried out on a set of 8-bit grayscale standard images of size 512×512 pixels, selected from the CVG-Granada database³ and Kodak dataset⁴. Note that 10 images, as shown in Fig. 7, are randomly selected from both databases for evaluation. The EC performance is assessed by the peak signal to noise ratio (PSNR). To evaluate the proposed methods, different types of loss patterns, *i.e.* *isolated loss*, *consecutive loss*, and *random loss* as shown in Fig. 1, are

³<http://decsai.ugr.es/cvg/dbimagenes/>

⁴<http://r0k.us/graphics/kodak/>



Fig. 7. 8-bit grayscale test images of size 512×512 . From left to right and top to down: *Lena*, *Peppers*, *Goldhill*, *Mandrill*, *Zelda*, *Barbara*, *Montreal*, *Pelican*, *Kodim05*, and *Kodim06*

considered with the lost blocks of size 8×8 and 16×16 pixels with different PLR, ranging from 10% up to 50%. In the case of random loss pattern, the PSNR values are obtained by running the proposed methods 10 times and the average results are reported as the final values.

A. Training Sets Description

The CVG-Granada database is used for training and test steps. The CVG-Granada database consists of 96 natural gray-level images of size 512×512 pixels. 20 images are randomly selected for training. A corrupted training image set is also generated by creating the isolated loss pattern for each image. The lost blocks are recovered by a simple interpolation algorithm [40]. Then, the training sets \mathbf{X} and \mathbf{Y} are generated following the procedure described in Section III-B. A total number of 100000 patches of size $\sqrt{T} \times \sqrt{T}$, which are rich in edges and textures, are randomly cropped from each set of training images. In practice, in order to make dictionaries more descriptive, the smooth patches are removed from the training datasets [56]. This guarantees that the meaningful patches with a certain amount of edge structures will be involved in the training algorithm. In this paper, only the patches with the intensity variance greater than a threshold $\sigma = 4$ are kept.

B. Experimental Setup

Several parameters should be selected carefully: dictionary size D , size of image patch T , values of the regularization parameters, and the related parameters in the NS and LR models, such as number of similar patches and number of clusters. By conducting a wide range of experiments, all the parameters are carefully tuned according to their best performance.

The parameters λ_x , λ_y , λ_m , γ in Eq. (5) are set to 0.01, 0.01, 0.1, 0.1, respectively. The parameter h in the Eq. (12) is empirically set to 65. The parameter λ_s in Eqs. (8), (14), and (16) is related to the number of non-zero entities (S) in the corresponding sparse representation vector. In our implementation, we set S to the nearest integer number to $T/4$. The parameters λ_{nl} , a and b are evaluated by varying one of the parameters while keeping the others constant. First, we fix λ_{nl} to 0.1. In the next step, we vary a from 0.5 to 1 and b from 0.05 to 0.5. By conducting a wide range of experiments, the parameters a and b are selected as 0.8 and 0.2 respectively.

In the JSR model, the dictionary size and patch size would jointly impact the quality of the bases in the dictionary pair and also the precision of the corresponding mappings, thus having a great effect on the EC quality. In the following sections, the impact of these parameters on the EC performance is analyzed. In each experiment, the test images are corrupted with the isolated loss pattern and restored by the proposed JSR-based EC method.

To evaluate the influence of dictionary size D on the EC quality, we set $T = 25$ and D as 64, 128, 256, 512, and 1024. For each size, the dictionary pair and the mapping matrices are learned using the joint dictionary-mapping learning algorithm. Table I shows the effect of dictionary size on the EC performance using the proposed JSR-based EC algorithm in terms of PSNR. As D increases, the PSNR value increases, but a larger value of D does not yield higher performance. It is easy to understand that the PSNR value first rises and then declines, when the dictionary size becomes larger. Further, for the highly textured images, like *Mandrill*, the improvement is negligible with respect to the dictionary size. We also evaluate the performance of the proposed JSR-based EC algorithm with different T from 16 to 64. Best dictionary pair and mappings are trained for each patch size. Table II shows the results according to different values of T . As can be seen, the PSNR values increase when T increases. Using a larger patch means that the area \mathbf{u} (see Fig. 2) contains more reliable information. Note that the size of area \mathbf{v} is always fixed (2×2 pixels). The EC quality increases because more reliable information is used for patch recovery.

Generally, larger D and T are more suitable for the proposed JSR model for the EC. The reason lies behind the fact that larger D and T will strengthen the representation ability of the dictionary pair and the matching precision of the mapping matrices. On the other hand, choosing a large patch size and dictionary size leads to more computational complexity. Therefore, we select an appropriate patch size to obtain good EC performance with lower computational costs, *i.e.* $T = 25$ and $D = 256$.

Other parameters: The number of similar patches in the NS model is set to $L = 10$ and the size of search window is set to 20×20 pixels. In addition, in order to avoid the data over-fitting, a LR model of the order 10 (*i.e.* $J = 10$) is used. The optimal selection of the number of clusters in the LR model is a nontrivial task. If the number of clusters decreases, the distinctiveness of the LR models is decreased. On the contrary, choosing a large number of clusters makes the LR models less representative and reliable. To select an appropriate value for the clusters number in the LR model, we conduct several experiments by changing the clusters number. We have found that the performance of the proposed JSR+NS method is stable when the cluster number is greater than 200. The last parameter that should be set is the iterations number. The EC performance of the proposed JSR+NS algorithm is monotonically improved by increasing the iteration number. In our implementation, when the difference of the PSNR value of the reconstructed image in successive iteration falls below a predefined threshold ($\epsilon = 0.005$), the recovery algorithm is stopped. All above-mentioned parameters are fixed for all the experiments throughout this paper.

C. EC Quality

The performance of the proposed JSR+NL algorithm is compared with those of other state-of-the-art EC techniques, including non-normative spatial EC for H.264 (AVC) [33], EC using projections onto convex sets (POCS) [34], content adaptive technique (CAD) [36], edge recovery technique

TABLE I
EFFECT OF DICTIONARY SIZE ON THE EC QUALITY USING THE JSR-BASED EC ALGORITHM IN TERMS OF PSNR (T IS SET TO 5)

| | Dictionary Size D | | | | |
|-----------------|---------------------|-------|-------|-------|-------|
| | 64 | 128 | 256 | 512 | 1024 |
| Lena | 34.59 | 34.90 | 34.92 | 34.76 | 34.68 |
| Peppers | 34.52 | 34.98 | 34.97 | 34.86 | 34.80 |
| Goldhill | 32.53 | 32.71 | 32.67 | 32.64 | 32.60 |
| Mandrill | 26.95 | 27.02 | 27.02 | 27.00 | 27.00 |
| Zelda | 37.23 | 37.49 | 37.51 | 37.39 | 37.36 |

TABLE II
EFFECT OF PATCH SIZE ON THE EC QUALITY USING THE JSR-BASED EC ALGORITHM IN TERMS OF PSNR (BEST DICTIONARY SIZE IS OBTAINED FOR EACH PATCH SIZE)

| | Patch Size T | | | | |
|-----------------|----------------|-------|-------|-------|-------|
| | 16 | 25 | 36 | 49 | 64 |
| Lena | 34.57 | 34.92 | 34.97 | 35.05 | 35.07 |
| Peppers | 37.72 | 34.98 | 35.05 | 35.17 | 35.19 |
| Goldhill | 32.55 | 32.67 | 32.69 | 32.73 | 32.80 |
| Mandrill | 26.96 | 27.01 | 27.02 | 27.05 | 27.08 |
| Mandrill | 37.31 | 37.51 | 37.52 | 37.59 | 37.63 |

based on visual clearness (VC) [35], Markov random fields approach (MRF) [40], multivariate kernel density estimation (MKDE) [41], sparse linear prediction (SLP) [38], learning sparse representation-based EC (LSR) [11], and adaptive linear prediction (ALP) based EC [39]⁵. The performance of the proposed method is also compared with one state-of-the-art inpainting (INP) based EC method proposed in [57]. All of the competing algorithms use a similar recovery order to fill the lost block. Some of these algorithms, such as AVC, POCS, CAD, and MRF recover the lost blocks pixel by pixel; the others recover the lost block patch by patch, similar to our approach. In order to see the effect of LR and NS models on the EC performance, the PSNR values of the reconstructed image using the JSR-based EC algorithm are also reported.

Tables III and IV provide the EC results of all the methods for the test images corrupted by different loss scenarios (isolated loss, consecutive loss, and random loss with 30% PLR) with the lost blocks of size 8×8 and 16×16 pixels, respectively. The best values are marked in bold. Three different modes of the proposed algorithm are evaluated in Tables III and IV: 1- recovery without employing the NS and LR models (JSR), 2- recovery by using only the NS model (JSR-N), and 3- recovery by both the NS and LR models (JSR-NL). It can be observed from Tables III and IV that the JSR-NL algorithm, by employing both the NS and LR models, performs better than the JSR-N and JSR algorithms. The reason is that the JSR-NL algorithm uses the most available information within the image to obtain an accurate estimation of the original patch.

In terms of PSNR, the performance of the proposed JSR-based EC algorithm is better than most of the competing methods, especially in the consecutive and random loss pat-

⁵The source code of all methods have been kindly provided by their authors. Also, the MATLAB implementation of some techniques are available online at <http://dtstc.ugr.es/jkoloda/download.html>

terns. The EC performance can be further improved by the proposed JSR-N and JSR+NL algorithms. As can be seen in Tables III and IV, the EC results of competing methods are significantly lower than those provided by the proposed JSR+NL algorithm for the images such as *Lena* and *Peppers*, which demonstrate its effectiveness for the error recovery by exploiting jointly the non-local self-similarity and local structural regularity. We do notice that for the images like *Barbara* and *Mandrill*, the proposed JSR+NL algorithm does not seem to outperform others. This is partially due to its complex local texture, making it difficult to find a reliable support area of the corrupted pixels. This type of texture calls for a new definition of the support area which is our future work. Generally, we can infer that the proposed method can recover the lost blocks of the images wherein there are global object contours. Although, the performance of the JSR+NL algorithm for the lost blocks of size 16×16 pixels is lower than that of the ALP method in terms of PSNR, as discussed later, the JSR+LN algorithm is much faster than the ALP algorithm.

Several concealed images *Lena*, reconstructed by different EC methods, are shown in Fig. 8 for visual comparisons. The images are corrupted by the random loss pattern (30% PLR). As it can be seen, the CAD algorithm fails to recover the lost blocks and the AVC and POCS cannot restore the lost blocks well. In comparison, the MRF, MKDE and SLP can preserve more image details. However, these algorithms blur the image edges. In contrast, the ALP algorithm is more effective in reconstruction of both the smooth area and the complex regions, including texture and edges. Similar to the ALP algorithm, the proposed JSR+NL method achieves much better results than the others. Evidently, it can preserve most of image details and sharper image edges and generates much less artifacts, leading to visually much more pleasant recovery. It can also be observed that the JSR-based EC scheme can well reconstruct the image. However, there are some artifacts around the edges.

The superior performance of the proposed JSR+NL algorithm comes from both the joint sparse representation modeling and the natural image priors of the non-local self-similarity and the local structural regularity. By analyzing the objective and subjective results, the following observations are made: 1) In general, the JSR-based EC approach performs better as compared to the interpolation based methods like SLP and statistical based methods like MKDE, although its performance is comparable with that of ALP. 2) By incorporating the local and non-local models into the recovery phase, the image quality can be further improved. Using the NS and LR models lead to the consistent good performance.

D. Computational Costs

The run time of the proposed JSR+NL algorithm is compared with the state-of-the-art EC methods in Table V for the random loss pattern (PLR=30%) with the lost blocks of size 8×8 and 16×16 pixels on a typical computer (Intel(R) Xeon(R) CPU @ 3.20 GHz 8 GB RAM) based on a non-optimized MATLAB implementation. A similar random loss pattern is used for all the above-mentioned algorithms and the average run time over 10 trials is given for each algorithm. It

TABLE III
PSNR VALUES OF THE CONCEALED IMAGES USING SEVERAL EC TECHNIQUES FOR DIFFERENT LOSS PATTERNS (ISOLATED LOSS, CONSECUTIVE LOSS AND RANDOM LOSS WITH 30% PLR) WITH THE LOST BLOCKS OF SIZE 8×8

| Loss Pattern | EC Technique | | | | | | | | | | | | |
|-----------------|--------------|--------|-------|-------|-------|--------------|-------|-------|-------|--------------|-------|-------|--------------|
| | AVC | POCS | CAD | VC | MRF | MKDE | INP | SLP | LSR | ALP | JSR | JSR+N | JSR+NL |
| Lena | | | | | | | | | | | | | |
| Isolated | 32.04 | 29.15 | 33.97 | 34.58 | 34.38 | 34.55 | 30.55 | 33.72 | 34.45 | 35.69 | 35.08 | 35.67 | 35.78 |
| Consecutive | 28.84 | 26.21 | 27.43 | 22.83 | 31.09 | 30.57 | 28.22 | 29.48 | 30.13 | 32.14 | 31.80 | 32.38 | 32.54 |
| Random | 28.92 | 26.94 | 26.45 | 18.18 | 31.55 | 31.45 | 29.60 | 30.62 | 31.35 | 32.61 | 31.71 | 32.04 | 32.20 |
| Peppers | | | | | | | | | | | | | |
| Isolated | 32.77 | 28.92 | 34.70 | 34.45 | 34.42 | 35.30 | 31.12 | 34.68 | 34.91 | 35.72 | 35.15 | 35.87 | 35.96 |
| Consecutive | 29.59 | 25.78 | 28.75 | 23.20 | 31.37 | 30.57 | 27.99 | 29.77 | 30.67 | 32.36 | 32.10 | 32.87 | 33.04 |
| Random | 28.98 | 26.70 | 26.97 | 18.40 | 31.44 | 30.95 | 28.56 | 30.23 | 31.25 | 32.29 | 31.52 | 31.86 | 32.01 |
| Goldhill | | | | | | | | | | | | | |
| Isolated | 32.54 | 29.38 | 32.72 | 32.45 | 32.34 | 32.99 | 29.88 | 32.46 | 32.40 | 33.24 | 32.77 | 33.16 | 33.31 |
| Consecutive | 29.43 | 26.44 | 29.04 | 24.34 | 29.32 | 28.99 | 27.01 | 28.49 | 29.11 | 30.12 | 29.73 | 30.11 | 30.23 |
| Random | 29.80 | 27.77 | 28.28 | 18.77 | 30.22 | 30.03 | 28.22 | 29.46 | 30.15 | 30.89 | 30.55 | 30.70 | 30.99 |
| Mandrill | | | | | | | | | | | | | |
| Isolated | 26.18 | 24.89 | 26.36 | 26.84 | 26.89 | 26.71 | 24.01 | 25.05 | 26.78 | 27.44 | 27.07 | 27.16 | 27.24 |
| Consecutive | 23.19 | 22.17 | 22.36 | 20.72 | 23.88 | 23.34 | 20.56 | 21.73 | 23.82 | 24.33 | 24.04 | 24.14 | 24.22 |
| Random | 24.83 | 24.09 | 22.10 | 17.55 | 25.57 | 25.18 | 22.87 | 23.62 | 25.23 | 25.87 | 25.86 | 25.90 | 25.95 |
| Zelda | | | | | | | | | | | | | |
| Isolated | 36.41 | 32.76 | 37.25 | 37.40 | 37.13 | 37.06 | 33.66 | 36.67 | 36.81 | 38.87 | 37.66 | 38.25 | 38.38 |
| Consecutive | 33.46 | 29.54 | 32.73 | 26.15 | 34.30 | 33.40 | 30.25 | 32.79 | 32.38 | 35.68 | 34.86 | 35.40 | 35.57 |
| Random | 32.95 | 30.47 | 31.76 | 20.65 | 34.40 | 33.42 | 31.02 | 32.95 | 33.64 | 35.51 | 35.16 | 35.60 | 35.77 |
| Barbara | | | | | | | | | | | | | |
| Isolated | 27.56 | 26.02 | 30.27 | 28.81 | 30.20 | 34.00 | 30.14 | 33.12 | 29.31 | 33.66 | 30.22 | 31.04 | 30.58 |
| Consecutive | 24.83 | 23.03 | 24.09 | 21.60 | 27.11 | 29.74 | 26.76 | 28.44 | 25.88 | 29.76 | 27.19 | 28.22 | 27.54 |
| Random | 25.99 | 27.78 | 22.85 | 18.41 | 28.15 | 30.03 | 27.10 | 28.60 | 27.58 | 30.22 | 28.13 | 28.62 | 28.29 |
| Montreal | | | | | | | | | | | | | |
| Isolated | 31.15 | 28.73 | 30.72 | 30.85 | 30.69 | 30.98 | 28.10 | 30.12 | 31.09 | 31.62 | 31.02 | 31.32 | 31.41 |
| Consecutive | 27.84 | 25.78 | 27.73 | 24.94 | 27.51 | 27.10 | 24.80 | 26.30 | 27.18 | 28.09 | 27.80 | 27.99 | 28.10 |
| Random | 28.88 | 27.39 | 27.81 | 21.62 | 28.85 | 28.34 | 26.10 | 27.53 | 28.81 | 29.19 | 28.98 | 29.05 | 29.22 |
| Pelican | | | | | | | | | | | | | |
| Isolated | 30.43 | 28.27 | 31.54 | 31.81 | 31.85 | 31.75 | 28.55 | 30.47 | 30.48 | 32.57 | 32.18 | 32.30 | 32.39 |
| Consecutive | 27.43 | 25.28 | 26.43 | 22.85 | 28.86 | 28.24 | 25.32 | 26.86 | 26.90 | 29.38 | 29.18 | 29.29 | 29.39 |
| Random | 28.99 | 27.22 | 25.82 | 16.02 | 30.60 | 30.00 | 27.12 | 28.76 | 30.42 | 31.20 | 30.78 | 30.84 | 31.32 |
| Kodim05 | | | | | | | | | | | | | |
| Isolated | 26.66 | 25.30 | 28.18 | 28.31 | 28.11 | 28.23 | 25.88 | 27.32 | 27.62 | 28.74 | 28.58 | 28.99 | 29.12 |
| Consecutive | 23.59 | 22.43 | 23.56 | 22.48 | 25.00 | 24.39 | 22.30 | 23.34 | 23.87 | 25.38 | 25.43 | 25.83 | 25.97 |
| Random | 24.64 | 23.75 | 23.11 | 20.21 | 25.96 | 25.24 | 22.90 | 24.34 | 26.10 | 25.86 | 26.19 | 26.37 | 26.44 |
| Kodim06 | | | | | | | | | | | | | |
| Isolated | 29.68 | 27.44 | 30.18 | 30.15 | 29.86 | 30.25 | 27.65 | 29.08 | 30.63 | 30.98 | 30.15 | 30.43 | 30.53 |
| Consecutive | 26.61 | 24.60 | 26.52 | - | 26.84 | 26.43 | 24.22 | 25.42 | 26.30 | 27.61 | 27.10 | 27.37 | 27.46 |
| Random | 28.31 | 26.58 | 25.98 | - | 28.45 | 28.34 | 26.01 | 27.31 | 28.18 | 28.98 | 28.58 | 28.67 | 28.75 |
| Average | | | | | | | | | | | | | |
| Isolated | 30.54 | 28.086 | 31.58 | 31.56 | 31.58 | 32.18 | 28.95 | 31.26 | 31.44 | 32.85 | 31.98 | 32.41 | 32.47 |
| Consecutive | 27.48 | 25.12 | 26.86 | 23.51 | 28.52 | 28.27 | 25.74 | 27.26 | 27.62 | 29.48 | 28.92 | 29.36 | 29.40 |
| Random | 28.22 | 26.86 | 26.11 | 19.48 | 29.51 | 29.29 | 26.95 | 28.34 | 29.27 | 30.26 | 29.74 | 29.96 | 30.09 |

can be seen that the proposed algorithm is much faster than the recently proposed ALP algorithm, which is the best among the competing algorithms. Most competitors recover the lost block in a pixel-by-pixel manner, wherein the pixels in the missing blocks are successively recovered on a small spatial window. However, our approach is block-based, in which all the missing pixels within a subblock (subblock of size $\sqrt{P} \times \sqrt{P}$ pixels) are recovered at the same time (Please see Fig. 2). This leads to a lower reconstruction time. Although, the proposed JSR+NL algorithm has a considerably large error

recovery time in comparison with the AVC, CAD and MRF methods, its advantage in precisely estimation of the corrupted information is obvious in terms of objective evaluations.

It should be noted that the dictionaries and mappings are trained offline and the computational cost of the proposed JSR+NL algorithm is usually dominated by the computation of the sparse representation of the input patches using the OMP algorithm. In the worst case, B^2 vectors should be obtained via Eq. (17) for each lost block of size $B \times B$. The computational cost of the sparse representation is also

TABLE IV
PSNR VALUES OF THE CONCEALED IMAGES USING SEVERAL EC TECHNIQUES FOR DIFFERENT LOSS PATTERNS (ISOLATED LOSS, CONSECUTIVE LOSS AND RANDOM LOSS WITH 30% PLR) WITH THE LOST BLOCKS OF SIZE 16×16

| Loss Pattern | EC Technique | | | | | | | | | | | | |
|-----------------|--------------|-------|-------|-------|-------|--------------|-------|-------|-------|--------------|-------|-------|--------------|
| | AVC | POCS | CAD | VC | MRF | MKDE | INP | SLP | LSR | ALP | JSR | JSR+N | JSR+NL |
| Lena | | | | | | | | | | | | | |
| Isolated | 29.25 | 24.30 | 31.51 | 32.45 | 31.52 | 33.35 | 29.89 | 32.44 | 31.57 | 33.90 | 33.24 | 33.47 | 33.59 |
| Consecutive | 26.26 | 23.76 | 22.64 | 24.85 | 28.44 | 29.12 | 26.78 | 28.12 | 27.86 | 29.86 | 29.32 | 29.50 | 29.63 |
| Random | 25.72 | 24.30 | 19.87 | 17.59 | 28.18 | 28.81 | 26.90 | 28.16 | 28.15 | 29.25 | 29.06 | 29.18 | 29.27 |
| Peppers | | | | | | | | | | | | | |
| Isolated | 29.19 | 23.87 | 30.07 | 30.96 | 30.16 | 31.83 | 29.41 | 31.33 | 30.29 | 31.98 | 31.86 | 32.12 | 32.18 |
| Consecutive | 26.40 | 23.47 | 22.65 | 24.88 | 27.56 | 28.68 | 26.59 | 28.00 | 27.14 | 29.49 | 28.88 | 29.14 | 29.25 |
| Random | 26.41 | 23.87 | 20.86 | 18.43 | 27.01 | 28.05 | 25.42 | 26.67 | 27.00 | 28.36 | 27.66 | 27.83 | 27.93 |
| Goldhill | | | | | | | | | | | | | |
| Isolated | 30.47 | 25.71 | 28.69 | 30.92 | 30.31 | 31.46 | 28.99 | 30.88 | 30.25 | 31.27 | 31.45 | 31.49 | 31.54 |
| Consecutive | 27.32 | 24.66 | 22.65 | 26.17 | 27.03 | 27.89 | 26.10 | 27.43 | 27.03 | 27.97 | 29.01 | 28.06 | 28.08 |
| Random | 27.55 | 25.71 | 21.39 | 18.82 | 27.55 | 27.23 | 25.01 | 26.53 | 27.55 | 27.70 | 27.77 | 27.82 | 27.86 |
| Mandrill | | | | | | | | | | | | | |
| Isolated | 25.40 | 23.86 | 24.82 | 26.01 | 26.04 | 26.33 | 23.50 | 24.79 | 26.05 | 26.42 | 26.26 | 26.27 | 26.29 |
| Consecutive | 22.57 | 21.98 | 19.40 | 21.81 | 23.29 | 23.22 | 21.20 | 21.97 | 23.08 | 23.49 | 23.35 | 23.37 | 23.38 |
| Random | 24.58 | 23.86 | 19.67 | 17.34 | 25.18 | 25.01 | 22.65 | 23.84 | 25.00 | 25.37 | 25.23 | 25.25 | 25.26 |
| Zelda | | | | | | | | | | | | | |
| Isolated | 33.81 | 27.32 | 33.71 | 35.13 | 34.32 | 35.11 | 30.10 | 34.70 | 34.93 | 36.38 | 35.46 | 35.73 | 35.83 |
| Consecutive | 30.67 | 27.00 | 23.84 | 28.16 | 31.07 | 31.47 | 29.67 | 31.11 | 30.38 | 32.84 | 31.99 | 32.30 | 32.41 |
| Random | 29.74 | 27.32 | 23.55 | 20.68 | 29.92 | 31.15 | 29.32 | 30.81 | 30.00 | 31.36 | 31.25 | 31.36 | 31.48 |
| Barbara | | | | | | | | | | | | | |
| Isolated | 27.09 | 23.21 | 28.00 | 28.58 | 28.46 | 32.35 | 29.43 | 31.56 | 28.43 | 31.46 | 30.96 | 30.98 | 30.99 |
| Consecutive | 23.94 | 21.54 | 21.20 | 23.13 | 25.21 | 28.38 | 26.03 | 27.62 | 24.94 | 27.84 | 27.38 | 27.39 | 27.39 |
| Random | 24.80 | 23.21 | 20.16 | 17.91 | 26.01 | 28.05 | 25.65 | 26.88 | 26.01 | 27.69 | 27.46 | 27.47 | 27.47 |
| Montreal | | | | | | | | | | | | | |
| Isolated | 29.86 | 26.63 | 27.69 | 29.66 | 29.52 | 30.36 | 28.22 | 29.82 | 31.05 | 30.26 | 30.50 | 30.53 | 30.55 |
| Consecutive | 26.84 | 25.04 | 22.85 | 25.84 | 26.38 | 26.79 | 25.07 | 26.39 | 26.19 | 27.00 | 27.14 | 27.18 | 27.19 |
| Random | 28.09 | 26.63 | 24.18 | 22.06 | 27.49 | 27.66 | 25.88 | 27.27 | 27.42 | 27.61 | 28.16 | 28.20 | 28.22 |
| Pelican | | | | | | | | | | | | | |
| Isolated | 28.89 | 24.92 | 30.28 | 30.95 | 30.39 | 30.53 | 27.76 | 29.09 | 30.48 | 31.50 | 30.57 | 30.60 | 30.62 |
| Consecutive | 25.78 | 23.86 | 22.03 | 24.87 | 27.45 | 27.52 | 25.00 | 26.46 | 26.79 | 28.36 | 27.82 | 27.87 | 27.92 |
| Random | 26.01 | 24.92 | 20.71 | 16.16 | 27.78 | 28.00 | 25.78 | 27.04 | 27.05 | 28.58 | 28.30 | 28.33 | 28.34 |
| Kodim05 | | | | | | | | | | | | | |
| Isolated | 25.44 | 22.79 | 25.21 | 26.79 | 26.34 | 27.47 | 25.22 | 26.75 | 26.23 | 27.41 | 27.60 | 27.68 | 27.71 |
| Consecutive | 22.16 | 21.24 | 20.17 | 22.46 | 23.23 | 23.67 | 21.87 | 22.77 | 23.35 | 23.89 | 23.91 | 24.00 | 24.05 |
| Random | 23.52 | 22.79 | 20.99 | 20.17 | 24.48 | 24.28 | 22.55 | 23.63 | 24.25 | 24.79 | 24.90 | 24.94 | 24.97 |
| Kodim06 | | | | | | | | | | | | | |
| Isolated | 28.82 | 25.73 | 28.32 | 29.43 | 28.82 | 29.18 | 26.89 | 28.45 | 28.83 | 29.57 | 29.47 | 29.52 | 29.55 |
| Consecutive | 25.64 | 24.55 | 24.58 | 23.84 | 25.69 | 25.84 | 23.65 | 25.11 | 25.57 | 26.32 | 26.20 | 26.25 | 26.28 |
| Random | 26.68 | 25.73 | 21.27 | 18.07 | 26.04 | 26.99 | 24.84 | 26.20 | 26.40 | 27.13 | 27.09 | 24.10 | 27.14 |
| Average | | | | | | | | | | | | | |
| Isolated | 28.82 | 24.83 | 28.83 | 30.09 | 29.59 | 30.80 | 27.94 | 29.98 | 29.81 | 31.01 | 30.74 | 30.84 | 30.89 |
| Consecutive | 25.76 | 23.71 | 22.20 | 24.60 | 26.53 | 27.26 | 25.20 | 26.50 | 26.23 | 27.71 | 27.50 | 27.51 | 27.56 |
| Random | 26.31 | 24.83 | 21.27 | 18.72 | 26.96 | 27.52 | 25.40 | 26.70 | 26.88 | 27.78 | 27.69 | 27.45 | 27.49 |

proportional to the dictionary and patch sizes. As discussed before, we set $D = 256$ and $T = 25$ to balance the EC quality and the reconstruction time. Further, the NS model adds significant computational costs which comes from finding the similar patches for the NS model for each patch. It should be taken into consideration that, given the recovery order of each missing block, (see Fig. 4), the reconstruction process can be done by starting the recovery of the subblocks located at 4 corners of the lost block at the same time. Therefore, the reconstruction of block can be done four times faster.

Moreover, in our implementation, similar patches for each block are not updated at each iteration, but after 100 iterations, thus reducing the reconstruction time significantly.

Our approach can be modified in order to reduce the recovery time as follows:

- When building patch y , one can consider more corrupted pixels P in the patch to be recovered. This approach reduces the number of vectors to be recovered for each lost block. In this paper we consider $P = 4$. It is also possible to decrease the overlap depth between the



Fig. 8. Subjective comparison between the proposed JSR+NL results and those of other EC techniques for random loss patterns with 30% PLR. Top to Down: *Lena*, *Montreal*, *Kodim06*.

adjacent subblocks (see Section III-D).

- The algorithm could be optimized by using a fast NS algorithm [58]. Further, for faster computation of the LR

models, one can obtain one LR model for all the pixels in the input patch, instead of finding different LR models for each pixel.

TABLE V
RECONSTRUCTION TIME (IN SECONDS) FOR THE CONCEALED IMAGE *Lena* USING SEVERAL EC TECHNIQUES FOR DIFFERENT LOSS PATTERNS (ISOLATED LOSS, CONSECUTIVE LOSS AND RANDOM LOSS WITH 30% PLR)

| Loss Pattern | EC Technique | | | | | | | | | | | |
|-----------------------------|--------------|-------|-------|------|-------|------|-------|-----|-----|-----|-----|--------|
| | AVC | POCS | CAD | VC | MRF | MKDE | INP | SLP | LSR | ALP | JSR | JSR+NL |
| Block Size (8 × 8) | | | | | | | | | | | | |
| Isolated | 0.09 | 6.07 | 4.10 | 559 | 9.23 | 236 | 10987 | 82 | 64 | 158 | 22 | 39 |
| Consecutive | 0.20 | 8.70 | 5.22 | 1079 | 16.73 | 363 | 21876 | 126 | 126 | 281 | 41 | 58 |
| Random | 0.15 | 5.83 | 3.90 | 586 | 10.34 | 253 | 13654 | 85 | 75 | 171 | 25 | 42 |
| Block Size (16 × 16) | | | | | | | | | | | | |
| Isolated | 0.09 | 7.45 | 5.98 | 2400 | 9.23 | 1189 | 41850 | 256 | 269 | 510 | 110 | 150 |
| Consecutive | 0.06 | 11.75 | 8.74 | 4010 | 16.63 | 1899 | 77980 | 406 | 510 | 910 | 223 | 300 |
| Random | 0.05 | 7.97 | 10.15 | 2480 | 10.62 | 1278 | 59876 | 301 | 310 | 580 | 130 | 183 |

TABLE VI
EFFECT OF MAPPING APPROACH ON THE EC PERFORMANCE (PSNR) FOR THE IMAGE *Lena* AT DIFFERENT PLRS

| Mapping | PLR (%) | | | |
|--------------|---------|-------|-------|-------|
| | 10 | 20 | 30 | 50 |
| JSR-C | 38.41 | 34.96 | 32.31 | 30.13 |
| JSR-D | 38.31 | 34.87 | 32.23 | 30.05 |
| JSR-I | 37.35 | 33.88 | 31.16 | 29.24 |

- The computational burden of the recovery process is substantially increased using the iterative shrinkage algorithm [55]. For decreasing the complexity, the receiver can early terminate the iterations, depending on its computational power, and yet obtain a visually satisfactory image reconstruction.
- The proposed method can be easily executed in parallel manner and obtain an acceptable computational cost using an array of multiple instructions multiple data (MIMD)-based parallel processors. Further, the difficulty in hardware implementation can be effectively handled using a CMOS technology based architecture [59] and learning a dictionary with binary or ternary values instead of a real ones [60], [61].

Finally, it should be noted that, in this paper, we use the algorithm in [40] for the initial recovery which gives us an acceptable reconstruction time for the JSRNL algorithm. This initial interpolation simplifies much of the subsequent work without a computational cost. How to choose the initial recovery algorithm to guarantee a low computational cost and how to make the algorithm less sensitive to the initial reconstruction, and more generally, to noise, will be part of our future research.

E. Performance Analysis of the Proposed Algorithm

The proposed method offers several benefits. Firstly, learning a separate dictionary for each domain preserves the main structure of the data in the both domains, which can be represented well by sparse linear combinations of the dictionary atoms. Further, learning a mapping function in the sparse representation domain makes it more accurate since the irrelevant information in the spatial domain is discarded. We have introduced several mapping approaches in Section III-B which relate the sparse representations of the original and

corrupted patches in different ways. In this section, the effect of the employed common space mapping, defined in Eq. (4), on the EC performance is evaluated in comparison with the direct mapping, defined by the term (3). We also evaluate the EC performance of the proposed coupling method in [11], wherein the mapping matrix \mathbf{M} in Eq. (3) is set as an identity matrix.

Table VI reports the experimental results in terms of PSNR, where the image *Lena* is corrupted with the random loss pattern at different PLRs, ranging from 10% until 50%. The image is concealed using the JSR-based EC algorithm with above coupling terms. “JSR-I”, “JSR-D”, and “JSR-C” means mapping with the identity matrix, direct mapping, and common space mapping, respectively. It can be observed that coupling of the sparse coefficients using a common space performs better error recovery than other coupling methods. The reason is that transferring the sparse representations into a common space provides more freedom to uncover the relationships between the sparse coefficients of the original and corrupted patches. It means that using an accurate coupling term in the objective function (5) plays an important role in the enhancement of error recovery.

VI. CONCLUSION

In this paper, a new image EC has been developed by integrating the LR and the NS properties of natural images into the JSR. The JSR model estimates the corrupted patch via a dictionary pair and two mapping matrices that are trained offline from two given training datasets. By using this model, the error concealment is achieved by transferring the error recovery problem into a common space via the two learned dictionaries and mappings. Such transformation provides more freedom and flexibility for error concealment. Incorporating this model with the local and non-local priors, as a new regularization term in the sparse representation domain, produces sharper edges and suppresses the visual artifacts. The performance of the proposed method has been evaluated and compared with those of the state-of-the-art methods, both quantitatively and perceptually.

REFERENCES

[1] A. Akbari, M. Trocan, and B. Granado, “Sparse recovery-based error concealment,” *IEEE Transactions on Multimedia*, vol. 19, no. 6, pp. 1339–1350, June 2017.

- [2] D. Song, L. Cao, and C. W. Chen, "Robust multiple description image coding over wireless networks based on wavelet tree coding, error resilient entropy coding, and error concealment," *Journal of Visual Communication and Image Representation*, vol. 19, no. 5, pp. 311–319, July 2008.
- [3] A. Akbari and M. Trocan, "Sparse recovery-based error concealment for multiview images," in *Proceedings of IEEE Workshop on Computational Intelligence for Multimedia Understanding (IWCIM)*, Prague, Czech Republic, Oct. 2015, pp. 1–5.
- [4] M. Kim, H. Lee, and S. Sull, "Spatial error concealment for H.264 using sequential directional interpolation," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 4, pp. 1811–1818, Nov. 2008.
- [5] A. Akbari, M. Trocan, and B. Granado, "Image compression using adaptive sparse representations over trained dictionaries," in *Proceedings of IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Montreal, Canada, Sep. 2016, pp. 1–6.
- [6] A. Akbari and M. Trocan, "Downsampling based image coding using dual dictionary learning and sparse representations," in *Proceedings of IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Vancouver, Canada, Aug. 2018, pp. 1–6.
- [7] A. Akbari, M. Trocan, and B. Granado, "Residual based compressed sensing recovery using sparse representations over a trained dictionary," in *Proceedings of International ITG Conference on Systems, Communications and Coding (SCC)*, Hamburg, Germany, Feb. 2017, pp. 1–6.
- [8] A. Akbari, D. Mandache, M. Trocan, and B. Granado, "Adaptive saliency-based compressive sensing image reconstruction," in *Proceedings of IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, Seattle, WA, July 2016, pp. 1–6.
- [9] G. Peyre, "A review of adaptive image representations," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 896–911, Sep. 2011.
- [10] R. Rubinstein, A. M. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1045–1057, June 2010.
- [11] A. Akbari, M. Trocan, and B. Granado, "Image error concealment using sparse representations over a trained dictionary," in *Proceedings of IEEE Picture Coding Symposium (PCS)*, Nuremberg, Germany, Dec. 2016.
- [12] —, "Joint-domain dictionary learning based error concealment using common space mapping," in *Proceedings of IEEE Conference on Digital Signal Processing (DSP)*, London, UK, Aug. 2017.
- [13] —, "Image error concealment based on joint sparse representation and non-local similarity," *Proceedings of IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Nov. 2017.
- [14] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3467–3478, Aug. 2012.
- [15] S. Wang, L. Zhang, Y. Liang, and Q. Pan, "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, June 2012, pp. 2216–2223.
- [16] D. A. Huang and Y. C. F. Wang, "Coupled dictionary and feature space learning with applications to cross-domain image synthesis and recognition," in *Proceedings of IEEE Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013, pp. 2496–2503.
- [17] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *proceedings of IEEE Conference on Computer Vision (ICCV)*, Kyoto, Japan, Sep. 2009, pp. 2272–2279.
- [18] J. Jiang, X. Ma, C. Chen, T. Lu, Z. Wang, and J. Ma, "Single image super-resolution via locally regularized anchored neighborhood regression and nonlocal means," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 15–26, Jan. 2017.
- [19] K. Zhang, D. Tao, X. Gao, X. Li, and J. Li, "Coarse-to-fine learning for single-image super-resolution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 5, pp. 1109–1122, May 2017.
- [20] K. Zhang, J. Li, H. Wang, X. Liu, and X. Gao, "Learning local dictionaries and similarity structures for single image super-resolution," *Signal Processing*, vol. 142, pp. 231 – 243, Jan. 2018.
- [21] K. Zhang, X. Gao, J. Li, and H. Xia, "Single image super-resolution using regularization of non-local steering kernel regression," *Signal Processing*, vol. 123, pp. 53 – 63, June 2016.
- [22] W. Dong, L. Zhang, G. Shi, and X. Wu, "Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization," *IEEE Transactions on Image Processing*, vol. 20, no. 7, pp. 1838–1857, July 2011.
- [23] X. Zhang and X. Wu, "Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation," *IEEE Transactions on Image Processing*, vol. 17, no. 6, pp. 887–896, June 2008.
- [24] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, June 2010.
- [25] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [26] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *Journal of Multiscale Modeling & Simulation*, vol. 7, no. 1, pp. 214–241, 2008.
- [27] M. Elad, M. A. T. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 972–982, June 2010.
- [28] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
- [29] S. Shekhar, V. M. Patel, H. V. Nguyen, and R. Chellappa, "Coupled projections for adaptation of dictionaries," *IEEE Transactions on Image Processing*, vol. 24, no. 10, pp. 2941–2954, Oct. 2015.
- [30] D. Mandal and S. Biswas, "Generalized coupled dictionary learning approach with applications to cross-modal matching," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3826–3837, Aug. 2016.
- [31] M. Gong, P. Zhang, L. Su, and J. Liu, "Coupled dictionary learning for change detection from multisource data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7077–7091, Dec. 2016.
- [32] J. Liu, M. Shah, B. Kuijpers, and S. Savarese, "Cross-view action recognition via view knowledge transfer," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, CO, June 2011, pp. 3209–3216.
- [33] V. Varsa, M. M. Hannuksela, and Y.-K. Wang, "Non-normative error concealment algorithms," *ITU-T SG16, VCEG-N62*, vol. 50, Sep. 2001.
- [34] H. Sun and W. Kwok, "Concealment of damaged block transform coded images using projections onto convex sets," *IEEE Transactions on Image Processing*, vol. 4, no. 4, pp. 470–477, Apr. 1995.
- [35] J. Koloda, V. Sánchez, and A. M. Peinado, "Spatial error concealment based on edge visual clearness for image/video communication," *Journal of Circuits, System, and Signal Processing*, vol. 32, no. 2, pp. 815–824, Apr. 2013.
- [36] Z. Rongfu, Z. Yuanhua, and H. Xiaodong, "Content-adaptive spatial error concealment for video communication," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 335–341, Feb. 2004.
- [37] X. Li and M. T. Orchard, "Novel sequential error-concealment techniques using orientation adaptive interpolation," *IEEE Transactions on Circuits System and Video Technology*, vol. 12, no. 10, pp. 857–864, Oct. 2002.
- [38] J. Koloda, J. Ostergaard, S. H. Jensen, V. Sanchez, and A. M. Peinado, "Sequential error concealment for video/images by sparse linear prediction," *IEEE Transactions on Multimedia*, vol. 15, no. 4, pp. 957–969, June 2013.
- [39] J. Liu, G. Zhai, X. Yang, B. Yang, and L. Chen, "Spatial error concealment with an adaptive linear predictor," *IEEE Transactions on Circuits System and Video Technology*, vol. 25, no. 3, pp. 353–366, Mar. 2015.
- [40] S. Shirani, F. Kossentini, and R. Ward, "An adaptive markov random field based error concealment method for video communication in an error prone environment," in *Proceedings of IEEE Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Phoenix, AZ, Mar. 1999, pp. 3117–3120.
- [41] J. Koloda, A. Peinado, and V. Sanchez, "Kernel-based MMSE multimedia signal reconstruction and its application to spatial error concealment," *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp. 1729–1738, June 2014.
- [42] G. Zhai, J. Cai, W. Lin, X. Yang, and W. Zhang, "Image error-concealment via block-based bilateral filtering," in *Proceedings of IEEE Conference on Multimedia Expo (ICME)*, Hannover, Germany, Apr. 2008, pp. 621–624.
- [43] G. Zhai, X. Yang, W. Lin, and W. Zhang, "Bayesian error concealment with dct pyramid for images," *IEEE Transactions on Circuits System and Video Technology*, vol. 20, no. 9, pp. 1224–1232, Sep. 2010.
- [44] X. Liu, D. Zhai, J. Zhou, S. Wang, D. Zhao, and H. Gao, "Sparsity-based image error concealment via adaptive dual dictionary learning and regularization," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 782–796, Feb. 2017.
- [45] D. Zhai, X. Liu, J. Zhou, D. Zhao, and W. Gao, "Spatial error concealment via model based coupled sparse representation," in *Proceeding of*

IEEE Conference on Multimedia and Expo Workshops (ICMEW), July 2013, pp. 1–4.

- [46] D. A. Huang and Y. C. F. Wang, “Coupled dictionary and feature space learning with applications to cross-domain image synthesis and recognition,” in *Proceedings of IEEE Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013, pp. 2496–2503.
- [47] K. Wang, R. He, W. Wang, L. Wang, and T. Tan, “Learning coupled feature spaces for cross-modal matching,” in *Proceedings of IEEE Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013, pp. 2088–2095.
- [48] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, “Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition,” in *Proceedings of IEEE Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Nov. 1993, pp. 40–44.
- [49] H. H. Chen and J. J. Ding, “Nonlocal context modeling and adaptive prediction for lossless image coding,” in *Proceedings of IEEE Picture Coding Symposium (PCS)*, San Jose, CA, Dec. 2013, pp. 133–136.
- [50] L. F. R. Lucas, N. M. M. Rodrigues, E. A. B. da Silva, C. L. Pagliari, and S. M. M. de Faria, “Image coding using generalized predictors based on sparsity and geometric transformations,” *IEEE Transactions on Image Processing*, vol. 25, no. 9, pp. 4046–4060, Sep. 2016.
- [51] W. Dong, L. Zhang, G. Shi, and X. Li, “Nonlocally centralized sparse representation for image restoration,” *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1620–1630, Apr. 2013.
- [52] A. Buades, B. Coll, and J. M. Morel, “A non-local algorithm for image denoising,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, San Diego, CA, June 2005, pp. 60–65.
- [53] X. Wu, X. Zhang, and J. Wang, “Model-guided adaptive recovery of compressive sensing,” in *Proceedings of Data Compression Conference (DCC)*, UT, USA, Mar. 2009, pp. 123–132.
- [54] J. A. Hartigan and M. A. Wong, “Algorithm AS 136: A k-means clustering algorithm,” *Journal of Applied Statistics*, vol. 28, no. 1, pp. 100–108, 1979.
- [55] I. Daubechies, M. DeFrise, and C. DeMol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint,” *Journal of Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, Mar. 2004.
- [56] R. Timofte, R. Rothe, and L. V. Gool, “Seven ways to improve example-based single image super resolution,” in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, June 2016, pp. 1865–1873.
- [57] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin, “Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images,” *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 130–144, Jan. 2012.
- [58] K. Zhang, D. Tao, X. Gao, X. Li, and Z. Xiong, “Learning multiple linear mappings for efficient single image super-resolution,” *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 846–861, Mar. 2015.
- [59] A. Akbari, M. Trevisi, M. Trocan, and R. Carmona-Galán, “Designing a hardware-friendly measurement matrix for compressed sensing of natural images based on cellular automata,” *IEEE Transactions on Circuits and Systems for Video Technology*.
- [60] A. Akbari and M. Trocan, “Robust image reconstruction for block-based compressed sensing using a binary measurement matrix,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Oct 2018, pp. 1832–1836.
- [61] A. Akbari, M. Trevisi, and M. Trocan, “Adaptive compressed sensing image reconstruction using binary measurement matrices,” in *2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, Dec 2018, pp. 659–660.



learning and dictionary learning.



dElectronique de Paris (ISEP). Her current research interests focus on image and video analysis and compression and sparse signal representations.



has served as an Associate Editor for the IEEE Signal Processing Letters, IEEE Signal Processing Magazine, and Journal of Computational Intelligence and Neuroscience.



for biomedical applications, reconfigurable and adaptable hardware, genetic algorithms, and neural network.

Ali Akbari (M’15) received the M.S. degree in Electrical Engineering from the Shiraz University of Technology, Iran in 2012 and the PhD degree in Telecommunications at the Institut suprieur d’lectronique de Paris (ISEP), Sorbonne Universit, Paris, France in March 2018. Since July 2018, he joined the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK as a research fellow to enrich his experiences in the field of face recognition. His research interests include image and video processing, computer vision, deep

Maria Trocan received her M.Eng. in Electrical Engineering and Computer Science from Politehnica University of Bucharest in 2004, the Ph.D. in Signal and Image Processing from Telecom ParisTech in 2007 and the Habilitation to Lead Researches (HDR) from Pierre and Marie Curie University (Paris 6) in 2014. She joined Joost Netherlands in 2007, where she worked as research engineer involved in the design and development of video transcoding systems. Since May 2009 she is firstly Associate Professor, then Professor (2016) at Institut Suprieur d’lectronique de Paris (ISEP). Her current research interests focus on image and video analysis and compression and sparse signal representations.

Saeid Sanei (SM’05) received the Ph.D. degree in signal processing from Imperial College London, London, U.K. He has been a member of academic staff in Iran, Singapore, and the U.K. He has published three monographs, a number of book chapters, and more than 330 papers in peer-reviewed journals and conference proceedings. His research interests include adaptive filtering, cooperative learning, and multiway, multimodal, and multichannel signal processing with applications to biomedical, audio, biometrics, and communication signals and images. He

Bertrand Geranado received the B.S., M.S., and PhD degrees from the University of Paris-Sud, Orsay, France, in 1991, 1994, and 1998, respectively, all in Computer Science. He had successively been assistant and associate professor at the Pierre and Marie Curie University, Paris, France, and a full professor at the National School of Electrical Engineering (ENSEA), Cergy, France, from 2007 and 2012. He is currently a full professor at the Pierre and Marie Curie University since 2012. His main research interests include embedded system design