

Image/video compressive sensing recovery using joint adaptive sparsity measure



Nasser Eslahi, Ali Aghagolzadeh*, Seyed Mehdi Hosseini Andargoli

Department of Electrical and Computer Engineering, Babol University of Technology, Babol, Iran

ARTICLE INFO

Article history:

Received 12 September 2015

Received in revised form

13 March 2016

Accepted 15 March 2016

Available online 28 March 2016

Keywords:

Compressive video sensing

Sparse recovery

Dictionary learning

Nonlocal self-similarity

ABSTRACT

Compressive sensing (CS) is a recently emerging technique and an extensively studied problem in signal and image processing, which enables joint sampling and compression into a unified approach. Recently, local smoothness and nonlocal self-similarity have both led to superior sparsity priors for CS image restoration. In this paper, first, a new sparsity measure called joint adaptive sparsity measure (JASM) is introduced. The proposed JASM enforces both local sparsity and nonlocal 3D sparsity in transform domain, concurrently, providing a powerful mechanism for characterizing the structured sparsities of natural image. More precisely, the local sparsity depicts the local smoothness redundancies exploited by an adaptively learned sparsifying basis, and the nonlocal 3D sparsity corresponds to the nonlocal self-similarity constraint achieved by a new proposed nonlocal statistical sparse modeling. Then, two novel techniques for high-fidelity CS image and video recovery via JASM are proposed. The proposed methods are formulated in the form of minimization functional under regularization-based framework which is solved via an efficient alternating minimization algorithm based on split Bregman framework. Comprehensive experimental results are reported to manifest the effectiveness of the proposed methods compared with the current state-of-the-art methods in CS image/video restoration.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

Due to great efforts by Candès et al. [1,2] and Donoho [3], compressive sensing (CS)—also called compressed sensing or compressive sampling—suggests a new framework for simultaneous sampling and compression of signals at a rate significantly below the Nyquist rate. It also permits that under certain conditions, the original signal can be reconstructed properly from a small set of measurements using sparsity-promoting nonlinear recovery algorithms.

Suppose we wish to recover a real value finite length signal $u \in \mathbb{R}^n$ from a finite length observation $f \in \mathbb{R}^m$ (with $m < n$) and there is a linear projection between them

$$f = \Phi u + e, \quad (1)$$

where $\Phi \in \mathbb{R}^{m \times n}$ is a sensing matrix and $e \in \mathbb{R}^{m \times 1}$ denotes the additive noise. Since the number of unknowns is much more than the observations, clearly we are not able to recover every u from f and it is generally considered as an *ill-posed* problem. However, if u is sufficiently sparse in the sense that it can be written as a

superposition of a small number of vectors taken from a known (sparsifying) transform domain basis ($t = n$) or frame ($t > n$) $\Psi \in \mathbb{R}^{t \times n}$ or even adaptively learned sparsifying (ALS) basis, such that Ψu contains only a small set of significant entries (e.g., $s < m \ll n$ nonzero coefficients), then the exact recovery of u is possible. In order to solve the reconstruction problem with a reasonable accuracy and robustness to the noise, the estimation of u is formulated as an unconstrained Lagrangian optimization problem as:

$$\min_u \left\{ \frac{1}{2} \|f - \Phi u\|_{\ell_2}^2 + \lambda \mathcal{R}(u) \right\}. \quad (2)$$

The term $\mathcal{R}(u)$ could be various choices, e.g., $\|u\|_{\ell_p}$, $\|\Psi u\|_{\ell_p}$ where $\ell_p \in \{\ell_0, \ell_1\}$, total variation $TV(u)$ [4] or Bregman distance [5]. The optimization problem given in Eq. (2) incorporates the prior information about the original signal. The first term in Eq. (2) is a penalty that represents the closeness of the solution to the observed scene and quantifies the “prediction error” with respect to the measurements. The second term in Eq. (2) is a regularization term that represents a priori sparse information of the original scene and also it is designed to penalize an estimate that would not exhibit the expected properties. Also, λ is a regularization parameter that balances the contribution of both terms. This minimizing problem can be solved easily by an iterative shrinkage/thresholding (IST) method (e.g., [6,7]) or Bregman iterative algorithms (e.g., [8,9]).

* Corresponding author.

E-mail addresses: nasser.eslahi@stu.nit.ac.ir (N. Eslahi), aghagol@nit.ac.ir (A. Aghagolzadeh), smh_andargoli@nit.ac.ir (S.M.H. Andargoli).

Much efforts have been made to develop an effective regularization term $\mathcal{R}(u)$, to reflect the image prior knowledge. The classical smoothing regularization terms, such as the quadratic Tikhonov regularization [10] and the total variation (TV) [4] regularization, utilize local structural patterns and are built based on the assumption that images are locally smooth except at edges. More specifically, these models favor piecewise constant image structures, and hence tend to smooth much the image details. Nonetheless, they cannot deal well with image details and fine structures (resulting in staircase artifacts and contrast losses), since they only exploit the local statistics, neglecting the nonlocal statistics of images [11,12].

Stemming from the sparsity and the local statistics, images are often composed of localized patterns (e.g., textures and structures) that repeat themselves at distant locations in the image domain. Hence, nonlocal regularizers can effectively model long-range dependencies and yield improvements in reconstruction results. Inspired by the success of nonlocal means (NLM) filtering for image denoising [13], many nonlocal regularization-based methods have also been proposed for various image processing applications [11,12,14–27], and also CS image restoration [28–34].

1.1. Related work on CS via nonlocal regularization

In recent works, the sparsity and the nonlocal self-similarity properties are usually combined into the final cost functional of image restoration solution to achieve better performance. In [28], a nonlocal total variation (NLTV) regularization model for CS image recovery is proposed, which is solved efficiently with Bregman iteration method. A combinational regularization parameter, using a reweighted TV and a weighted-based nonlocal sparse constraint, for CS image recovery is proposed in [29]. The work in [30] proposed an adaptive sparsity regularization term for CS image recovery process, which incorporated the local piecewise autoregressive model and a weighted-based nonlocal self-similarity constraint. In [31], the sparsity regularization parameters (which are locally estimated), together with a weighted-based nonlocal self-similarity constraint, are incorporated into the overall cost functional of image restoration solution to improve the image quality. A model-assisted adaptive recovery of CS (MARX-PC) is proposed in [32], which exploits both the local structural sparsity and nonlocal self-similarity, leading to an efficient CS recovery scheme. In [33], a nonlocal low-rank regularization approach toward exploiting the structured sparsity for CS image recovery is proposed. The proposed model in [33] consists of two components: patch grouping for characterizing the self-similarity of the signal and low-rank approximation for sparsity enforcement. The work in [34] proposed a strategy for CS image recovery via collaborative sparsity (RCoS) modeling. The local 2D sparsity and the nonlocal 3D sparsity are simultaneously imposed in RCoS enabling a natural image to be highly sparse in an adaptive hybrid space-transform domain.

1.2. Compressive video sensing (CVS)

Recently, the idea of CS for imaging (single pixel camera [35,36]) has been extended to the conventional predictive/distributed video coding, to develop highly desirable compressive video sensing (CVS)/distributed compressive video sensing (DCVS). CVS employs both data acquiring (video sensing) and compression into a unified task which emerges a new procedure to directly acquiring compressed video data via random projection (without temporally storing the complete raw data) for each individual frame in a low complexity encoder. In this case, the majority of computational burden is shifted from the encoder side to the decoder side, which is more suitable to deploy in modern

video applications, e.g., video surveillance systems and wireless multimedia sensor networks.

1.3. Related work on CVS

Several CVS recovery methods have already been proposed. Wakin et al. [37] proposed an intuitive (motion JPEG motivated) approach which extends compressive image sensing to video applications by considering each frame of the video sequence independently, and recovers each frame using the 2D discrete wavelet transform (2D DWT), individually. Since compressed image sensing techniques explore the spatial redundancy within an image, this simple extension fails to address the temporal redundancy in video. To enhance the signal sparsity in both spatial and temporal domains and achieve higher sampling efficiency, several frames can be jointly considered as a signal and recovered under a 3D transform (e.g., 3D DWT) [37]. Park and Wakin [38] proposed a multi-scale recovery approach, where several CS measurements are taken independently for each frame, and also the motion estimation is applied at the decoding step. The recovered video at coarse scales (low spatial resolution) is used to estimate motion which is then used to enhance the recovery at finer scales (high spatial resolutions). The same approach based on using two-step to iteratively update the estimates for the images in the video and the inter-frame motion was proposed in [39]. Also, Cossalter et al. [40] considered the motion estimation in their proposed joint compressive video coding and analysis scheme. Stanković et al. [41] and Prades-Nobet et al. [42] proposed a block-based selective video sampling scheme which firstly divides frames of the video sequence into key and non-key frames; then each frame is divided into the small non-overlapping blocks of equal sizes. In the decoding process, each block is approximated by a linear combination of blocks of previously reconstructed frames. Zheng and Jacobs [43] explored the sparsity of small inter-frame difference to remove the temporal redundancy. A multi-hypothesis (MH) prediction approach for CVS was proposed in [44], where different MH predictions of the current frame are generated from one or more previously reconstructed reference frames, and then combined to yield a composite prediction superior to any of the constituent single-hypothesis predictions. Ma et al. [45] proposed a CVS recovery method by introducing a modification of the approximated message passing algorithm and incorporating the 3D dual-tree complex wavelet transform during the recovering. In [46], each frame of a compressed-sensed video sequence is reconstructed iteratively using Karhunen–Loève transform (KLT) bases trained from adjacent previously reconstructed frame(s). There also exist other research works about CVS recovery based on dictionary learning (DL) [47–50]. In [50], we proposed a block-based CVS recovery method where key frames are reconstructed using ALS basis via ℓ_0 minimization method of [51]. For recovering of non-key frame, its prediction is achieved by using the previous reconstructed frame (to exploit the temporal redundancy) and incorporated into an optimization problem to refine the frame. Shu et al. [52] proposed a 3D CS approach, which decodes a video from incomplete compressive measurements by exploiting its 3D piecewise smoothness and temporal low-rank property. Yang et al. [53] proposed a Gaussian mixture model (GMM)-based inversion algorithm for CVS recovery from temporally compressed video measurements. The GMM is used to represent each 3D patch in a data set, with the assumption that the subspace of each patches lives on a union of subspaces and each patch is drawn from one subspace. Hosseini and Palataniotis [54] proposed an alternative model to the TV regularization to regulate the spatial and temporal redundancy in CVS by means of a tensorial decomposition.

1.4. Contributions

Inspired by the promising results of the above-mentioned techniques, in this paper, the nonlocal self-similarity and the local sparsity prior are both incorporated in a combinational regularization term, to adopt in a regularization-based framework for CS image/video restoration. The main contributions of this paper are listed as follows.

- We introduce a new sparsity measure, called *joint adaptive sparsity measure* (JASM). The proposed JASM coincidentally enforces both the local sparsity constraint and the nonlocal 3D sparsity in transform domain, in a unified manner, suggesting a powerful mechanism for characterizing the structured sparsities of natural images. In fact, the local sparsity depicts the local smoothness redundancies exploited by ALS basis, and the nonlocal 3D sparsity corresponds to the nonlocal self-similarity constraint achieved by a nonlocal statistical sparse modeling closely related to the ones proposed in [12,34]. Unlike those introduced in [12,34], our used nonlocal statistical sparse modeling has some functional modifications which makes it much more superior for CS image recovery—as introduced and examined in Sections 3 and 5.3, respectively.
- We propose two novel techniques for high-fidelity CS image and video restoration using JASM. The proposed CS image recovery problem via JASM (we refer to it as CS-JASM) is formulated in the form of minimization functional under regularization-based framework. Based on split Bregman framework [8,9], a powerful method for solving various variational models, an efficient alternating minimization algorithm is developed to solve the above severely underdetermined inverse problem efficiently. The proposed CVS recovery method (we refer to it as CVS-JASM) splits the video sequence into the key and non-key frames followed by dividing each frame into small non-overlapping blocks of equal sizes. The key frames are recovered using the proposed CS-JASM method, in order to exploit the spatial redundancy. For recovery of the non-key frames, a prediction of the current frame is initialized, by using the previous reconstructed frame to exploit the temporal redundancy. The prediction is employed in a proper optimization problem to recover the current non-key frame. Furthermore, we investigate the effectiveness of three well-known DL algorithms for adopting the best one in our proposed scheme.

Extensive numerical results on benchmark test images/video sequences clearly demonstrate that our proposed methods substantially outperform many of the conventional and state-of-the-art techniques for CS image/video restoration.

1.5. Outline

The rest of this paper is organized as follows. Section 2 provides a brief background on sparse representation and DL. Also, three well-known techniques of DL, accompanied by a recently proposed modeling for nonlocal self-similarity are introduced, briefly. The proposed regularization term, JASM, and its relation to previous works is described and discussed in Section 3. Section 4 shows how JASM is incorporated into the framework of image/video CS recovery, and gives the implementation details of solving the ensuing optimization problems. Numerical results and comparisons for our proposed methods are given in Section 5 and finally, Section 6 concludes the paper.

2. Background

2.1. Sparse representation and dictionary learning

One crucial problem in a sparse-representation problem is how to choose an efficient dictionary. There are many pre-specified (non-adaptive analytically designed) sparsifying dictionaries (basis or frame), e.g., Fourier transform, discrete cosine transform, wavelets, ridgelets, curvelets, contourlets and shearlets. In spite of being simple and having fast computation, the analytically designed dictionaries are not able to efficiently (sparsely) represent a given class of signals, and they lack the adaptivity to the image local structures. However, learning the atoms from a set of training signals belonging to signal class of interest would result in dictionaries with the capability of better matching the content of the signals. It has been experimentally shown that these adaptive dictionaries outperform the non-adaptive ones in many signal processing applications such as image compression [31,51,55], denoising [19,57], deblurring [17,19], interpolation [18,21], and super-resolution [17,24,56].

DL algorithms iteratively perform two steps of *sparse coding* and *dictionary update*. In the first step, which is actually the clustering of the signals into a union subspace, the sparse approximation of the signals is computed using the current dictionary. The second step is for updating the current dictionary. In fact, most DL algorithms differ mainly in the way of updating the dictionaries [58–60]. Some algorithms such as K-singular value decomposition (K-SVD) [59] are based on updating the atoms one-by-one, while some others such as method of optimal directions (MOD) [58] update the whole set of atoms at once. In [60], a MOD-like algorithm was proposed in which more than one atom along with the nonzero entries in their associated row vectors in coefficient matrix are updated at a time. We refer to this algorithm as the multiple dictionary update (MDU) algorithm.

2.2. Patch-based redundant sparse recovery

Owing to the curse of dimensionality, proposing a global model for the whole image is often found to be hard, and especially so if we are dealing with learned models. To tackle this problem (besides exploiting self-similarity of patches), in recent years, patch-based modeling approaches have been widely adopted in many image restoration (e.g., [14,17–21,30,31,55–57,61,62]), recognition (e.g., [63,64]), and classification methods (e.g., [65,66]). The main idea is to decompose the target image into overlapped patches, process each of which separately, and then merge the results by a plain averaging. Suppose the vector representation of the original image is denoted by $u \in \mathbb{R}^n$; accordingly, $u_{p_l} \in \mathbb{R}^{B_s}$ represents an image patch of size $\sqrt{B_s} \times \sqrt{B_s}$ at location l , $l = 1, 2, \dots, J$. Then we have:

$$u_{p_l} = R_l u, \quad (3)$$

where $R_l \in \mathbb{R}^{B_s \times n}$ is a binary matrix operator that extracts the square patch u_{p_l} from u , forming the output patch as the column vector. Patches are usually overlapped (to suppress the boundary artifacts), and such patch based representation is highly redundant and also significant for achieving high recovery quality. Therefore, the recovery of u from $\{u_{p_l}\}$ becomes an over-determined system, which is straightforward to obtain the following least-square solution: $u = (\sum_{l=1}^J R_l^T R_l)^{-1} \sum_{l=1}^J (R_l^T u_{p_l})$, which indicates that the overall image is reconstructed by averaging all the overlapped patches. Given dictionary D , the sparse coding process of each patch u_{p_l} over D is to find a sparse vector α_l such that each patch can be approximated by a linear combination of dictionary atoms $u_{p_l} \approx D\alpha_l$. Then the entire image can be sparsely represented by the set of

sparse codes $\{\alpha_l\} : u \approx D \circ \alpha \triangleq (\sum_{l=1}^J R_l^T R_l)^{-1} \sum_{l=1}^J (R_l^T D \alpha_l)$, where α denotes the concatenation of all $\{\alpha_l\}$. By given a set of training image patches $U = [u_{p_1}, u_{p_2}, \dots, u_{p_J}]$, where J' is the number of training image patches, the goal of DL is to jointly optimize the sparsifying basis D , and the representation coefficient matrix $\Lambda = [\alpha_1, \alpha_2, \dots, \alpha_{J'}]$, such that $u_{p_l} = D \alpha_l$ and $\|\alpha_l\|_{\ell_p} \leq L$, where ℓ_p is either ℓ_0 or ℓ_1 . This can be formulated by the following minimization problem:

$$\{\hat{D}, \hat{\Lambda}\} = \arg \min_{D, \Lambda} \|U - DA\|_F^2 \quad \text{s.t. } \|\alpha_l\|_{\ell_p} \leq L \quad \forall l, \quad (4)$$

where $\|\cdot\|_F$ denotes the Frobenius norm and the requirement of $\|\alpha_l\|_{\ell_p} \leq L \ll n$ indicates that the sparse representation step uses no more than L atoms from the dictionary for every image patch instance.

Many a time, since ℓ_0 minimization problem is nonconvex and NP-hard, the usual routine is to solve its optimal convex approximation, i.e., ℓ_1 minimization. However, for some practical problems (e.g., image inverse problems), the conditions guaranteeing the equivalence of ℓ_0 and ℓ_1 minimization are not necessarily satisfied. Although the above minimization problem in (4) is large-scale and highly nonconvex even when ℓ_p is equal to ℓ_1 , some heuristic approaches (e.g., MOD [58], K-SVD [59] and MDU [60]) have been developed to optimize D and Λ alternatively, leading state-of-the-art results in many fundamental signal and image processing tasks. More precisely, these approaches typically alternate between finding the dictionary D by fixing Λ (dictionary update step), and the sparse representation Λ by fixing D (sparse coding step).

2.3. Nonlocal 3D sparsity and nonlocal self-similarity modeling in transform domain

Motivated by the success of NLM [13] in self-similarity and BM3D [61] in image restoration, Zhang et al. [34] proposed a model for nonlocal self-similarity prior information, which is employed efficiently in CS image restoration. The proposed model explores the nonlocal self-similarity by means of the sparsity of the transform coefficients, which are obtained by transforming the 3D array generated by stacking similar image patches. The detailed description of nonlocal 3D sparsity (N3D) in transform domain [34] is delineated in Table 1. To elucidate on, first, the image is divided into overlapped patches of equal sizes. Then, for each patch the best matched patches are found within a searching window followed by stacking these similar patches into a 3D array, which is called a group. Next, a 3D transform is applied to the group to obtain its transform coefficients, followed by arranging them in a lexicographic order. At last, the number of nonzero coefficients is used to measure the N3D sparsity of this patch while the N3D sparsity of the whole image is obtained by summing all the ones of each patch [34]. The mathematical formulation of the

Table 1

A description of both the nonlocal 3D sparsity (N3D) in transform domain [34] (**step 1-step 4**), and the nonlocal self-similarity modeling (NLSM) in 3D transform domain [12] (**step 1-step 5**).

- Step 1:** Split the image u (of size $\sqrt{n} \times \sqrt{n}$) into P overlapped patches of equal sizes $u_{p_l} \in \mathbb{R}^{B_p}$ at location l' , $l' = 1, 2, \dots, P$
- Step 2:** Define $S_{u_{p_l}} = \{S_{u_{p_l} \otimes q}\}_{q=1}^c$ the set including the c best matched patches (based on Euclidean distance) in a search window of size $\omega \times \omega$
- Step 3:** For each $S_{u_{p_l}}$, Stack the c patches belonging to $S_{u_{p_l}}$ into a 3D array to form a group $G_{u_{p_l}}$
- Step 4:** Find the transform coefficients of each group $T^{3D}(G_{u_{p_l}})$, and arrange all the $T^{3D}(G_{u_{p_l}})$ in the lexicographic order to form $\Theta_u \in \mathbb{R}^{K_\theta}$; the column vector of all the transform coefficients of image u
- Step 5:** Analyze the histogram of the transform coefficients

Note: (1) $T^{3D}(\cdot)$ denotes the operator of the orthogonal 3D transform. (2) $K_\theta = B_p * c * P$.

N3D for self-similarity in transform domain can be written as:

$$\Psi_{\text{N3D}}(u) = \|\Theta_u\|_{\ell_0} = \sum_{l'=1}^P \|\mathcal{T}^{3D}(G_{u_{p_l}})\|_{\ell_0}, \quad (5)$$

where $\Psi_{\text{N3D}}(\cdot)$ corresponds to the N3D operator.

After obtaining Θ_u , it is split into P groups (3D arrays) of transform coefficients, which are then inverted to generate estimates for each patch in the group. The patch-wise estimates are returned to their original positions and the final estimated image is achieved by averaging all of the above patch-wise estimates [34]. To put it simply and briefly, after obtaining Θ_u , the new estimate of u is achieved by $\hat{u} = \Omega_{\text{N3D}}(\Theta_u)$, where $\Omega_{\text{N3D}}(\cdot)$ corresponds to the inverse operator of Ψ_{N3D} .

Besides, inspired by the success of NLM and BM3D, a nonlocal statistical modeling (NLSM) for self-similarity in 3D transform is proposed by Zhang et al. [12], which is used in image restoration application—except CS. The NLSM, which is closely related to N3D, characterizes the nonlocal self-similarity of natural images by means of the distribution of the transform coefficients Θ_u , i.e., $\Psi_{\text{NLSM}}(u) = \|\Theta_u\|_{\ell_1}$.

3. Joint adaptive sparsity measure (JASM)

As stated previously, using only the local sparsity constraint $\|\alpha\|_{\ell_p}$ in Eq. (2) may not lead to an enough accurate CS image restoration. An alternative approach for superior incorporating the prior knowledge about images is via sparse representation and nonlocal self-similarity which has led to highly competent works of sparsity-based CS image restoration (see e.g., [31,34]). As can be seen in Fig. 1, the distribution of transform coefficients Θ_u is characterized by a very sharp peak at zero amplitude and the vast majority of coefficients are concentrated near zero (*leptokurtic distribution*) implying Θ_u has a very sparse representation. Considering the above point, NLSM for self-similarity in 3D transform domain motivated us to adopt it, by some modifications, as an adaptive *nonlocal statistical sparsity measure* (NSSM) in our proposed scheme to depict the repetitiveness of the textures and structures in natural images within nonlocal regions. The major similarities and differences between NSSM and those proposed in Section 2.3 will be discussed in the following.

The proposed *joint adaptive sparsity measure* (JASM) is defined by integrating both the local sparsity constraint (depicting the local smoothness achieved by ALS basis) and nonlocal 3D sparsity in transform domain (corresponding to the nonlocal self-similarity prior achieved by a method similar to the one introduced in Section 2.3, that is NSSM). In a mathematical expression, the proposed JASM prior knowledge is expressed as:

$$\mathcal{R}_{\text{JASM}}(u) = \Psi_{\text{LS}}(u) + \tau \Psi_{\text{NSSM}}(u) = \|\alpha\|_{\ell_0} + \tau \|\Theta_u\|_{\ell_1}. \quad (6)$$

In the proposed JASM prior (6), the first and the second terms represent the image local smoothness prior and nonlocal self-similarity prior, respectively. Also, τ is a regularization parameter which controls the trade-off between two competing sparsity terms.

The similarities and differences between the proposed JASM, and those proposed in [12] (JSM) and [34] (CoSM) are discussed as the following.

1. Resemble to NLSM, the proposed NSSM characterizes the nonlocal self-similarity by means of the distributions of the transform coefficients, statistical modeling, while the nonlocal self-similarity in CoSM (N3D) is characterized by means of the sparsity of transform coefficients—summing all the number of nonzero coefficients of each group.

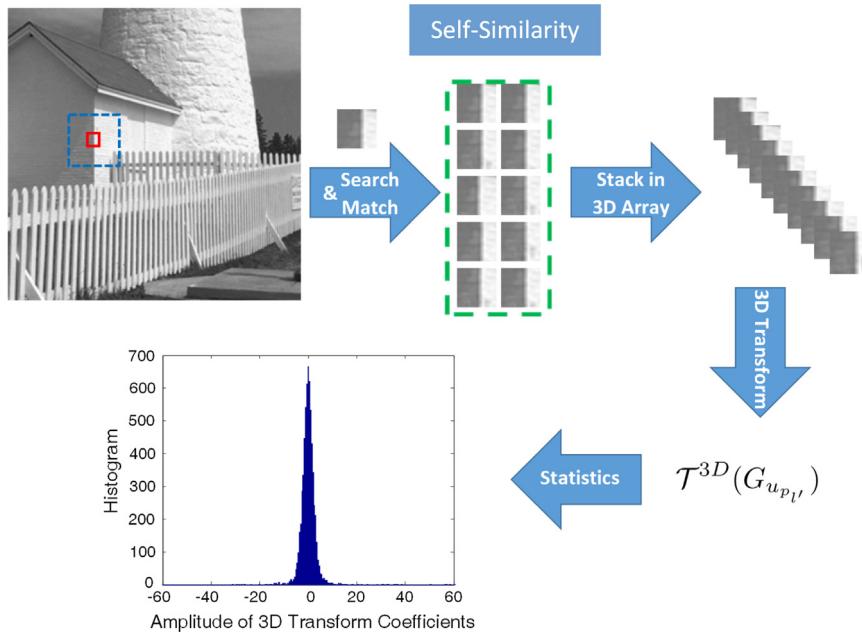


Fig. 1. Illustration of NLSM for self-similarity in 3D transform domain [12].

2. In both NLSM and N3D, the 3D transform denoted by \mathcal{T}^{3D} is composed of 2D DCT and 1D wavelet transform, however in NSSM in the proposed JASM, \mathcal{T}^{3D} is composed of 3D wavelet transform. More specifically, on the 2D wavelet coefficients of each patch in a group $G_{u_{p_f}}$, 1D wavelet transform is conducted along the third axis.
3. CoSM is developed in order to recover CS images; however, JSM is incorporated into the regularization-based framework for the image inverse problems such as image deblurring, image inpainting and mixed Gaussian plus impulse noise removal, except CS. In this paper, though the proposed JASM is developed for CS image restoration, it can be used appropriately for the other image inverse problems like those mentioned for JSM.
4. RCoS and JSM, both employ an anisotropic TV regularization to characterize the local smoothness prior of natural images. Nevertheless, JASM takes the advantages of local sparsity achieved by ALS basis for better characterizing of the local smoothness prior and also for better edge and detail preserving.

4. Image/video CS recovery via JASM

4.1. Encoding

As mentioned before, the proposed method, firstly, divides the video sequence into the key and non-key frames followed by dividing each frame/image, of size $I_r \times I_c$, into small non-overlapping blocks of equal sizes (i.e., size $B \times B$), and then the same sensing matrix Φ_B ¹ (i.e., size $m_B \times B^2$, where $m_B = \lfloor \frac{m \times B^2}{n} \rfloor$, $n = I_r \times I_c$) is applied for sampling of each block. In this case, we have:

$$f_i = \Phi_B u_i, \quad (7)$$

where u_i is the (column) vector representing block i of the input frame/image, f_i is its corresponding measurement vector and Φ_B independently samples blocks within a frame/image. Using this technique has several benefits compared to the use of a random

sampling operator to the entire frame/image; i.e., the encoder does not need to wait until the entire frame/image is measured but each block is sent after its linear projection. In addition, at the decoder side, each block is processed independently; therefore the speed of encoding and decoding procedure is increased. Also in this case, we just need to store a $m_B \times B^2$ sensing matrix instead of a $m \times n$ sensing matrix. More precisely, the global sensing matrix takes a block-diagonal structure, $\Phi = \text{diag}(\Phi_B, \dots, \Phi_B)$, where $\text{diag}(\cdot)$ represents a diagonal matrix.

4.2. CS image recovery via JASM—recovery of key frame(s)

By substituting the proposed JASM (6) for the regularization term $\mathcal{R}(u)$ in the regularization-based framework of Eq. (2), the proposed optimization problem for CS image recovery is expressed as follows:

$$\min_u \left\{ \frac{1}{2} \|f - \Phi u\|_{\ell_2}^2 + \lambda \Psi_{\text{LS}}(u) + \tau \Psi_{\text{NSSM}}(u) \right\}. \quad (8)$$

The first term represents the observation constraint and the second and the third term indicate the image local smoothness and nonlocal self-similarity prior information.

In order to solve the above optimization problem efficiently, the framework of the split Bregman iteration (SBI) algorithm [8] is employed; due to its efficiencies in efficiently solving the large-scale optimization problems, small memory footprint requirement, and also, ease of programming by users. For this purpose, first, Eq. (8) is converted into its equivalent constrained-form, and then the variable splitting technique is introduced along with invoking the Bregman iteration. Consider the equivalent constrained-form of Eq. (8):

$$\begin{aligned} \min_{u, \alpha, w} \quad & \left\{ \frac{1}{2} \|f - \Phi u\|_{\ell_2}^2 + \lambda \|\alpha\|_{\ell_0} + \tau \|\Theta_w\|_{\ell_1} \right\} \\ \text{s.t.} \quad & u = D \circ \alpha, u = w, w = \Omega_{\text{NSSM}}(\Theta_w), \end{aligned} \quad (9)$$

where D stands for ALS basis and α denotes the patch-based redundant sparse representation for the whole image over D which can be found by the method introduced in Section 2.2.

¹ Note that, m/n denotes the measurement ratio; the measurement ratio for key and non-key frame may differ. In this context, MR_K (MR_{NK}) and Φ_{BK} (Φ_{BNK}) are used as the measurement ratio and the sensing matrix of key (non-key) frames, respectively. In this paper, we assume that: $\Phi_{BK} = \Phi_{BNK}$.

In order to solve the above minimization problem, an alternating SBI algorithm is applied. We finally obtain the following schemes:

$$u^{k+1} = \arg \min_u \frac{1}{2} \|f - \Phi u\|_{\ell_2}^2 + \frac{\mu_1}{2} \|u - D \circ \alpha^k - b^k\|_{\ell_2}^2 + \frac{\mu_2}{2} \|u - w^k - c^k\|_{\ell_2}^2 \quad (10a)$$

$$\alpha^{k+1} = \arg \min_{\alpha} \|\alpha\|_{\ell_0} + \frac{\mu_1}{2} \|u^{k+1} - D \circ \alpha - b^k\|_{\ell_2}^2 \quad (10b)$$

$$w^{k+1} = \arg \min_w \|\Theta_w\|_{\ell_1} + \frac{\mu_2}{2} \|u^{k+1} - w - c^k\|_{\ell_2}^2 \quad (10c)$$

$$b^{k+1} = b^k - u^{k+1} + D \circ \alpha^{k+1} \quad (10d)$$

$$c^{k+1} = c^k - u^{k+1} + w^{k+1}. \quad (10e)$$

Here, μ_1 and μ_2 are fixed value parameters for improving the numerical stability of the algorithm.

Given α^k and w^k , the sub-problem of (10a) consists of minimizing a strictly convex quadratic function that can be solved easily. By setting the gradient of the objective function in Eq. (10a) to zero, a closed-form solution for Eq. (10a) is achieved:

$$u^{k+1} = (\Phi^T \Phi + (\mu_1 + \mu_2) I)^{-1} (\mu_1 (D \circ \alpha^k + b^k) + \mu_2 (w^k + c^k) + \Phi^T f). \quad (11)$$

Since for image CS recovery, Φ is a random projection matrix with no especial structures ($\Phi^T \Phi \neq I$), it is too costly to solve the minimization of the quadratic function in Eq. (10a) directly by using Eq. (11), because of the existence of the matrix inverse in Eq. (11). Here, in order to avoid computing the inversion of matrix, the steepest descent method with the optimal step is utilized to solve the minimization of the quadratic function in Eq. (10a), which can be expressed as:

$$u^{k+1} = u^k - \eta^k g^k, \quad \eta^k > 0. \quad (12)$$

Here, g is the gradient direction of the objective function and $\eta = \text{abs}(g^T g / g^T (\Phi^T \Phi + (\mu_1 + \mu_2) I) g)$ represents the optimal step size.

In order to efficiently solve Eq. (10b), an alternative approach via ALS basis and ℓ_0 minimization is proposed in [51], where in this part, we adopt the general scheme of [51] to solve Eq. (10b) effectively. Now, let us go back to Eq. (10b) and point out how to solve it. By given u in hand (according to Eq. (10a)), considering $r = u - b$, $v = D \circ \alpha$ (for simplicity, the superscript k is dropped without confusion) and $\ell_p = \ell_0$ (sparsity is strictly measured), the sub-problem of Eq. (10b) becomes:

$$\min_{\alpha} \frac{1}{2} \|v - r\|_{\ell_2}^2 + \frac{\lambda}{\mu_1} \|\alpha\|_{\ell_0}. \quad (13)$$

By these transformations, we regard r as some type of the noisy observation of v . However, it is worth to note that, owing to the complicated definition of α , it is difficult to solve Eq. (13) directly. In order to solve Eq. (13) amenable, in this paper, a reasonable assumption is used which leads to obtain a closed-form solution of Eq. (13). Such that this assumption is efficiently employed in [34]; hence, we utilize the same approach to solve the above and ensuing minimization problems.

Let us denote the error vector by $e = v - r$ (such that $v, r \in \mathbb{R}^n$) and each element of e by $e(i), i = 1, \dots, n$. Also, we suppose that all elements of e are independent and identically distributed (i.i.d.) with zero-mean, and variance σ^2 . It is worth emphasizing that the corresponding distribution does not need to be Gaussian, Laplacian, or generalized Gaussian distribution process (GGD), which are more general. By invoking the law of large numbers in probability theory,

for any $\epsilon > 0$, it leads to $\lim_{n \rightarrow \infty} \Pr\{|\frac{1}{n} \sum_{i=1}^n e^2(i) - \sigma^2| < \frac{\epsilon}{2}\} = 1$, i.e.,

$$\lim_{n \rightarrow \infty} \Pr\left\{\left|\frac{1}{n} \|v - r\|_{\ell_2}^2 - \sigma^2\right| < \frac{\epsilon}{2}\right\} = 1, \quad (14)$$

where $\Pr\{\cdot\}$ represents the probability.

Let v_c and r_c denote the concatenation of all the patches v_{p_l} and r_{p_l} , $l = 1, 2, \dots, J$, respectively, and each element of $e_c = v_c - r_c$ is denoted by $e_c(j), j = 1, \dots, K$, where $K = B_s * J$. Based on the given assumptions, it is concluded that all elements of e_c are i.i.d. with zero mean and variance σ^2 . Thanks to the law of the large numbers, by doing the same manipulation as Eq. (14) to $e_c^2(j)$, it yields $\lim_{K \rightarrow \infty} \Pr\left\{|\frac{1}{K} \sum_{j=1}^K e_c^2(j) - \sigma^2| < \frac{\epsilon}{2}\right\} = 1$, i.e.,

$$\lim_{K \rightarrow \infty} \Pr\left\{\left|\frac{1}{K} \sum_{l=1}^J \|v_{p_l} - r_{p_l}\|_{\ell_2}^2 - \sigma^2\right| < \frac{\epsilon}{2}\right\} = 1. \quad (15)$$

Therefore, according to Eqs. (14) and (15), the following property is concluded:

$$\lim_{n \rightarrow \infty} \Pr_{K \rightarrow \infty} \left\{ \left| \frac{1}{n} \|v - r\|_{\ell_2}^2 - \frac{1}{K} \sum_{l=1}^J \|v_{p_l} - r_{p_l}\|_{\ell_2}^2 \right| < \epsilon \right\} = 1, \quad (16)$$

in which the relationship between $\|v - r\|_{\ell_2}^2$ and $\sum_{l=1}^J \|v_{p_l} - r_{p_l}\|_{\ell_2}^2$ (almost sure) is described as:

$$\frac{1}{n} \|v - r\|_{\ell_2}^2 = \frac{1}{K} \sum_{l=1}^J \|v_{p_l} - r_{p_l}\|_{\ell_2}^2. \quad (17)$$

Now, by substituting Eq. (17) into Eq. (13), J sub-problems for all the patches v_{p_l} are achieved that they can be solved more efficiently. Each patch based sub-problem is formulated as:

$$\arg \min_{\alpha_l} \frac{1}{2} \|v_{p_l} - r_{p_l}\|_{\ell_2}^2 + \theta \|\alpha_l\|_{\ell_0}, \quad (18)$$

where $\theta = \frac{\lambda K}{\mu_1 n}$. Consider $v_{p_l} = D \alpha_l$, where D is the adaptively learned dictionary from r_{p_l} using patch-based DL method described in Section 2.2, we have:

$$\hat{D} = \arg \min_D \sum_{l=1}^J \|r_{p_l} - D \alpha_l\|_{\ell_2}^2 \quad \text{s.t. } \|\alpha_l\|_{\ell_0} \leq L, \quad \forall l. \quad (19)$$

Since r is regarded as a good approximation of v at each iteration, here we conduct adaptive sparsifying basis learning using all the patches extracted from r .

Now, by considering $v_{p_l} = D \alpha_l$, the sub-problem of Eq. (18) can be considered as the sparse coding problem as:

$$\arg \min_{\alpha_l} \frac{1}{2} \|D \alpha_l - r_{p_l}\|_{\ell_2}^2 + \theta \|\alpha_l\|_{\ell_0}. \quad (20)$$

Also, in order to achieve higher sparsity, Eq. (20) can be formulated in its constrained form as:

$$\min_{\alpha_l} \|\alpha_l\|_{\ell_0} \quad \text{s.t. } \|D \alpha_l - r_{p_l}\|_{\ell_2}^2 \leq \delta \quad (21)$$

where $\delta = \omega' \theta$ is a small constant controlling the approximation error, and ω' is a control factor. Now, Eq. (21) can be solved efficiently using orthogonal matching pursuit (OMP) [67] algorithm. However, if ℓ_0 pseudo-norm in Eq. (21) is relaxed with ℓ_1 convex norm, it can be solved with basis pursuit [68], lasso [69] and ℓ_1 -regularized least square [70], but may lead to less sparse solution. For all J overlapped patches, this process is employed to achieve α (concatenation of all $\{\alpha_l\}, l = 1, \dots, J$) which is the solution of Eq. (13).

Given u , analogously, the sub-problem of (10c) can be written as:

$$\min_w \frac{1}{2} \|w - z\|_{\ell_2}^2 + \frac{\tau}{\mu_2} \|\Theta_w\|_{\ell_1}, \quad (22)$$

where $z = u - c$ (for simplicity the superscript k is omitted without confusion). Due to the complicated definition of Θ_w , it seems difficult to solve Eq. (22) directly. Therefore, to enable solving Eq. (22) tractable, here we use a procedure which is similar to the one used

for solving Eq. (10c). Accordingly, by viewing z as some type of the noisy observation w , we denote the error vector by $e' = w - z$ (such that $w, z \in \mathbb{R}^n$) and each element of e' by $e'(i), i = 1, \dots, n$. Then an assumption is made that all elements of e' are i.i.d. with zero-mean and variance σ'^2 . It is worth stressing again that the above assumption does not need to be Gaussian, Laplacian or GGD process. Based on the assumption, and by invoking the *law of large numbers*, for any $\epsilon' > 0$ (similar to the one used to obtain Eq. (14)), we have:

$$\lim_{n \rightarrow \infty} \Pr \left\{ \left| \frac{1}{n} \|w - z\|_{\ell_2}^2 - \sigma'^2 \right| < \frac{\epsilon'}{2} \right\} = 1. \quad (23)$$

Let us denote the error vector in 3D transform domain by $\Theta_e = \Theta_w - \Theta_z$ (such that $\Theta_w, \Theta_z \in \mathbb{R}^{K_\Theta}$), and each element of Θ_e by $\Theta_e(j), j = 1, \dots, K_\Theta$. Due to the orthogonal property of 3D transform T^{3D} , and definition of the error vector in 3D transform domain, it is concluded that all elements of Θ_e are i.i.d. with zero-mean and variance σ'^2 . Therefore, by invoking the *law of large numbers*, and doing the same manipulations with Eq. (23) to $\Theta_e(j)$, for any $\epsilon' > 0$, it yields:

$$\lim_{K_\Theta \rightarrow \infty} \Pr \left\{ \left| \frac{1}{K_\Theta} \|\Theta_w - \Theta_z\|_{\ell_2}^2 - \sigma'^2 \right| < \frac{\epsilon'}{2} \right\} = 1. \quad (24)$$

According to Eqs. (23) and (24), and by using the property introduced in Eq. (16), the relationship between $\|w - z\|_{\ell_2}^2$ and $\|\Theta_w - \Theta_z\|_{\ell_2}^2$ is described as follows:

$$\frac{1}{n} \|w - z\|_{\ell_2}^2 = \frac{1}{K_\Theta} \|\Theta_w - \Theta_z\|_{\ell_2}^2. \quad (25)$$

Incorporating Eq. (25) into Eq. (22) leads to:

$$\arg \min_{\Theta_w} \frac{1}{2} \|\Theta_w - \Theta_z\|_{\ell_2}^2 + \frac{K_\Theta \tau}{\mu_2 n} \|\Theta_w\|_{\ell_1}. \quad (26)$$

Since the unknown variable Θ_w is component-wise separable in Eq. (26), each of its component $\Theta_w(j)$ can be independently obtained by a component-wise (soft) shrinkage procedure; i.e.: $\Theta_w(j) = \arg \min_{\Theta_w(j)} \left\{ \frac{1}{2} \|\Theta_w(j) - \Theta_z(j)\|^2 + \frac{K_\Theta \tau}{\mu_2 n} |\Theta_w(j)| \right\}$, where $0 \in \frac{\Theta_w(j)}{|\Theta_w(j)|} \left(\frac{K_\Theta \tau}{\mu_2 n} \right) + (\Theta_w(j) - \Theta_z(j))$. Indeed, in vector form, this shrinkage can be written shortly as: $\Theta_w = S_{\frac{K_\Theta \tau}{\mu_2 n}}(\Theta_z)$, where $S_\rho(x) := \text{sign}(x) \odot \max\{|x| - \rho, 0\}$ is the soft shrinkage operator and \odot denotes the component-wise product. Also, the sign, max and absolute value functions are applied in component-wise fashion. Thus, the closed-form solution of Eq. (22) is written as:

$$w = \Omega_{\text{NSSM}}(\Theta_w) = \Omega_{\text{NSSM}} \left(S_{\frac{K_\Theta \tau}{\mu_2 n}}(\Theta_z) \right). \quad (27)$$

Either achieving adaptive sparsifying basis (by training image patches) or exploring nonlocal self-similarity of an image (by stacking the similar image patches into multiple groups and then imposing the correlation prior within each group) requires the original image. Nevertheless, in practice, the original image is unknown in CS, except for its random measurements f . Such a problem with a *chicken-and-egg* flavor is usually solved by an iterative scheme; obtaining both the estimate of the original image and its patch-grouping information (in NSSM); and also obtaining the estimate of the original image and ALS basis, alternately. In this paper, we set the initial estimate for the original image by MH prediction method [44]. Then in the subsequent iterations, since r in Eq. (13) (z in Eq. (22)) is regarded as a good available approximation of the original image, it is conducted to obtain ALS basis (patch-grouping information for NSSM) using all patches extracted from r (z). The proposed algorithm for CS image (and also key frame) recovery via JASM is summarized in detail in Table 2.

Table 2

The detailed description of the proposed CS image recovery framework via JASM.

Input: $f, \Phi_B, B_s, B_p, \omega, \tau, \mu_1, \mu_2, iin$: inloop iteration number, k_{max} : maximum iteration number, Tol : Tolerance Output: u^* : Recovered image;
Initialization: Set $k=0$; $(\alpha^0, b^0, c^0) = (0, 0, 0)$; $u_{init} = v^0 = \text{MH_recovery}(f, \Phi_B)$ using method of [44]; While a stop criterion is not satisfied do
1. $u^{k+1} = \tilde{u} = u^k; r^{k+1} = u^{k+1} - b^k; z^{k+1} = u^{k+1} - c^k$;
2. update D^{k+1} using Eq. (19)
3. for each patch r_{pi} do
compute α_l^{k+1} using Eq. (21)
end for
4. update α^{k+1} by concatenating all $\{\alpha_l^{k+1}\}$
5. for each patch z_{pi} do
find $G_{z_{pi}}$ and compute $\mathcal{T}^{3D}(G_{z_{pi}})$ (see Table 1)
end for
6. arrange all $\mathcal{T}^{3D}(G_{z_{pi}})$ in lexicographic order to form Θ_z
7. update w^{k+1} using Eq. (27)
8. for $i = 1 : iin$
$g_i^k \leftarrow (\Phi^T(\Phi u^{k+1} - f) + \mu_1(u^{k+1} - D^{k+1} \circ \alpha^{k+1} - b^k) + \mu_2(u^{k+1} - w^{k+1} - c^k))$
$n_i^k \leftarrow \text{diag} \left(\text{abs} \left(\frac{g_i^k T g_i^k}{g_i^k T (\Phi^T(\Phi u^{k+1} - f) + (\mu_1 + \mu_2) I) g_i^k} \right) \right)$
$u^{k+1} \leftarrow u^{k+1} - n_i^k g_i^k$ MPS- NO-SPC
end for
9. compute $s^{k+1} = \text{SSIM}(u^{k+1}, \tilde{u})$
10. compute $diff = \text{abs}(s^{k+1} - s^k)$
11. update b^k, c^k using Eqs. (10d) and (10e), respectively
12. $k \leftarrow k+1$
end While
stopping criterion: $k = k_{max}$ or $diff \leq Tol$.

4.3. Recovery of non-key frame(s)

While spatial domain compression is performed by CS, the temporal redundancy is not exploited fully since no motion estimation and compensation is performed at the CVS encoder. To incorporate the temporal redundancy for efficient recovery of the non-key frames, the temporal correlation between adjacent frames is exploited through the inter-frame sparsity model. Here, an iterative approach for the reconstruction of the non-key frames is adopted where the approach initially estimates an approximation of the non-key frame using the previous reconstructed frame. Then the initially estimated frame is utilized in an optimization problem to recover and refine the current non-key frame.

Assume that $u_{t-1}^* = D \circ \alpha_{t-1}^*$, where u_{t-1}^* is the previous reconstructed frame using ALS basis D , and α_{t-1}^* denotes the patch-based redundant sparse representation for the whole previous reconstructed frame over D . The initialization step can be formulated as:

$$\min_{\alpha} \frac{1}{2} \|f - \Phi D \circ \alpha\|_{\ell_2}^2 + \lambda \|\alpha\|_{\ell_1} + \tau \|\alpha - \alpha_{t-1}^*\|_{\ell_1}. \quad (28)$$

The first term keeps the solution close to the measurements; the second term promotes sparsity in the spatial transform of the current frame; and the third term promotes the sparsity in the inter-frame difference to achieve the temporal redundancy between the current frame and the previous reconstructed frame. In this paper, the problem of Eq. (28) is solved efficiently by an alternating SBI-based framework via an iterative/shrinkage method using surrogate functions.

Assume that $v = D \circ \alpha$, then Eq. (28) can be formulated in its constrained form:

$$\min_{\alpha, v} \frac{1}{2} \|f - \Phi v\|_{\ell_2}^2 + \lambda \|\alpha\|_{\ell_1} + \tau \|\alpha - \alpha_{t-1}^*\|_{\ell_1} \quad \text{s.t. } v = D \circ \alpha. \quad (29)$$

Now, the above equation can be solved easily by an alternating

SBI-based framework as follows:

$$v^{k+1} = \arg \min_v \frac{1}{2} \|f^k - \Phi v\|_{\ell_2}^2 + \frac{\mu}{2} \|D \circ \alpha^k - v - b^k\|_{\ell_2}^2, \quad (30a)$$

$$\alpha^{k+1} = \arg \min_{\alpha} \frac{\mu}{2} \|D \circ \alpha - v^{k+1} - b^k\|_{\ell_2}^2 + \lambda \|\alpha\|_{\ell_1} + \tau \|\alpha - \alpha_{t-1}^*\|_{\ell_1}, \quad (30b)$$

$$b^{k+1} = b^k + v^{k+1} - D \circ \alpha^{k+1}, \quad (30c)$$

$$f^{k+1} = f^k + f - \Phi v^{k+1}. \quad (30d)$$

The term in Eq. (30d) adds a residual feedback to the algorithm.

Given α^k , b^k and f^k , the sub-problem of (30a) is essentially a minimization problem of strictly convex quadratic function. Setting the gradient of the objective function in Eq. (30a) to zero leads to a closed-form solution:

$$v^{k+1} = (\Phi^T \Phi + \mu I)^{-1} (\mu(D \circ \alpha^k - b^k) + \Phi^T f^k). \quad (31)$$

Similar to Eq. (11), it is too costly to compute the matrix inverse in Eq. (31) directly, owing to the randomness of Φ with no especial structures. Therefore, again, the steepest descent method is used to solve Eq. (30a) efficiently.

By given v^{k+1} , and assuming $u = D \circ \alpha$ and $r = v + b$ (for the simplicity the superscript k is omitted without confusion), the α sub-problem of (30b) can be formulated as:

$$\min_{\alpha} \frac{1}{2} \|u - r\|_{\ell_2}^2 + \frac{\lambda}{\mu} \|\alpha\|_{\ell_1} + \frac{\tau}{\mu} \|\alpha - \alpha_{t-1}^*\|_{\ell_1}. \quad (32)$$

Now, by utilizing the same assumption which is used in Section 2.2, the sub-problem of (32) can be formulated as:

$$\min_{\alpha_l} \sum_{l=1}^J \left\{ \frac{1}{2} \|u_{p_l} - r_{p_l}\|_{\ell_2}^2 + \frac{\lambda K}{\mu n} \|\alpha_l\|_{\ell_1} + \frac{\tau K}{\mu n} \|\alpha_l - \alpha_{t-l}^*\|_{\ell_1} \right\}. \quad (33)$$

Obviously, Eq. (33) can be solved efficiently by solving J sub-problems for all patches. By assuming $u_{p_l} = D \alpha_l$, $\theta_1 = \frac{\lambda K}{\mu n}$ and $\theta_2 = \frac{\tau K}{\mu n}$, for a single patch r_{p_l} , we have:

$$\arg \min_{\alpha_l} \left\{ \frac{1}{2} \|D \alpha_l - r_{p_l}\|_{\ell_2}^2 + \theta_{1,l} \|\alpha_l\|_{\ell_1} + \theta_{2,l} \|\alpha_l - \alpha_{t-l}^*\|_{\ell_1} \right\}, \quad (34)$$

where $\theta_\chi = \text{diag}((\theta_{\chi,l})_{l=1}^J)$, $\chi \in \{1, 2\}$. The sub-problem of (34) can be solved adeptly via an iterative/shrinkage algorithm using surrogate functions. Such method is based on the works of Daubechies et al. [6], and was used proficiently by Dong et al. [71] in order to solve a double-header ℓ_1 optimization. It is worth mentioning that, to the best of our knowledge, the proposed method in [71] updates the dictionary using K-means and principle component analysis; however, as will be discussed in Section 5.1, in our approach, K-SVD is utilized for updating the dictionary.

We utilize the solution of Eq. (29) into a refined (post processed) optimization problem for enhancing the recovery of current frame, u_t . Suppose v^* is the solution of Eq. (29), then the refined optimization problem can be formulated as:

$$\begin{aligned} \{u_t^*, \{\alpha_{t_l}^*\}_{l=1}^J, D\} = \min_{u_t, \alpha_{t_l}, D} & \left\{ \frac{1}{2} \|f - \Phi u_t\|_{\ell_2}^2 + \frac{1}{2} \|u_t - v^*\|_{\ell_2}^2 \right. \\ & \left. + \sum_{l=1}^J \mu'_l \|\alpha_{t_l}\|_{\ell_0} + \frac{\lambda'}{2} \sum_{l=1}^J \|D_l \alpha_{t_l} - R_l u_t\|_{\ell_2}^2 \right\}, \end{aligned} \quad (35)$$

where $\mu_l, l = 1, \dots, J$, are some positive regularization parameters that control the image patch sparsity. Also, λ' is a weight parameter which controls the trade-off between the data fidelity and the image priority. Here, D_l is the ALS basis of the current frame that can be learned initially using v^* and be updated via the methods of [58–60]. Indeed, the second term measures the distance between the estimated frame and the enhanced one. The

Table 3

The detailed description of the proposed non-key frame recovery framework.

Input: f , Φ_B , α_{t-1}^* , u_{t-1}^* , B_s , λ , τ , μ , iin : inloop iteration number, k_{max} , K_{max} : maximum iteration number, Tol : Tolerance
Output: α_t^* ; u_t^* : Recovered frame;
Initialization: Set $k=0$; $(\alpha^0, b^0, f^0) = (0, 0, 0)$;
 $v^0 = u_{t-1}^*$ (The previous reconstructed frame);
While the stop criterion (1) is not satisfied **do**
1: $v^{k+1} = \tilde{v} = v^k$; $r^{k+1} = v^{k+1} - b^k$;
2: update D^{k+1} using K-SVD [59]
3: **for** each patch r_{p_l} **do**
 compute α_l^{k+1} (see Eq. (34)) using iterative/shrinkage algorithm via
 surrogate function
end for
4: update α^{k+1} by concatenating all $\{\alpha_l^{k+1}\}$ MPS- NO-SPC
5: **for** $i = 1 : iin$
 $g_i^k \leftarrow (\Phi^T(\Phi v^{k+1} - f^k) + \mu(v^{k+1} - D^{k+1} \circ \alpha^{k+1} - b^k))$
 $n_i^k \leftarrow \text{diag} \left(\text{abs} \left(\frac{g_i^k T g_i^k}{g_i^k T (\Phi^T \Phi + \mu I) g_i^k} \right) \right)$
 $v^{k+1} \leftarrow v^{k+1} - n_i^k g_i^k$
end for
6: compute $s^{k+1} = \text{SSIM}(v^{k+1}, \tilde{v})$
7: compute $diff = \text{abs}(s^{k+1} - s^k)$
8: update b^k, f^k using Eqs. (30c) and (30d), respectively
9: $v^* \leftarrow v^{k+1}$
10: $k \leftarrow k+1$
end While
11: $u_t^l \leftarrow v^*$
While the stop criterion (2) is not satisfied **do**
12: **for** each patch $R_l u_t^k$ **do**
 compute $\alpha_{t_l}^{k+1}$ using OMP [67]
end for
13: update α_t^{k+1} by concatenating all $\{\alpha_{t_l}^{k+1}\}$
14: update D_t^{k+1} using K-SVD [59]
15: compute u_t^{k+1} using Eq. (36)
16: $u_t^k \leftarrow u_t^{k+1}$; $\alpha_t^* \leftarrow \alpha_t^{k+1}$
17: $k' \leftarrow k' + 1$
end While

stopping criterion (1): $k = k_{max}$ or $diff \leq Tol$.
stopping criterion (2): $k' = K_{max}$.

problem of Eq. (35) can be solved in an iterative way, by decoupling it into three sub-problems of *sparse coding*, *dictionary learning*, and *reconstruction*. The sparse coding problem can be solved using OMP [67], and D_t (in DL step) is updated by the methods of [58–60]. Also, the reconstruction step has a closed-form solution as follows:

$$u_t^{k+1} = \left(\Phi \Phi^T + I + \lambda' \sum_{l=1}^J R_l^T R_l \right)^{-1} \left(v^* + \Phi^T f + \lambda' \sum_{l=1}^J R_l^T D_l \alpha_{t_l}^* \right), \quad (36)$$

where the superscript k' denotes the iteration number that is dropped in right side of Eq. (36) without confusion. In fact, not only is Eq. (35) used to update the variables which are used for the recovery of the next frame, but also to achieve an additional enhancement for the recovered frame by Eq. (29). The proposed non-key frame recovery algorithm is summarized in detail in Table 3.

5. Experimental results

In this section, we evaluate the performance of the proposed method, and compare it with benchmark methods. To evaluate our simulation results, we use two applicable quality assessors, the peak signal-to-noise ratio (PSNR) in dB and the structural

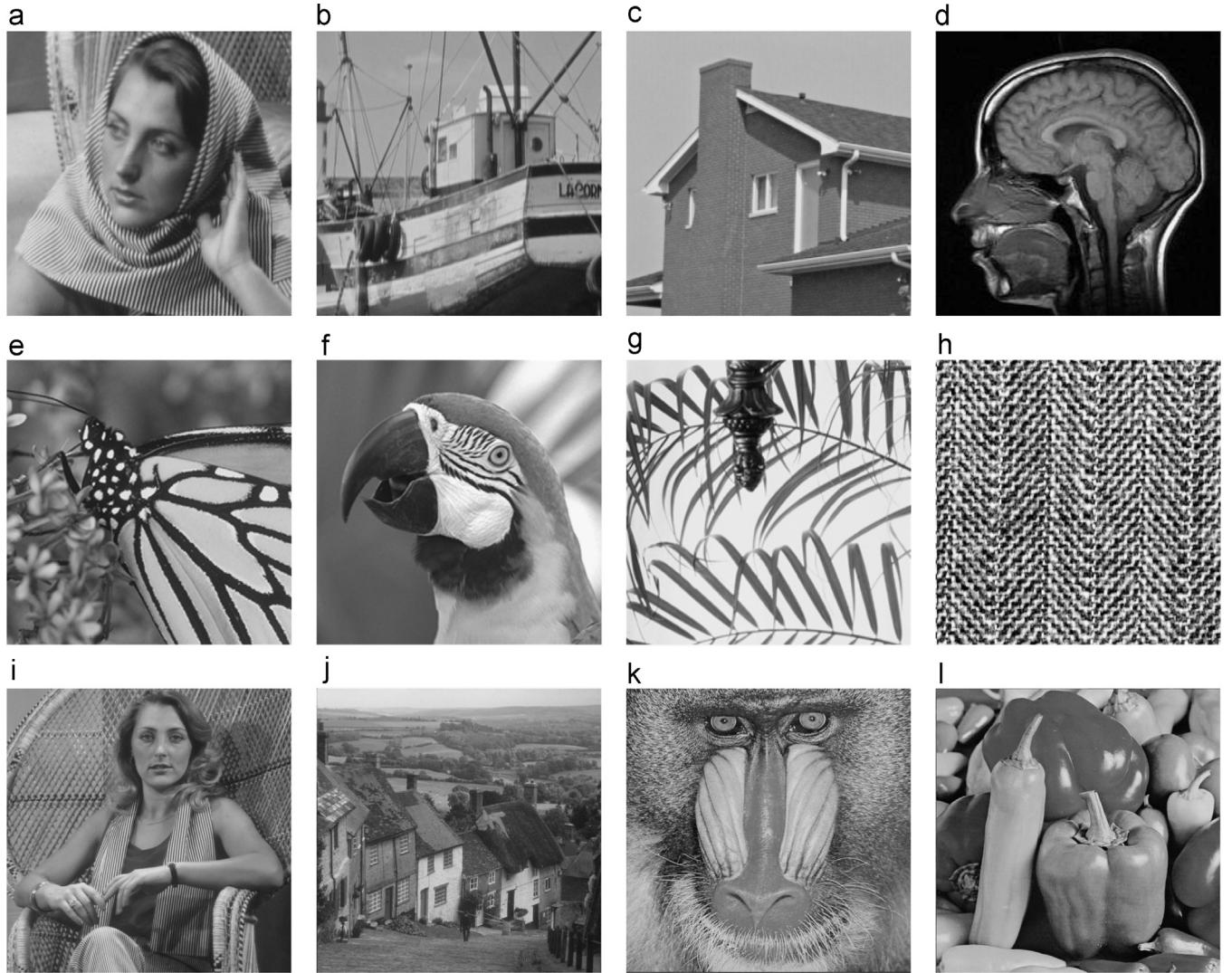


Fig. 2. The test images used in the experiments. (a)–(h) 256 × 256 test images of *Barbara*, *Boat*, *House*, *MRI*, *Monarch*, *Parrot*, *Leaves* and *Texture*, respectively. (i)–(l) 512 × 512 test images of *Barbara2*, *Goldhill*, *Mandrill* and *Peppers*, respectively.

similarity (SSIM) [72]. The performances of our experiments are evaluated on the luminance component of the test images shown in Fig. 2. Also, the performances of our experiments for CVS recovery are evaluated on the luminance component of eight well-known video test sequences shown in Fig. 3. These test images/videos are commonly used in other related publication, which enables a fair comparison of results. Note that, in this paper we do not consider quantization and entropy encoding of measurements, since they are beyond the scope of this paper. Also, in all experiments we use the block size of 32×32 and the CS measurement of each block is obtained by applying a dense Gaussian projection matrix to each of them. All experiments were performed using MATLAB 2012a, on a computer equipped with Intel® core™ i7, 3.5 GHz processor, with 16 GB of RAM, and running on Windows 7. Due to the space limitations, only parts of the experimental results are shown in this paper. Interested readers may contact the corresponding author for all the images, the video frames and the source codes.

5.1. Which one should be adopted in dictionary learning step: MOD, K-SVD or MDU?

There are several DL methods which can be adopted in our scheme. However, in this experiment, we evaluate the performance of

the three well-known DL methods of [58–60]. Toward this approach, we do not consider the stopping criterion $\text{diff} \leq \text{Tol}$ and the steps 5–7 in Table 2, and set $\mu_2 = 0$ in step 8. The CS image is reconstructed initially using the method of [44], in order to use as the initial training image in the process of learning an ALS basis (dictionary). The ALS basis is obtained by each of the methods introduced in [58–60], in which the default parameter setting is as follows: the size of sparsifying basis (dictionary) is 256 and the number of training iterations is 20. The size of each patch ($\sqrt{B_s} \times \sqrt{B_s}$), in process of obtaining an ALS basis, is set to 8×8 . Also, in the corresponding recovery algorithm (see Table 2), $iin = 200$, $k_{max} = 40$, $\text{Tol} = 10^{-4}$, $\mu_1 = 2.5 \times 10^{-3}$ and λ is set, empirically, between 4 and 8.

In Fig. 4, a forty-time-iteration of our recovery method is illustrated for recovery of the test images, *Boat* and *House*, with different measurement ratios (MR), via DL algorithms of MOD, K-SVD and MDU. Also, the average running times of each DL algorithm per iteration (corresponding to Fig. 4) are shown in Table 4. Obviously, the results show that MDU provides a better recovery performance (in quality) compared to the other two methods, but with higher computational complexity. The extra cost is only generated by the dictionary update step in each iteration. Also, the PSNR performance of K-SVD and MOD is very close, but the first one has lower computational complexity. Thus, it seems reasonable to adopt K-SVD as the DL algorithm in our method. The same



Fig. 3. The test videos used in the experiments: (a)–(d) "Foreman", "Coastguard", "Mobile and Calendar" and "Hall Monitor" with a CIF resolution of 352×288 pixels registered in 30 frames/second; (e)–(h) "Crew", "City", "Harbor" and "Soccer" with a 4CIF resolution of 702×576 pixels registered in 30 frames/second. The 25th frame of each video sequence is shown.

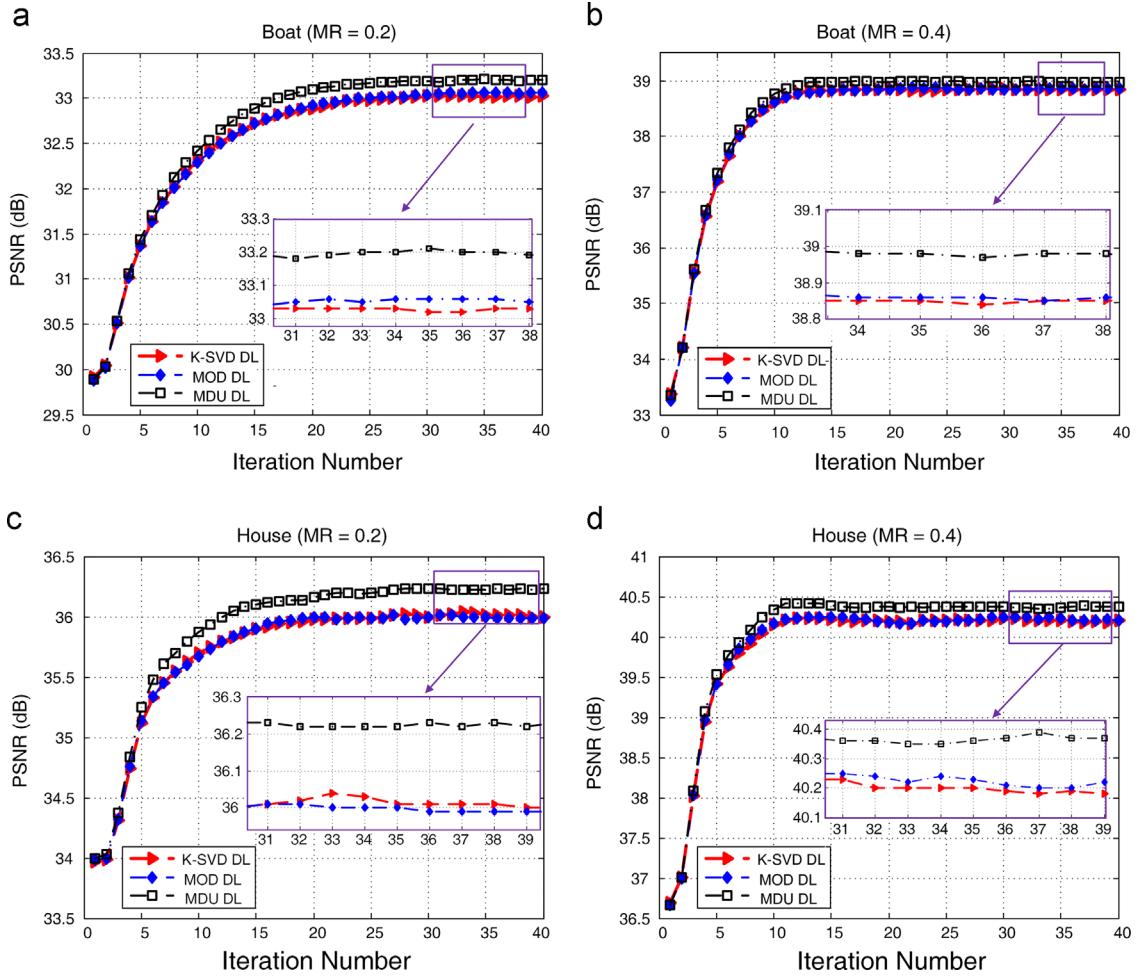


Fig. 4. Performance of different DL algorithms for the proposed method, for *Boat* and *House* with $MR=0.2$ and 0.4 , in terms of PSNR vs. iteration number.

Table 4

Average execution time (in seconds) of different DL algorithms for each iteration (corresponding to Fig. 4).

DL algorithm	Boat		House	
	MR=0.2	MR=0.4	MR=0.2	MR=0.4
MOD [58]	8.706	10.440	6.315	7.436
K-SVD [59]	8.149	9.875	5.628	6.799
MDU [60]	53.904	56.970	46.810	49.515

experiments were run on other test images. The obtained results confirm the suitability of the adaption of K-SVD.

5.2. Comparison with state-of-the-art techniques

To demonstrate the effectiveness of the proposed method in CS image recovery, we have compared it with several competitive CS image recovery techniques: (i) CS image recovery using TV via augmented Lagrangian method (TVAL3) [73]; (ii) block-based CS image recovery using smoothed projected Landweber (BCS-SPL) [74]; (iii) TV based CS image recovery via nonlocal regularization (TVNLR) [75]; (iv) MH prediction method [44]; (v) RCoS [34]; (vi) model-assisted CS recovery method (MARX-PC) [32]; and (vii) CS image recovery using ALS basis (we refer to it as CS-ALS) [51]. It is worth emphasizing that the last four of these methods are known as the current state-of-the-art methods for CS image recovery. The source codes of all benchmark methods of [32,34,44,51,73–75] were

obtained from their respective authors' websites. To make a fair comparison among the competing methods, we have carefully tuned their parameters to achieve the best performance.

In our implementation, in process of finding NSSM for self-similarity in 3D transform domain, the size of each patch (i.e., $\sqrt{B_p} \times \sqrt{B_p}$) is set to 8×8 . The size of training window for searching matched blocks (i.e., $\omega \times \omega$) is set to 20×20 , and the number of best matched blocks (i.e., c) is set to 10. The other main parameters of the proposed algorithm are empirically set, $k_{max} = 20$, $\mu_1 = 0.2\mu$, $\mu_2 = 0.8\mu$, and $\mu = 2.5 \times 10^{-3}$. Also, the orthogonal 3D transform denoted by T^{3D} is composed of 2D Bior 1.5 and 1D Haar transform.

Tables 5 and 6 list the average PSNR and SSIM results of the recovered images by competing CS image recovery methods, for 10%, 30% and 50% of measurements (best results are emphasized in bold face). All experimental results are averaged over 10 independent trials. Note that, for the sake of fair comparison, the same test conditions are used in all experiments; i.e., the same sensing matrices are applied for BCS-SPL, TVNLR, MH, RCoS, CS-ALS and the proposed CS-JASM. From Tables 5 and 6, we can see that TVAL3 obtains the lowest PSNR and SSIM among all compared algorithms, and the proposed CS-JASM considerably outperforms the other competing methods in most cases. In the case 256×256 (512×512) CS image recovery, the average PSNR gain of CS-JASM over BCS-SPL, TVNLR, MH, RCoS, MARX-PC and CS-ALS can be up to 8.62 dB (3.63 dB), 5.9 dB (2.82 dB), 3.94 dB (1.75 dB), 3.3 dB (1.7 dB), 2.51 dB (1.25 dB) and 2.02 dB (0.68 dB), respectively.

Table 5PSNR/SSIM comparisons of the proposed CS-JASM with the various CS image recovery methods for recovery of test images (of size 256×256) in different MR scenarios.

Test image	MR	Algorithm							
		TVAL3 [73]	BCS-SPL [74]	TVNLR [75]	MH [44]	RCoS [34]	MARX-PC [32]	CS-ALS [51]	CS-JASM (proposed)
Barbara	0.1	21.19/0.556	22.65/0.608	22.48/0.622	27.46/0.833	23.77/0.683	24.25/0.652	27.85/0.828	29.54/0.885
	0.3	23.97/0.757	25.54/0.752	26.04/0.819	33.35/0.941	30.80/0.931	33.25/0.933	34.92/0.960	36.26/0.968
	0.5	25.42/0.888	28.43/0.853	30.13/0.922	36.86/0.968	35.46/0.964	37.88/0.969	39.51/0.982	40.34/0.989
Boat	0.1	23.70/0.664	24.49/0.662	25.49/0.743	26.39/0.759	27.39/0.801	27.80/0.788	28.29/0.829	30.27/0.877
	0.3	27.55/0.863	28.44/0.798	31.95/0.911	32.18/0.906	34.78/0.934	34.81/0.937	36.31/0.960	37.35/0.968
	0.5	27.12/0.931	31.39/0.872	37.54/0.960	35.82/0.948	38.62/0.972	39.03/0.969	40.89/0.981	41.45/0.986
Fence	0.1	19.06/0.438	18.71/0.527	19.58/0.560	24.09/0.749	24.13/0.704	25.36/0.755	24.91/0.781	25.93/0.800
	0.3	23.24/0.706	24.02/0.744	25.02/0.807	28.91/0.872	29.28/0.871	30.02/0.879	30.66/0.907	31.25/0.915
	0.5	23.70/0.858	26.95/0.848	30.90/0.926	32.13/0.928	33.31/0.936	33.59/0.934	34.90/0.958	35.32/0.963
House	0.1	19.21/0.736	26.77/0.769	29.74/0.834	29.60/0.815	30.75/0.847	32.90/0.845	32.16/0.860	34.48/0.888
	0.3	23.05/0.880	32.92/0.879	35.24/0.909	35.75/0.917	37.12/0.921	36.89/0.912	38.48/0.946	39.26/0.953
	0.5	23.67/0.933	36.49/0.928	38.71/0.948	38.74/0.955	40.31/0.957	40.40/0.956	41.87/0.971	42.55/0.978
Leaves	0.1	16.79/0.596	18.26/0.553	19.42/0.712	20.44/0.720	25.48/0.883	24.27/0.793	25.13/0.851	28.82/0.913
	0.3	24.03/0.872	22.33/0.733	27.07/0.935	31.01/0.924	30.87/0.965	34.09/0.976	31.91/0.938	34.31/0.981
	0.5	31.04/0.959	25.62/0.827	31.63/0.975	35.56/0.955	36.28/0.988	38.61/0.990	37.16/0.969	39.39/0.993
Monarch	0.1	21.11/0.695	21.73/0.692	22.99/0.773	23.05/0.762	24.20/0.802	27.11/0.839	25.26/0.837	27.05/0.877
	0.3	23.40/0.889	27.30/0.845	29.71/0.926	29.34/0.903	32.47/0.951	34.12/0.947	31.24/0.943	33.82/0.960
	0.5	22.81/0.936	31.27/0.910	31.46/0.952	32.96/0.941	37.04/0.977	38.88/0.975	37.11/0.978	38.72/0.982
MRI	0.1	16.69/0.472	22.63/0.687	23.59/0.689	26.89/0.758	26.91/0.764	27.88/0.799	28.01/0.813	30.03/0.856
	0.3	24.15/0.727	31.19/0.875	32.14/0.902	33.54/0.911	34.13/0.918	34.32/0.922	34.49/0.929	35.52/0.939
	0.5	28.07/0.828	34.68/0.927	35.11/0.942	36.63/0.948	37.07/0.955	38.61/0.963	38.60/0.965	39.28/0.969
Texture	0.1	1.95/0.244	11.25/0.180	11.94/0.328	13.01/0.620	12.02/0.414	11.31/0.212	13.10/0.628	14.82/0.729
	0.3	6.20/0.587	12.81/0.484	15.14/0.701	17.51/0.839	16.30/0.750	15.25/0.689	17.74/0.849	18.71/0.884
	0.5	9.46/0.775	14.95/0.705	17.50/0.836	20.26/0.917	18.92/0.861	17.94/0.884	21.03/0.933	21.70/0.942
Average	0.1	17.47/0.551	20.81/0.558	21.90/0.657	23.19/0.738	24.32/0.736	25.11/0.710	25.60/0.803	27.62/0.863
	0.3	21.95/0.785	25.57/0.764	27.41/0.808	29.90/0.902	30.66/0.906	31.51/0.900	31.96/0.929	33.31/0.946
	0.5	23.91/0.889	28.72/0.858	31.62/0.933	33.40/0.945	34.63/0.951	35.61/0.955	36.38/0.967	37.34/0.976

Table 6PSNR/SSIM comparisons of the proposed CS-JASM with the various CS image recovery methods for recovery of test images (of size 512×512) in different MR scenarios.

Test image	MR	Algorithm							
		TVAL3 [73]	BCS-SPL [74]	TVNLR [75]	MH [44]	RCoS [34]	MARX-PC [32]	CS-ALS [51]	CS-JASM (proposed)
Barbara2	0.1	23.46/0.585	23.04/0.667	23.63/0.631	26.68/0.792	24.43/0.684	24.97/0.711	27.22/0.801	27.76/0.820
	0.3	27.58/0.808	28.31/0.842	27.67/0.842	31.98/0.918	30.52/0.899	31.13/0.906	32.69/0.932	33.18/0.938
	0.5	28.09/0.904	31.67/0.914	31.22/0.926	35.03/0.954	34.31/0.940	35.78/0.959	37.32/0.972	37.71/0.974
Goldhill	0.1	25.80/0.654	24.17/0.640	27.16/0.692	27.07/0.706	27.44/0.718	28.17/0.714	27.83/0.727	28.61/0.756
	0.3	27.36/0.825	30.66/0.825	31.35/0.847	31.82/0.864	32.03/0.871	32.19/0.848	32.64/0.880	33.04/0.889
	0.5	27.46/0.901	33.12/0.892	33.84/0.911	34.33/0.914	34.75/0.928	34.80/0.922	35.86/0.937	36.28/0.943
Mandrill	0.1	16.67/0.403	19.54/0.456	20.07/0.442	19.61/0.470	19.57/0.453	19.70/0.476	19.76/0.471	20.39/0.535
	0.3	18.29/0.613	22.88/0.674	22.25/0.664	23.98/0.745	24.11/0.724	24.07/0.751	24.13/0.757	24.53/0.776
	0.5	18.98/0.787	25.15/0.803	24.61/0.810	26.64/0.854	27.02/0.851	26.95/0.862	27.15/0.874	27.46/0.883
Peppers	0.1	26.56/0.755	27.16/0.764	29.92/0.816	29.93/0.812	30.17/0.824	30.28/0.820	30.90/0.829	31.65/0.839
	0.3	31.10/0.781	33.68/0.854	33.71/0.881	34.35/0.885	34.55/0.883	34.71/0.887	35.16/0.896	35.51/0.903
	0.5	32.64/0.914	36.63/0.922	36.51/0.921	36.80/0.925	37.10/0.933	37.35/0.929	38.08/0.940	38.34/0.944
Average	0.1	23.12/0.599	23.47/0.632	25.20/0.645	25.82/0.695	25.40/0.670	25.78/0.680	26.42/0.707	27.10/0.737
	0.3	26.08/0.757	28.88/0.799	28.74/0.808	30.53/0.853	30.30/0.844	30.52/0.848	31.14/0.866	31.56/0.877
	0.5	26.79/0.876	31.64/0.883	31.54/0.892	33.20/0.912	33.29/0.913	33.72/0.918	34.60/0.931	34.95/0.936

For visual comparison, Figs. 5–7 show some CS recovered images by the competing methods. Better visual comparison can be made by zooming the images on the screen. Although the CS recovered images by CS-ALS, MARX-PC and RCoS achieve much better visual quality than those of TVAL3, BCS-SPL, TVNLR and MH, they still suffer from some undesirable artifacts (such as ringing, jaggy and staircase artifacts) and lost details. It can be observed that the proposed CS-JASM produces not only sharp large-scale

edges but also fine-scale image details. It preserves the image structures and textures better than the other competing methods and reproducing clearer images. The high performance of the proposed algorithm is attributed to the efficient employment of both locally adaptive sparsity and nonlocal self-similarity constraint in the proposed JASM model which offers a powerful mechanism for characterizing the structured sparsities of natural images.



Fig. 5. CS recovered *Barbara* images with 10% of measurements ($MR=0.1$). The recovered results by: (a) TVAL3 [73]; (b) BCS-SPL [74]; (c) TVNLR [75]; (d) MH [44]; (e) RCoS [34]; (f) MARX-PC [32]; (g) CS-ALS [51]; and (i) the proposed CS-JASM. For numerical comparison, see Table 5.

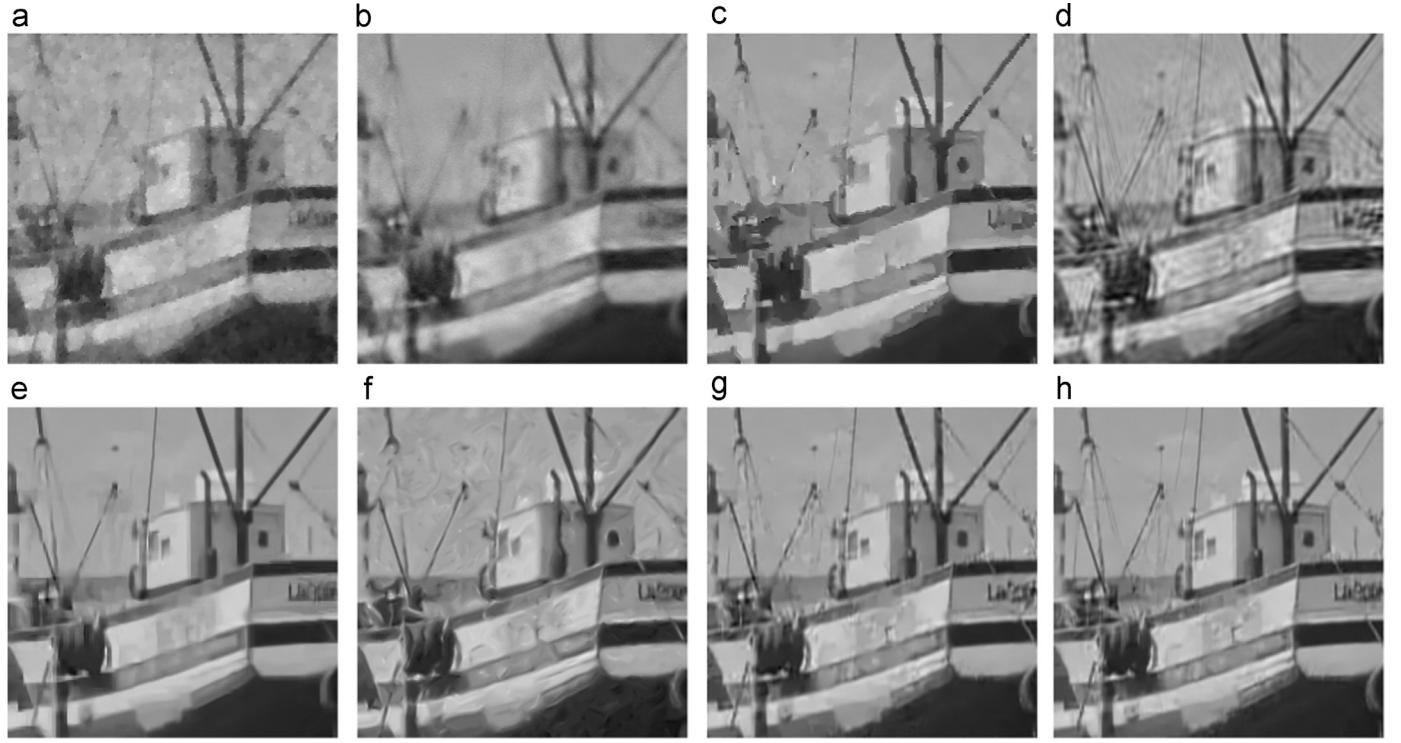


Fig. 6. CS recovered *Boat* images with 10% of measurements ($MR=0.1$). The recovered results by: (a) TVAL3 [73]; (b) BCS-SPL [74]; (c) TVNLR [75]; (d) MH [44]; (e) RCoS [34]; (f) MARX-PC [32]; (g) CS-ALS [51]; and (i) the proposed CS-JASM. For numerical comparison, see Table 5.

5.3. Discussions: comparison between ℓ_0 and ℓ_1 minimization, algorithm convergence, and the 3D transform T^{3D}

In the proposed JASM prior, described in Eq. (6), the nonlocal self-similarity constraint is found by summing all the ℓ_1 norm of

the transform coefficients (i.e., $\Psi_{\text{NSSM}}(u) = \|\Theta_u\|_{\ell_1}$). Therefore, the related sub-problem described in Eq. (22) is solved via soft-thresholding. However, if the nonlocal self-similarity constraint was written as sum of all ℓ_0 pseudo-norm of the transform coefficients (number of nonzero coefficients), it would be solved via

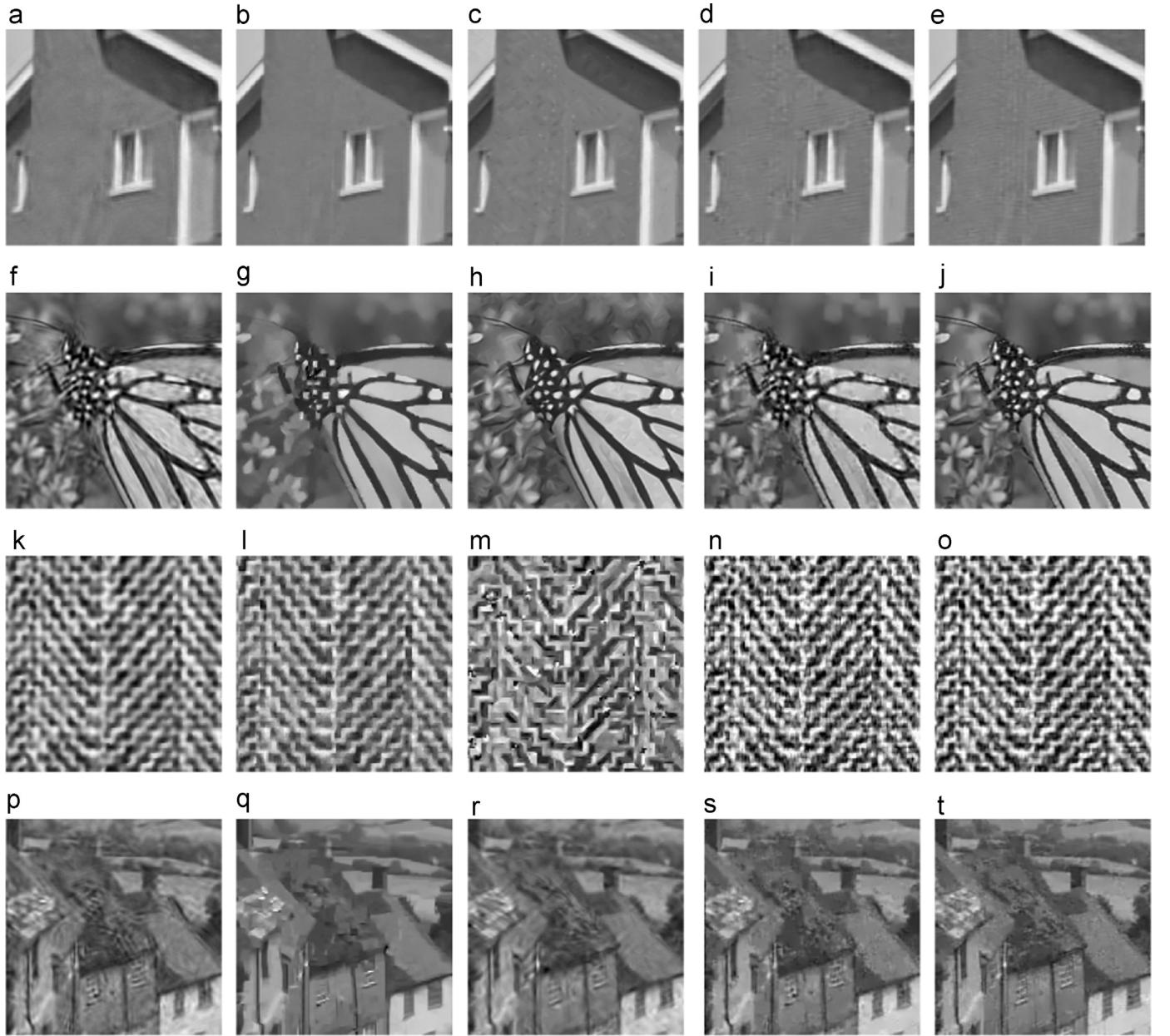


Fig. 7. CS recovered *House*, *Monarch*, *Texture* and *Goldhill* images with 20%, 10%, 30% and 10% of measurements, respectively. From left to right column, the recovered results by: MH [44]; RCos [34]; MARX-PC [32]; CS-ALS [51]; and the Proposed CS-JASM. For numerical comparison, see Tables 5, 6 and 9. Cropped portions of the reconstructed *House*, *Texture* and *Goldhill* images are shown.

hard-thresholding—similar to that carried out in [34]. In this experiment, we make a comparison between using ℓ_1 and ℓ_0 minimization in solving Eq. (22) for the proposed method. For comparing the efficiency of utilizing ℓ_1 norm over ℓ_0 pseudo-norm in achieving the nonlocal self-similarity constraint, Fig. 8 plots some progression curves of PSNR vs. the iteration number, just by considering $k_{max} = 80$ as the stopping criterion. As can be seen, the results obtained by using ℓ_1 norm (solid line curve; the proposed CS-JASM) has better performance than those obtained by using ℓ_0 pseudo-norm (dashed line curve). This corroborates the convenience and superiority of using ℓ_1 over ℓ_0 in the used nonlocal self-similarity constraint, and validates the effectiveness of our approach to solve Eq. (22) via soft-thresholding.

Since the JASM prior is composed of mixed ℓ_0 and ℓ_1 norm, consequently the objective function described in (8) is nonconvex; hence, it is difficult to give its theoretical proof for convergence.

For the convergence study of the proposed method, we only provide empirical evidence to illustrate it. From Fig. 8, it can be observed that with the growth of iteration number, all the PSNR curves increase monotonically and ultimately become flat and stable which fully demonstrates the convergence of the proposed algorithm. This fact motivated us to introduce the second stopping criterion, as illustrated in Table 2, to prevent further iterations after convergence of the algorithm.

As mentioned previously, the 3D transform used in JSM [12] and CoSM [34] is composed of 2D DCT and 1D Haar transform; however, in this paper, we employ a T^{3D} which is composed of 2D Bior 1.5 and 1D Haar transform. The results corresponding to the effects of using these two various T^{3D} are presented in Table 7. As can be seen, the results obtained by utilizing the latter T^{3D} are slightly better than the first one, which fully demonstrate the convenience of this adoption.

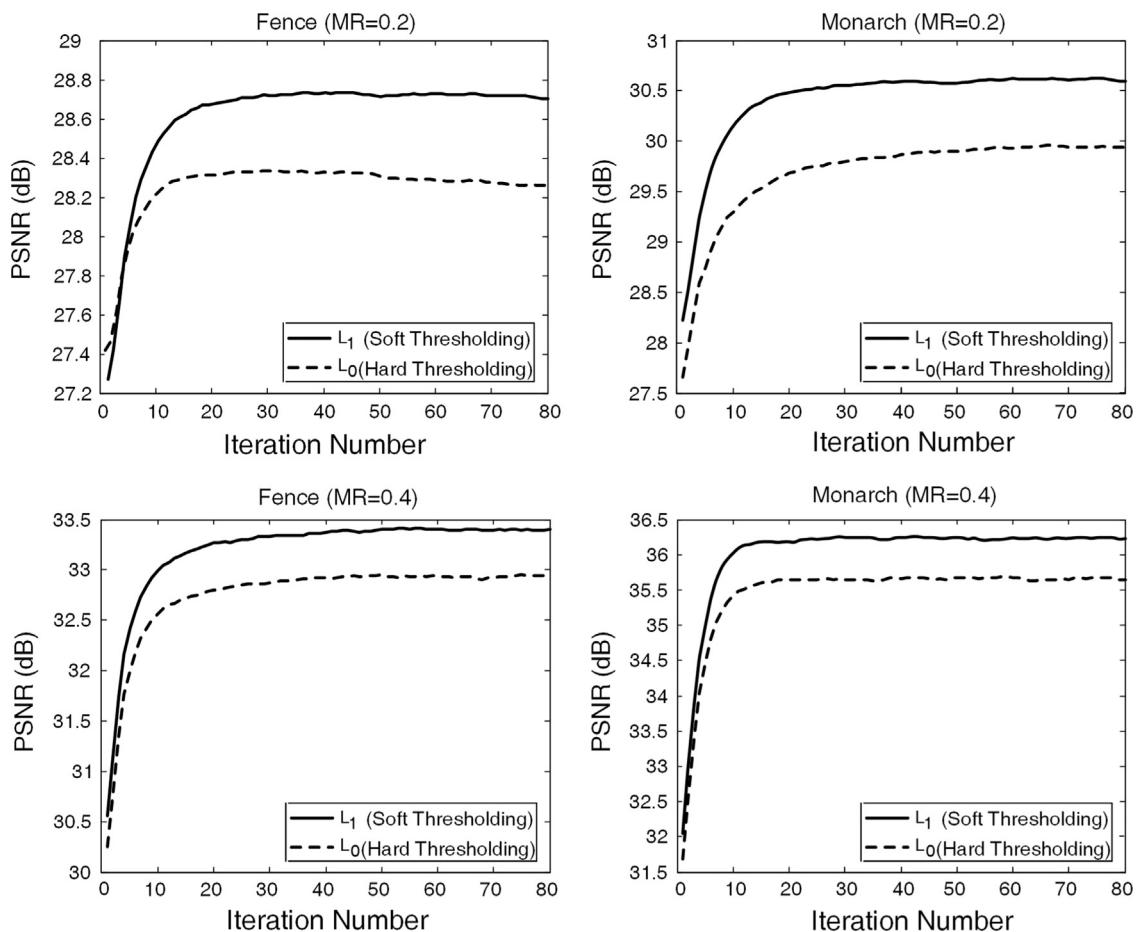


Fig. 8. Comparison between the efficiency of utilizing ℓ_1 and ℓ_0 for the nonlocal self-similarity constraint, in the proposed JASM prior. Progression of the PSNR vs. iteration number, for images *Fence* $\text{MR} \in \{0.2, 0.4\}$ (left plots), and *Monarch* $\text{MR} \in \{0.2, 0.4\}$ (right plots).

Table 7
PSNR/SSIM and NSSM time (seconds) of the recovered *Boat* and *House* images by the proposed method with two different 3D transforms. The NSSM time corresponds to the time consumed for solving the *w* sub-problem of (10c) in each iteration. Those of NSSM time are reported in parentheses.

T^{3D}	Boat		House	
	MR=0.2	MR=0.4	MR=0.2	MR=0.4
2D DCT	33.24/0.929	39.19/0.974	36.28/0.919	40.55/0.965
+1D Haar	(8.325)	(8.870)	(7.664)	(8.317)
2D Bior 1.5	33.45/0.931	39.28/0.975	36.44/0.921	40.62/0.966
+1D Haar	(8.048)	(8.418)	(7.347)	(8.254)

5.4. Performance evaluations of CVS-JASM

In this subsection, we conduct experiments to evaluate the performance of our proposed CVS recovery method (CVS-JASM) for decoding of the first 50 frames of the test video sequences with different MR scenarios. To verify the performance of the proposed CVS-JASM, we compare it with several competitive CVS recovery methods, including (i) the 2D dual-tree discrete wavelet transform (DDWT) [76] basis intra-frame decoder [74] (with 3 levels of decomposition), (ii) MH Inter-frame decoder [44], (iii)–(v) the methods proposed in [50,52,54] (referred to them as CVS-DL, 3DCS and BC-ADMM, respectively). The group of the pictures (GOP) size is set to 5, $k_{\max} = 14$, $k'_{\max} = 4$ and the default parameter setting is the same as the experiments ran in Section 5.2.

Fig. 9 shows the frame-by-frame variations of PSNR and SSIM values of the recovered *Foreman* and *Mobile and Calendar* video

frames. From the PSNR and SSIM results shown in Fig. 9, it is observed that the proposed CVS-JASM consistently performs best among all competitive methods for recovery of *Foreman* and *Mobile and Calendar* video sequences. However, the second best method in recovery of *Mobile and Calendar*, i.e., 3DCS, weakly performs on boundary frames and has a lower performance compared to the CVS-JASM—see Fig. 9, bottom figures. As can be seen, the proposed CVS-JASM surpasses the other methods, especially in recovery of some frames which are known as key frames and also their immediate neighboring frames. This implies that by reducing the GOP size, the performance of CVS-JASM will be increased (see Table 8) but at the cost of higher computational cost since more frames should be recovered as the key frames. To balance the performance and computational cost, we choose $\text{GOP}=5$ for the proposed CVS-JASM. For better visual comparison, Figs. 10–12 show the different decodings of the 18th frame of *Foreman*, 28th frame of *Mobile and Calendar*, and 43rd frame of *City*. As can be seen, the improvements via CVS-JASM are highly evident in recovering edge information, and superior reducing the artifacts in the recovered frames.

Figs. 13 and 14 show the PSNR and SSIM performance of the proposed CVS-JASM compared with the mentioned competitive CVS recovery methods for recovering of CIF and 4CIF video frames under various MR scenarios, respectively. The PSNR and SSIM values shown in Figs. 13 and 14 are averaged over all PSNR and SSIM values of the reconstructed frames. As can be seen from Fig. 13, for the video sequences which contain the camera movements with the objects motion (i.e., *Foreman*, *Coastguard* and *Mobile and Calendar*), CVS-JASM outperforms significantly in both

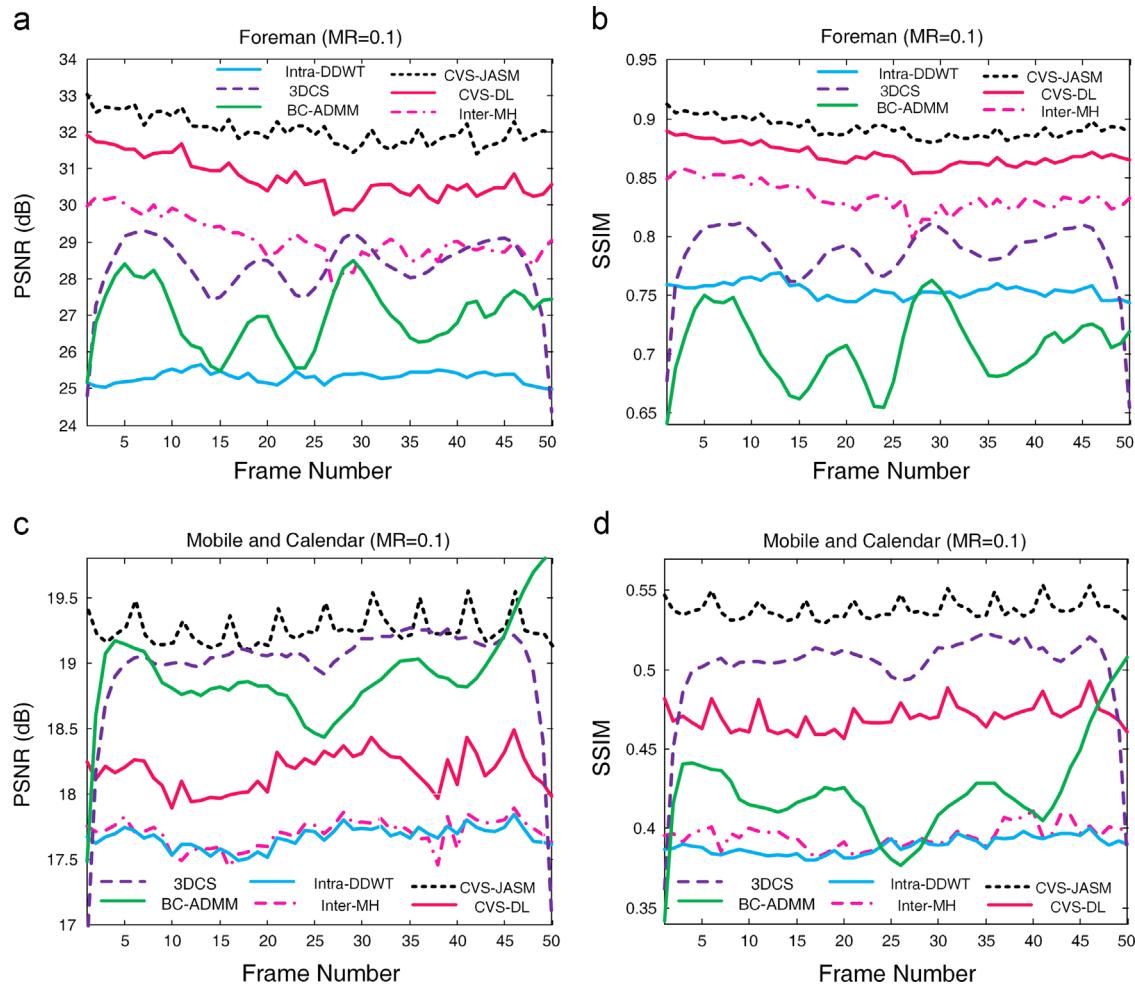


Fig. 9. Frame-by-frame PSNR and SSIM comparisons of the proposed CVS-JASM with the other competitive CVS recovery methods for recovery of (a), (b): *Foreman* and (c), (d): *Mobile and Calendar* video sequences (MR=0.1).

Table 8

Average PSNR/SSIM comparison of the recovered *Foreman* and *Mobile and Calendar* video sequences (MR = 0.1) via the proposed CVS-JASM using different GOP sizes.

Sequence	GOP Size			
	2	5	8	12
Foreman	32.23/0.904	31.98/0.896	31.57/0.889	31.35/0.881
Mobile	19.47/0.551	19.27/0.540	19.01/0.528	18.65/0.516

PSNR and SSIM values compared with the mentioned competitive methods. However, for *Hall Monitor* video sequence which contains statistic camera with moving objects, the BC-ADMM outperforms the CVS-JASM in low MRs. Nevertheless, by growing the MR, the CVS-JASM outperforms the BC-ADMM in both terms of quantitative metrics. In addition, the results shown in Fig. 14 reveal that the CVS-JASM considerably outperforms the mentioned competitive methods with notably large margins which attests the superiority of the proposed method compared with each of which.

5.5. Computational complexity and running time

The computational cost of the proposed CS-JASM algorithm (illustrated in Table 2) mainly comes from two sources:

1. **α sub-problem:** Finding the ALS basis from a fraction of all n patches and exploiting it to obtain sparse approximation of the n patches—steps 2–4.
2. **w sub-problem:** Finding similar patches by searching within search window, stacking into a 3D array, the operation of 3D transform, soft thresholding, the operation of inverse 3D transform, and the final averaging—steps 5–7.

In evaluating the computational complexity of the α sub-problem, steps 2–4 utilize 3 steps of dictionary update, sparse coding (by OMP process), and the final averaging process. The overall cost of all these steps is $\mathcal{O}(P'B_sJLJ')$, where $P' = (\sqrt{n} - \sqrt{B_s} + 1)^2$ is the number of all overlapped image patches of size $\sqrt{B_s} \times \sqrt{B_s}$, extracted from image u . Also, J is the number of iterations in process of DL, L is the number of nonzero elements in each coefficient vector, and J' is the number of atoms in the dictionary. Solving w sub-problem needs $\mathcal{O}(PB_p(c + c \log B_p + \omega^2 + \log \omega^2))$ operations, where P is the number of all overlapped patches u_{p_i} of size $\sqrt{B_p} \times \sqrt{B_p}$, and c is the number of best matched patches in a search window of size $\omega \times \omega$.

The computational time for CS encoding of images (frames) of sizes 256×256 (352×288) is about 0.003 (0.004) s. Also, a running time comparison between the proposed CS-JASM and the other competing state-of-the-art methods in CS image recovery is shown in Table 9. Based on the experiments, the proposed CS-JASM requires about 2–5 min to recover a CS image of size 256×256 , under mentioned environment. Also, the proposed CVS-JASM requires about 12–28 (8–16) s to recover a CS key (non-key)

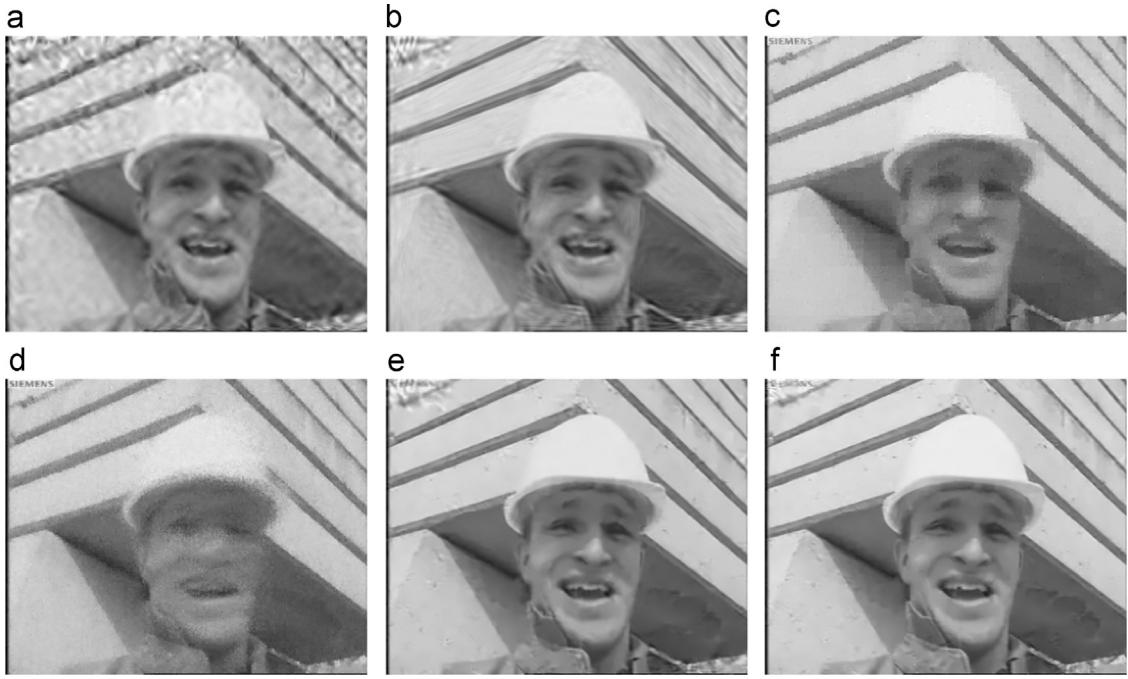


Fig. 10. Different decodings of the 18th frame of *Foreman* w.r.t. recovery as a non-key frame ($MR=0.1$). The reconstructed frame by: (a) The 2D DDWT basis intra-frame decoder [74] ($PSNR=25.40$ dB, $SSIM=0.750$); (b) The MH inter-frame decoder [44] ($PSNR=29.14$ dB, $SSIM=0.829$); (c) 3DCS [52] ($PSNR=28.36$ dB, $SSIM=0.787$); (d) BC-ADMM [54] ($PSNR=26.33$ dB, $SSIM=0.684$); (e) CVS-DL [50] ($PSNR=30.65$ dB, $SSIM=0.873$); and (f) CVS-JASM ($PSNR=32.01$ dB, $SSIM=0.894$).

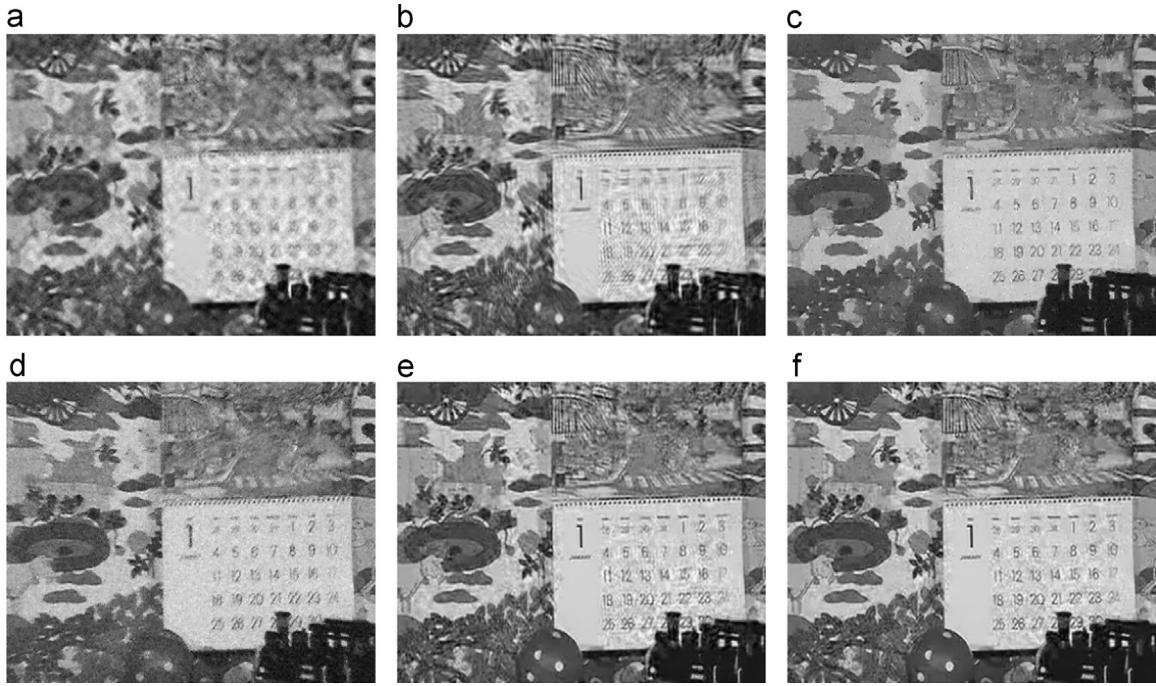


Fig. 11. Different decodings of the 28th frame of *Mobile and Calendar* w.r.t. recovery as a non-key frame ($MR=0.2$). The reconstructed frame by: (a) The 2D DDWT basis intra-frame decoder [74] ($PSNR=19.58$ dB, $SSIM=0.501$); (b) the MH inter-frame decoder [44] ($PSNR=20.55$ dB, $SSIM=0.574$); (c) 3DCS [52] ($PSNR=22.12$ dB, $SSIM=0.670$); (d) BC-ADMM [54] ($PSNR=20.27$ dB, $SSIM=0.512$); (e) CVS-DL [50] ($PSNR=21.59$ dB, $SSIM=0.682$); and (f) CVS-JASM ($PSNR=22.56$ dB, $SSIM=0.722$).

frame of size 352×288 , in each iteration. In our proposed methods, since each patch is reconstructed independently, the α and w subproblems and also the simultaneous sparse coding (for each exemplar patch in non-key frame recovery problem) can be computed in parallel to severely speed up the algorithm (e.g., by processing of multiple patches in parallel with GPU).

6. Conclusion

The motivation of this paper is to introduce a novel sparsity measure called joint adaptive sparsity measure (JASM), and a new strategy for high-fidelity CS image/video restoration via JASM. The proposed JASM efficiently characterizes the intrinsic sparsities of

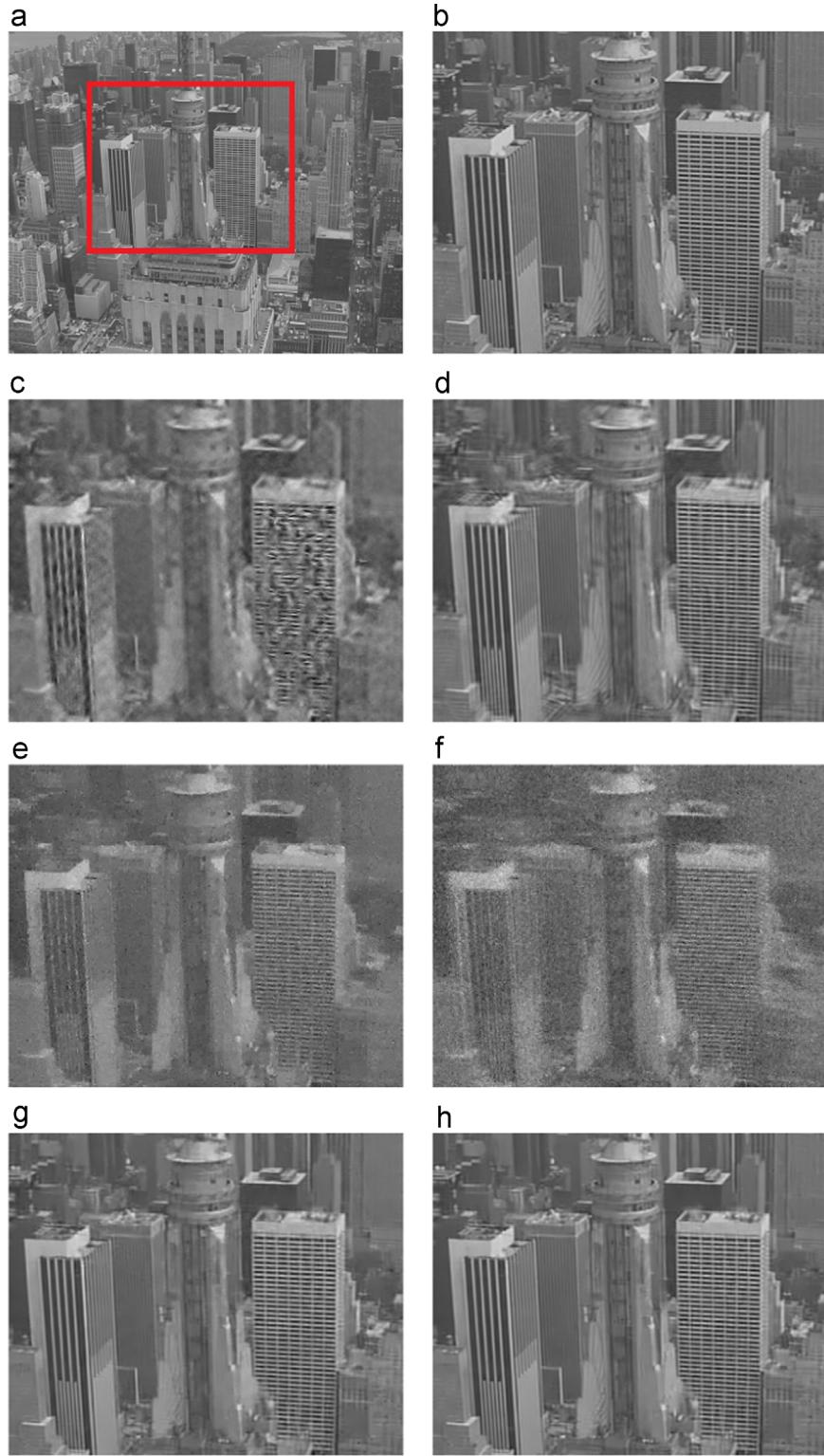


Fig. 12. Different decodings of the 43rd frame of City w.r.t. recovery as a non-key frame ($MR=0.2$). (a) Original frame. (b) Cropped portion of the original frame. The reconstructed frame by: (c) The 2D DDWT basis intra-frame decoder [74] ($PSNR=23.67$ dB, $SSIM=0.671$); (d) the MH inter-frame decoder [44] ($PSNR=29.56$ dB, $SSIM=0.867$); (e) 3DCS [52] ($PSNR=23.28$ dB, $SSIM=0.560$); (f) BC-ADMM [54] ($PSNR=21.19$ dB, $SSIM=0.408$); (g) CVS-DL [50] ($PSNR=31.15$ dB, $SSIM=0.902$); and (h) CVS-JASM ($PSNR=31.49$ dB, $SSIM=0.919$).

natural images by exploiting both local sparsity and nonlocal 3D sparsity, simultaneously. In order to obtain the ALS basis, we investigated the effectiveness of three well-known DL algorithms of MOD, K-SVD and MDU. We found out that MDU provides a better recovery performance (in quality) compared to the K-SVD and MOD, but with higher computational complexity. We found

that it is reasonable to use the K-SVD as the DL algorithm in our proposed scheme; however, the effectiveness of the other DL algorithms can be investigated. Also, we demonstrated the superiority of utilizing ℓ_1 norm over ℓ_0 pseudo-norm in obtaining the nonlocal self-similarity constraint. In addition, we found that, utilizing a \mathcal{T}^{3D} composed of 2D Bior 1.5 and 1D Haar transform

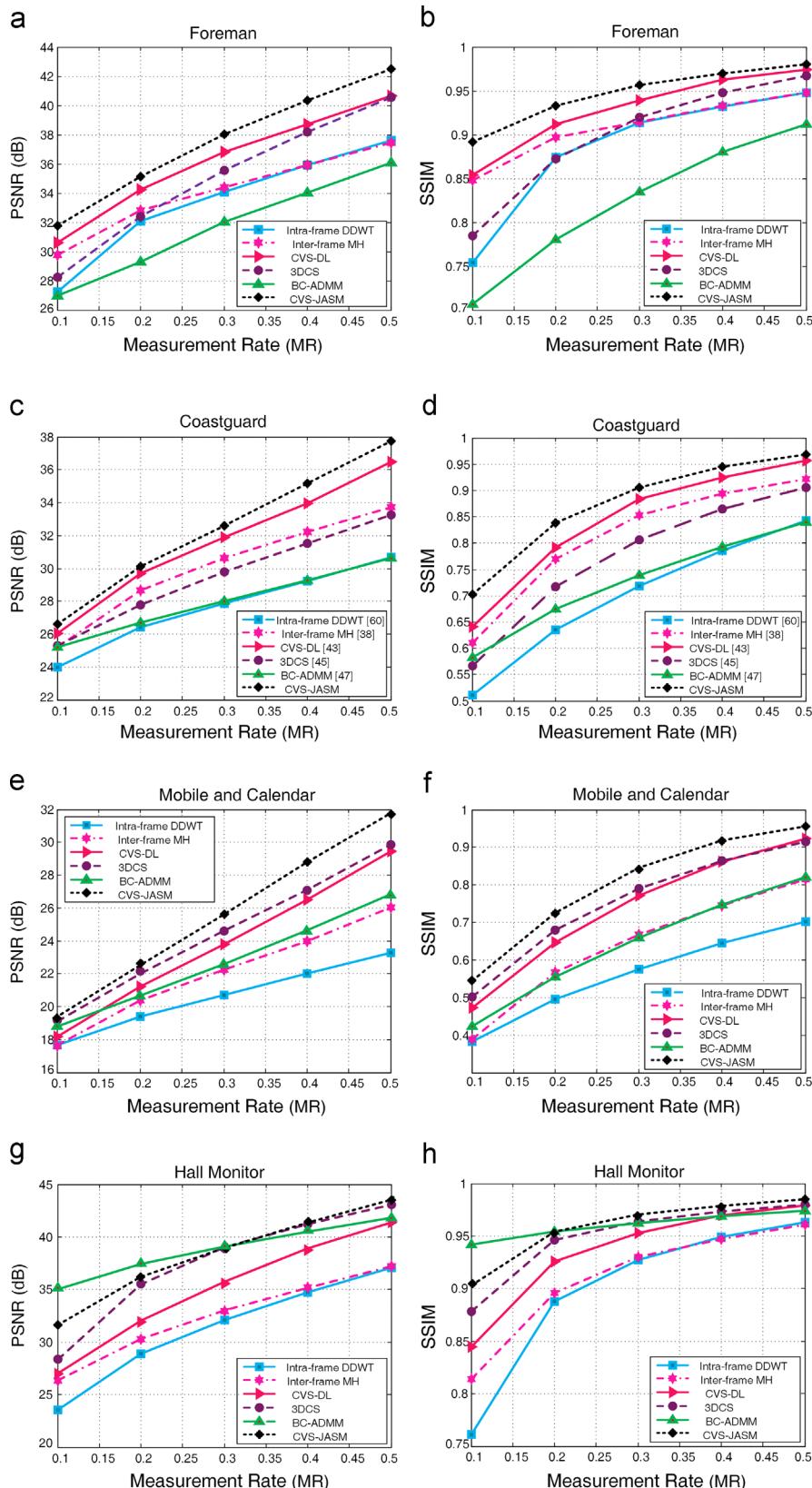


Fig. 13. The average PSNR and SSIM performance of CVS-JASM compared with the competitive CVS recovery methods for recovery of the first 50 frames of the CIF test video sequences. Left column: PSNR vs. MR; Right column: SSIM vs. MR.

gains better performance than utilizing a \mathcal{T}^{3D} composed of 2D DCT and 1D Haar transform. Our extensive experimental results demonstrated that the proposed methods significantly surpasses the current state-of-the-art CS image/video restoration methods

with remarkably large margins in terms of both quantitative metrics and subjective visual quality.

For future work, due to the *ill-posed* nature of image restoration, we would like to extend the proposed sparsity measure,

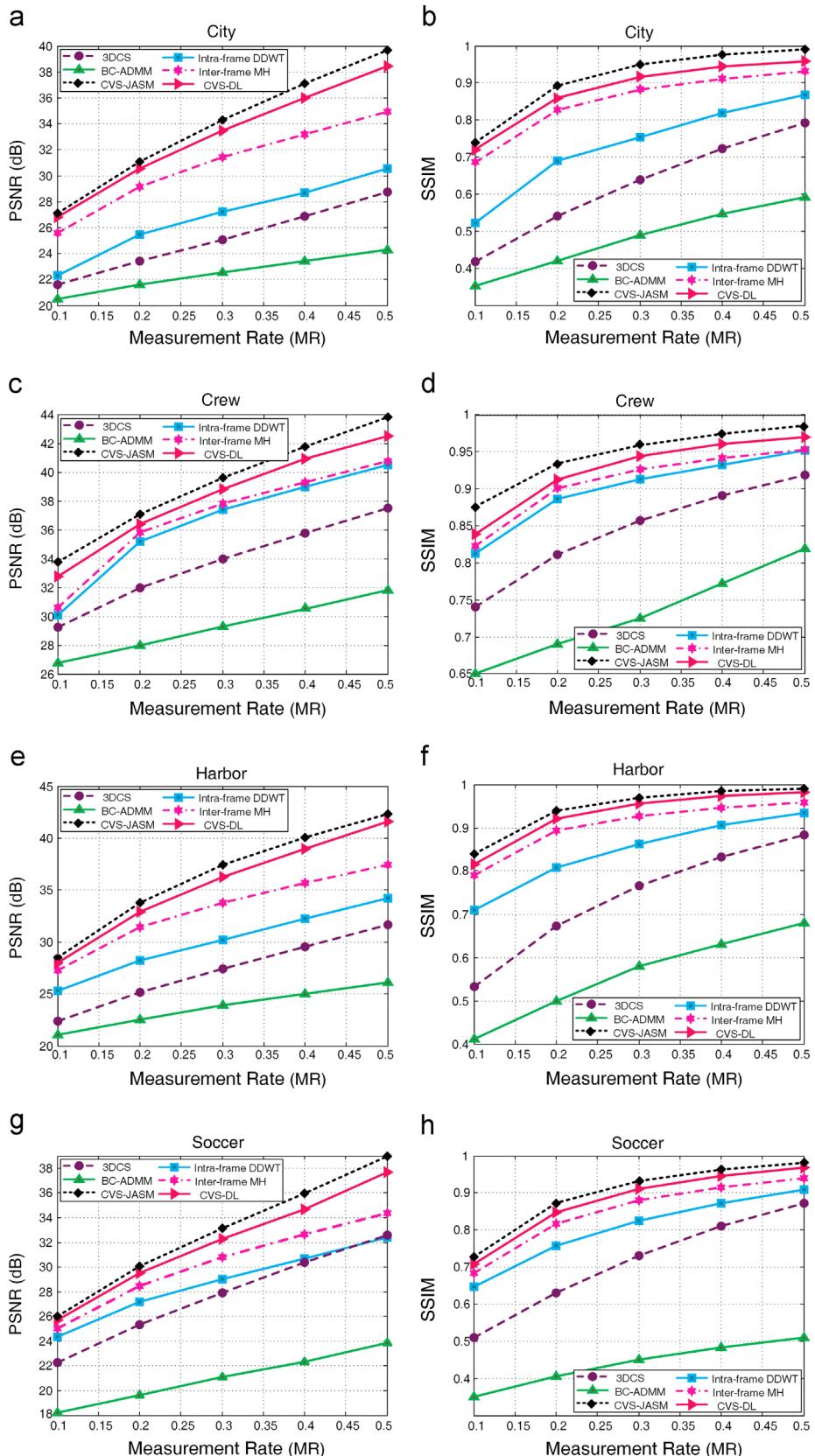


Fig. 14. The average PSNR and SSIM performance of CVS-JASM compared with the competitive CVS recovery methods for recovery of the first 50 frames of the 4CIF test video sequences. Left column: PSNR vs. MR; Right column: SSIM vs. MR.

Table 9

Running time comparison between the proposed CS-JASM and the other competing state-of-the-art methods on image *House* of Size 256×256 with MR = 0.2 (unit: second).

Algorithm	PSNR (dB)	SSIM	Time (s)
MH [44]	33.795	0.882	21
RCoS [34]	35.151	0.894	1170
MARX-PC [32]	35.366	0.885	633
CS-ALS [51]	35.987	0.912	218
CS-JASM	37.112	0.929	289

JASM, to the other image restoration applications, e.g., denoising, inpainting, deblurring, and super-resolution.

Acknowledgments

The authors would like to thank the editors and the anonymous reviewers, whose comments helped improve this paper greatly. They also would like to express their gratitude to Prof. W. Dong (Xidian University) and Dr. J. Zhang (Peking University) for many fruitful discussions; and the authors of [12,32,34,44,51,52,54,73–75] for sharing the source code of their papers used in Section 5.

References

- [1] E. Candès, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, *IEEE Trans. Inf. Theory* 52 (12) (2006) 5406–5425.
- [2] E. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, *IEEE Trans. Inf. Theory* 52 (2) (2006) 489–509.
- [3] D.L. Donoho, Compressed sensing, *IEEE Trans. Inf. Theory* 52 (4) (2006) 1289–1306.
- [4] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D* 60 (1992) 259–268.
- [5] L. Bregman, The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming, *Comput. Math. Math. Phys.* 7 (1967) 200–217.
- [6] I. Daubechies, M. Defrise, C.D. Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint, *Commun. Pure Appl. Math.* 57 (11) (2004) 1413–1457.
- [7] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sci.* 2 (1) (2009) 183–202.
- [8] T. Goldstein, S. Osher, The split Bregman method for ℓ_1 regularized problems, *SIAM J. Imaging Sci.* 2 (2) (2009) 323–343.
- [9] W. Yin, S. Osher, D. Goldfarb, J. Darbon, Bregman iterative algorithms for ℓ_1 minimization with applications to compressed sensing, *SIAM J. Imaging Sci.* 1 (1) (2008) 143–168.
- [10] A.N. Tikhonov, V.Y. Arsenin, *Solutions of Ill-Posed Problems*, V. H. Winston & Sons, Washington, DC, USA, 1977.
- [11] X. Li, Image recovery via hybrid sparse representation: a deterministic annealing approach, *IEEE J. Sel. Top. Signal Process.* 5 (5) (2011) 953–962.
- [12] J. Zhang, D. Zhao, R. Xiong, S. Ma, W. Gao, Image restoration using joint statistical modeling in space-transform domain, *IEEE Trans. Circuits Syst. Video Technol.* 24 (6) (2014) 915–928.
- [13] A. Buades, B. Coll, J.M. Morel, A review of image denoising algorithms, with a new one, *SIAM Multiscale Model. Simul.* 4 (2) (2005) 490–530.
- [14] J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman, Non-local sparse models for image restoration, in: International Conference on Computer Vision, 2009, pp. 2272–2279.
- [15] G. Peyré, S. Bougleux, L.D. Cohen, Non-local regularization of inverse problems, *Inverse Prob. Image* 5 (2) (2011) 511–530.
- [16] M. Jung, X. Bresson, T.F. Chan, L.A. Vese, Nonlocal Mumford–Shah regularizers for color image restoration, *IEEE Trans. Image Process.* 20 (6) (2011) 1583–1598.
- [17] W. Dong, L. Zhang, G. Shi, X. Wu, Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization, *IEEE Trans. Image Process.* 20 (7) (2011) 1838–1857.
- [18] W. Dong, L. Zhang, R. Lukac, G. Shi, Sparse representation based image interpolation with nonlocal autoregressive modeling, *IEEE Trans. Image Process.* 22 (4) (2013) 1382–1394.
- [19] W. Dong, L. Zhang, G. Shi, X. Li, Nonlocally centralized sparse representation for image restoration, *IEEE Trans. Image Process.* 22 (4) (2013) 1620–1630.
- [20] J. Jiang, L. Zhang, J. Yang, Mixed noise removal by weighted encoding with sparse nonlocal regularization, *IEEE Trans. Image Process.* 23 (6) (2014) 2651–2662.
- [21] Y. Romano, M. Protter, M. Elad, Single image interpolation via adaptive non-local sparsity-based modeling, *IEEE Trans. Image Process.* 23 (7) (2014) 3085–3098.
- [22] L. Li, Y. Xie, W. Hu, W. Zhang, Single image super-resolution using combined total variation regularization by split Bregman iteration, *Neurocomputing* 142 (2014) 551–560.
- [23] J. Yu, Y. Rui, D. Tao, Click prediction for web image reranking using multimodal sparse coding, *IEEE Trans. Image Process.* 23 (5) (2014) 2019–2032.
- [24] T. Wen, F. Yang, J. Gu, L. Wang, A novel Bayesian-based nonlocal reconstruction for freehand 3D ultrasound imaging, *Neurocomputing* 168 (2015) 104–118.
- [25] J. Li, J. Wu, H. Deng, J. Liu, A self-learning image super-resolution method via sparse representation and non-local similarity, *Neurocomputing* 184 (2016) 196–206.
- [26] N. Eslahi, H. Mahdavinataj, A. Aghagolzadeh, Mixed Gaussian-impulse noise removal from highly corrupted images via adaptive local and nonlocal statistical priors, in: 9th Iranian Machine Vision and Image Processing (MVIP), 2015, pp. 70–75.
- [27] H. Wang, X. Gao, K. Zhang, J. Li, Image super-resolution using non-local Gaussian process regression, *Neurocomputing* (2016), <http://dx.doi.org/10.1016/j.neucom.2016.01.073>.
- [28] X. Zhang, M. Burger, X. Bresson, S. Osher, Bregmanized nonlocal regularization for deconvolution and sparse reconstruction, *SIAM J. Imaging Sci.* 3 (3) (2010) 253–276.
- [29] W. Dong, X. Yang, G. Shi, Compressive sensing via reweighted TV and nonlocal sparsity regularisation, *IET Electron. Lett.* 49 (3) (2013) 184–186.
- [30] W. Dong, G. Shi, X. Wu, L. Zhang, A learning-based method for compressive image recovery, *J. Vis. Commun. Image Represent.* 24 (7) (2013) 1055–1063.
- [31] W. Dong, G. Shi, X. Li, L. Zhang, X. Wu, Image reconstruction with locally adaptive sparsity and nonlocal robust regularization, *Signal Process. Image Commun.* 27 (10) 1109–1122.
- [32] X. Wu, W. Dong, X. Zhang, G. Shi, Model-assisted adaptive recovery of compressed sensing with imaging applications, *IEEE Trans. Image Process.* 21 (2) (2012) 451–458.
- [33] W. Dong, G. Shi, X. Li, Y. Ma, F. Huang, Compressive sensing via nonlocal low-rank regularization, *IEEE Trans. Image Process.* 23 (8) (2014) 3618–3632.
- [34] J. Zhang, D. Zhao, C. Zhao, R. Xiong, S. Ma, W. Gao, Image compressive sensing recovery via collaborative sparsity, *IEEE J. Emerg. Sel. Top. Circuits Syst.* 2 (3) (2012) 380–391.
- [35] M.B. Wakin, J.N. Laska, M.F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K.F. Kelly, R.G. Baraniuk, An architecture for compressive imaging, in: IEEE ICIP, 2006, pp. 1273–1276.
- [36] M.F. Duarte, M.A. Davenport, D. Takhar, J.N. Laska, T. Sun, K.F. Kelly, R.G. Baraniuk, Single-pixel camera via compressive sampling, *IEEE Signal Process. Mag.* 25 (2) (2008) 83–91.
- [37] M.B. Wakin, J.N. Laska, M.F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K.F. Kelly, R.G. Baraniuk, Compressive imaging for video representation and coding, in: PCS, 2006, pp. 711–716.
- [38] J.Y. Park, M.B. Wakin, A multiscale framework for compressive sensing of video, in: PCS, 2009, pp. 197–200.
- [39] D. Reddy, A. Veeraraghavan, R. Chellappa, P2C2: programmable pixel compressive camera for high speed imaging, in: IEEE CVPR, 2011, pp. 329–336.
- [40] M. Cossalter, G. Valenzise, M. Tagliasacchi, S. Tabaro, Joint compressive video coding and analysis, *IEEE Trans. Multimedia* 12 (3) (2010) 168–183.
- [41] V. Stanković, L. Stanković, S. Cheng, Compressive video sampling, in: 16th European Signal Processing Conference (EUSIPCO), 2008.
- [42] J. Prades-Nobet, Y. MA, T. Huang, Distributed video coding using compressive sampling, in: PCS, 2009, pp. 165–168.
- [43] J. Zheng, E. Jacobs, Video compressive sensing using spatial domain sparsity, *Opt. Eng.* 48 (8) (2009) 1–10.
- [44] C. Chen, E.W. Tramel, J.E. Fowler, Compressed sensing recovery of images and video using multihypothesis predictions, in: 45th Asilomar Conference on Signals, Systems and Computers, 2011, pp. 1193–1198.
- [45] J. Ma, G. Plonka, M.Y. Hussaini, Compressive video sampling with approximate message passing decoding, *IEEE Trans. Circuits Syst. Video Technol.* 22 (9) (2012) 1354–1364.
- [46] Y. Liu, M. Li, D.A. Pados, Motion-aware decoding of compressed-sensing video, *IEEE Trans. Circuits Syst. Video Technol.* 23 (3) (2013) 438–444.
- [47] H.W. Chen, L.W. Kang, C.S. Lu, Dictionary Learning-based distributed compressive sensing, in: PCS, 2010, pp. 210–213.
- [48] H. Liu, B. Song, H. Qin, Z. Qiu, Dictionary learning based reconstruction for distributed compressed video sensing, *J. Vis. Commun. Image Represent.* 24 (8) (2013) 1232–1242.
- [49] X. Gao, F. Jiang, S. Liu, W. Che, X. Fan, D. Zhao, Hierarchical frame based spatial-temporal recovery for video compressive sensing coding, *Neurocomputing* 174 (2015) 404–412.
- [50] N. Eslahi, A. Aghagolzadeh, S.M.H. Andargoli, Recovery of compressive video sensing via dictionary learning and forward prediction, in: 7th International Symposium on Telecommunications (IST), 2014, pp. 833–838.
- [51] J. Zhang, C. Zhao, D. Zhao, W. Gao, Image compressive sensing recovery using adaptively learned sparsifying basis via ℓ_0 minimization, *Signal Process.* 103 (2014) 114–126.
- [52] X. Shu, N. Ahuja, Imaging via three-dimensional compressive sampling (3DCS), in: IEEE ICCV, 2011, pp. 439–446.

- [53] J. Yang, X. Yuan, X. Liao, P. Llull, D.J. Brady, G. Sapiro, L. Carin, Video compressive sensing using Gaussian mixture models, *IEEE Trans. Image Process.* 23 (11) (2014) 4863–4878.
- [54] M.S. Hosseini, K.N. Plataniotis, High-accuracy total variation with application to compressed video sensing, *IEEE Trans. Image Process.* 23 (9) (2014) 3869–3884.
- [55] Y. Shen, J. Li, Z. Zhu, W. Cai, Y. Song, Image reconstruction algorithm from compressed sensing measurement by dictionary learning, *Neurocomputing* 151 (2015) 1153–1162.
- [56] S. Yang, Z. Li, M. Wang, F. Sun, L. Jiao, Multitask dictionary learning and sparse representation based single-image super-resolution reconstruction, *Neurocomputing* 74 (17) (2015) 3193–3203.
- [57] M. Elad, M. Aharon, Image denoising via sparse and redundant representations over learned dictionaries, *IEEE Trans. Image Process.* 15 (12) (2006) 3736–3745.
- [58] K. Engan, S.O. Aase, J. Hakon Husoy, Method of optimal directions for frame design, in: *IEEE ICASSP*, 1999, pp. 2443–2446.
- [59] M. Aharon, M. Elad, A. Bruckstein, K-SVD: an algorithm for designing over-complete dictionaries for sparse representation, *IEEE Trans. Signal Process.* 54 (11) (2006) 4311–4322.
- [60] L.N. Smith, M. Elad, Improving dictionary learning: multiple dictionary updates and coefficient reuse, *IEEE Signal Process. Lett.* 20 (1) (2013) 79–82.
- [61] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3D transform-domain collaborative filtering, *IEEE Trans. Image Process.* 16 (8) (2007) 2080–2095.
- [62] K. Zeng, J. Yu, R. Wang, C. Li, D. Tao, Coupled deep autoencoder for single image super-resolution, *IEEE Trans. Cybern.* (2015), <http://dx.doi.org/10.1109/TCYB.2015.2501373>.
- [63] B. Xiao, X. Gao, D. Tao, Y. Yuan, J. Li, Photo-sketch synthesis and recognition based on subspace learning, *Neurocomputing* 73 (2010) 840–852.
- [64] C. Ding, C. Xu, D. Tao, Multi-task pose-invariant face recognition, *IEEE Trans. Image Process.* 24 (3) (2015) 980–993.
- [65] J. Yu, R. Hong, M. Wang, J. You, Image clustering based on sparse patch alignment framework, *Pattern Recognit.* 47 (11) (2014) 3512–3519.
- [66] J. Li, H. Zhang, Y. Huang, L. Zhang, Hyperspectral image classification by nonlocal joint collaborative representation with a locally adaptive dictionary, *IEEE Trans. Geosci. Remote Sens.* 25 (6) (2014) 3707–3719.
- [67] J.A. Tropp, A. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, *IEEE Trans. Inf. Theory* 53 (12) (2007) 4655–4666.
- [68] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, *SIAM J. Sci. Comput.* 20 (1) (1998) 33–61.
- [69] R. Tibshirani, Regression shrinkage and selection via the lasso, *J. R. Stat. Soc. B* 58 (1) (1996) 267–288.
- [70] H. Lee, A. Battle, R. Raina, A.Y. Ng, Efficient sparse coding algorithms, in: *Advances in Neural Information Processing Systems*, 2007, pp. 801–808.
- [71] W. Dong, X. Li, L. Zhang, G. Shi, Sparsity-based image denoising via dictionary learning and structural clustering, in: *IEEE CVPR*, 2011, pp. 457–464.
- [72] Z. Wong, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.* 13 (4) (2004) 600–612.
- [73] C. Li, W. Win, H. Jing, Y. Zhang, An efficient augmented Lagrangian method with applications to total variation minimization, *Comput. Optim. Appl.* 56 (3) (2013) 507–530.
- [74] S. Mun, J.E. Fowler, Block compressed sensing of images using directional transforms, in: *IEEE ICIP*, 2009, pp. 3021–3024.
- [75] J. Zhang, S. Liu, R. Xiong, S. Ma, D. Zhao, Improved total variation based image compressive sensing recovery by nonlocal regularization, in: *IEEE International Symposium on Circuits and Systems*, 2013, pp. 2836–2839.
- [76] N.G. Kingsbury, Complex wavelets for shift invariant analysis and filtering of signals, *J. Appl. Comput. Harmon. Anal.* 10 (3) (2001) 234–253.



Nasser Eslahi received the B.S. and M.S. degrees in electrical engineering from Imam Khomeini International University and Babol University of Technology, Iran, in 2012 and 2015, respectively.

He is currently a research assistant in Machine Vision & Image Processing Laboratory, Babol University of Technology, Babol, Iran. His research interests include image and video processing, statistical signal processing, sparse representation/approximation, compressive sensing, image inverse problems and convex optimization.



Ali Aghagolzadeh received the B.S. degree in electrical and electronic engineering from Tabriz University, Tabriz, Iran, in 1985. He received the M.S. and the Ph.D. degrees in electrical engineering from the Illinois Institute of Technology, Chicago, IL, USA, and Purdue University, West Lafayette, IN, USA, in 1988 and 1991, respectively.

He is currently a Professor with the Department of Electrical and Computer Engineering, Babol University of Technology, Babol, Iran. His research interests include image processing, video coding and compression, information theory, and computer vision.



Seyed Mehdi Hosseini Andargoli received the B.S. degree in electronics engineering from Shahed University, Tehran, Iran, in 2004 and the M.S. and Ph.D. degrees in telecommunication systems engineering from K. N. Toosi University of Technology, Tehran, in 2009 and 2011, respectively.

He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Babol University of Technology, Babol, Iran. His research interests include signal processing, compressive sensing, convex optimization, resource allocation of cellular networks, cognitive radio networks, relay networks, sensor networks, and MIMO-OFDM systems.