Received: 27 September 2024 | Revised: 11 December 2024 | Accepted: 16 January 2025

DOI: 10.1049/ipr2.70004

IET Image Processing

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

# Sparse representation for restoring images by exploiting topological structure of graph of patches

Yaxian Gao[1] | Zhaoyuan Cai[2] | Xianghua Xie[3] | Jingjing Deng[4] | Zengfa Dou[1] | Xiaoke Ma[2]

[1] School of Information Engineering, Shaanxi Xueqian Normal University, Xi'an, Shaanxi, China

[2] School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi, China

[3] Department of Computer Science, Swansea University, Swansea, UK

[4] Department of Computer Science, Durham University, Durham, UK

**Correspondence**
Xiaoke Ma, School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi, China.
Email: xkma@xidian.edu.cn

**Funding information**
Shaanxi Key Research and Development Program, Grant/Award Number: 2021ZDLGY02-02

**Abstract**

Image restoration poses a significant challenge, aiming to accurately recover damaged images by delving into their inherent characteristics. Various models and algorithms have been explored by researchers to address different types of image distortions, including sparse representation, grouped sparse representation, and low-rank self-representation. The grouped sparse representation algorithm leverages the prior knowledge of non-local self-similarity and imposes sparsity constraints to maintain texture information within images. To further exploit the intrinsic properties of images, this study proposes a novel low-rank representation-guided grouped sparse representation image restoration algorithm. This algorithm integrates self-representation models and trace optimization techniques to effectively preserve the original image structure, thereby enhancing image restoration performance while retaining the original texture and structural information. The proposed method was evaluated on image denoising and deblocking tasks across several datasets, demonstrating promising results.

## 1 | INTRODUCTION

Image restoration serves as a fundamental task in image processing, aiming to reconstruct or recover the original image from degraded or corrupted signals [1]. This field has garnered extensive research attention and can generally be formulated as follows:

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{E} \tag{1}$$

Here, **X**, **Y**, and **E** represent the original, degraded, and noise components of the image, respectively, while **H** denotes the degradation matrix. The restoration problem represented by Equation (1) can vary significantly depending on the degradation matrix **H**. For instance, an identity matrix for **H** corresponds to image denoising [2], a diagonal masking corresponds to image inpainting [3], and a blurring operator corresponds to image deblurring [4].

Image priors play a crucial role in image restoration, including total variation (TV) [5–7], sparsity [2, 8], low-rank [9–11], and deep image prior [12–20]. Particularly, sparsity prior is considered remarkable for natural images [2, 8, 21–24]. Current algorithms, based on strategies for manipulating sparsity prior, are roughly divided into two classes: patch-based [2, 25, 26] and group-based approaches [8, 22, 27–29].

Patch-based image restoration has received considerable attention over the past decades [2, 30]. These algorithms aim to identify low-dimensional representations (patch codes) under the assumption that each patch can be modelled with a linear combination of learned basis elements, known as a dictionary [2]. Dictionary strategies typically fall into two categories: analytic and learning-based. Analytic approaches include discrete cosine transform (DCT), wavelet, and curvelet [31]. Compared to traditional analytic methods, dictionaries learned from images are more adaptive and accurate since they comprehensively depict the local structure of images. For instance, the widely-used dictionary learning method K-SVD [30] exhibits strong adaptability and has been successfully applied to tasks like image denoising [2, 30]. Furthermore, by imposing sparse constraints on patch representations, patch-based sparse representation

(PSR) achieves excellent performance for image restoration, where each patch is represented with a linear combination of a few atoms from the learned dictionary.

However, patch-based methods have been criticized for independently learning dictionaries and representations for each patch, leading to two significant limitations. Firstly, these methods are computationally time-consuming, hindering their application to large-scale image datasets. Secondly, they only exploit the intrinsic structure of each patch, disregarding the correlation among various patches, namely non-local self-similarity (NSS). To address these issues, group-based approaches, such as group sparse representation (GSR) [27, 28, 32], learn sparse coding and dictionaries from groups of similar patches, where strong correlations among them can be captured. In recent years, with the continuous development of deep neural networks (DNN), many image restoration methods based on DNN have emerged. Ref.[33] proposes a retractable transformer architecture based on attention mechanisms, which dynamically adjusts attention across different layers to restore images details more precisely. Ref.[34] proposes sparse transformer to solve deraining problems adaptively. The model leverage multi-scale features to improve the efficiency of removing rain streaks. Ref. [35] efficiently captures long-rang dependencies and preserves fine image details, enabling effective image restoration while reducing computational complexity. The image restoration algorithm based on DNN essentially achieves implicit patch similarity computation through the combination and cascading of linear layers (especially convolutional neural networks) and non-linear layers.

Compared to patch-based methods, GSR models [25, 26] demonstrate outstanding performance in image restoration. For example, BM3D [26] performs collaborative filtering on groups of 3D patches. Mairal et al. [32] proposed LSSC, which simultaneously sparse encodes similar patches in a certain transform domain to enforce similar coefficients. Zhang et al. [27] introduced a GSR-based model for image restoration, designing a self-adaptive dictionary for image patch groups and solving sparse coding with $\ell_0$ minimization. Xu et al. [36] learned an NSS prior for patch groups based on external image databases before image denoising, achieving excellent results when the distribution of external patch groups and target image patch groups is similar. To preserve the characteristics of the target image itself, a series of models combining internal and external priors are proposed [37, 38]. To obtain more correct sparsity solutions for image restoration, Wang et al. [29] incorporated non-convex weighted $\ell_p$ minimization into the GSR framework for image denoising. To avoid learning dictionaries from image patches, principal component analysis (PCA) is adopted to construct dictionaries [27, 29]. Recently, Zha et al. [39] proposed the LGSR model, utilizing low-rankness to guide dictionary learning.

However, these group sparse representation models simply group similar image patches without fully exploiting the relationships between these patches and ignoring the specificity among patches within the same group. To address these issues, we propose a graph learning-guided group sparse representation image restoration algorithm. Firstly, this algorithm characterizes the similarity relationships between image patches through graph learning and performs initial reconstruction of the image to enhance the performance of subsequent sparse representation learning. Secondly, low-rank constraints are imposed during graph learning to fully explore the sub-group structure of the same group of image patches. Finally, to ensure that the learned representation satisfies sparsity while preserving the original similarity structure between image patches, the algorithm introduces trace optimization regularization. Extensive experiments are conducted to validate the superiority of the proposed algorithm over some currently popular image restoration algorithms.

The following is a summary of this research's main contributions.

- To enhance the quality of sparse representation learning, this study utilizes a graph learning model to characterize the similarity relationships between image patches and employs this model for the initial reconstruction of the image.
- In order to fully exploit the relationships between image patches while preserving the specificity of each patch, low-rank constraints are imposed during the graph learning process to identify sub-group structures within the same group of image patches.
- To ensure that the learned representation maintains sparsity while preserving the original similarity structure between image patches, this paper introduces a structural preservation regularization term into the model, thereby further improving the interpretability of sparse representation.
- Extensive experiments on two image restoration tasks, namely image denoising and inpainting, are conducted to thoroughly validate the effectiveness and superiority of the proposed algorithm.

The remaining sections of this article are arranged as follows: Section 2 introduces the preliminaries, Section 3 elaborates on the proposed algorithm for image restoration in detail, Section 4 presents the experimental results, and conclusions are drawn in Section 5.

## 2 | PRELIMINARIES

In this section, we will present the notations and preliminaries that are going to be used for the rest of the paper.

### 2.1 | Notations

Let the bold upper, bold lower, and lower-case letters denote matrices, vectors, and scalars, respectively. Let $\mathbf{X} \in R^{n \times m}$ be a $n \times m$ matrix, and $\mathbf{x} \in R^d$ be a vector with $d$ elements, respectively. $\mathbf{X}'$ is the transpose of matrix $\mathbf{X}$.

The Frobenius norm of matrix $\mathbf{X}$ is defined as

$$\|\mathbf{X}\| = \sqrt{tr(\mathbf{X}'\mathbf{X})} = \sqrt{tr(\mathbf{X}\mathbf{X}')}, \tag{2}$$

where $tr(\mathbf{X})$ is the trace of matrix $\mathbf{X}$. $\ell_0$-norm of vector $\mathbf{x}$ is defined as the number of non-zero elements in $\mathbf{x}$, that is,

$$\|\mathbf{x}\|_0 = \sum_i |x_i|^0. \tag{3}$$

$\ell_1$-norm of vector $\mathbf{x}$ is the sum of absolute values of elements in $\mathbf{x}$, that is,

$$\|\mathbf{x}\|_1 = \sum_i |x_i|. \tag{4}$$

$\ell_p$-norm $(0 < p < 1)$ of vector $\mathbf{x}$ is defined as

$$\|\mathbf{x}\|_p = \left(\sum_i |x_i|^p\right)^{1/p}. \tag{5}$$

$\|\mathbf{X}\|_0$, $\|\mathbf{X}\|_1$ and $\|\mathbf{X}\|_p$ denotes imposing $\ell_0$-norm, $\ell_1$-norm, and $\ell_p$-norm on each column of matrix $\mathbf{X}$, respectively. Nuclear norm of matrix $\mathbf{X}$ is defined as

$$\|\mathbf{X}\|_* = \sum_{i=1}^{\min(m,n)} |\lambda_i|, \tag{6}$$

where $\lambda_i$ is the $i$-th singular value of matrix $\mathbf{X}$.

## 2.2 | Image restoration

To simplify the model, we set the degradation matrix $\mathbf{H}$ as the identity matrix. Then, given a degraded image $\mathbf{Y}$, image restoration is formulated as

$$\mathbf{Y} = \mathbf{X} + \mathbf{E}, \tag{7}$$

where $\mathbf{X}$ and $\mathbf{E}$ denote the original image and additive noise, respectively. Without loss of generality, image prior is denoted by $\theta$ and then maximum a posteriori (MAP) framework [8, 27, 40] is employed, that is, a posteriori function of the form $\log p(\mathbf{X}|\mathbf{Y}, \theta)$ is maximized

$$\log p(\mathbf{X}|\mathbf{Y}) = \log p(\mathbf{Y}|\mathbf{X}, \theta) + \log p(\mathbf{X}|\theta). \tag{8}$$

The likelihood term is the Gaussian distribution [8]

$$p(\mathbf{Y}|\mathbf{X}, \theta) = \frac{1}{\sqrt{2\pi}\sigma_E} \exp\left(-\frac{1}{2\sigma_E^2}\|\mathbf{Y} - \mathbf{X}\|^2\right), \tag{9}$$

where $\sigma_E^2$ is the noise variance. And then Equation (8) is equal to

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{Y} - \mathbf{X}\|^2 + \sigma_E^2 \Theta(\mathbf{X}), \tag{10}$$

where $\Theta(\mathbf{X})$ is regularization term derived from prior $\theta$.

## 2.3 | Sparse representation

Given features $\mathbf{d}_1, \dots, \mathbf{d}_n$, representation learning for a vector $\mathbf{x}$ aims to obtain a linear function such that

$$\mathbf{x} \approx a_1\mathbf{d}_1 + \cdots + a_n\mathbf{d}_n, \tag{11}$$

where $a_i$ is the coefficient for feature $\mathbf{x}_i$. Equation (11) is solved by minimizing approximation, that is,

$$\min \frac{1}{2}\|\mathbf{x} - \mathbf{Da}\|^2, \tag{12}$$

where $D = [\mathbf{x}_1, \dots, \mathbf{x}_n]$, and $\mathbf{a} = (a_1, \dots, a_n)'$, respectively. The sparse representation learning expects most of the coefficients are 0, where Equation (11) is formulated as

$$\min \frac{1}{2}\|\mathbf{x} - \mathbf{Da}\|^2 + \alpha\|\mathbf{a}\|_0, \tag{13}$$

where $\alpha$ is a parameter.

Furthermore, an extension for sparse representation learning is needed. When multiple objects involve, that is, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$, GSR simultaneously handles $n$ objects into an objective function, where Equation (13) is re-written as,

$$\min \frac{1}{2}\|\mathbf{X} - \mathbf{DA}\|^2 + \alpha\|\mathbf{A}\|_0, \tag{14}$$

where $\|\mathbf{A}\|_0$ is the regularization item, denotes imposing $\ell_0$-norm on each column of $\mathbf{A}$.

There are various strategies for constructing sparsity, that is, $\ell_1$-norm [41, 42], and $\ell_p$-norm $(0 < p < 1)$, to bridge $\ell_0$ and $\ell_1$ [43, 44]. In summary, sparse representation methods assume that image patches or pixels can be represented by a small number of basis elements (atoms). And graph-based sparse representation methods decompose the image into sparse components while considering the graph structure of the image. Graph-based sparse coding exploits the graph structure to encourage similarity between adjacent patches, allowing for the recovery of missing parts of the image by leveraging the underlying relationships in the graph.

# 3 | PROPOSED METHOD

In this section, we present the proposed method in detail, encompassing the restoration model, optimization, parameter selection, and discussion on its computational complexity.

The overview of the proposed algorithm is illustrated in Figure 1, which comprises four major components: patch grouping, sparse representation learning, low-rank self-representation, and structure preservation. Patch grouping divides sub-blocks of the original images into different classes, where patches within the same groups exhibit high similarity. The low-rank self-representation module conducts self-representation learning through original image blocks and

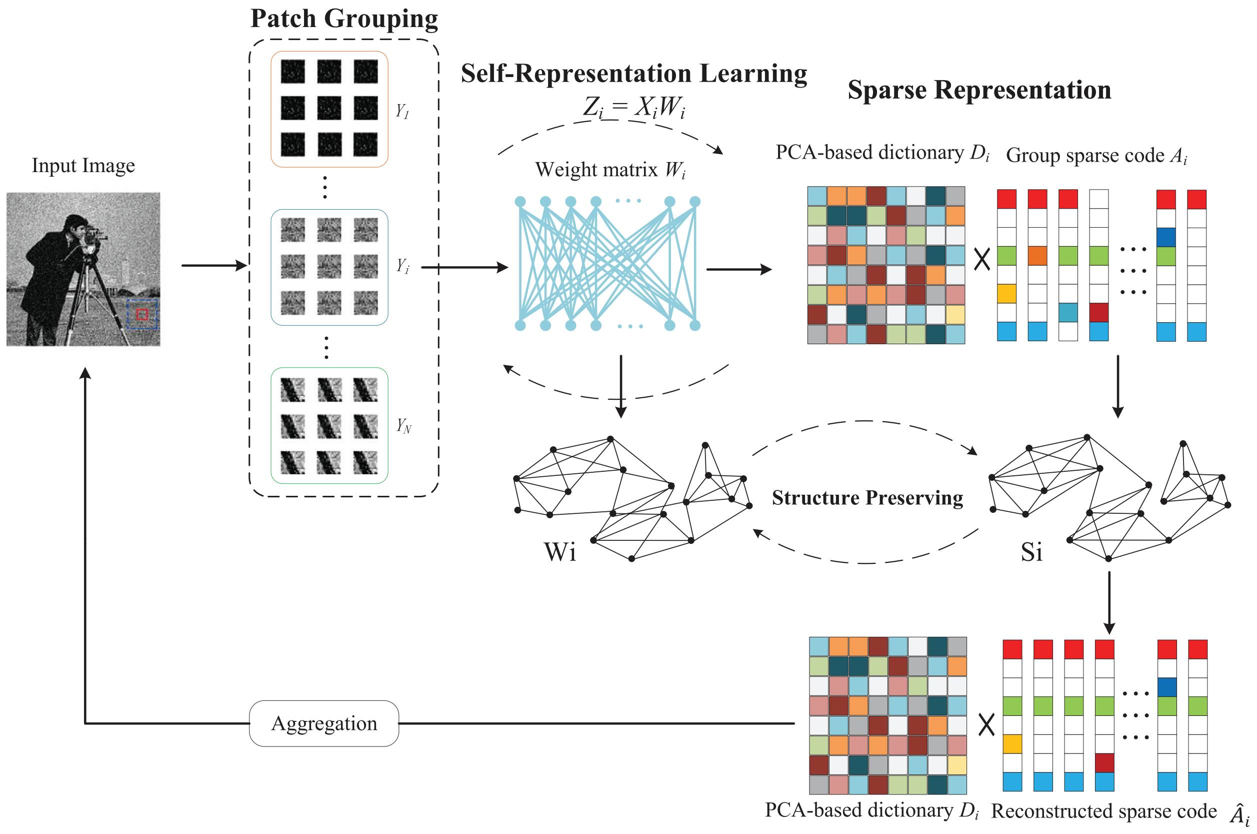

**FIGURE 1** Overview of the proposed image restoration algorithm, which consists of three major parts, including patch grouping, sparse representation learning and group residual learning.

initiates the reconstruction of the image blocks. Group sparse representation learning projects each group of image blocks into a subspace spanned by dictionary matrix columns to obtain the representation of the image block, while the structure preservation module aims to ensure that the learned sparse representation maintains the original similarity structure of the image.

## 3.1 | Restoration model

In the patch grouping block, like other GSR-based restoration models [27, 28, 32], a patch-matching based approach is utilized. Specifically, the degraded image $\mathbf{Y}$ is divided into patches, where the size of patches varies with downstream applications. For each reference patch, the closest $m$ patches within window of $l \times l$ are selected as a group, where patches belonging to multiple groups are allowed. To ensure the quality of groups, the step size of selected reference patches is small, where window size is large. In general, we set step size of selection reference patches as 3 or 4, and that of windows as $25 \times 25$. By stacking pixels each reference patch is denoted as $\overline{\mathbf{y}}_i$, and the corresponding patch group is $\mathbf{Y}_i$, where each column corresponds to a patch within the group.

In the sparse representation learning block, the most intuitive strategy is to project each group of patches into a subspace, where the low-dimensional representation of patches is obtained. Specifically, given patch group $\mathbf{Y}_i$, the low-dimensional representation of patches is learned by minimizing the approximation, which is formulated as:

$$\mathcal{O}(\mathbf{Y}_i) = \frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}_i\mathbf{A}_i\|^2, \tag{15}$$

where $\mathbf{D}_i$ and $\mathbf{A}_i$ denotes the dictionary and coefficient matrix of $\mathbf{Y}_i$, respectively. Sparse representation learning [30] expects the learned $\mathbf{A}_i$ is sparse, that is, the most elements are 0, which improves computational efficiency and interpretability of solutions. By imposing $\ell_1$-norm constraint to coefficient matrix $\mathbf{A}_i$, Equation (15) is reformulated as:

$$\mathcal{O}(\mathbf{Y}_i) = \frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}_i\mathbf{A}_i\|^2 + \beta\|\mathbf{A}_i\|_1, \tag{16}$$

where parameter $\beta$ determines the relative importance of sparsity constraint. Recently, evidence [43, 44] demonstrates that $\ell_p$-norm overcomes limitation of $\ell_1$-norm to fulfil sparsity of representation. Therefore, Equation (16) is re-written as,

$$\mathcal{O}(\mathbf{Y}_i) = \frac{1}{2}\|\mathbf{Y}_i - \mathbf{D}_i\mathbf{A}_i\|^2 + \beta\|\mathbf{A}_i\|_p. \tag{17}$$

In order to achieve better image restoration effects, in addition to utilizing image sparsity, the non-local self-similarity of the image should also be considered. This chapter uses low-rank self-representation to characterize the non-local self-similarity of images, that is, an image block can be represented by a linear combination of similar image blocks, and the coefficient matrix satisfies the block diagonal structure (low rank). Based on the group sparse representation model, this chapter introduces the low-rank self-representation feature, and obtains,

$$\begin{aligned} O(Y_i) = \frac{1}{2}||Y_i - D_iA_i||^2 + \beta||A_i||_p \\ + \frac{\gamma}{2}||D_iA_i - Z_i||^2 + \tau||W_i||_* \\ \text{s.t.} \quad Z_i = Z_iW_i,\ W_i = W_i{}', \end{aligned} \tag{18}$$

where $Z_i$ is an intermediate auxiliary variable, $W_i$ is a self-represented sparse matrix, and $||W_i||_*$ represents the nuclear norm of $W_i$. Different from the method proposed in the previous chapter, low-rank self-representation learning is used here to guide the learning of dictionary matrices and sparse representations at the same time, thereby further improving the quality and interpretability of sparse representation learning.

In order to prevent over-smoothing, this paper hopes that the learned sparse representation satisfies the sparsity constraints while still maintaining the similarity structure between the original image blocks. First, we perform self-representation learning on the sparse representation $A_i$, and the obtained self-representation matrix is as close as possible to the self-representation of the original image block, that is, minimizing

$$O(A_i) = ||A_i - A_iS_i||^2 - Tr(W_i{}'S_i), \tag{19}$$

where $S_i$ is the self-representation matrix of sparse coding $A_i$. $Tr(\cdot)$ represents the trace of the matrix, $Tr(W_i{}'S_i)$ measures the similarity between matrices $W_i$ and $S_i$. The combined expressions (18) and (19) can be obtained

$$\begin{aligned} O(Y_i) = \frac{1}{2}||Y_i - D_iA_i||^2 + \beta||A_i||_p + \lambda(||A_i - A_iS_i||^2 \\ - Tr(W_i{}'S_i)) + \frac{\gamma}{2}||D_iA_i - Z_i||^2 + \tau||W_i||_* \\ \text{s.t.} \quad Z_i = Z_iW_i,\ W_i = W_i{}', \end{aligned} \tag{20}$$

Without loss of generality, the above prior model can be substituted into the general image restoration framework (10) to obtain

$$\begin{aligned} O(Y) = \frac{1}{2}||Y - X|| + \sum_i \Big\{ \frac{\alpha}{2}||Q_iX - D_iA_i||^2 + \beta||A_i||_p \\ + \frac{\gamma}{2}||D_iA_i - Z_i||^2 + \tau||W_i||_* + \lambda(||A_i - A_iS_i||^2 \\ - Tr(W_i{}'S_i)) \Big\} \\ \text{s.t.} \quad Z_i = Z_iW_i,\ W_i = W_i{}',\ \forall i \end{aligned} \tag{21}$$

where $Q_i$ represents the matrix operator for extracting the $i$th group of image blocks in image $X$, that is, $Q_iX = X_i$. Given the degradation image $Y$, the restored image $X$ can be obtained by solving the above equation. The next section will introduce the optimization process of solving this objective function.

## 3.2 | Optimization

Due to the non-convex nature of the nuclear norm and $l_p$ norm, the objective function (21) cannot directly yield an analytical solution. Therefore, this paper employs an alternating iteration strategy for optimization, wherein a single variable is optimized while keeping other variables fixed until convergence or reaching the termination condition.

1) Update $Z_i$ and weight matrix $W_i$: Fixing $X$ and $A_i$, and simultaneously eliminating irrelevant terms, the objective function (21) for $W_i$ can be equivalently expressed as,

$$\frac{\gamma}{2}||D_iA_i - Z_i||^2 + \tau||W_i|| * \quad \text{s.t. } Z_i = Z_iW_i,, W_i = W'i. \tag{22}$$

According to the literature [45], Equation (22) can be efficiently solved by performing singular value decomposition on $D_iA_i$, that is, $D_iA_i = U_i\Sigma_iV'i$, where $\Lambda_i = diag(\lambda_i)$ is a diagonal matrix containing singular values, and $U_i$ and $V_i$ represent the left and right singular matrices, respectively. Then, the optimal solution for $Z_i$ in Equation (22) can be expressed as,

$$\hat{Z}_i = Ui1\Sigma i1V'i1, \tag{23}$$

where $\Sigma_{i1}$ contains singular values greater than $\sqrt{\frac{2\tau}{\gamma}}$, while $U_{i1}$ and $V_{i1}$ contain the corresponding singular vectors. The optimal solution for $W_i$ is

$$\hat{W}_i = V_{i1}V_{i1}'. \tag{24}$$

For the proof of the optimal solution of the above formula, please refer to the literature [45] .

2) Update the weight matrix $S_i$: Eliminate terms irrelevant to $S_i$, and the objective function (21) can be simplified to

$$O(A_i) = ||A_i - A_iS_i||^2 - \text{Tr}(W_i{}'S_i). \tag{25}$$

Taking the partial derivative of $S_i$ yields

$$A_i'A_i + A_i'A_iS_i - W_i = 0. \tag{26}$$

Setting it to 0, we obtain the updated formula for $S_i$ as

$$\hat{S}_i = (A_i'A_i)^{-1}(A_i'A_i + W_i). \tag{27}$$

3) Update group sparse representation $A_i$: Eliminate terms irrelevant to $A_i$, and the objective function (21) can be simplified to

$$\frac{\alpha}{2}||X_i - D_iA_i||^2 + \frac{\gamma}{2}||D_iA_i - Z_i||^2 + \frac{\eta}{2}||A_i - A_iS_i||^2 + \beta||A_i||_p. \tag{28}$$

By merging the first two terms, we obtain

$$\frac{1}{2}||G_i - D_iA_i||^2 + \frac{\eta}{2}||A_i - A_iS_i||^2 + \beta||A_i||_p, \tag{29}$$

where $G_i = (\alpha X_i + \gamma Z_i)/(\alpha + \gamma)$. To better adapt to the local structure of the image, this section adopts the PCA sub-dictionary strategy, that is, learning an orthogonal dictionary through each group $G_i$.

After obtaining the dictionary $D_i$, the $l_p$ norm makes the solution of Equation (29) non-convex. Therefore, this paper utilizes the generalized soft threshold (GST) algorithm [46], an efficient iterative strategy to obtain approximate solutions. Specifically, the update rule for $A_i$ can be expressed as

$$\hat{A}_i = \mathrm{GST}(P_i, \mu, p, t), \tag{30}$$

where $t$ represents the number of iterations of the GST algorithm, and the specific definitions of $P_i$ and $\mu$ are:

$$\begin{cases} P_i = \alpha(A_i - D'_{i}X_i) + \gamma A_i(I - S_i - S'_{i} + S_iS'_{i}), \\ mu = \dfrac{\beta}{\alpha + \gamma||I - S_i||^2}. \end{cases} \tag{31}$$

4) Update the restored image $X$: By fixing the matrices $A_i$ and $W_i$, the objective function (21) can be simplified to

$$\frac{1}{2}||Y - X||^2 + \frac{\alpha}{2}\sum_i ||Q_iX - D_iA_i||^2. \tag{32}$$

Since Equation ((32)) is convex with respect to $X$, setting the partial derivative of (32) with respect to $X$ to 0 yields the exact solution as

$$\hat{X} = \left(I + \alpha\sum_i Q_i'Q_i\right)^{-1}\left(Y + \alpha\sum_i Q_i'D_iA_i\right), \tag{33}$$

where $D_iA_i$ represents the reconstruction of the image patch group $X_i$, and $Q_i'$ can be regarded as a matrix operator that puts the reconstructed image patch group back into the original image. In fact, $(I + \alpha\sum_i Q_i'Q_i)$ is a diagonal matrix, and its inverse can be obtained by element-wise division. Therefore, Equation (33) can be regarded as superimposing the reconstructed image patches and performing a weighted average with the degraded image to obtain the restored image.

## 3.3 | Parameter selection

In order to obtain the best performance results, this algorithm adopts an adaptive parameter adjustment strategy to enable the proposed algorithm to adapt to various image structures. First, we update the noise variance $\sigma_E^2$ [6] using an iterative regularization strategy,

$$\sigma_E^{(k)} = c0\sqrt{\left(\sigma_E^2 - ||Y - \hat{X}^{(k)}||^2\right)}, \tag{34}$$

Among them, $k$ represents the current number of iterations, and $c_0$ is a positive constant.

Inspired by the maximum posterior probability framework [8], it is assumed here that the sparse encoding $A_i$ obeys the Laplace distribution [29, 47], and the sparse residual $R_i$ obeys the Gaussian distribution, and then the parameters can be obtained. The update strategy of $\beta$ and $\eta$ is

$$\beta = \frac{\sigma_E^2}{\delta_i + \epsilon}, \tag{35}$$

$$\eta = \frac{\sigma_E^2}{\delta_i^2 + \epsilon}, \tag{36}$$

Among them, $\delta_i$ represents the standard deviation of $A_i$, and its estimation method can be found in the document [10]. At the same time, $\epsilon$ represents a very small constant to prevent the denominator from being zero.

In addition, the parameters $\alpha$ and $\gamma$ adopt the following update strategy:

$$\alpha = c_1\sigma_E^2, \tag{37}$$

$$\gamma = c_2\sigma_E^2. \tag{38}$$

where $c_1$ and $c_2$ both represent positive constants. Formulas (37) and (38) mean that the parameters $\alpha$ and $\gamma$ are respectively proportional to the standard deviation $\delta_i$ of the noise variance $\sigma_E^2$.

## 3.4 | Method overview

To sum up, the image restoration algorithm proposed in this chapter can be realized through the above-mentioned alternating update steps and parameter adaptive adjustment mechanism. The pseudocode of the proposed algorithm is shown in Algorithm 1.

## 3.5 | Computational complexity analysis

In this section, we analyse the computational complexity of the proposed method theoretically. Concerning the spatial complexity, the proposed algorithm requires space complexity of $O(m^2n)$. The matrices $W_i$ and $S_i$ for each image block group

**ALGORITHM 1** The proposed algorithm for image restoration.

```
Input: The degraded image $\mathbf{Y}$.
Initialize $\hat{\mathbf{X}}^{(0)} = \mathbf{Y}$, $k = 0$, $\sigma_E^{(0)}$.
Set the parameters $c_0$, $c_1$, $c_2$ and $p$.
while $k <=$ Max-Iter do
    for each reference patch $\mathbf{x}_i$ do in $\hat{\mathbf{X}}^{(k)}$
        Search similar patches to construct patch group $\mathbf{X}_i$.
        Update $Z_i$ and $W_i$ by (23) and (24).
        Update $\alpha$ and $\gamma$ by (37) and (38).
        Construct the dictionary $\mathbf{D}_i$ through PCA on $\mathbf{G}_i$ by (29).
        Update $\mathbf{A}_i$ by $\mathbf{A}_i = \mathbf{D}_i{}' \mathbf{X}_i$.
        Update $S_i$ by (27).
        Update $\beta$ and $\eta$ by (35) and (36).
        Update $\mathbf{A}_i$ by (30).
    end for
    Update $\hat{\mathbf{X}}^{(k)}$ by (33).
    Update $\sigma_E^{(k)}$ by (34).
    Until convergence conditions are met.
end while
Output: The restored image $\hat{\mathbf{X}}$.
```

require space of $O(m^2)$. The space complexity for $A_i$, $X_i$, and $Z_i$ for each image block group is $O(dm)$, where $d$ represents the number of rows in $A_i$, $X_i$, and $Z_i$. Therefore, the total spatial complexity of the proposed algorithm is $O(m^2 n)$, where $n$ is the number of image block groups.

Regarding the time complexity, it comprises four main components: (1) Low-rank self-representation learning; (2) structure preservation; (3) sparse representation learning; and (4) image reconstruction. The time complexity for updating $W_i$ is $O(tnm^3)$, where $t$ is the number of iterations. The time complexity for updating $S_i$ is $O(tnm^3)$. The time complexity for updating $A_i$ is $O(tndm)$. The time complexity for image group reconstruction is $O(tnb^2m)$. Hence, the total time complexity of the algorithm is $O(tnm^3)$.

## 4 | EXPERIMENTAL RESULTS

To fully validate the performance of the proposed algorithms, extensive experiments are conducted on two typical image restoration tasks: denoising and deblocking.

### 4.1 | Experimental setting

1) Benchmark: As the proposed algorithm follows a self-supervised learning approach, only test datasets are required to validate the performance of the proposed algorithm.

   For the image denoising task, experiments are conducted on three datasets: commonly used test images, the Set12 benchmark dataset [48] (consisting of 12 greyscale images), and the DND dataset [49]. DND consists of 50 high-resolution images with realistic image noise, and the DND images have been resized to 256 × 256. For the scene with man-made noise (the first two datasets), Gaussian noise at various levels is added to the original images to generate noisy images, which are then used for testing. For DND, there is no need to generate noisy images via manually adding Gaussian noise. Additionally, several real noisy images are also selected for experiments to thoroughly validate the algorithm's effectiveness.

   For the image deblocking task, two widely used datasets are employed: the LIVE1 dataset [50] and the Classic5 dataset [51], comprising 29 and five natural images, respectively. Each test image is first encoded using the MATLAB JPEG encoder at different compression quality levels $Q$. Subsequently, the compressed images are decoded using a standard JPEG decoder to obtain the input images for experimentation. Besides the LIVE1 and Classic5 datasets, eight fingerprint images are also used to further validate the superiority of the proposed algorithm.

   This experimental setup ensures comprehensive evaluation of the proposed algorithm's performance across various image restoration tasks.

2) Parameter setting: For image denoising, the parameter settings of the proposed algorithm are as follows: when the noise level $\sigma_E$ is $\leq 30$, $30 < \sigma_E \leq 50$, and $50 < \sigma_E \leq 100$, the patch sizes are set to 7 × 7, 8 × 8, and 9 × 9, respectively. When $\sigma_E \leq 30$, $30 < \sigma_E \leq 40$, $40 < \sigma_E \leq 50$, $50 < \sigma_E \leq 75$, and $75 < \sigma_E \leq 100$, the number of patches per group is set to 60, 70, 80, 90, and 100, respectively. When $\sigma_E \leq 30$, $30 < \sigma_E \leq 40$, and $40 < \sigma_E \leq 100$, the parameter $p$ is set to 0.8, 0.85, and 0.9, respectively.

   For image deblocking, the patch size is set to 7 × 7. The number of similar patches per group is set to 60. When the compression quality $Q$ is $\leq 10$, $10 < Q \leq 20$, and $20 < Q \leq 40$, the parameter $p$ is set to 0.9, 0.8, and 0.2, respectively.

### 4.2 | Compared methods

In the image denoising task, the proposed algorithm is compared with several state-of-the-art denoising methods, including BM3D [26], LSSC [32], EPLL [52], LPCA [53], NCSR [8], aGMM [54], NLN-CDR [55], SNSS [56], and LGSR [39]. Among them, algorithms such as BM3D [26], LSSC [32], EPLL [52], LPCA [53], aGMM [54], and NLN-CDR [55] utilize the prior of non-local self-similarity in images. The SNSS [57] algorithm further incorporates non-local self-similarity prior knowledge obtained through an external image database. Particularly, the NCSR [8] algorithm and the proposed algorithm in this paper are both based on sparse representation models and utilize non-local self-similarity in images for algorithmic improvement. Additionally, the proposed algorithm is compared with several deep learning-based denoising models, including TRND [58], DnCNN [48], and S2S [59]. Among these models, TRND and DnCNN are supervised learning models,

**TABLE 1** Average PSNR (dB) results of image denoising compared with classical methods on the test image dataset.

| $\sigma_E$ | BM3D | LSSC | EPLL | LPCA | NCSR | aGMM | NLN-CDR | SNSS | LGSR | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| 20 | 31.87 | 31.98 | 31.44 | 31.31 | 31.85 | 31.78 | 31.21 | 32.09 | 32.15 | **32.20** |
| 30 | 29.86 | 29.88 | 29.88 | 29.40 | 29.72 | 29.69 | 29.02 | 30.11 | 30.21 | **30.25** |
| 40 | 28.25 | 28.41 | 27.95 | 27.48 | 28.29 | 28.22 | 27.84 | 28.68 | 28.65 | **28.77** |
| 50 | 27.26 | 27.26 | 26.82 | 26.25 | 27.16 | 27.10 | 26.73 | 27.62 | 27.62 | **27.68** |
| 75 | 25.31 | 25.16 | 24.82 | 24.09 | 25.08 | 25.02 | 24.82 | 25.65 | 25.67 | **25.69** |
| 100 | 23.92 | 23.69 | 23.46 | 22.61 | 23.60 | 23.63 | 23.58 | 24.33 | **24.35** | 24.30 |
| Average | 27.75 | 27.73 | 27.40 | 26.86 | 27.62 | 27.57 | 27.20 | 28.08 | 28.11 | **28.15** |

while S2S is a self-supervised learning algorithm. These deep learning models serve as widely adopted baseline models.

For the image deblocking task, the proposed method is compared with various classical image deblocking methods, including BM3D [26], SA-DCT [60], PC-LRM [61], WNNM [62], ANCE [63], SSR-QC [64], COGL [65], JPG-SR [57], NSSRC [22], as well as with deep learning-based deblocking models such as AR-CNN [66], TRND [58], DnCNN [48], DCSC [67], and MDDU [68]. Among these comparison methods, AR-CNN is a commonly used deep learning baseline model for compression artefact removal, while TRND and DnCNN are general-purpose image restoration models. Lastly, DCSC and MDDU are the latest and most advanced image deblocking models.

It is worth noting that the experiments in this section were conducted with the default parameters set by the original authors for the compared methods. For deep learning models, the experiments were conducted using the pre-trained models provided by the official sources.

## 4.3 | Image denoising

Image denoising is the most common and fundamental task in image restoration. To validate the effectiveness of the proposed image restoration algorithm in this chapter, experiments were conducted using MATLAB's random number generator to synthesize Gaussian white noise (GWN) images for testing. Additionally, several real images were selected for denoising testing.

1) Comparison with classical image denoising methods: The proposed method and other classical denoising methods were evaluated at noise levels $\sigma_E$ of 20, 30, 40, 50, 75, and 100, respectively. To quantify the effectiveness of the algorithms, two evaluation metrics were used to assess the quality of the restored images: Peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [69]. Table 1 (PSNR results) and Table 2 (SSIM results) show the denoising average results of all compared methods on 13 commonly used test images, with the best results highlighted in bold. It is evident that the proposed algorithm outperforms all other compared methods overall in both PSNR and SSIM metrics. Particularly in terms of the SSIM metric, the proposed algorithm significantly outperforms other methods. The SSIM metric primarily focuses on the structural information of images, simulating human perception of image structure, and providing a more accurate assessment of image quality.

Experiments setup on real-word dataset DND follows the main approach outlined in [49]. The algorithms are applied to the space of linear raw intensity(RAW data) and RAW data with a variance stabilizing transformation (VST). After denoising, the results are compared with RAW and sRGB for evaluation, respectively. Therefore, there are four separate scenarios in the experiment results, as reported in Table 3 for average PSNR and Table 4 for average SSIM, respectively, where the best performance is highlighted in bold. It is obvious that performance is better in scenarios where algorithms are evaluated on the RAW space, regardless of which space the algorithm is applied to. Our proposed method outperforms most other baseline methods in nearly all scenarios, except for the third scenario, where both PSNR and SSIM are slightly weaker than BM3D.

Human visual perception is the most intuitive judgment of image quality, which is crucial for evaluating image denoising algorithms. Figures 2 and 3, respectively illustrate the denoising visualization results of the proposed algorithm and other classical algorithms on images *Airplane* and *Miss* at noise level $\sigma_E = 75$. Among them, BM3D [26] adopts collaborative filtering for denoising, and it can be observed from the images that its result suffers from oversmoothing, leading to the blurring of the original texture structure. The EPLL algorithm [52] denoises based on image distribution, but the denoising result is not ideal, as there are still many artefacts remaining. The NCSR algorithm [8], like the algorithm proposed in this chapter, is based on group sparse representation model. However, it can be seen from the images that although NCSR can effectively remove noise, the image does not retain the original structure clearly, whereas the proposed algorithm in this chapter has addressed this issue as much as possible. Overall, both the SSIM results and the visualization results demonstrate the superiority of the proposed algorithm. This is attributed to the adoption of low-rank self-representation for graph learning and guiding the learning of sparse

**TABLE 2** Average SSIM results comparing image denoising with classical methods on test image dataset.

| $\sigma_E$ | BM3D | LSSC | EPLL | LPCA | NCSR | aGMM | NLN-CDR | SNSS | LGSR | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| 20 | 0.9014 | 0.9013 | 0.8950 | 0.8935 | 0.9004 | 0.8994 | 0.8905 | 0.9029 | 0.9056 | **0.9062** |
| 30 | 0.8659 | 0.8648 | 0.8549 | 0.8526 | 0.8645 | 0.8607 | 0.8357 | 0.8712 | 0.8760 | **0.8764** |
| 40 | 0.8303 | 0.8330 | 0.8177 | 0.8145 | 0.8346 | 0.8256 | 0.8183 | 0.8439 | 0.8456 | **0.8477** |
| 50 | 0.8058 | 0.8047 | 0.7836 | 0.7789 | 0.8079 | 0.7939 | 0.7879 | 0.8195 | 0.8202 | **0.8231** |
| 75 | 0.7440 | 0.7398 | 0.7096 | 0.6988 | 0.7518 | 0.7197 | 0.7320 | 0.7660 | 0.7658 | **0.7687** |
| 100 | 0.6925 | 0.6907 | 0.6477 | 0.6277 | 0.7042 | 0.6566 | 0.6964 | 0.7230 | 0.7260 | **0.7262** |
| Average | 0.8067 | 0.8057 | 0.7848 | 0.7777 | 0.8106 | 0.7927 | 0.7935 | 0.8211 | 0.8232 | **0.8247** |

**TABLE 3** Average PSNR (dB) results of image denoising compared with classical methods on DND dataset.

| Applied | Evaluated | BM3D | LSSC | EPLL | LPCA | NCSR | aGMM | NLN-CDR | SNSS | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| RAW | RAW | 46.52 | 45.04 | 46.32 | 46.22 | 42.04 | 45.58 | 42.47 | 43.58 | **46.63** |
| RAW | sRGB | 37.91 | 37.11 | 37.37 | 36.55 | 36.28 | 36.02 | 36.23 | 35.99 | **37.99** |
| RAW+VST | RAW | **47.05** | 46.98 | 46.85 | 46.72 | 45.58 | 45.09 | 45.39 | 44.98 | 47.01 |
| RAW+VST | sRGB | 36.78 | 36.85 | 35.89 | 36.56 | 36.12 | 36.84 | 36.28 | 36.44 | **36.91** |

**TABLE 4** Average SSIM results of image denoising compared with classical methods on DND dataset.

| Applied | Evaluated | BM3D | LSSC | EPLL | LPCA | NCSR | aGMM | NLN-CDR | SNSS | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| RAW | RAW | 0.9701 | 0.9655 | 0.9583 | 0.958 | 0.9537 | 0.962 | 0.9674 | 0.9532 | **0.9724** |
| RAW | sRGB | 0.9218 | 0.919 | 0.9012 | 0.9242 | 0.9273 | 0.9242 | 0.9101 | 0.9154 | **0.9313** |
| RAW+VST | RAW | **0.9542** | 0.9172 | 0.9143 | 0.9111 | 0.9204 | 0.9044 | 0.9174 | 0.9077 | 0.9502 |
| RAW+VST | sRGB | 0.9135 | 0.8995 | 0.9045 | 0.9235 | 0.9005 | 0.9141 | 0.9006 | 0.9133 | **0.9258** |

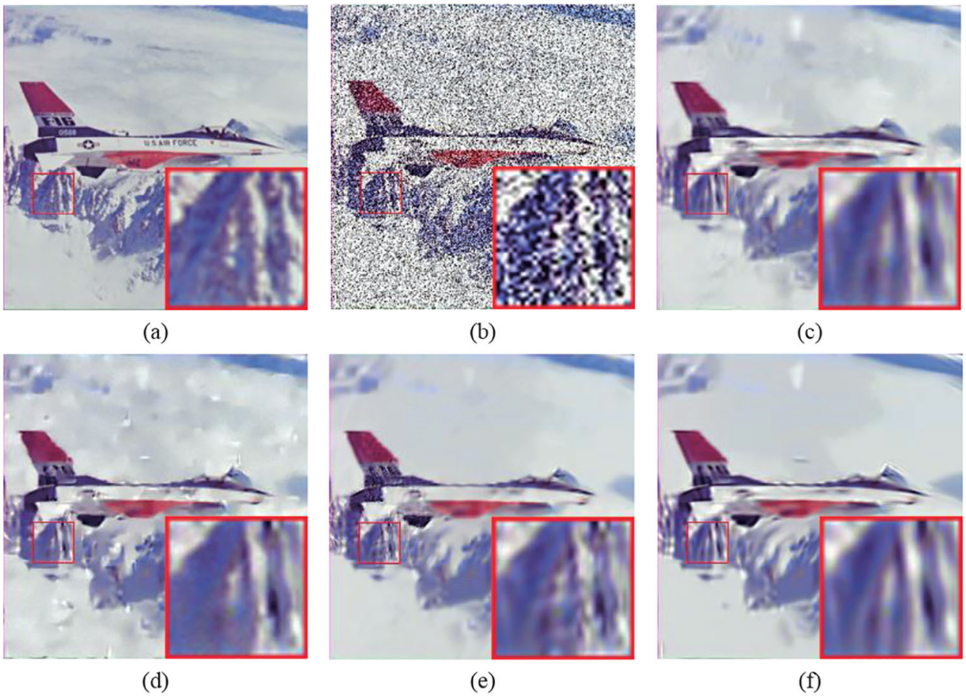

**FIGURE 2** Visualization of algorithms for denoising image *airplane* under $\sigma_E = 75$ noise: (a) Original image, (b) noise image, (c) BM3D (PSNR = 23.99 dB, SSIM = 0.7488), (d) EPLL (PSNR = 23.94 dB, SSIM = 0.7168), (e) NCSR (PSNR = 23.77 dB, SSIM = 0.7551), and (f) the proposed algorithm (PSNR = 24.25 dB, SSIM = 0.7690).

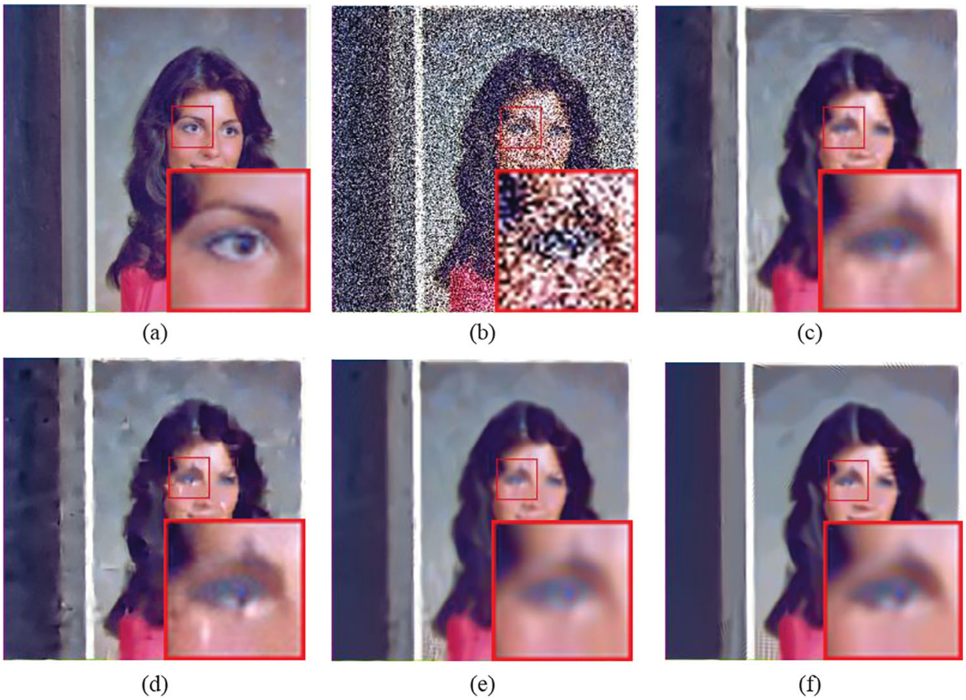

**FIGURE 3** Visualization of algorithms for denoising image*Miss* under $\sigma_E = 75$ noise: (a) Original image, (b) noisy image, (c) BM3D (PSNR = 27.34 dB, SSIM = 0.7722), (d) EPLL (PSNR = 26.69 dB, SSIM = 0.7422), (e) NCSR (PSNR = 27.01 dB, SSIM = 0.7927), and (f) the proposed algorithm (PSNR = 27.55 dB, SSIM = 0.7956).

**TABLE 5** Average PSNR (dB)/SSIM results comparing image denoising with DNN-based methods on the Set12 dataset.

| Methods | $\sigma_E = 15$ | $\sigma_E = 25$ | $\sigma_E = 50$ | Average |
|---|---|---|---|---|
| TRND | 32.51 | 30.04 | 26.78 | 29.78 |
| | 0.8970 | 0.8523 | 0.7672 | 0.8388 |
| DnCNN | 32.50 | 30.17 | 26.98 | 29.88 |
| | **0.8966** | **0.8549** | 0.7700 | 0.8405 |
| S2S | 32.07 | 29.94 | 26.12 | 29.38 |
| | 0.8891 | 0.8475 | 0.7382 | 0.8249 |
| **Ours** | **32.51** | **30.17** | **27.01** | **29.90** |
| | 0.8941 | 0.8538 | **0.7794** | **0.8424** |

representation in this algorithm, which allows the learned sparse representation to maintain the original graph structure, thus avoiding oversmoothing to a certain extent.

2) Comparison with DNN-based image denoising models: Deep neural networks (DNNs) have achieved significant success in both high-level understanding and basic processing tasks of images. Therefore, this section compares the proposed algorithm with several mainstream DNN-based image denoising models, including TRND [58], DnCNN [48], and S2S [59]. The average results (PSNR and SSIM) on the Set12 dataset are shown in Table 5.

The results indicate that the proposed method outperforms some popular deep image denoising models. For better visualization, this section selects some denoising results at $\sigma_E = 50$ for visual display, as shown in Figures 4 and 5. The denoising results of TRND, DnCNN, S2S, and the proposed method are displayed in the figures. It can be observed that deep learning-based methods tend to produce artefacts or oversmoothing during denoising, while the proposed method can avoid such issues and more clearly restore the details of the image. The results indicate that although supervised deep models can be trained on large-scale datasets to fit the distribution of images as much as possible, the generalization ability of this distribution fitting is usually insufficient, resulting in unsatisfactory performance on images dissimilar to the training dataset distribution. These supervised deep models overlook the inherent structural priors of images, such as sparsity and NSS, while the proposed algorithm can effectively utilize these priors to achieve desirable results on various images. Although the S2S model and the proposed method are both self-supervised models, the deep network parameters of the S2S model lead to longer learning times compared to the proposed method.

To thoroughly validate the effectiveness of the proposed algorithm, experiments were conducted using real noisy images. As the model proposed in this chapter requires the noise variance of the images as a prior parameter, a fast noise estimation method [70] was employed to obtain the noise variance of the real images in advance. The denoising results of real noisy images are shown in Figure 6. The proposed method is compared with the S2S [59] model, which is also a self-supervised model based on deep learning. It can be clearly observed that

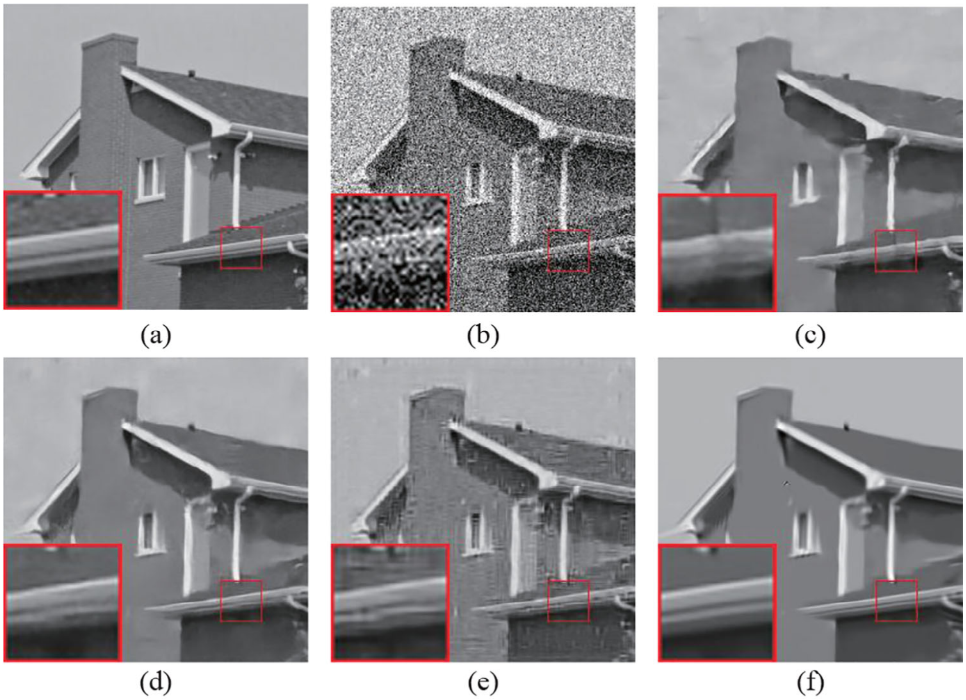

**FIGURE 4** Visualization of denoising results of algorithms for image *House* in the Set12 dataset under $\sigma_E = 50$ noise: (a) Original image, (b) noisy image, (c) TRND (PSNR = 29.40 dB, SSIM = 0.8058), (d) DnCNN (PSNR = 29.74 dB, SSIM = 0.8059), (e) S2S (PSNR = 27.47 dB, SSIM = 0.7032), and (f) the proposed algorithm (PSNR = 30.40 dB, SSIM = 0.8293).

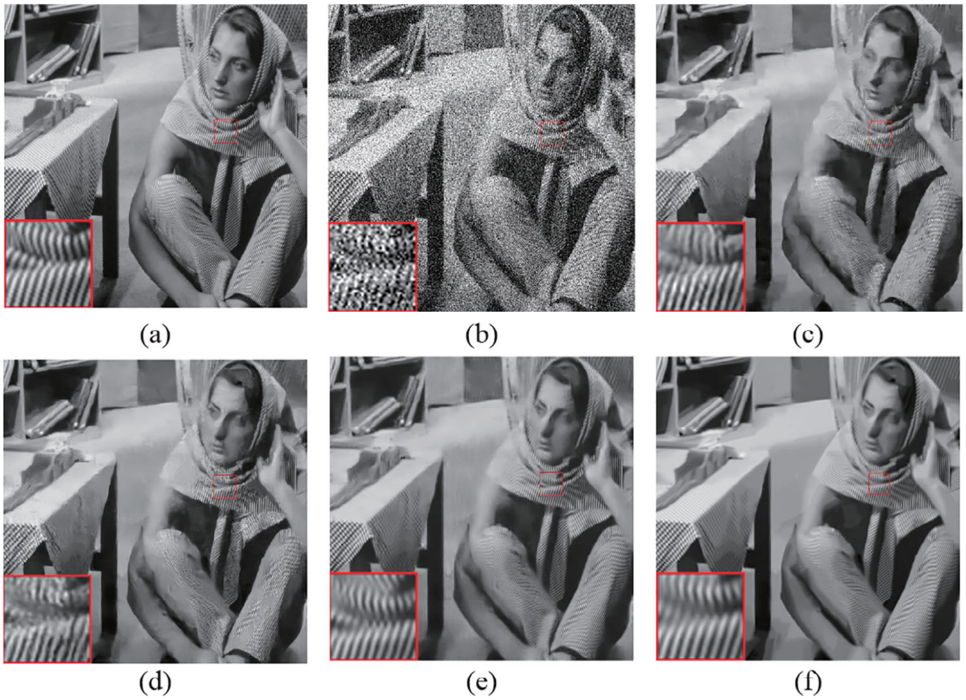

**FIGURE 5** Visualization of denoised image *Barbara* in the Set12 dataset under $\sigma_E = 50$ noise: (a) Original image, (b) noisy image, (c) TRND (PSNR = 25.78 dB, SSIM = 0.7450), (d) DnCNN (PSNR = 25.53 dB, SSIM = 0.7361), (e) S2S (PSNR = 26.82 dB, SSIM = 0.7840), and (f) the proposed algorithm (PSNR = 27.88 dB, SSIM = 0.8243).

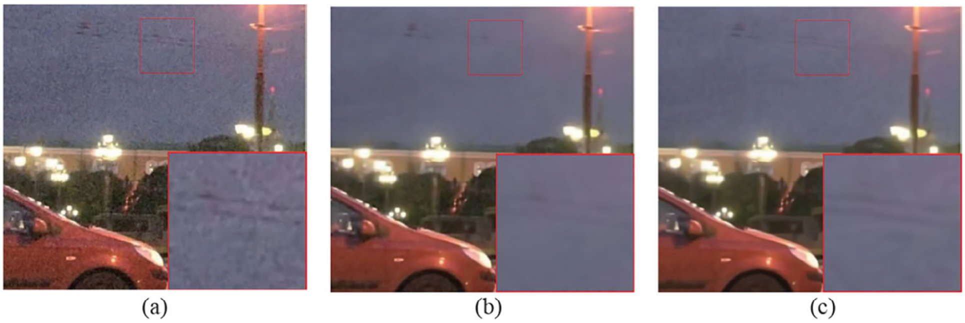

**FIGURE 6** Visualization of denoising for real images by various algorithms: (a) Real image, (b) S2S, and (c) the proposed algorithm.

**TABLE 6** Average PSNR (dB) results of image deblocking compared with the classic method on the datasets LIVE1 and Classic5 (image size: 256 × 256).

**LIVE1 dataset (Image size: 256 × 256)**

| Q | JPEG | BM3D | SA-DCT | PC-LRM | ANCE | WNNM | SSR-QC | COGL | JPG-SR | NSSRC | Ours |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 26.37 | 27.16 | 27.23 | 27.24 | 27.24 | 27.25 | 27.26 | 27.38 | 27.29 | 27.43 | **27.45** |
| 20 | 28.55 | 29.21 | 29.24 | 29.28 | 29.29 | 29.29 | 29.33 | 29.46 | 29.37 | 29.53 | **29.55** |
| 30 | 29.86 | 30.45 | 30.48 | 30.54 | 30.57 | 30.55 | 30.60 | 30.74 | 30.75 | 30.85 | **30.86** |
| 40 | 30.80 | 31.35 | 31.37 | 31.45 | 31.51 | 31.46 | 31.57 | 31.66 | 31.71 | **31.82** | 31.81 |
| Average | 28.90 | 29.54 | 29.58 | 29.63 | 29.65 | 29.64 | 29.69 | 29.81 | 29.78 | 29.91 | **29.92** |

**Classic5 dataset (Image size: 256 × 256)**

| Q | JPEG | BM3D | SA-DCT | PC-LRM | ANCE | WNNM | SSR-QC | COGL | JPG-SR | NSSRC | Ours |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 27.57 | 28.69 | 28.72 | 28.79 | 28.77 | 28.78 | 28.83 | 28.93 | 28.78 | 28.97 | **29.03** |
| 20 | 29.90 | 30.87 | 30.89 | 30.98 | 30.96 | 30.98 | 31.07 | 31.13 | 31.12 | 31.23 | **31.26** |
| 30 | 31.21 | 32.07 | 32.09 | 32.21 | 32.22 | 32.21 | 32.34 | 32.39 | 32.50 | **32.55** | 32.54 |
| 40 | 32.14 | 32.94 | 32.96 | 33.09 | 33.16 | 33.10 | 33.30 | 33.29 | 33.46 | **33.54** | 33.51 |
| Average | 30.21 | 31.14 | 31.17 | 31.27 | 31.28 | 31.27 | 31.39 | 31.43 | 31.47 | 31.57 | **31.59** |

the restoration results of S2S exhibit oversmoothing, while the proposed method preserves more image details.

## 4.4 | Image deblocking

To further comprehensively validate the effectiveness of the proposed algorithm, experiments were conducted on the JPEG compression artifact removal problem [60, 64, 66], which involves removing blocky artefacts from JPEG compressed images. Unlike image denoising tasks, in image deblocking, the additive noise $E$ is quantization noise. Therefore, classic Gaussian models [60] were employed to estimate the noise standard deviation $\sigma_E$, characterizing the noise quantization process.

1) Comparison with classical image deblocking methods: To evaluate the performance of all classical deblocking methods involved in the comparison, experiments were conducted on two commonly used benchmark datasets: the LIVE1 dataset [50] and the Classic5 dataset [51]. Similar to image denoising, experiments utilized two evaluation metrics, PSNR and SSIM. The results are shown in Tables 6 and 7. It is evident that the proposed method outperforms other classical methods on the Classic5 dataset at a compression quality of $Q = 40$. Particularly, the proposed method significantly outperforms other comparison methods on low compression quality images ($Q = 10, 20, 30$) and approaches or even surpasses current state-of-the-art methods on high compression quality images ($Q = 40$).

To provide a more intuitive demonstration of the superiority of the proposed algorithm, Figures 7 and 8, respectively illustrate the deblocking results of the images *buildings* and *sailing3* from the LIVE1 dataset at a compression quality of $Q = 10$. A visual comparison is made between the proposed algorithm and the popular SA-DCT image compression algorithm. From the images, it can be observed that the SA-DCT algorithm fails to fully restore the edge information of the images during the deblocking process. Portions of the edges still exhibit blocky artefacts, as highlighted by the red boxes in the figures. In contrast, the proposed algorithm is able to effectively remove the blocky artefacts while preserving the edge details of the images.

**TABLE 7** Average SSIM results of image deblocking compared with the classic method on the datasets LIVE1 and Classic5 (image size: 256 × 256).

**LIVE1 dataset (Image size: 256 × 256)**

| Q | JPEG | BM3D | SA-DCT | PC-LRM | ANCE | WNNM | SSR-QC | COGL | JPG-SR | NSSRC | Ours |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 0.7611 | 0.7877 | 0.7869 | 0.7835 | 0.7879 | 0.7824 | 0.7859 | 0.7957 | 0.7931 | 0.7956 | **0.7971** |
| 20 | 0.8423 | 0.8591 | 0.8571 | 0.8550 | 0.8585 | 0.8542 | 0.8576 | 0.8642 | 0.8630 | 0.8645 | **0.8651** |
| 30 | 0.8791 | 0.8917 | 0.8903 | 0.8892 | 0.8913 | 0.8888 | 0.8913 | 0.8952 | 0.8967 | 0.8963 | **0.8970** |
| 40 | 0.8998 | 0.9103 | 0.9093 | 0.9089 | 0.9102 | 0.9087 | 0.9099 | 0.9129 | 0.9145 | 0.9148 | **0.9144** |
| Average | 0.8456 | 0.8622 | 0.8609 | 0.8592 | 0.8620 | 0.8585 | 0.8612 | 0.8670 | 0.8668 | 0.8678 | **0.8684** |

**Classic5 dataset (Image size: 256 × 256)**

| Q | JPEG | BM3D | SA-DCT | PC-LRM | ANCE | WNNM | SSR-QC | COGL | JPG-SR | NSSRC | Ours |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 0.7715 | 0.8087 | 0.8060 | 0.8043 | 0.8081 | 0.8033 | 0.8094 | 0.8134 | 0.8134 | 0.8168 | **0.8195** |
| 20 | 0.8519 | 0.8753 | 0.8728 | 0.8723 | 0.8730 | 0.8714 | 0.8740 | 0.8751 | 0.8796 | 0.8802 | **0.8807** |
| 30 | 0.8844 | 0.9018 | 0.9002 | 0.9003 | 0.9002 | 0.8998 | 0.9017 | 0.9012 | 0.9063 | 0.9060 | **0.9061** |
| 40 | 0.9036 | 0.9178 | 0.9168 | 0.9170 | 0.9172 | 0.9167 | 0.9180 | 0.9175 | 0.9225 | **0.9226** | 0.9217 |
| Average | 0.8529 | 0.8759 | 0.8740 | 0.8735 | 0.8746 | 0.8728 | 0.8758 | 0.8768 | 0.8805 | 0.8814 | **0.8820** |

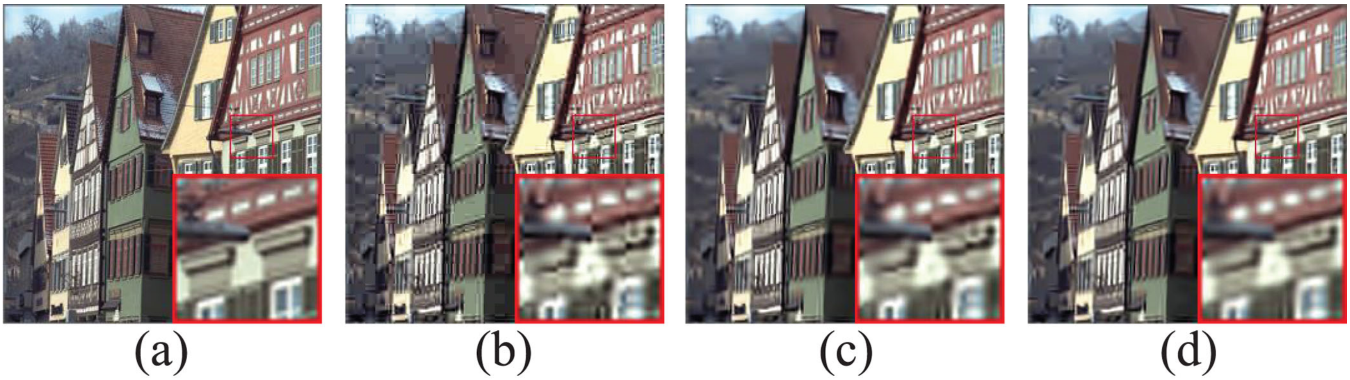

**FIGURE 7** Visualization of image *buildings* in the LIVE1 dataset (image size: 256 × 256) under compression quality $Q = 10$ deblocked by various algorithms: (a) Original image, (b) JPEG compressed image (PSNR = 23.83 dB, SSIM = 0.8232), (c) SA-DCT (PSNR = 24.66 dB, SSIM = 0.8177), and (d) the proposed algorithm (PSNR = 25.11 dB, SSIM = 0.8311).

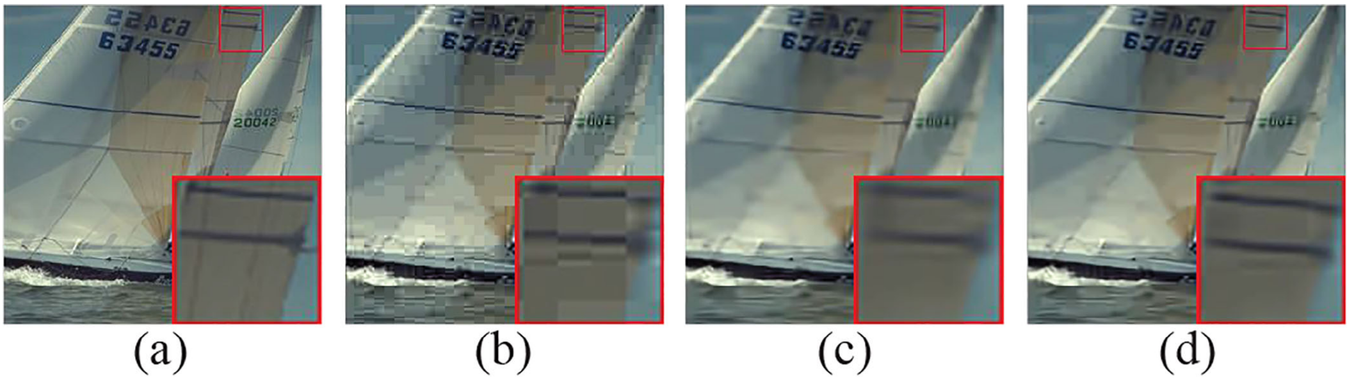

**FIGURE 8** Visualization of performance of various algorithms for deblocking image *sailing3* in the LIVE1 dataset (image size: 256 × 256) under compression quality $Q = 10$: (a) Original image, (b) JPEG compressed image (PSNR = 28.61 dB, SSIM = 0.7561), (c) SA-DCT (PSNR = 29.62 dB, SSIM = 0.8310), and (d) the proposed algorithm (PSNR = 29.96 dB, SSIM = 0.8457).

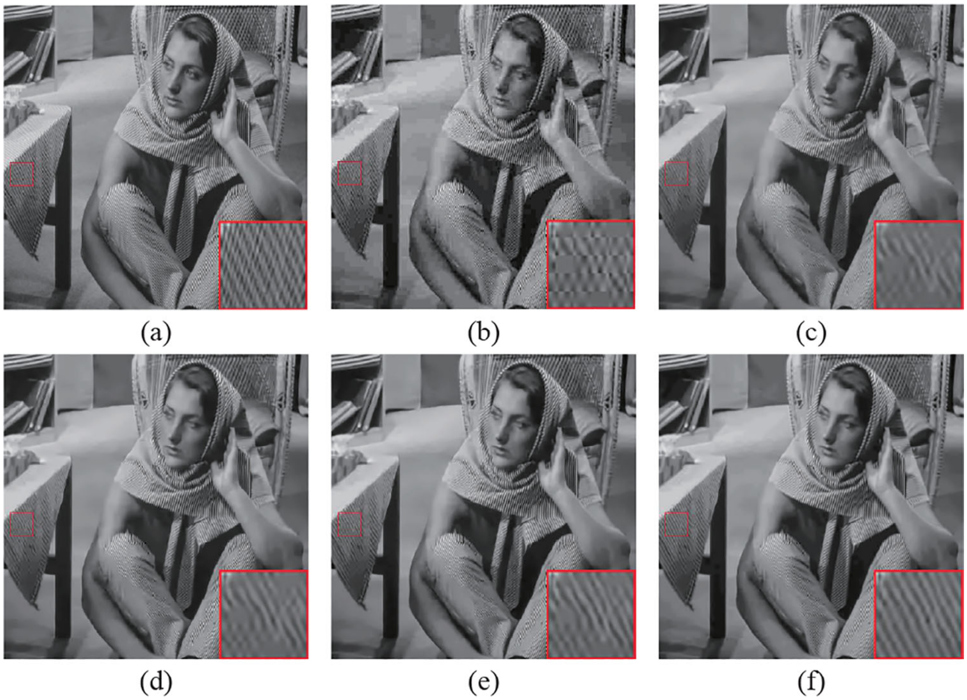

**FIGURE 9** Deblocking image *Barbara* in the Classic5 dataset under compression quality $Q = 10$: (a) Original image, (b) JPEG compressed image (PSNR = 25.78 dB, SSIM = 0.7621), (c) ARCNN (PSNR = 26.89 dB, SSIM = 0.7934), (d) TRND (PSNR = 27.24 dB, SSIM = 0.8104), (e) DnCNN (PSNR = 27.59 dB, SSIM = 0.8161), and (f) the proposed algorithm (PSNR = 28.26 dB, SSIM = 0.8335).

**TABLE 8** Average PSNR (db)/SSIM results comparing image deblocking with DNN-based methods on dataset Classic5.

| Methods | $Q = 10$ | $Q = 20$ | $Q = 30$ | Average |
|---|---|---|---|---|
| AR-CNN | 29.08 | 31.25 | 32.60 | 30.98 |
| | 0.7909 | 0.8514 | 0.8808 | 0.8410 |
| TRND | 29.29 | 31.48 | 32.79 | 31.19 |
| | 0.7996 | 0.8581 | 0.8841 | 0.8473 |
| DnCNN | 29.40 | 31.63 | 32.91 | 31.31 |
| | 0.8026 | 0.8610 | 0.8861 | 0.8499 |
| DCSC | 29.62 | 31.81 | 33.06 | 31.50 |
| | 0.8096 | 0.8641 | 0.8882 | 0.8540 |
| MDDU | 29.95 | 32.11 | 33.33 | 31.80 |
| | 0.8171 | 0.8689 | 0.8916 | 0.8592 |
| **Ours** | 29.43 | 31.65 | 32.88 | 31.32 |
| | 0.8047 | 0.8608 | 0.8855 | 0.8503 |

2) Comparison with DNN-based image deblocking models: To further demonstrate the superiority of the proposed method in the image deblocking task, experiments were conducted to compare it with several deep learning-based methods, including AR-CNN [66], TRND [58], DnCNN [48], DCSC [67], and MDDU [68]. The comparison experiment was conducted on the Classic5 [51] dataset, which is a popular benchmark dataset in the field of image deblocking. Table 8 presents the average PSNR and SSIM results at different compression qualities $Q$.

The results indicate that the proposed method achieves better results compared to AR-CNN and TRND, while performing comparably to DnCNN, DCSC, and MDDU. It is worth noting that these supervised deep learning methods require large-scale image datasets to train the image deblocking models. It can be observed that if the training image dataset and the distribution of test images are similar or identical, then deep learning models can effectively adapt to different image structures.

However, it was observed in the experimental results that deep learning methods tend to cause excessive smoothing in the restored images, especially for texture-rich images, as shown in Figure 9. To further validate this finding, experiments were conducted using eight fingerprint images collected from the NIST dataset as the test benchmark, as shown in Figure 10. The average deblocking results for the eight fingerprint images are presented in Table 9. The proposed method outperforms all other deep learning-based image deblocking methods. Visual comparison results are shown in Figure 11, where it can be observed that the proposed method reconstructs better texture details compared to other methods.

## 4.5 | Convergence

Since the proposed algorithm involves block grouping operations, non-convex optimization, and parameter updates, it is challenging to provide theoretical proof for the local

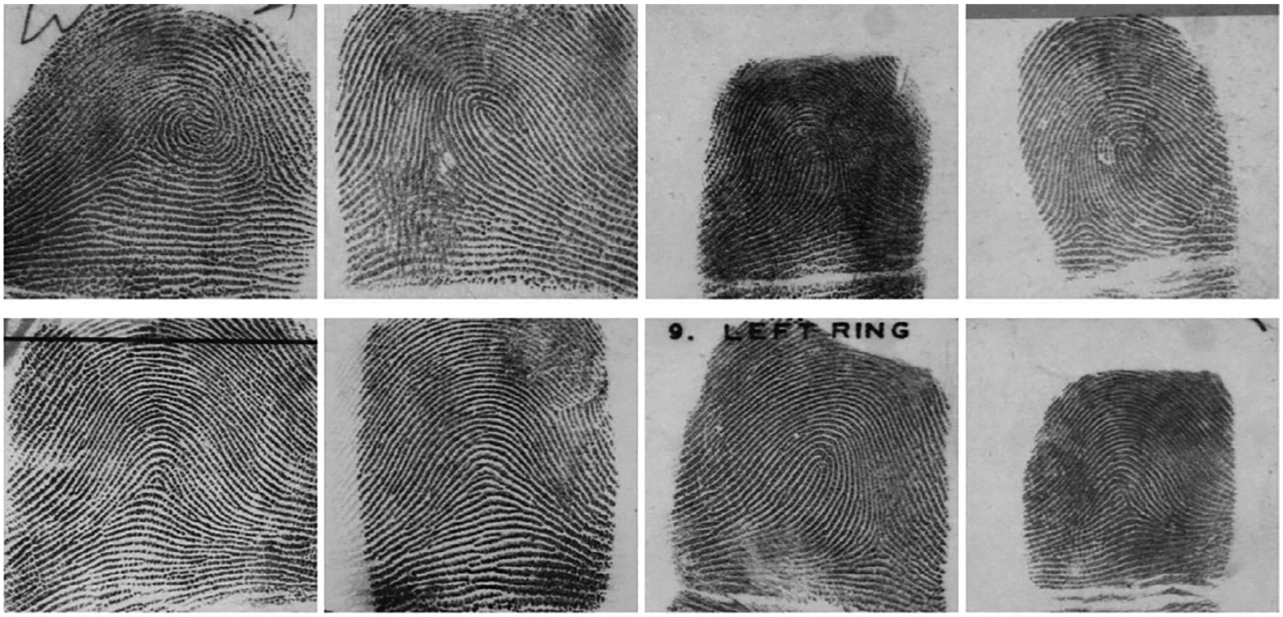

**FIGURE 10** Eight fingerprint test images selected from the NIST dataset.

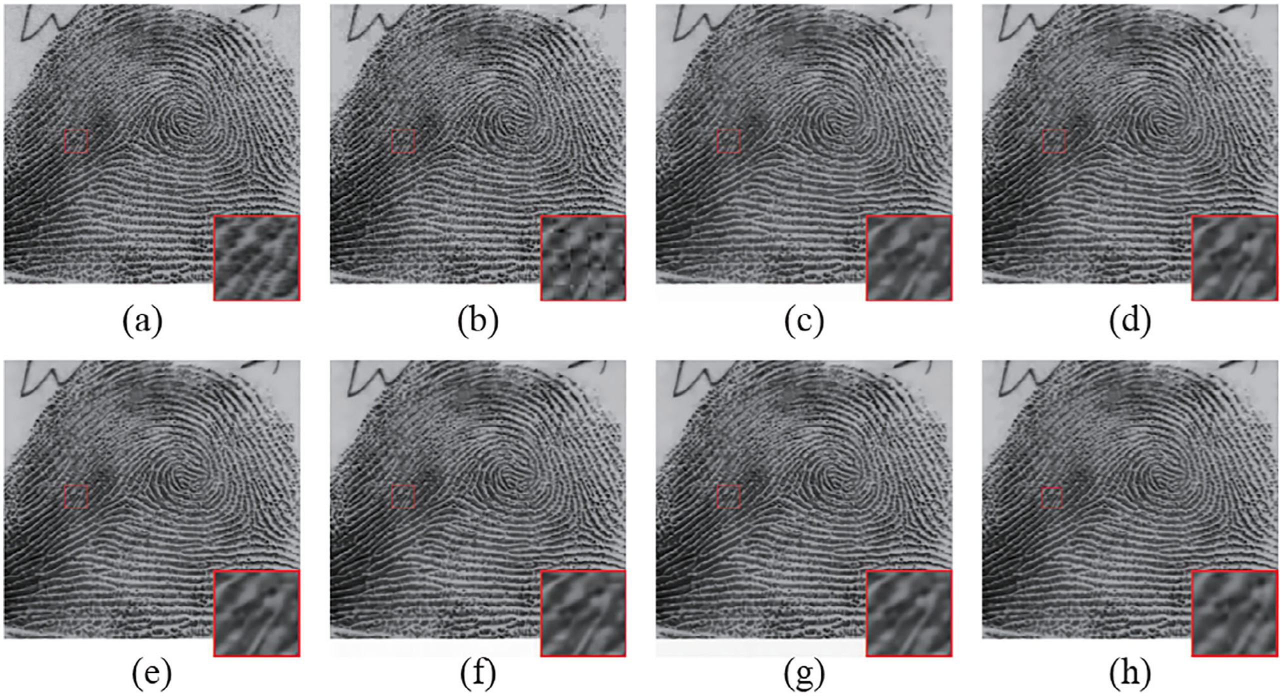

**FIGURE 11** Performance of algorithms for deblocking image *image 01* in the Classic5 dataset under compression quality $Q = 10$: (a) Original image, (b) JPEG compressed image (PSNR = 28.41 dB, SSIM = 0.8737), (c) ARCNN (PSNR = 29.57 dB, SSIM = 0.8969), (d) TRND (PSNR = 29.73 dB, SSIM = 0.9008), (e) DnCNN (PSNR = 29.72 dB, SSIM = 0.9019), (f) DCSC (PSNR = 29.82 dB, SSIM = 0.9045), (g) MDDU (PSNR = 29.82 dB, SSIM = 0.9081), and (h) the proposed algorithm (PSNR = 30.13 dB, SSIM = 0.9083).

convergence of the proposed algorithm. Therefore, this section provides experimental evidence to validate the convergence of the proposed algorithm. Five test images were selected from the test dataset, and the process of restoring these images was recorded. Figure 12a,b shows the variation curves of PSNR values during the iterations of the image denoising with noise level $\sigma_E = 50$ and image deblocking with compression quality $Q = 10$ algorithms, respectively. It can be clearly observed that as the number of algorithm iterations increases, the PSNR curves of all restored images first monotonically increase and

**TABLE 9** Average PSNR (db)/SSIM results comparing image deblocking with DNN-based methods on fingerprint image dataset.

| Methods | $Q = 10$ | $Q = 20$ | $Q = 30$ | Average |
|---|---|---|---|---|
| AR-CNN | 30.23 | 33.04 | 34.76 | 32.68 |
| | 0.8859 | 0.9291 | 0.9480 | 0.9210 |
| TRND | 30.42 | 33.19 | 34.87 | 32.83 |
| | 0.8899 | 0.9317 | 0.9492 | 0.9236 |
| DnCNN | 30.31 | 33.07 | 34.73 | 32.70 |
| | 0.8894 | 0.9308 | 0.9485 | 0.9229 |
| DCSC | 30.52 | 33.13 | 34.78 | 32.81 |
| | 0.8934 | 0.9330 | 0.9497 | 0.9254 |
| MDDU | 30.45 | 32.95 | 34.35 | 32.58 |
| | 0.8961 | **0.9349** | **0.9508** | 0.9273 |
| **Ours** | **30.81** | **33.54** | **35.20** | **33.18** |
| | **0.8967** | 0.9344 | 0.9507 | **0.9273** |

**TABLE 10** Average PSNR (dB) results of image denoising and ablation experiments on 13 commonly used test images.

| Modules | 20 | 30 | 40 | 50 | 75 | 100 | Average |
|---|---|---|---|---|---|---|---|
| SR | 30.26 | 27.61 | 27.27 | 26.70 | 24.90 | 23.72 | 26.74 |
| LR | 25.70 | 23.23 | 17.59 | 15.70 | 12.30 | 9.18 | 17.28 |
| SR+LR | **32.24** | 30.13 | 28.56 | 27.60 | 25.66 | 24.23 | 28.07 |
| SR+LR+SP | 32.20 | **30.25** | **28.77** | **27.68** | **25.69** | **24.30** | **28.15** |

then gradually stabilize. Therefore, it can be proved that the proposed algorithm exhibits good convergence.

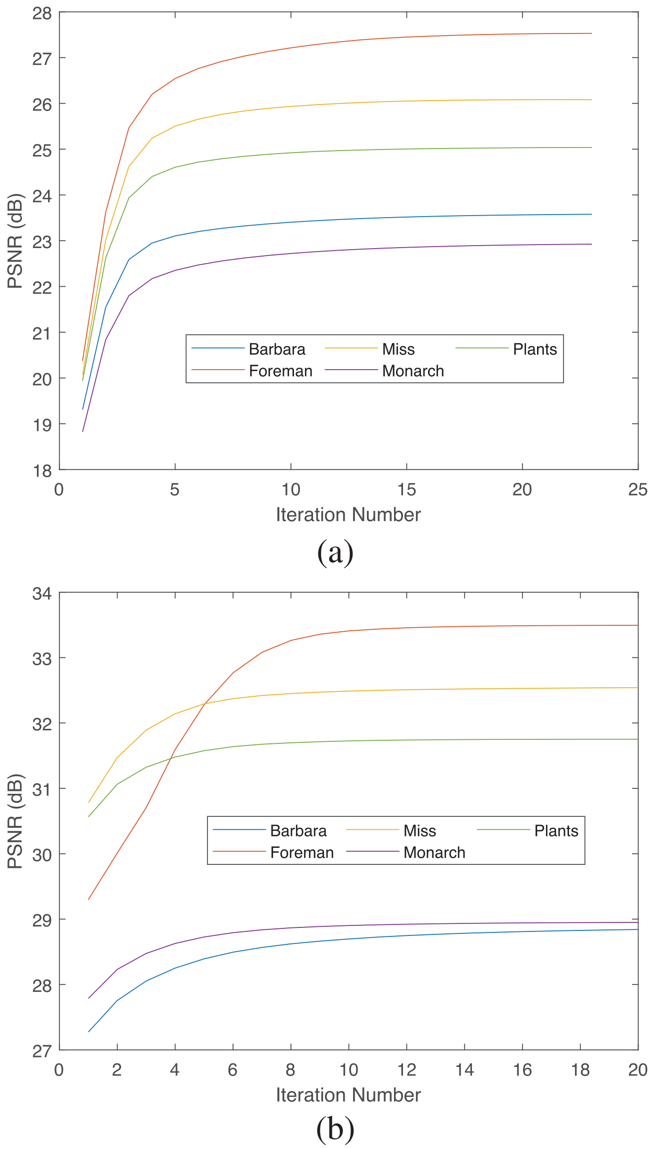

**FIGURE 12** Convergence of the proposed algorithm with various strategies: (a) how PSNR changes as the number of iterations increases with noise level $\sigma_E = 50$ and (b) how PSNR changes as the number of iterations increases with compression quality $Q = 10$.

## 4.6 | Ablation study

From the objective function 21, it can be seen that the proposed algorithm consists of three main modules: group sparse representation (SR), low-rank self-representation (LR), and structure preservation (SP). In order to investigate the effectiveness of these different modules in the proposed algorithm, this section conducts ablation experiments by separately removing the low-rank self-representation guidance module ($\gamma = 0$), the sparse constraint ($\beta = 0$), and the structure preservation term ($\lambda = 0$), to verify the roles played by each module. The ablation experiments are conducted using 13 widely used test images and applying these modules to image denoising. The average PSNR results are shown in Table 10. It can be observed that the low-rank self-representation also has the effect of noise removal, and the group sparse representation model guided by low-rank self-representation achieves a significant improvement in denoising performance compared to the single group sparse representation model. Additionally, the introduced structure preservation module in this chapter also contributes to the improvement in performance.

To further reveal the roles of each module in the proposed algorithm, Figures 13 and 14, respectively demonstrate the denoising results of each module on images *Lena* and *Plants*. As shown in Figures 13c and 14c, the sparse representation module indeed serves as an effective tool for image restoration, but it is susceptible to noise, resulting in some undesirable artefacts such as pseudo structures. Although the low-rank self-representation has a minor effect on denoising, combining the sparse representation model with the low-rank self-representation significantly improves the denoising performance, as illustrated in Figures 13e and 14e. Similarly, by introducing the structure preservation module, it can be observed from Figures 13f and 14f that the images retain well-preserved texture details.

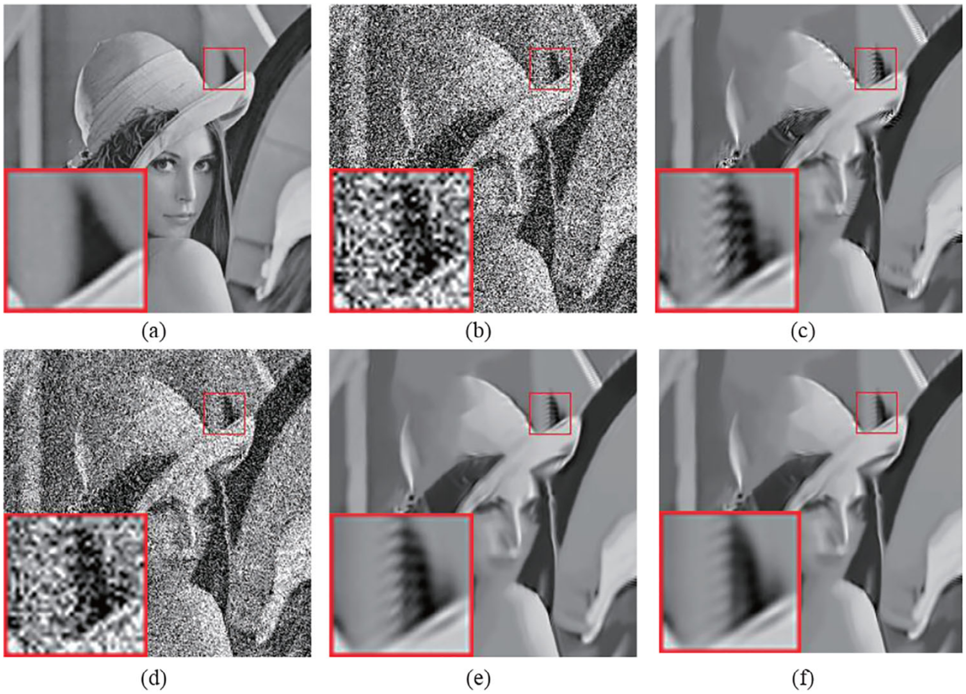

**FIGURE 13** Visualization of algorithms for denoising image *Lena* under noise $\sigma_E = 75$: (a) Original image, (b) noisy image, (c) SR (PSNR = 25.04 dB, SSIM = 0.7487), (d) LR (PSNR = 12.31 dB, SSIM = 0.1234), (e) SR+LR (PSNR = 25.50 dB, SSIM = 0.7554), and (f) SR+LR+SP (PSNR = 25.51 dB, SSIM = 0.7562).

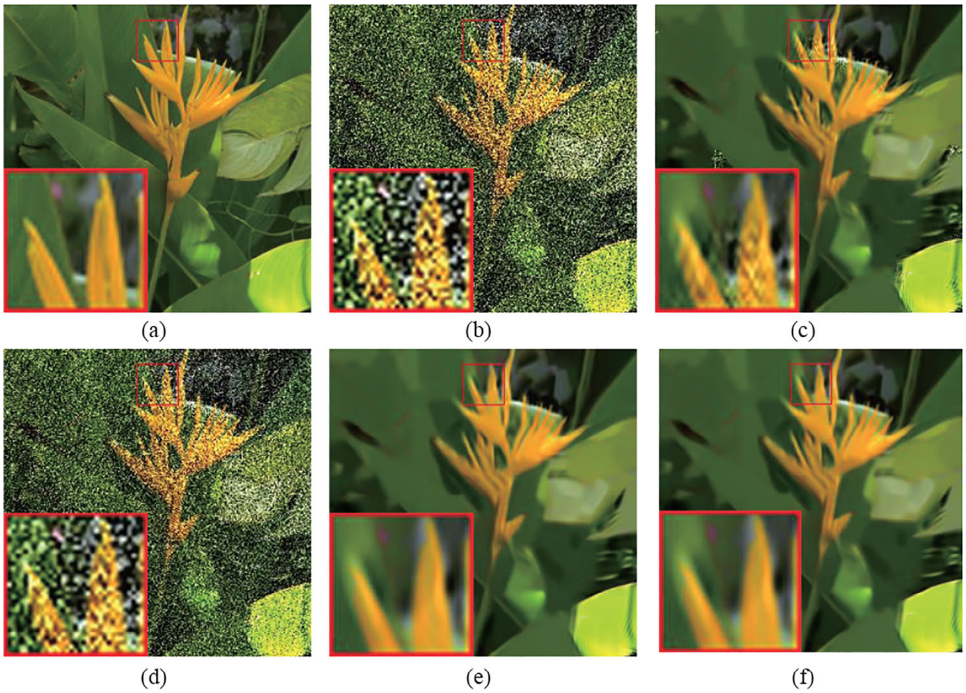

**FIGURE 14** Visualization of algorithms for denoising image *Plants* under noise $\sigma_E = 75$: (a) Original image, (b) noisy image, (c) SR (PSNR = 25.51 dB, SSIM = 0.7116), (d) LR (PSNR = 12.32 dB, SSIM = 0.0820), (e) SR+LR (PSNR = 26.46 dB, SSIM = 0.7270), and (f) SR+LR+SP (PSNR = 26.50 dB, SSIM = 0.7278).

In addition, our previous work [71] also integrated the sparsity and low-rank self-representation properties of images. However, that method was based on sparse coding for self-representation learning, leading to suboptimal solutions because each set of sparse coding coefficients could not guarantee low-rank self-representation properties. In this ablation experiment, we compared the proposed method with the method proposed in our previous work. The PSNR and SSIM results are shown

**TABLE 11** Comparison with the method proposed in [71] in the image denoising task.

| Methods | 20 | 30 | 40 | 50 | 75 | 100 |
|---|---|---|---|---|---|---|
| SRLR | 32.23 | 30.24 | 28.81 | 27.73 | 25.72 | 24.40 |
| | 0.9046 | 0.8711 | 0.8450 | 0.8198 | 0.7662 | 0.7255 |
| LRSR-SP | 32.20 | 30.25 | 28.77 | 27.68 | 25.69 | 24.30 |
| | **0.9062** | **0.8764** | **0.8477** | **0.8231** | **0.7687** | **0.7262** |

in Table 11. From the table, it can be seen that the proposed algorithm outperforms the previous method in terms of SSIM, which is closer to the human visual system's intuitive evaluation mechanism for image quality. This further demonstrates the effectiveness of the proposed method.

## 5 | CONCLUSION

Most existing group sparse representation models overlook the similarity relationships between non-local image blocks, while leveraging these relationships can effectively preserve texture information in images. Group sparse representation models apply simple sparsity constraints only to each image block within a group, neglecting other beneficial characteristics of images. To further explore the intrinsic properties of natural images, this chapter proposes a low-rank self-representation guided group sparse representation image restoration algorithm. Specifically, in addition to utilizing the group sparse representation regularization term, this algorithm also utilizes the low-rank self-representation property to jointly estimate the reconstructed image block groups. This low-rank self-representation model can better characterize the intrinsic properties of natural images, namely the correlation between similar image blocks. Additionally, to ensure that the learned sparse representation also preserves the similarity structure between image blocks, the algorithm also performs self-representation learning on the sparse representation, making the self-representation obtained as close as possible to the original self-representation between image blocks. Extensive experimental results demonstrate that this algorithm performs excellently in image restoration tasks such as image denoising and image deblocking.

However, this research still has limitations because it can only achieve excellent performance in scenarios with additive Gaussian noise. In future research, we will consider other noise distributions or multiplicative noise. Furthermore, since the proposed algorithm is self-supervised, it only utilizes the intrinsic information of the degraded images themselves without leveraging external prior knowledge, which limits the improvement of the current algorithm's performance. This is also a challenge that needs to be addressed in future work.

## AUTHOR CONTRIBUTIONS

**Yaxian Gao**: Data curation; formal analysis; writing—original draft. **Zhaoyuan Cai**: Data curation; formal analysis; writing—original draft. **Xianghua Xie**: Conceptualization; writing—review and editing. **Jingjing Deng**: Conceptualization; writing—review and editing. **Zengfa Dou**: Resources; validation. **Xiaoke Ma**: Conceptualization; funding acquisition; supervision; writing—review and editing.

## ACKNOWLEDGEMENTS

This work was supported by the Shaanxi Key Research and Development Program (Program No. 2021ZDLGY02-02).

## CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

## DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## ORCID

*Zengfa Dou* https://orcid.org/0000-0002-5162-6244
*Xiaoke Ma* https://orcid.org/0000-0002-5604-7137

## REFERENCES

1. Banham, M.R., Katsaggelos, A.K.: Digital image restoration. IEEE Signal Process Mag. 14(2), 24–41 (1997)
2. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. Image Process. 15(12), 3736–3745 (2006)
3. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE, Piscataway, NJ (2008)
4. Jung, C., Jiao, L., Qi, H., Sun, T.: Image deblocking via sparse representation. Signal Process. Image Commun. 27(6), 663–677 (2012)
5. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D 60(1-4), 259–268 (1992)
6. Osher, S., Burger, M., Goldfarb, D., Xu, J., Yin, W.: An iterative regularization method for total variation-based image restoration. Multiscale Model. Simul. 4(2), 460–489 (2005)
7. Bappy, D., Jeon, I.: Combination of hybrid median filter and total variation minimisation for medical X-ray image restoration. IET Image Proc. 10(4), 261–271 (2016)
8. Dong, W., Zhang, L., Shi, G., Li, X.: Nonlocally centralized sparse representation for image restoration. IEEE Trans. Image Process. 22(4), 1620–1630 (2012)
9. Ji, H., Huang, S., Shen, Z., Xu, Y.: Robust video restoration by joint sparse and low rank matrix approximation. SIAM J. Imag. Sci. 4(4), 1122–1142 (2011)
10. Dong, W., Shi, G., Li, X.: Nonlocal image restoration with bilateral variance estimation: A low-rank approach. IEEE Trans. Image Process. 22(2), 700–711 (2012)
11. Gu, S., Zhang, L., Zuo, W., Feng, X.: Weighted nuclear norm minimization with application to image denoising. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2862–2869. IEEE, Piscataway, NJ (2014)
12. Liu, D., Wen, B., Fan, Y., Loy, C.C., Huang, T.S.: Non-local recurrent network for image restoration. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, pp. 1680–1689. Curran Associates Inc., Red Hook, NY (2018)
13. Mastan, I.D., Raman, S.: Deepcfl: Deep contextual features learning from a single image. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2897–2906. IEEE, Piscataway, NJ (2021)

14. Zha, Z., Yuan, X., Zhou, J.T., Zhou, J., Wen, B., Zhu, C.: The power of triply complementary priors for image compressive sensing. In: Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), pp. 983–987. IEEE, Piscataway, NJ (2020)
15. Mataev, G., Milanfar, P., Elad, M.: Deepred: Deep image prior powered by red. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, IEEE, Piscataway, NJ (2019). arXiv:1903.10176
16. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5728–5739. IEEE, Piscataway, NJ (2022)
17. Shen, L., Pauly, J., Xing, L.: NERP: Implicit neural representation learning with prior embedding for sparsely sampled image reconstruction. IEEE Trans. Neural Networks Learn. Syst. 35(1), 770–782 (2022)
18. Chen, L., Chu, X., Zhang, X., Sun, J.: Simple baselines for image restoration. In: Proceedings of the European Conference on Computer Vision, pp. 17–33. Springer, Berlin (2022)
19. Zha, Z., Wen, B., Yuan, X., Zhou, J., Zhu, C.: Simultaneous nonlocal low-rank and deep priors for Poisson denoising. In: Proceedings of the ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2320–2324. IEEE, Piscataway, NJ (2022)
20. Wu, W., Chen, M., Xiang, Y., Zhang, Y., Yang, Y.: Recent progress in image denoising: A training strategy perspective. IET Image Proc. 17(6), 1627–1657 (2023)
21. Deeba, F., Kun, S., Dharejo, F.A., Zhou, Y.: Sparse representation based computed tomography images reconstruction by coupled dictionary learning algorithm. IET Image Proc. 14(11), 2365–2375 (2020)
22. Zha, Z., Yuan, X., Wen, B., Zhang, J., Zhu, C.: Nonconvex structural sparsity residual constraint for image restoration. IEEE Trans. Cybern. 52(11), 12440–12453 (2021)
23. Yuan, W., Liu, H., Liang, L.: Image restoration via exponential scale mixture-based simultaneous sparse prior. IET Image Proc. 16(12), 3268–3283 (2022)
24. Xu, W., Zhu, Q., Qi, N., Chen, D.: Deep sparse representation based image restoration with denoising prior. IEEE Trans. Circuits Syst. Video Technol. 32(10), 6530–6542 (2022)
25. Buades, A., Coll, B., Morel, J.M.: A non-local algorithm for image denoising. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2, pp. 60–65. IEEE, Piscataway, NJ (2005)
26. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. IEEE Trans. Image Process. 16(8), 2080–2095 (2007)
27. Zhang, J., Zhao, D., Gao, W.: Group-based sparse representation for image restoration. IEEE Trans. Image Process. 23(8), 3336–3351 (2014)
28. Li, X., Shen, H., Li, H., Zhang, L.: Patch matching-based multitemporal group sparse representation for the missing information reconstruction of remote-sensing images. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 9(8), 3629–3641 (2016)
29. Wang, Q., Zhang, X., Wu, Y., Tang, L., Zha, Z.: Nonconvex weighted $\ell_p$ minimization based group sparse representation framework for image denoising. IEEE Signal Process Lett. 24(11), 1686–1690 (2017)
30. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. Signal Process. 54(11), 4311–4322 (2006)
31. Daubechies, I.: The wavelet transform, time-frequency localization and signal analysis. IEEE Trans. Inf. Theory 36(5), 961–1005 (1990)
32. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Non-local sparse models for image restoration. In: Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, pp. 2272–2279. IEEE, Piscataway, NJ (2009)
33. Zhang, J., Zhang, Y., Gu, J., Zhang, Y., Kong, L., Yuan, X.: Accurate image restoration with attention retractable transformer. arXiv:2210.01427 (2022). Available from: https://api.semanticscholar.org/CorpusID:252693111
34. Chen, X., Li, H., Li, M., Pan, J.: Learning a sparse transformer network for effective image deraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5896–5905. IEEE, Piscataway, NJ (2023)
35. Zhou, S., Chen, D., Pan, J., Shi, J., Yang, J.: Adapt or perish: Adaptive sparse transformer with attentive feature refinement for image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2952–2963. IEEE, Piscataway, NJ (2024)
36. Xu, J., Zhang, L., Zuo, W., Zhang, D., Feng, X.: Patch group based nonlocal self-similarity prior learning for image denoising. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 244–252. IEEE, Piscataway, NJ (2015)
37. Zha, Z., Wen, B., Yuan, X., Zhou, J., Zhu, C., Kot, A.C.: A hybrid structural sparsification error model for image restoration. IEEE Trans. Neural Networks Learn. Syst. 33(9), 4451–4465 (2021)
38. Yuan, W., Liu, H., Liang, L.: Joint group dictionary-based structural sparse representation for image restoration. Digital Signal Process. 137, 104029 (2023)
39. Zha, Z., Wen, B., Yuan, X., Zhou, J., Zhu, C., Kot, A.C.: Low-rankness guided group sparse representation for image restoration. IEEE Trans. Neural Networks Learn. Syst. 34(10), 7593–7607 (2022)
40. Niknejad, M., Rabbani, H., Babaie-Zadeh, M.: Image restoration using gaussian mixture models with spatially constrained patch clustering. IEEE Trans. Image Process. 24(11), 3624–3636 (2015)
41. Candes, E.J., Tao, T.: Near-optimal signal recovery from random projections: Universal encoding strategies? IEEE Trans. Inf. Theory 52(12), 5406–5425 (2006)
42. Donoho, D.L.: For most large underdetermined systems of linear equations the minimal $\ell_1$-norm solution is also the sparsest solution. Commun. Pure Appl. Math. 59(6), 797–829 (2006)
43. Chartrand, R., Yin, W.: Iteratively reweighted algorithms for compressive sensing. In: Proceedings of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3869–3872. IEEE, Piscataway, NJ (2008)
44. Mazumder, R., Friedman, J.H., Hastie, T.: Sparsenet: Coordinate descent with nonconvex penalties. J. Am. Stat. Assoc. 106(495), 1125–1138 (2011)
45. Vidal, R., Favaro, P.: Low rank subspace clustering (LRSC). Pattern Recognit. Lett. 43, 47–61 (2014)
46. Zuo, W., Meng, D., Zhang, L., Feng, X., Zhang, D.: A generalized iterated shrinkage algorithm for non-convex sparse coding. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 217–224. IEEE, Piscataway, NJ (2013)
47. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In: Proceedings of the 23rd International Conference on Neural Information Processing Systems, pp. 1033–1041. Curran Associates Inc., Red Hook, NY (2009)
48. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE Trans. Image Process. 26(7), 3142–3155 (2017)
49. Plotz, T., Roth, S.: Benchmarking denoising algorithms with real photographs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1586–1595. IEEE, Piscataway, NJ (2017)
50. Sheikh, H.: Live image quality assessment database release 2. http://liveeceutexasedu/research/quality (2005). Accessed 15 Apr 2024
51. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Proceedings of the International Conference on Curves and Surfaces, pp. 711–730. Springer, Berlin (2010)
52. Zoran, D., Weiss, Y.: From learning models of natural image patches to whole image restoration. In: Proceedings of the 2011 International Conference on Computer Vision, pp. 479–486. IEEE, Piscataway, NJ (2011)
53. Zhang, L., Dong, W., Zhang, D., Shi, G.: Two-stage image denoising by principal component analysis with local pixel grouping. Pattern Recognit. 43(4), 1531–1549 (2010)
54. Luo, E., Chan, S.H., Nguyen, T.Q.: Adaptive image denoising by mixture adaptation. IEEE Trans. Image Process. 25(10), 4489–4503 (2016)
55. Liu, H., Tan, S.: Image regularizations based on the sparsity of corner points. IEEE Trans. Image Process. 28(1), 72–87 (2018)

56. Zha, Z., Yuan, X., Zhou, J., Zhu, C., Wen, B.: Image restoration via simultaneous nonlocal self-similarity priors. IEEE Trans. Image Process. 29, 8561–8576 (2020)
57. Zha, Z., Yuan, X., Wen, B., Zhang, J., Zhou, J., Zhu, C.: Image restoration using joint patch-group-based sparse representation. IEEE Trans. Image Process. 29, 7735–7750 (2020)
58. Chen, Y., Pock, T.: Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. IEEE Trans. Pattern Anal. Mach. Intell. 39(6), 1256–1272 (2016)
59. Quan, Y., Chen, M., Pang, T., Ji, H.: Self2self with dropout: Learning self-supervised denoising from single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1890–1898. IEEE, Piscataway, NJ (2020)
60. Foi, A., Katkovnik, V., Egiazarian, K.: Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images. IEEE Trans. Image Process. 16(5), 1395–1411 (2007)
61. Ren, J., Liu, J., Li, M., Bai, W., Guo, Z.: Image blocking artifacts reduction via patch clustering and low-rank minimization. In: Proceedings of the 2013 Data Compression Conference, pp. 516–516. IEEE, Piscataway, NJ (2013)
62. Gu, S., Xie, Q., Meng, D., Zuo, W., Feng, X., Zhang, L.: Weighted nuclear norm minimization and its applications to low level vision. Int. J. Comput. Vision 121(2), 183–208 (2017)
63. Zhang, X., Xiong, R., Fan, X., Ma, S., Gao, W.: Compression artifact reduction by overlapped-block transform coefficient estimation with block similarity. IEEE Trans. Image Process. 22(12), 4613–4626 (2013)
64. Zhao, C., Zhang, J., Ma, S., Fan, X., Zhang, Y., Gao, W.: Reducing image compression artifacts by structural sparse representation and quantization constraint prior. IEEE Trans. Circuits Syst. Video Technol. 27(10), 2057–2071 (2016)
65. Young, S.I., Naman, A.T., Taubman, D.: COGL: Coefficient graph Laplacians for optimized JPEG image decoding. IEEE Trans. Image Process. 28(1), 343–355 (2018)
66. Dong, C., Deng, Y., Loy, C.C., Tang, X.: Compression artifacts reduction by a deep convolutional network. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 576–584. IEEE, Piscataway, NJ (2015)
67. Fu, X., Zha, Z.J., Wu, F., Ding, X., Paisley, J.: JPEG artifacts reduction via deep convolutional sparse coding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2501–2510. IEEE, Piscataway, NJ (2019)
68. Fu, X., Wang, M., Cao, X., Ding, X., Zha, Z.J.: A model-driven deep unfolding method for JPEG artifacts removal. IEEE Trans. Neural Networks Learn. Syst. 33(11), 6802–6816 (2021)
69. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. 13(4), 600–612 (2004)
70. Immerkaer, J.: Fast noise variance estimation. Comput. Vision Image Understanding 64(2), 300–302 (1996)
71. Cai, Z., Xie, X., Deng, J., Dou, Z., Tong, B., Ma, X.: Image restoration with group sparse representation and low-rank group residual learning. IET Image Proc. 18(3), 741–760 (2024)

**How to cite this article:** Gao, Y., Cai, Z., Xie, X., Deng, J., Dou, Z., Ma, X.: Sparse representation for restoring images by exploiting topological structure of graph of patches. IET Image Process. 19, e70004 (2025). https://doi.org/10.1049/ipr2.70004