

Reweighted Low-Rank Matrix Analysis With Structural Smoothness for Image Denoising

Hengyou Wang[✉], Yigang Cen, *Member, IEEE*, Zhiqian He, Zhihai He, *Fellow, IEEE*, Ruizhen Zhao, and Fengzhen Zhang

Abstract—In this paper, we develop a new low-rank matrix recovery algorithm for image denoising. We incorporate the total variation (TV) norm and the pixel range constraint into the existing reweighted low-rank matrix analysis to achieve structural smoothness and to significantly improve quality in the recovered image. Our proposed mathematical formulation of the low-rank matrix recovery problem combines the nuclear norm, TV norm, and l_1 norm, thereby allowing us to exploit the low-rank property of natural images, enhance the structural smoothness, and detect and remove large sparse noise. Using the iterative alternating direction and fast gradient projection methods, we develop an algorithm to solve the proposed challenging non-convex optimization problem. We conduct extensive performance evaluations on single-image denoising, hyper-spectral image denoising, and video background modeling from corrupted images. Our experimental results demonstrate that the proposed method outperforms the state-of-the-art low-rank matrix recovery methods, particularly for large random noise. For example, when the density of random sparse noise is 30%, for single-image denoising, our proposed method is able to improve the quality of the restored image by up to 4.21 dB over existing methods.

Index Terms—Low-rank matrix recovery, reweighted, TV norm, smooth, bound constraint.

I. INTRODUCTION

LOW-RANK and sparse matrix analysis [1]–[8] has been an important research topic, resulting in broad applica-

Manuscript received May 23, 2017; revised September 30, 2017 and November 6, 2017; accepted November 26, 2017. Date of publication December 8, 2017; date of current version January 12, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61502024, Grant 61572067, and Grant 61611530710, in part by the China Scholarship Council under Grant 201708110139, in part by the Beijing Municipal Natural Science Foundation under Grant 4162050, in part by the Natural Science Foundation of Guangdong Province under Grant 2016A030313708, in part by the Science and Technology Plan of Beijing Municipal Education Commission under Grant SQKM201610016009, and in part by the Talent Program of Beijing University of Civil Engineering and Architecture. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jie Liang. (*Corresponding author: Zhiqian He*)

H. Wang is with the School of Science, Beijing University of Civil Engineering and Architecture, Beijing 100044, China, also with the Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China, and also with the Key Laboratory of Advanced Information Science and Network Technology of Beijing, Beijing Jiaotong University, Beijing 100044, China.

Y. Cen, R. Zhao, and F. Zhang is with the Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China, and also with the Key Laboratory of Advanced Information Science and Network Technology of Beijing, Beijing Jiaotong University, Beijing 100044, China.

Z. He is with the College of Information Engineering, Shenzhen University, Shenzhen, 518060, China (e-mail: zhiqian@szu.edu.cn).

Z. He is with the Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO 65211 USA.

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

Digital Object Identifier 10.1109/TIP.2017.2781425

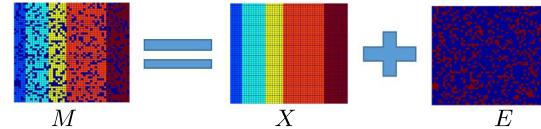


Fig. 1. Illustration of low-rank matrix recovery.

tions in image processing [9]–[14], computer vision [15]–[17], recommendation systems [18] and machine learning [19], [20]. This analysis aims to identify low-dimensional structures from high-dimensional data. The models can be formulated as the joint low-rank and sparse matrix minimization problem, also called *robust principal component analysis* (RPCA) [21], which minimizes the matrix rank function and l_0 norm. However, due to the highly nonlinear and nonconvex properties of the matrix rank function and l_0 norm, the minimization problem is difficult to solve directly.

With the development of low-rank matrix recovery, the nuclear norm and l_1 norm have become popular choices for low-rank and sparse matrix minimization, with theoretical guarantees and competitive performance in practice. This modification is known as the tightest convex relaxation of the rank minimization problem [22], [23]. In [24]–[26], it was demonstrated that the nuclear norm minimization achieves the same solution as the original rank minimization approach with overwhelming probability.

As shown in Fig.1, let M be the observed matrix. The objective of low-rank matrix recovery is to decompose M into a low-rank matrix X and a sparse error matrix E . In nuclear norm minimization, the low-rank matrix recovery is formulated as a minimization problem, which minimizes the sum of the nuclear norm of X and the l_1 norm of E . This optimization problem is referred to as principal component pursuit (PCP). Several convex optimization algorithms have been proposed to solve this type of convex optimization problem, such as the semi-definite programming (SDP) method [27], the accelerated proximal gradient (APG) algorithm [28], singular value thresholding (SVT) [29], inexact augmented Lagrangian multipliers (IALM) [30], and so forth. The complexity of SDP is $O(n^6)$ for an $n \times n$ matrix [31], which is prohibitive for large-scale data analysis. The APG method requires that at least one term of the objective function has a Lipschitz continuous gradient. This assumption is often not reasonable in practice. Compared with SDP, APG and SVT, the IALM method is the most widely used method. However, this method typically requires introducing several auxiliary variables that correspond to nonsmooth terms. These auxiliary variables will slow the convergence process and occasionally result in divergence when there are too many variables. To solve this problem, Lu *et al.* [31] proposed making the objective

function smoother by introducing regularization terms. Then, they proposed the iteratively reweighted least squares (IRLS) method for solving the relaxed smooth problem by alternately updating the variable and its weight. Although these classical algorithms perform well in many low-rank matrix recovery scenarios, their performance degrades when the target matrix has complicated structures. If the intrinsic rank of the data matrix is high or the errors become dense, these methods may not be able to successfully recover the data matrix. Furthermore, these methods mainly exploit the low-rank structure of the matrix; the structural smoothness in the data has not been adequately addressed.

To improve the performance of low-rank matrix recovery, inspired by [32], Peng *et al.* [33] proposed a reweighted low-rank matrix recovery method. They introduced an iterative reweighting approach for low-rank matrix recovery and applied the non-uniform singular value thresholding (NSVT) operation for matrix enhancement. Their method has successfully improved the performance in image restoration. However, this method still suffers from significant performance degradation when addressing matrices with high intrinsic rank structures or high noise density. For example, this method is not able to efficiently remove sparse large noise with high density due to the absence of a proper regularization scheme. Note that the image structural smoothness has not been addressed in this method.

Researchers have long recognized that total variation (TV) regularization is a useful tool for enhancing the smoothness in image processing. It can effectively preserve the smoothness and edge feature in image denoising [34], [35]. The TV norm has also been explored for many other image processing applications, e.g., segmentation, super-resolution [36], and image reconstruction [37]. For example, Shi *et al.* [11], [12] proposed image super-resolution with low-rank and total variation regularizations, and successfully improved the performance on image super-resolution based on low-rank matrix theory. However, the model in [11] and [12] is more emphasis on the low-rank matrix recovery problem. In this paper, we propose a new model that incorporating the TV norm into the low-rank matrix recovery analysis framework to ensure structural smoothness in the recovered image, which is more suitable for the case that the matrix has a higher rank and corrupted by the large sparse noise. Fig.2 shows one example of image recovery from dense noise with and without using the TV norm. As shown in this figure, including the TV norm in the objective function is able to significantly improve the structural smoothness and image quality by approximately 2.64 dB in the output image.

Another issue that has not been carefully addressed in the existing low-rank matrix analysis for image restoration is the pixel value range constraint. As we known, each pixel in the restored image must range from 0 to 255 or 0 to 1 if the pixel value is normalized. SVT operation is often used in low-rank matrix recovery algorithms, and it does not guarantee the bound constraint. In existing methods, this constraint is applied to the final output of the low-rank matrix analysis as a post-processing step. As shown in Fig. 5, some pixels are out of the bound [0,1] when the bound constraint is not included. It may not have a significant effect on the quality of the restored image, but it does exist and has a certain effect

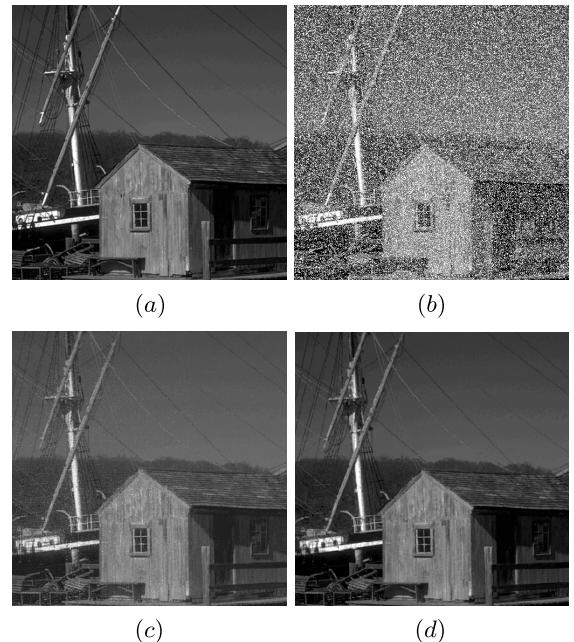


Fig. 2. Comparison of the denoising results of the with TV regular or not. 30% pixels of the frames are corrupted by large sparse noise. (a) original image. (b) The corrupted image. (c) Restored by model without TV regular. PSNR = 26.61(dB). (d) Restored by model with TV regular. PSNR = 29.25(dB).

on the output. In this work, we will propose a new image denoising method that incorporates the pixel range constraint in the low-rank matrix analysis.

The main contributions of this paper can be summarized as follows.

- 1) We incorporate the TV norm into the reweighted low-rank matrix recovery model, which is able to effectively separate the clean low-rank matrix from the high-density sparse noise while being able to maintain the smoothness of image structures.
- 2) We incorporate the pixel range constraint into the low-rank image recovery model to regulate the recovery of image pixels, which is able to improve the performance of image de-noising.
- 3) The optimization problem with the TV norm added into the objective function and the pixel range constraint becomes a challenging non-convex optimization problem. To effectively solve this problem, we have developed a new numerical solution based on inexact augmented Lagrangian multipliers (IALM) and non-uniform singular value thresholding (NSVT). Our experimental results show that the proposed method clearly improves both the objective and subjective quality of the restored images.

The remainder of this paper is organized as follows. Section 2 reviews related work on reweighted low-rank matrix recovery. Section 3 presents our proposed method. The solution of our proposed method is provided in Section 4. Section 5 summarizes our algorithm and analyzes its performance. The experimental results are presented in Section 6. Finally, Section 7 concludes the paper.

II. RELATED WORK

In this section, we review the existing work related to our proposed method.

A. Nuclear Norm Minimization-Based Low-Rank Matrix Recovery

It has been proven that the nuclear norm is the convex hull of the rank function of matrix [26]. The general sparse and low-rank minimization problem can be formulated as:

$$\begin{aligned} \min_{X,E} \quad & \|X\|_* + \lambda \|E\|_1, \\ \text{s.t. } & M = X + E, \end{aligned} \quad (1)$$

where $M \in R^{m \times n}$ is the observed matrix. $\|X\|_*$ is the nuclear norm of matrix X . Suppose that $U\Sigma V$ is the SVD of an arbitrary real matrix $X \in R^{m \times n}$, where $m \geq n$ is assumed in this paper. Σ is a diagonal matrix with diagonal entries, $\Sigma_{jj} = \tilde{\sigma}_j$ with decreasing magnitudes, where $\{\tilde{\sigma}_j\}$ are singular values of matrix X . The nuclear norm of X is denoted by $\|X\|_*$, which is the sum of the singular values of X .

B. Reweighted Low-Rank Matrix Recovery

The reweighted l_1 norm low-rank matrix recovery method was recently proposed by Deng *et al.* to enhance the sparsity of the error matrix [40]. This method aims to minimize the following function:

$$\begin{aligned} \min_{X,E} \quad & \|X\|_* + \lambda \|W_E \odot E\|_1, \\ \text{s.t. } & M = X + E, \end{aligned} \quad (2)$$

where $W_E \in R^{m \times n}$ is the weight of matrix $E = \{e_{ij}\} \in R^{m \times n}$. If the weight $w_{E,ij}$ equals the inverse of the absolute value of e_{ij} , i.e., $w_{E,ij} = 1/|e_{ij}|$ (for $e_{ij} \neq 0$), we have $\|W_E \odot E\|_1 = \|E\|_0$, where the symbol \odot denotes the component-wise product of two matrices. $\|\bullet\|_0$ is the number of nonzero entries of a matrix.

Similar to the method of the reweighted l_1 norm minimization problem, the low-rank matrix recovery problem based on reweighted nuclear norm [33] is formulated as:

$$\begin{aligned} \min_{X,E} \quad & \sum_{j=1}^n w_{X,j} \cdot \tilde{\sigma}_j + \lambda \|W_E \odot E\|_1 \\ \text{s.t. } & M = X + E \end{aligned} \quad (3)$$

where $W_X = \{w_{X,j}\}$ and $W_E \in R^{m \times n}$ are the weights for the set of $\{\tilde{\sigma}_j\}$ and entries of E , respectively. $\sum_j w_{X,j} \tilde{\sigma}_j$ is the reweighted nuclear norm of X . Furthermore, if $w_j = 1/\tilde{\sigma}_j$ (for $\tilde{\sigma}_j \neq 0$), it has $\sum_j w_{X,j} \tilde{\sigma}_j = \text{rank}(X)$. Thus, in problem (2), minimizing $\|X\|_*$ is equivalent to minimizing $\|\tilde{\sigma}\|_0$, where $\tilde{\sigma} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \dots, \tilde{\sigma}_n)$.

C. TV Norm-Based Image Restoration

The TV norm was used by Beck and Teboulle [38] to solve the image de-noising problem due to its capability in preserving edge information and enhance region smoothness. For a gray-level image $X \in R^{m \times n}$, the isotropic TV norm is defined in [38] as follows:

$$\begin{aligned} TV(X) = & \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} \sqrt{(x_{i,j} - x_{i+1,j})^2 + (x_{i,j} - x_{i,j+1})^2} \\ & + \sum_{i=1}^{m-1} |x_{i,n} - x_{i+1,n}| + \sum_{j=1}^{n-1} |x_{m,j} - x_{m,j+1}| \end{aligned} \quad (4)$$

The problem of image de-noising using TV norm can be formulated as a constrained convex problem [38]:

$$\begin{aligned} \min_X \quad & \|X - D\|_F^2 + 2\rho \|X\|_{TV} \\ \text{s.t. } & X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}, \end{aligned} \quad (5)$$

where D is the noising image and $B_{l,u}$ is a bounded set, which set the range of pixel values, for example [0, 255] with $l = 0$ and $u = 255$. The parameter ρ is used to balance these two norm terms in the objective function.

D. Non-Uniform Soft Thresholding Operator

In this subsection, the non-uniform soft thresholding (NST) operator which is used for reweighted l_1 norm minimization [32], will first be presented. Then, the non-uniform singular value thresholding (NSVT) operator for low-rank enhancement will be introduced. Finally, two more matrix operators are introduced, which will be useful in our work.

Operator 1 (Non-Uniform Soft Thresholding, NST [32]): Let $X \in R^n$, $W \in R_{++}^n$, the non-uniform soft thresholding operator S_W can be defined as follows:

$$S_W\{X\} = \{\text{sign}(x_i) \max(|y_i| - w_i, 0)\} \quad (6)$$

Operator 2 (Non-Uniform Singular Value Thresholding, NSVT [33]): Suppose that $X \in R^{m \times n}$ is a matrix of rank r , its Singular Value Decomposition (SVD) is denoted as

$$X = U\Sigma V^T, \quad \Sigma = \text{diag}(\{\tilde{\sigma}_i\}_{1 \leq i \leq r}) \quad (7)$$

where $\Sigma \in R^{r \times r}$ is a diagonal matrix with $\tilde{\sigma}_1, \dots, \tilde{\sigma}_r$ on the diagonal and zeros elsewhere. $\tilde{\sigma}_i > 0$ is the singular value of matrix X . $U \in R^{m \times r}$ and $V \in R^{n \times r}$ are left and right singular vectors, respectively. Given the threshold weights vector $W \in R_{++}^r$, the non-uniform singular value thresholding operator D_W can be defined as follows:

$$D_W\{X\} = US_W\{\Sigma\}V^T \quad (8)$$

where $S_W\{\Sigma\} = \text{diag}(\max((\tilde{\sigma}_i - w_i), 0))$.

Operator 3 [32]: Let $X \in R^{m \times n}$, $W_E \in R^{m \times n}$ and $Z \in R^{m \times n}$; then, the optimization problem

$$\min_X \varepsilon \|W_E \odot X\|_1 + \|X - Z\|_F^2/2 \quad (9)$$

has the optimal solution $X^* = S_{\varepsilon W_E}(Z)$, where $S(\bullet)$ is defined as operator 1.

Operator 4 [33]: Let $X \in R^{m \times n}$, $Z \in R^{m \times n}$ and $W_X = \{w_{X,j}\} \in R^r$; then, the optimization problem

$$\min_X \varepsilon \sum_{j=1}^r w_{X,j} \cdot \tilde{\sigma}_j + \|X - Z\|_F^2/2 \quad (10)$$

has the optimal solution $X^* = D_{\varepsilon W_X}(Z)$, where $D(\bullet)$ is defined as operator 2, and $\tilde{\sigma}_j$ is the singular value of matrix X .

III. FORMULATION OF THE PROPOSED METHOD

The reweighted low-rank matrix recovery algorithm has successfully improved the performance of the conventional low-rank matrix recovery, particularly for the matrices with a high intrinsic rank structure. However, it suffers from performance degradation for high-density image noise. In this section, inspired by the low-rank matrix recovery methods

in [33] and [38], we develop a smoothed low-rank matrix recovery algorithm using the re-weighted nuclear norm, which incorporates the TV norm into the low-rank matrix analysis framework. This new formulation is able to capture the local image correlation and structure. Mathematically, our new method can be formulated as follows:

$$\begin{aligned} & \min_{X, E} \sum_{j=1}^n w_{X,j} \cdot \tilde{\sigma}_j + \lambda \|W_E \odot E\|_1 + \eta \|X\|_{TV} \\ & \text{s.t. } X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}, \\ & \quad M = X + E, \end{aligned} \quad (11)$$

where $W_X = \{w_{X,j}\}$ and $W_E \in R^{m \times n}$ are weights for $\{\tilde{\sigma}_j\}$ and E , respectively. $\{\tilde{\sigma}_j\}$ are singular values of matrix X . The constraint $B_{l,u}$ state that pixel values are bounded, for example, $[0, 255]$.

IV. THE PROPOSED SOLUTION

Note that including the TV norm into the objective function and the pixel range constraint make the problem considerably more challenging. The mathematical solutions developed in existing methods [33], [38] can no longer be used to solve the optimization problem. In this section, we will develop an iterative method to solve our low-rank matrix recovery problem. To solve the optimization problem, a new variable is added to the model as follows:

$$\begin{aligned} & \min_{H, X, E} \sum_{j=1}^n w_{H,j} \cdot \sigma_j + \lambda \|W_E \odot E\|_1 + \eta \|X\|_{TV} \\ & \text{s.t. } X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}, \\ & \quad M = H + E, \quad H = X. \end{aligned} \quad (12)$$

where $W_H = \{w_{H,j}\}$ are the weights for $\{\sigma_j\}$, $\{\sigma_j\}$ are singular values of matrix H , and $w_{H,j} = w_{X,j}$, $j = 1, \dots, n$.

Thus, the augmented Lagrangian function of (12) is constructed:

$$\begin{aligned} f(H, X, E, Y_1, Y_2) = & \sum_{j=1}^n w_{H,j} \cdot \sigma_j + \lambda \|W_E \odot E\|_1 + \eta \|X\|_{TV} \\ & + \langle Y_1, M - H - E \rangle + \langle Y_2, X - H \rangle \\ & + \frac{\mu}{2} (\|M - H - E\|_F^2 + \|X - H\|_F^2) \\ & \text{s.t. } X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}, \end{aligned} \quad (13)$$

where $\langle \cdot, \cdot \rangle$ is the inner product of two matrices. $\{\sigma_j\}$ are singular values of matrix H . $W_H = \{w_{H,j}\}$ and $W_E \in R^{m \times n}$ are weights for $\{\sigma_j\}$ and entries of E , respectively. To simplify the objective function, we consider W_H and W_E to be fixed.

To solve the problem, we use the iterative alternating direction method, which optimizes one variable while fixing the remaining optimization variables in an iterative manner. In this way, the original complicated multi-variable optimization problem can be reduced to a single-variable optimization problem, whose solution can be obtained analytically. In our problem, there are five major sets of variables (H, X, E, Y_1, Y_2) . In the following, we will explain how each of these four variables can be optimized.

A. Optimizing H

If we fix variables (X, E, Y_1, Y_2) , then H can be optimized by minimizing $f(H, X, E, Y_1, Y_2)$ with respect to H . Specifically,

$$\begin{aligned} & \arg \min_H f(H, X, E, Y_1, Y_2) \\ & = \arg \min_H \sum_{j=1}^n w_{H,j} \cdot \sigma_j + \lambda \|W_E \odot E\|_1 + \eta \|X\|_{TV} \\ & \quad + \langle Y_1, M - H - E \rangle + \langle Y_2, X - H \rangle \\ & \quad + \frac{\mu}{2} (\|M - H - E\|_F^2 + \|X - H\|_F^2) \\ & = \arg \min_H \sum_{j=1}^n w_{H,j} \cdot \sigma_j \\ & \quad + \langle Y_1, M - H - E \rangle + \langle Y_2, X - H \rangle \\ & \quad + \frac{\mu}{2} (\|M - H - E\|_F^2 + \|X - H\|_F^2) \\ & = \arg \min_H \sum_{j=1}^n w_{H,j} \cdot \sigma_j + \mu \|H - L\|_F^2, \end{aligned} \quad (14)$$

where $L = \frac{1}{2}(M + X - E + Y_1/\mu + Y_2/\mu)$. This minimization problem (14) is similar to (10), which can be solved by non-uniform singular value thresholding (NSVT) defined by operator 2 as follows:

$$H = \mathcal{D}_{(2\mu)^{-1} W_H}(L). \quad (15)$$

B. Optimizing X

If we fix variables (H, E, Y_1, Y_2) , then X can be optimized by minimizing $f(H, X, E, Y_1, Y_2)$ with respect to X . Specifically,

$$\begin{aligned} & \arg \min_X f(H, X, E, Y_1, Y_2) \\ & = \arg \min_X \sum_{j=1}^n w_{H,j} \cdot \sigma_j + \lambda \|W_E \odot E\|_1 + \eta \|X\|_{TV} \\ & \quad + \langle Y_1, M - H - E \rangle + \langle Y_2, X - H \rangle \\ & \quad + \frac{\mu}{2} (\|M - H - E\|_F^2 + \|X - H\|_F^2) \\ & = \arg \min_X \eta \|X\|_{TV} + \langle Y_2, X - H \rangle \\ & \quad + \frac{\mu}{2} \|X - H\|_F^2 \\ & = \arg \min_X \eta \|X\|_{TV} + \frac{\mu}{2} \|X - (H - Y_2/\mu)\|_F^2 \\ & = \arg \min_X \eta \|X\|_{TV} + \frac{\mu}{2} \|X - R\|_F^2. \end{aligned} \quad (16)$$

where $R = (H - Y_2/\mu)$, $X \in B_{l,u} \equiv \{x_{i,j}, l \leq x_{i,j} \leq u\}$.

To let our algorithm be easily understood, we show how to solve this problem with a theorem. First, we introduce some notations as follows, which will be useful in our theorem.

- Let \mathcal{P} denote the set of the matrix pairs (p, q) , where $p \in R^{(m-1) \times n}$, $q \in R^{m \times (n-1)}$; they satisfy the following:

$$\begin{aligned} p_{i,j}^2 + q_{i,j}^2 & \leq 1, \quad \text{for } 1 \leq i \leq m-1, \quad 1 \leq j \leq n-1, \\ |p_{i,n}| & \leq 1, \quad \text{for } 1 \leq i \leq m-1, \\ |q_{m,j}| & \leq 1, \quad \text{for } 1 \leq j \leq n-1. \end{aligned} \quad (17)$$

- Linear operator \mathcal{L} , which maps an element (p, q) of \mathcal{P} to an m -by- n matrix, is defined as follows:

$$\begin{aligned} (\mathcal{L}(p, q))_{i,j} &= p_{i,j} - p_{i-1,j} + q_{i,j} - q_{i,j-1}, \\ \text{for } 1 \leq i \leq m, \quad 1 \leq j \leq n, \end{aligned} \quad (18)$$

where $p_{0,j} = p_{m,j} = q_{i,0} = q_{i,n} = 0$ for $i = 1, \dots, m$ and $j = 1, \dots, n$.

- \mathcal{L}^T , which denotes the adjoint of operator \mathcal{L} , is a linear map from an m -by- n matrix X to a matrix pair (p, q) of \mathcal{P} , defined by the following:

$$\begin{aligned} \mathcal{L}^T(X) &= (p, q), \\ p_{i,j} &= x_{i,j} - x_{i+1,j}, \quad \text{for } 1 \leq i \leq m-1, \quad 1 \leq j \leq n, \\ q_{i,j} &= x_{i,j} - x_{i,j+1}, \quad \text{for } 1 \leq i \leq m, \quad 1 \leq j \leq n-1. \end{aligned} \quad (19)$$

- P_C denotes the orthogonal projection operator onto set C . If C is replaced by set $B_{l,u}$ in (16), then we have the following:

$$(P_{B_{l,u}}(X))_{ij} = \begin{cases} l, & \text{if } x_{i,j} < l, \\ x_{i,j}, & \text{if } l \leq x_{i,j} \leq u, \\ u, & \text{if } x_{i,j} > u. \end{cases}$$

With these notations, we have the following theorem, which will be used to solve the problem in (16).

Theorem 1: For the optimization problem (16), if μ and η are given, then ρ can be calculated as η/μ . Then, the solution of problem (16) is given by the following:

$$X = P_{B_{l,u}}(R - \rho\mathcal{L}(p, q)), \quad (20)$$

where $R = (S - Y_2/\mu)$, \mathcal{P} , \mathcal{L} and P are the set, linear operator and orthogonal projection operator, respectively, which are defined above. (p, q) is the element of \mathcal{P} .

Proof: First, inspired by [38], we have the following fact:

$$\begin{aligned} \sqrt{x^2 + y^2} &= \max_{p,q} \{px + qy : p^2 + q^2 \leq 1\} \\ |x| &= \max_p \{|px| : |p| \leq 1\} \end{aligned} \quad (21)$$

Thus, the isotropic TV norm introduced in Section II can be rewritten as follows:

$$\|X\|_{TV} = \max_{(p,q) \in \mathcal{P}} T(X, p, q), \quad (22)$$

where

$$\begin{aligned} T(X, p, q) &= \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} [p_{i,j}(x_{i,j} - x_{i+1,j}) + q_{i,j}(x_{i,j} - x_{i,j+1})] \\ &\quad + \sum_{i=1}^{m-1} p_{i,n}(x_{i,n} - x_{i+1,n}) + \sum_{j=1}^{n-1} q_{m,j}(x_{m,j} - x_{m,j+1}) \end{aligned}$$

Then, we have

$$T(X, p, q) = Tr(\mathcal{L}(p, q)^T X). \quad (23)$$

With this, problem (16) can be reformulated as

$$\min_{X \in B_{l,u}} \max_{(p,q) \in \mathcal{P}} \frac{\mu}{2} \|X - R\|_F^2 + \eta Tr(\mathcal{L}(p, q)^T X). \quad (24)$$

As discussed in [38], the objective function in (24) is convex in X and concave in (p, q) . Therefore, it allows us to switch

the order of minimization and maximization. For any given η and μ , a constant ρ can be calculated as $\rho = \eta/\mu$; then, the problem can be converted into the following:

$$\max_{(p,q) \in \mathcal{P}} \min_{X \in B_{l,u}} \frac{1}{2} \|X - R\|_F^2 + \rho Tr(\mathcal{L}(p, q)^T X), \quad (25)$$

which can be rewritten as

$$\begin{aligned} \max_{(p,q) \in \mathcal{P}} \min_{X \in B_{l,u}} & \left\{ \frac{\mu}{2} (\|X - (R - \rho\mathcal{L}(p, q))\|_F^2 \right. \\ & \left. - \|R - \rho\mathcal{L}(p, q)\|_F^2 + \|R\|_F^2) \right\}. \end{aligned} \quad (26)$$

As illustrated in [38], the minimization problem for X in (26) has the solution $X = P_{B_{l,u}}(R - \rho\mathcal{L}(p, q))$, which can be computed using the fast gradient projection (FGP) algorithm [38]. This completes the proof. \square

C. Optimizing E

Next, we discuss how to optimize E while the other variables (H, X, Y_1, Y_2) are fixed. This can be achieved by minimizing $f(H, X, E, Y_1, Y_2)$ with respect to E . Specifically,

$$\begin{aligned} \arg \min_E f(H, X, E, Y_1, Y_2) &= \arg \min_E \sum_{j=1}^n w_{H,j} \cdot \sigma_j \\ &\quad + \lambda \|W_E \odot E\|_1 + \eta \|X\|_{TV} \\ &\quad + \langle Y_1, M - H - E \rangle + \langle Y_2, X - H \rangle \\ &\quad + \frac{\mu}{2} (\|M - H - E\|_F^2 + \|X - H\|_F^2) \\ &= \arg \min_E \lambda \|W_E \odot E\|_1 + \langle Y_1, M - H - E \rangle \\ &\quad + \frac{\mu}{2} \|M - H - E\|_F^2 \\ &= \arg \min_E \lambda \|W_E \odot E\|_1 \\ &\quad + \frac{\mu}{2} \|E - (M - H + Y_1/\mu)\|_F^2. \end{aligned} \quad (27)$$

This minimization problem is similar to (9), which can be solved using the non-uniform soft thresholding (NST) defined by **operator 1** as follows:

$$E = \mathcal{S}_{\lambda\mu^{-1}W_E}[M - H + Y_1/\mu]. \quad (28)$$

D. Optimizing Y_1 and Y_2

Y_1 and Y_2 are the Lagrange multiplier matrices of the original optimization problem. They should be updated after the other variables. If Y_1 is unknown and the other variables are fixed, then Y_1 can be updated as follows:

$$Y_1 = Y_1 + \mu(M - H - E). \quad (29)$$

If Y_2 is unknown and the other variables are fixed, then Y_2 can be updated as follows:

$$Y_2 = Y_2 + \mu(X - H). \quad (30)$$

V. ALGORITHM FOR SMOOTHED REWEIGHTED LOW-RANK MATRIX RECOVERY

In this section, we summarize the proposed algorithm for smoothed reweighted low-rank matrix recovery, and then we analyze its computational complexity.

Algorithm 1 Smoothed and Reweighted Low-Rank Matrix Recovery (SRLRMR)

Require: Data matrix $M \in R^{m \times n}$ (assuming $m \geq n$).
Ensure: Initialize $W_X = (w_{X,j}) \in R^n$, $W_E = (w_{E,ij}) \in R^{m \times n}$, $\varepsilon = 10^{-7}$, $k = 0$ and $maxiter$.
Compute $X^{(0)} = U\Sigma V^T$ and $E^{(0)}$ using IALM [39]. Set $w_{X,j}^{(0)} = \frac{1}{diag(\Sigma) + \epsilon_X}$, $w_{E,ij}^{(0)} = \frac{1}{|e_{ij}^{(k)}| + \epsilon_E}$.
while $\|M - X - E\|_F / \|M\|_F > \varepsilon$ and $k < maxiter$ **do**
 step 1: Using Algorithm 2 with weights $W_X^{(k)}$ and $W_E^{(k)}$ to compute $X^{(k)} \leftarrow X^*$ and $E^{(k)} \leftarrow E^*$;
 step 2: Update weights: The weights for each $i = 1, \dots, m$ and $j = 1, \dots, n$ are updated by
 $w_{X,j}^{(k+1)} = \frac{1}{\sigma_j^{(k)} + \epsilon_X}$, $w_{E,ij}^{(k+1)} = \frac{1}{|e_{ij}^{(k)}| + \epsilon_E}$.
 where ϵ_X and ϵ_E are predetermined positive constants, and the singular value matrix
 $\Sigma^{(k)} = diag([\sigma_1^{(k)}, \dots, \sigma_n^{(k)}]) \in R^{n \times n}$
 with $[U^{(k)}, \Sigma^{(k)}, V^{(k)}] = svd(X^{(k)})$.
end while
Output $X^{(k)}$, $E^{(k)}$.

A. Summary of Algorithm

Note that W_X and W_E are assumed to be fixed in the above solution for our proposed model. However, to obtain better performance, W_X and W_E should be adaptive. Thus, to make the structure of our proposed algorithm clear, it is presented as two algorithm modules. The first is the main body, which is used to describe how W_X and W_E are updated. The second is the algorithm module, which is utilized to obtain the solution for any given W_X and W_E . Now, the question is how to initialize and update the weights W_X and W_E such that the performance of our model can be optimized.

In [33], W_X is chosen to be the inverse of the magnitudes of singular values. W_E is set to be the inverse of the signal magnitude for reweighted l_1 norm minimization. In each iteration, the weights W_X and W_E are updated based on the new values of X and E .

Finally, the main body of our proposed algorithm, referred to as smoothed and reweighted low-rank matrix recovery (SRLRMR), is summarized in Algorithm 1. In addition, in each iteration of the SRLRMR Algorithm 1, with given weighted matrices W_X and W_E , the algorithm for solving the inner optimization problem of smoothed and reweighted low-rank matrix recovery is summarized in Algorithm 2.

B. Algorithm Complexity Analysis

We analyze the complexity of our proposed algorithm. Suppose that $M \in R^{m \times n}$ is the observed matrix. In our algorithm, the major computation is the singular value decomposition of matrix M and matrix addition operations. The complexity of SVD for matrix M is $O(mn^2)$ [42]. The complexity of matrix addition is $O(mn)$. Thus, the total complexity of our method is $O(t(mn^2 + mn))$, where t is the number of iterations. The major computation in the non-uniform singular value thresholding (NSVT) method [33] is also singular value decomposition of the matrix and matrix addition operations. The complexity of NSVT is also $O(t(mn^2 + mn))$ [42]. Thus, in terms of algorithm complexity, our proposed algorithm has the same complexity as NSVT.

Algorithm 2 IALM Algorithm for Solving the Problem of Smoothed and Reweighted Low-Rank Matrix Recovery

Require: Data matrix $M \in R^{m \times n}$, λ , η and δ .
Ensure: Initialize $X_0 \in R^{m \times n}$, $E_0 \in R^{m \times n}$, $H_0 \in R^{m \times n}$, $Y_{1,0} \in R^{m \times n}$, $Y_{2,0} \in R^{m \times n}$, $\mu_0 > 0$, $\xi = 10^{-7}$, $t = 0$ and $inneriter = 100$.
while $\|M - H - E\|_F / \|M\|_F > \xi$, $\|M - H - E\|_F / \|M\|_F > \xi$ and $t < inneriter$ **do**
 step 1: Let $L_{t+1} = M + X_t - E_t + Y_{1,t} / \mu_t + Y_{2,t} / \mu_t$, $\rho = \eta / \mu_t$; then, $H_{t+1} = \mathcal{D}_{\mu_t^{-1} W_X}(L_{t+1})$;
 step 2: Let $R_{t+1} = H_{t+1} - Y_{2,t}$ using the FGP Algorithm [38] to compute $X_{t+1} = P_{B_{t,u}}(R_{t+1} - \rho \mathcal{L}(p, q))$;
 step 3: $E_{t+1} = \mathcal{S}_{\lambda \mu_t^{-1} W_E}[M - H_{t+1} + Y_{1,t} / \mu_t]$;
 step 4: $Y_{1,t+1} = Y_{1,t} + \mu_t(M - H_{t+1} - E_{t+1})$;
 step 5: $Y_{2,t+1} = Y_{2,t} + \mu_t(X_{t+1} - H_{t+1})$;
 step 6: $\mu_{t+1} = \delta \mu_t$, $t \leftarrow t + 1$;
end while
Output X^* , E^* .

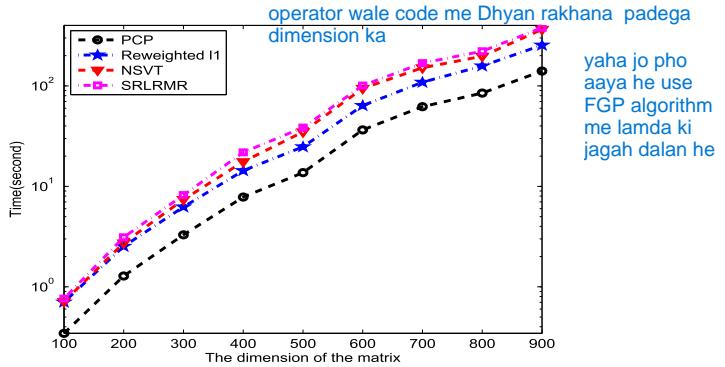


Fig. 3. The running time comparison of the PCP, Reweighted l_1 , NSVT and our proposed SRLRMR methods.

In general, our proposed method only has an extra addition operation when compared to the NSVT method. To verify this, we test our method on 9 images, as shown in Fig. 6. We resize these images from 100×100 to 900×900 . We compare the complexity of our algorithm against existing methods, including the principal component pursuit (PCP) algorithm [30], the reweighted l_1 algorithm [40] and the non-uniform singular value thresholding (NSVT) method [33]. As shown in Fig. 3, our running time is only slightly higher than that of NSVT.

VI. EXPERIMENTAL RESULTS

In this section, we conduct extensive experiments to evaluate the proposed SRLRMR algorithm and compare its performance with existing state-of-the-art methods, including the PCP algorithm [30], the reweighted l_1 algorithm [40] and the NSVT method [33]. We conduct performance evaluations on different image restoration scenarios, including single-image denoising, hyperspectral image denoising, and video background reconstruction from corrupted images. The noise used in our experiments is large sparse additive noise, which means that the ratio of the corrupted pixels number to the number of all pixels in the image is p and the values of these noisy pixels are all 255. The density of the noise ranges from 20% to 45%. Two objective image quality metrics, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [41], are used to evaluate performance.

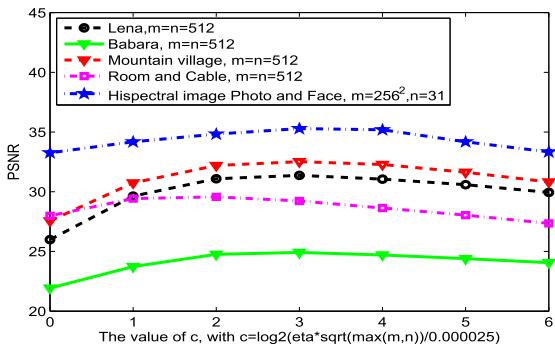


Fig. 4. The effect of η changes on the results. η also denoted as 'eta' in this figure. When 'c' changes from 0 to 6, η changes from $0.000025/\sqrt{\max(m,n)}$ to $0.0016/\sqrt{\max(m,n)}$.

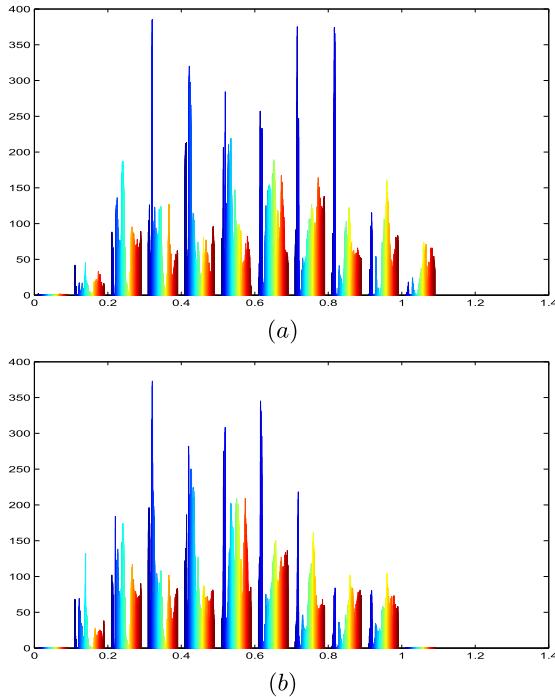


Fig. 5. The gray histograms of the recovered Lena images. The input pixel value is normalized to $[0,1]$. The noised image is that 40% pixels of the Lena image are corrupted by large sparse additive noise. (a) The result is obtained by our method without the bound constraint. (b) The result is obtained by our method with the bound constraint.

A. Experimental Settings

In all experiments, the parameters are set as follows. The size of the input matrix is $m \times n$. The max iteration of the fast gradient projection (FGP) algorithm [38] is set to be 2. In Algorithm 2, we set $X_0 = 0$, $E_0 = 0$, $H_0 = 0$, $Y_{1,0} = M / \max(\|M\|, \lambda^{-1} \|M\|_\infty)$, and $Y_{2,0} = 0$, where $\|M\|$ is the spectral norm of matrix M and $\|M\|_\infty$ is the maximum absolute value of the entries in matrix M . In addition, $\delta = 1.5$. The max iterations $inneriter$ in Algorithm 2 are set to be 100. In Algorithm 1, the constants $\epsilon_X = 0.01$ and $\epsilon_E = 0.01$ are set the as same as in [33]. The bounds of the constraint are set to be $[l, u] = [0, 255]$. The parameters of PCP, reweighted l_1 and NSVT are set as their default values.

In (11), λ and η represent the trade-off between those three components in the objective function. Specifically, they

control the sensitivity level of the model to sparse errors, TV norm, and rank of the approximation matrix. How to determine their values becomes an important task. Following the practice in [21] and [33], we set the weighting parameter λ to be $1/\sqrt{\max(m, n)}$. When η increases, the impact of TV regularization becomes more significant. When η is set to 0, our model reduces to the re-weighted low-rank matrix recovery model.

When λ is fixed, we choose the value of η from $\{(2^c)\tau, c = 0, 1, \dots, 6\}$, with $\tau = 0.000025/\sqrt{\max(m, n)}$. We test these values on the first 4 single images of Fig. 6 and hyperspectral image *Photo and Face*. The results are shown in Fig. 4. (In Fig. 4, η is denoted as 'eta'.) The images of mountain village and of room and cable are the third and fourth images in Fig. 6, respectively. As shown in Fig. 4, our proposed method obtains better image restoration when $c = 3$ and $\eta = (2^c)\tau$. Thus, we set $\eta = 0.0002/\sqrt{\max(m, n)}$. Finally, in Algorithm 2, ρ can be computed by $\rho = \eta/\mu$, where μ is the parameter of the problem (16). Simultaneously, μ corresponds to parameter μ_t in Algorithm 2.

All algorithms are implemented using MATLAB running on a desktop computer with an Intel core i5-2400 3.0 GHz CPU, 4 GB of RAM, and Windows 7.

B. Single Image Restoration

First, we evaluate the performance on single-image restoration. The original image size is 512×512 . During our experiments, we find that the singular values of natural images decrease very fast, which implies that the image has a low-rank structure. The max iteration number $maxiter$ is set to be 1, 2 or 3 in Algorithm 1. In addition, for different noise densities, such as 20%, 30% and 40%, μ is set as $\mu_0 = 0.25/\|M\|$, $\mu_0 = 1.25/\|M\|$ and $\mu_0 = 2.5/\|M\|$, respectively.

To evaluate the effect of the bound constraint, the experiments with and without bound constraint are performed. The relative error (RE) is used to evaluate the performances, which is defined as $RE = \|\hat{X} - X\|_F/\|X\|_F$, where $\|\bullet\|_F$ is the Frobenius norm of matrix. As shown in Fig. 5, some pixels are out of the range $[0,1]$ without bound constraint, and pixels are always within the range $[0,1]$ when the bound constraint is added. Furthermore, we present the recovery results of the 9 images in Fig. 6 by our method with and without bound constraint in TABLE I. As shown in TABLE I, our bound constraint may not have a significant effect on the quality of the restored image because the pixels exceeding the bound are only a small portion of all pixels in the image, these pixels will not affect the quantitative quotas such as the Relative Error or PSNR significantly. However, it does have a certain effect on the output.

In Figs. 8 to 12, the input images are corrupted by large random sparse noise. The noise density is 20%, 30% or 40%. The $maxiter$ is set to be 1 for our proposed Algorithm 1. Additional experiments are performed on various natural images of Fig. 6.¹ The results are summarized in Tables II and III. As shown in Table II, when the density of the noise is set to 0.2 and 0.3, the proposed algorithm has significantly improved structural smoothness and image quality by an average of approximately 0.72 dB and 4.21 dB, respectively.

¹These images are downloaded from <http://decsai.ugr.es/cvg/CG/base.htm>



Fig. 6. The 9 single images are used in our experiments. The images *No.* is given from left to right as 1 to 9.

TABLE I

RECOVERY RESULTS COMPARISON OF THE MODEL WITH AND WITHOUT BOUND CONSTRAINT FOR THE 9 IMAGES IN FIG.6.
THE IMAGES *No.* IS USED TO DENOTE THE IMAGES IN FIG.6. 40% PIXELS OF THE IMAGE ARE CORRUPTED BY LARGE SPARSE NOISE. TRAILS ARE REPEATED 10 TIMES AND THE AVERAGE PSNR VALUES ARE LISTED IN THIS TABLE

with bound or not	Relative Error(RE)								
	No.1	No.2	No.3	No.4	No.5	No.6	No.7	No.8	No.9
with bound	0.2183	0.2931	0.2011	0.2617	0.3239	0.1879	0.2255	0.3169	0.2254
without bound	0.2190	0.2943	0.2015	0.2622	0.3247	0.1879	0.2262	0.3180	0.2266

TABLE II

PSNR COMPARISON OF THE PCP, REWEIGHTED l_1 NORM MINIMIZATION, NSVT AND SRLRMR FOR THE 9 IMAGES IN FIG.6.
THE IMAGES *No.* IS USED TO DENOTE THE IMAGES IN FIG.6. THE *Maxiter* OF OUR ALGORITHM IS SET AS 1.
TRAILS ARE REPEATED 10 TIMES AND THE AVERAGE PSNR VALUES ARE LISTED IN THE TABLE

<i>p</i> value	Images <i>No.</i>	PSNR(dB)				Δ PSNR
		PCP	Reweighted l_1	NSVT	SRLRMR	
<i>p</i> = 0.2	1	26.75	23.84	32.15	32.47	+0.32
	2	24.23	22.33	26.28	26.36	+0.08
	3	28.41	26.75	31.99	33.71	+1.72
	4	25.90	22.79	28.10	29.64	+1.54
	5	29.77	26.43	34.63	34.88	+0.25
	6	29.98	27.96	34.01	33.73	-0.28
	7	25.71	23.49	28.78	29.31	+0.53
	8	21.72	20.43	22.12	23.71	+1.59
	9	25.22	23.01	27.65	28.35	+0.70
	Average	26.41	24.11	29.52	30.24	+0.72
<i>p</i> = 0.3	1	21.27	22.17	23.84	30.67	+6.83
	2	19.10	20.85	20.77	24.35	+3.58
	3	23.00	25.42	25.56	31.15	+5.59
	4	22.52	23.04	26.61	27.36	+0.75
	5	25.20	26.69	28.73	33.31	+4.58
	6	27.29	26.57	30.57	33.01	+2.44
	7	20.92	22.01	22.93	27.60	+4.67
	8	18.09	19.21	18.12	21.90	+3.78
	9	21.25	21.62	22.01	27.66	+5.65
	Average	22.07	23.06	24.35	28.56	+4.21
<i>p</i> = 0.4	1	13.18	16.53	13.57	20.53	+6.96
	2	12.25	15.10	12.50	17.88	+5.38
	3	13.04	19.39	13.80	21.66	+7.86
	4	12.32	19.61	15.69	22.58	+6.89
	5	11.39	18.65	13.21	20.04	+6.83
	6	13.94	24.49	18.04	28.12	+10.08
	7	13.28	17.24	13.50	20.55	+7.05
	8	12.65	15.00	12.06	17.07	+5.01
	9	13.82	17.03	13.84	19.64	+5.80
	Average	12.88	18.12	14.02	20.90	+6.88

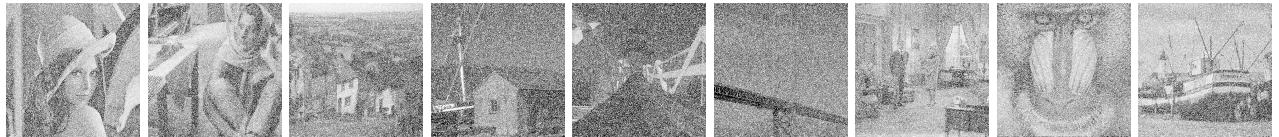


Fig. 7. The noisy images corresponding to the Fig.6. The density of the large sparse noise is *p* = 0.4.

For the Lena image, the quality has been improved by 6.83 dB with 30% of large and sparse noise. When the density of the noise is set to 0.4, which is shown in Fig. 7, although

our method has not obtained the expected results, it is considerably better than the other three methods. Note that for an objective and fair comparison with NSVT, the max

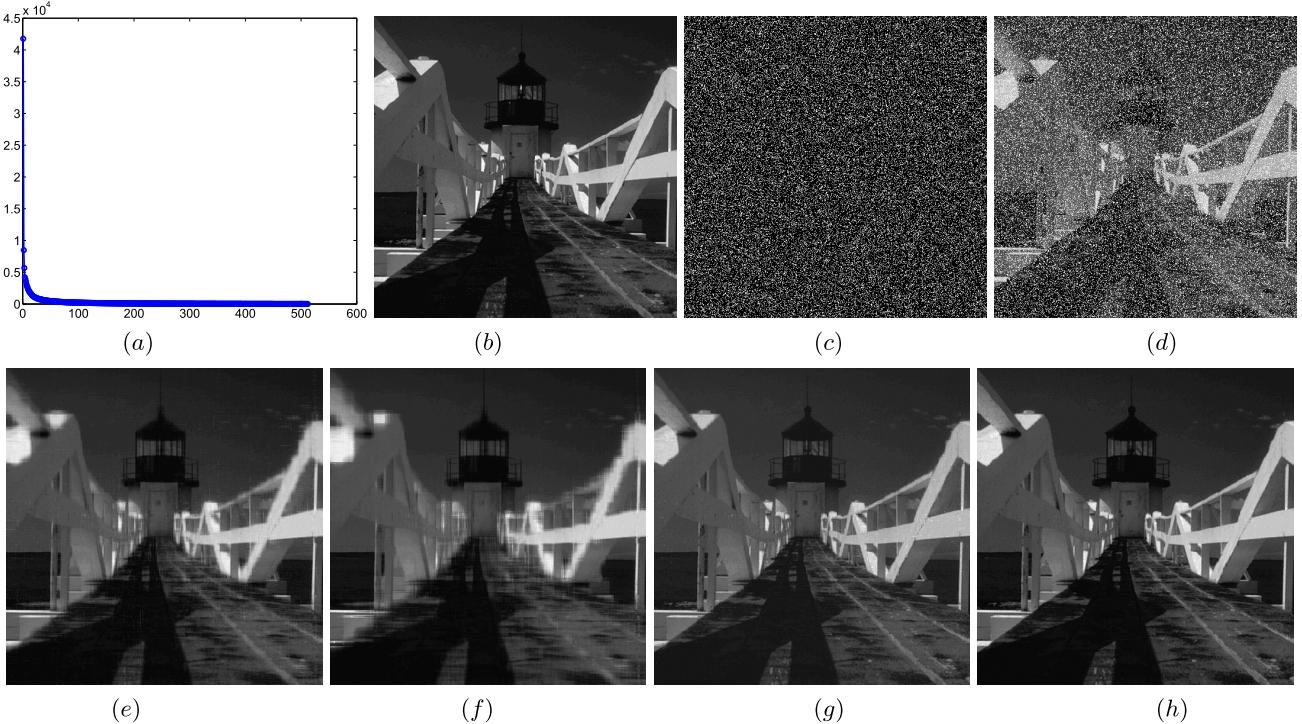


Fig. 8. Restoration results comparison of the single image Bridge of the PCP, Reweighted l_1 norm minimization, NSVT and SRLRMR. 20% pixels of the image are corrupted by large sparse noise. (a) Singular values distribution. (b) The original Bridge image, and the image size is 512×512 . (c) The large sparse noise. (d) The input corrupted image, PSNR = 9.99dB and SSIM = 0.167, which are abbreviated as (9.99dB/0.167). The rest of this paper is also abbreviated as this. (e) Restoration result by PCP (29.77dB/0.923). (f) Restoration result by Reweighted l_1 norm minimization (26.43dB/0.873). (g) Restoration result by NSVT (34.63dB/0.978). (h) Restoration result by SRLRMR (34.88dB/0.981) with *maxiter* = 1.

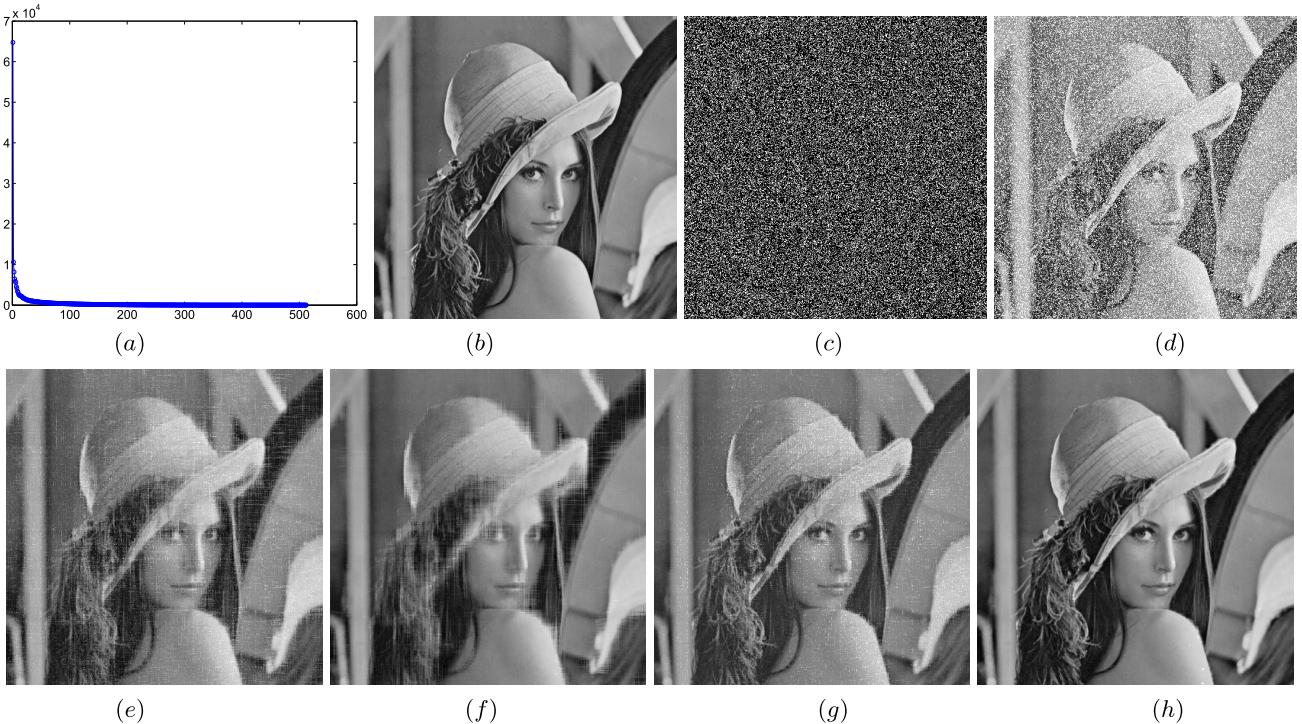


Fig. 9. Restoration results comparison of the single image Lena of the PCP, Reweighted l_1 , NSVT and SRLRMR. 30% image pixels are corrupted by large sparse noise. (a) Singular value distribution. (b) The original Boat image, and the image size is 512×512 . (c) The large sparse noise. (d) The input corrupted image(11.47dB/0.251). (e) Restoration result by PCP (21.27dB/0.695). (f) Restoration result by Reweighted l_1 norm minimization (22.17dB/0.804). (g) Restoration result by NSVT (23.84dB/0.778). (h) Restoration result by SRLRMR (30.67dB/0.957) with *maxiter* = 1.

iteration number *maxiter* is set to be 1 in Algorithm 1. In fact, the proposed algorithm has better performance with the increase of the *maxiter* in Algorithm 1. For example,

additional experimental results are summarized in Table IV and Fig. 10 when *maxiter* is set as 1, 2 and 3. Finally, stripe and large sparse noise patterns are tested in Fig. 13. The

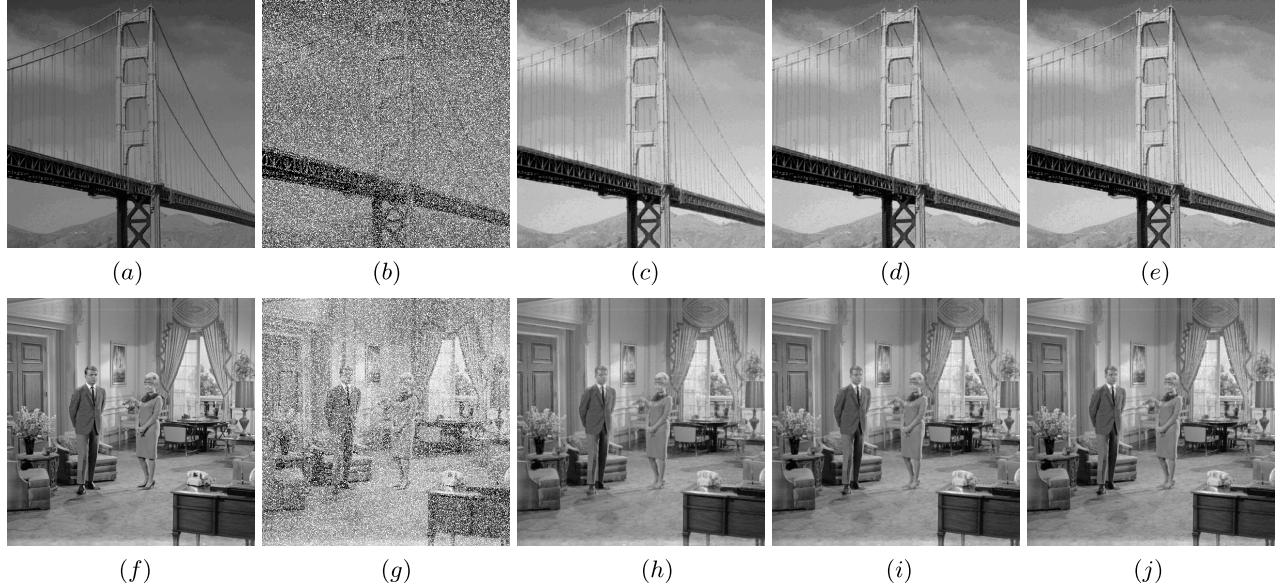


Fig. 10. Restoration results comparison of the single image No.6 and 7 of our proposed SRLRMR with different *maxiter* in Algorithm 1. 30% image pixels are corrupted by large sparse noise. (a)The original image No.6. (b) The input corrupted image. (c) Restoration result by SRLRMR (33.01dB/0.948) with *maxiter* = 1. (d) Restoration result by SRLRMR (33.77dB/0.954) with *maxiter* = 2. (e) Restoration result by SRLRMR (34.09dB/0.957) with *maxiter* = 3. (f) The original image No.7. (g) The input corrupted image. (h) Restoration result by SRLRMR (27.60dB/0.935) with *maxiter* = 1. (i) Restoration result by SRLRMR (28.21dB/0.944) with *maxiter* = 2. (j) Restoration result by SRLRMR (28.38dB/0.947) with *maxiter* = 3.

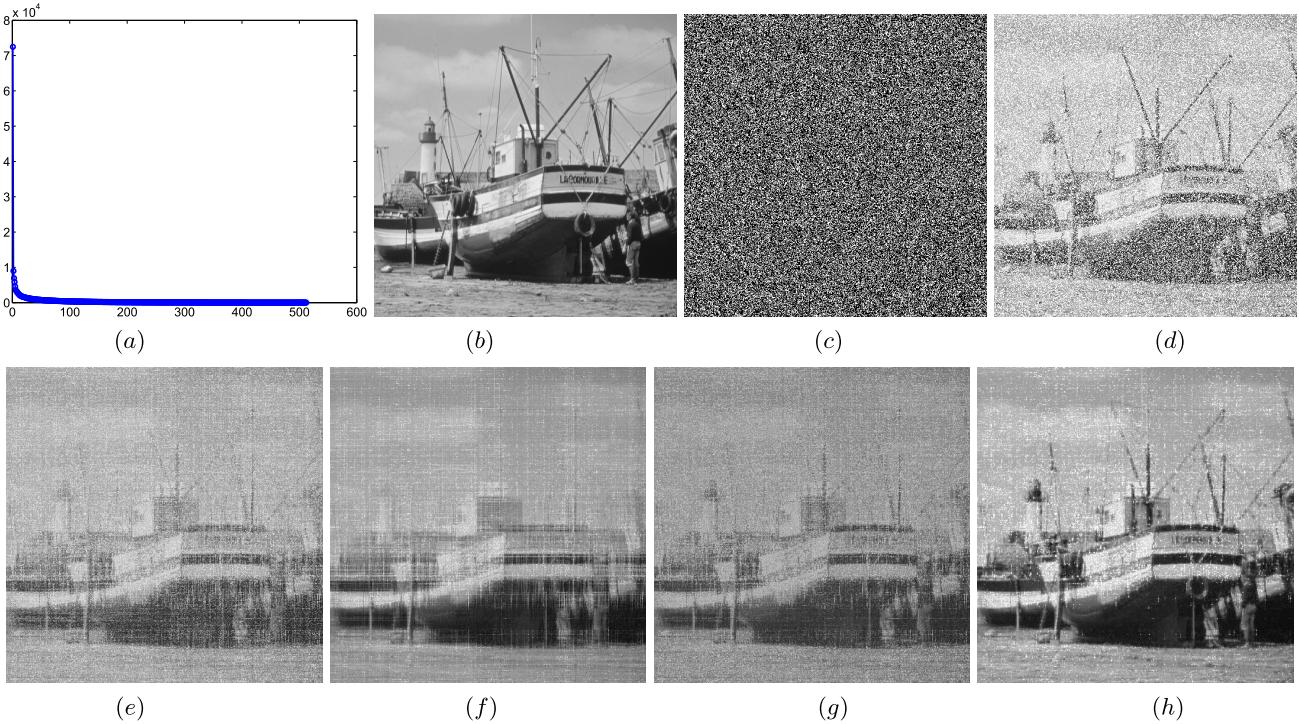


Fig. 11. Restoration results comparison of the single image Boat of the PCP, Reweighted l_1 , NSVT and SRLRMR. 40% image pixels are corrupted by large sparse noise. (a) Singular value distribution. (b) The original Boat image, and the image size is 512×512 . (c) The large sparse noise. (d) The input corrupted image(9.86dB/0.274). (e) Restoration result by PCP (13.82dB/0.343). (f) Restoration result by Reweighted l_1 norm minimization (17.03dB/0.452). (g) Restoration result by NSVT (13.84dB/0.334). (h) Restoration result by SRLRMR (19.64dB/0.600) with *maxiter* = 1.

maxiter is set as 3 in this experiment. As shown in Fig. 13, the proposed SRLRMR method achieves the best performance among these four methods. The PCP algorithm fails to restore most of those stripes. Meanwhile, the rank of the restoration result of the reweighted l_1 method is too low; thus, most of

the details are lost. The NSVT method also fails to restore some stripes. Note that the restoration performances are different for different test images. This result occurs because each image has its own special structure. However, our proposed method consistently outperforms the other methods.

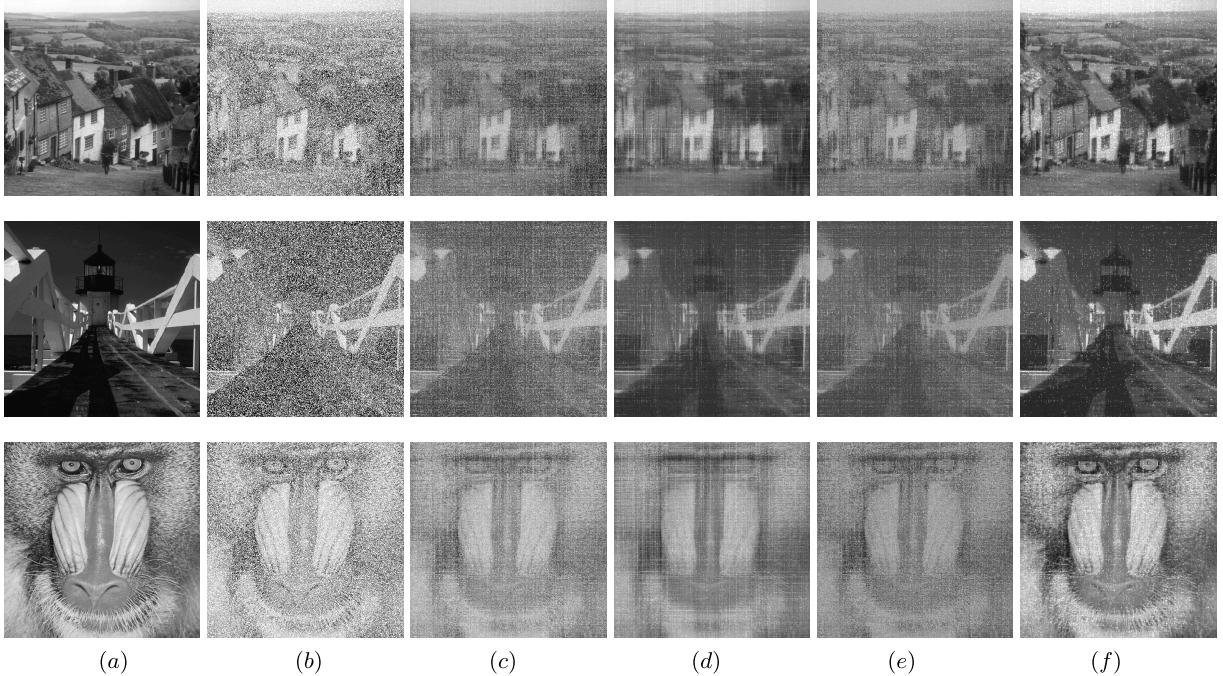


Fig. 12. Restoration results comparison of the image of the PCP, Reweighted l_1 , NSVT and SRLRMR. 40% image pixels are corrupted by large sparse noise. (a) The original images. (b) The corrupted images with large sparse noise. (c) Restored by PCP . (d) Restored by Reweighted l_1 norm minimization. (e) Restored by NSVT. (f) Restored by SRLRMR.

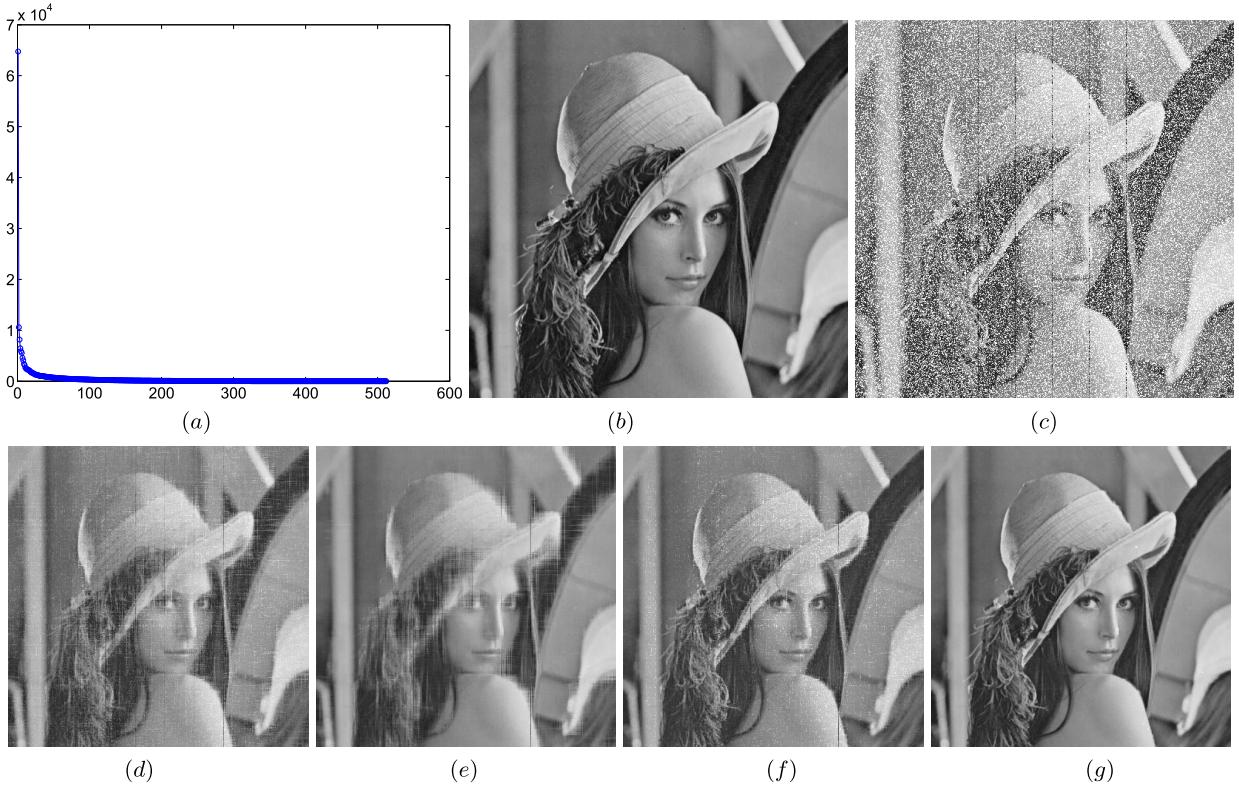


Fig. 13. Restoration results comparison of the Lena image of the PCP, Reweighted l_1 , NSVT and SRLRMR. 30% pixels of each image are corrupted by large sparse noise. In addition, it is also corrupted by stripes. (a) Singular value distribution. (b) The original Lena image with size 512×512 . (c) The corrupted image(10.47dB/0.250). (d) Restored result by PCP (21.23dB/0.678). (e) Restored result by Reweighted l_1 norm minimization (22.17dB/0.800). (f) Restored result by NSVT (23.68dB/0.774). (g) Restored result by SRLRMR (32.12dB/0.970) with $\text{maxiter} = 3$.

C. Hyperspectral Image Restoration

Next, we evaluate the performance of our SRLRMR algorithm on the restoration of hyperspectral images.² There

²These hyperspectral images are downloaded from <http://www.cs.columbia.edu/CAVE/databases/multispectral>

are 31 frequency bands in total, and each original hyperspectral frequency band is downsampled to 256×256 . We reshape each frequency band of the hyperspectral image into a vector and arrange all these vectors into one large matrix. The matrix columns are linearly correlated. Thus, this large matrix has a

TABLE III

SSIM COMPARISON OF THE PCP, REWEIGHTED l_1 NORM MINIMIZATION, NSVT AND SRLRMR FOR THE 9 IMAGES IN FIG.6. THE IMAGES $No.$ IS USED TO DENOTE THE IMAGES IN FIG.6. THE *Maxiter* OF OUR ALGORITHM IS SET AS 1. TRAILS ARE REPEATED 10 TIMES AND THE AVERAGE SSIM VALUES ARE LISTED IN THE TABLE

p value	Images $No.$	SSIM				Δ SSIM
		PCP	Reweighted l_1	NSVT	SRLRMR	
$p = 0.2$	1	0.903	0.839	0.973	0.982	+0.009
	2	0.851	0.780	0.918	0.922	+0.004
	3	0.904	0.851	0.963	0.976	+0.013
	4	0.871	0.794	0.954	0.958	0.004
	5	0.923	0.873	0.978	0.981	+0.003
	6	0.897	0.851	0.957	0.955	-0.002
	7	0.892	0.816	0.956	0.962	+0.006
	8	0.788	0.663	0.845	0.888	+0.043
	9	0.875	0.797	0.935	0.945	+0.010
	Average	0.878	0.807	0.942	0.954	+0.012
$p = 0.3$	1	0.695	0.804	0.778	0.957	+0.179
	2	0.644	0.727	0.746	0.874	+0.128
	3	0.761	0.819	0.861	0.951	+0.090
	4	0.726	0.794	0.902	0.941	+0.039
	5	0.762	0.872	0.899	0.975	+0.076
	6	0.842	0.834	0.909	0.948	+0.039
	7	0.729	0.775	0.833	0.935	+0.102
	8	0.615	0.595	0.658	0.822	+0.164
	9	0.722	0.757	0.750	0.924	+0.174
	Average	0.722	0.775	0.815	0.930	+0.115
$p = 0.4$	1	0.309	0.447	0.311	0.613	+0.302
	2	0.321	0.390	0.320	0.572	+0.251
	3	0.324	0.549	0.347	0.711	+0.364
	4	0.219	0.590	0.370	0.753	+0.383
	5	0.173	0.422	0.213	0.520	+0.307
	6	0.205	0.735	0.376	0.845	+0.469
	7	0.333	0.484	0.343	0.692	+0.349
	8	0.325	0.343	0.292	0.546	+0.254
	9	0.343	0.452	0.334	0.600	+0.226
	Average	0.283	0.490	0.323	0.650	+0.327

TABLE IV

COMPARISON OF SRLRMR WHEN MAX ITERATION NUMBER *Maxiter* OF OUR ALGORITHM 1 RANGES FROM 1 TO 3 FOR THE 9 IMAGES IN FIG.6. THE IMAGES $No.$ IS USED TO DENOTE THE IMAGES IN FIG.6. TRAILS ARE REPEATED 10 TIMES AND THE AVERAGE PSNR AND SSIM VALUES ARE LISTED IN THE TABLE

p value	Images $No.$	PSNR(dB)			SSIM		
		<i>maxiter</i> = 1	<i>maxiter</i> = 2	<i>maxiter</i> = 3	<i>maxiter</i> = 1	<i>maxiter</i> = 2	<i>maxiter</i> = 3
$p = 0.2$	1	32.47	34.92	36.21	0.982	0.990	0.992
	2	26.27	26.00	25.82	0.922	0.920	0.918
	3	33.71	34.42	34.63	0.976	0.980	0.981
	4	29.64	30.75	31.16	0.958	0.969	0.973
	5	34.88	36.63	37.32	0.981	0.986	0.988
	6	33.73	35.23	36.02	0.955	0.969	0.975
	7	29.31	30.08	30.39	0.962	0.968	0.970
	8	23.71	23.68	23.63	0.888	0.892	0.893
	9	28.35	29.36	29.95	0.945	0.957	0.963
	Average	30.24	31.23	31.68	0.950	0.959	0.961
$p = 0.3$	1	30.67	31.88	32.28	0.957	0.970	0.972
	2	24.35	24.40	24.37	0.874	0.880	0.881
	3	31.15	31.55	31.58	0.951	0.955	0.955
	4	27.36	26.90	26.77	0.941	0.945	0.946
	5	33.31	33.52	33.60	0.975	0.978	0.978
	6	33.01	33.77	34.09	0.948	0.954	0.957
	7	27.60	28.21	28.38	0.935	0.944	0.947
	8	21.90	22.08	22.10	0.822	0.835	0.839
	9	27.66	28.39	28.63	0.924	0.936	0.939
	Average	28.56	28.97	29.09	0.930	0.933	0.935

low-rank structure. The max iteration number *maxiter* is set to be 1 in Algorithm 1. μ is set as $\mu_0 = 0.50/\|M\|$.

In Fig. 14, the input hyperspectral image is restored. Moreover, 40% of the pixels of the hyperspectral image are

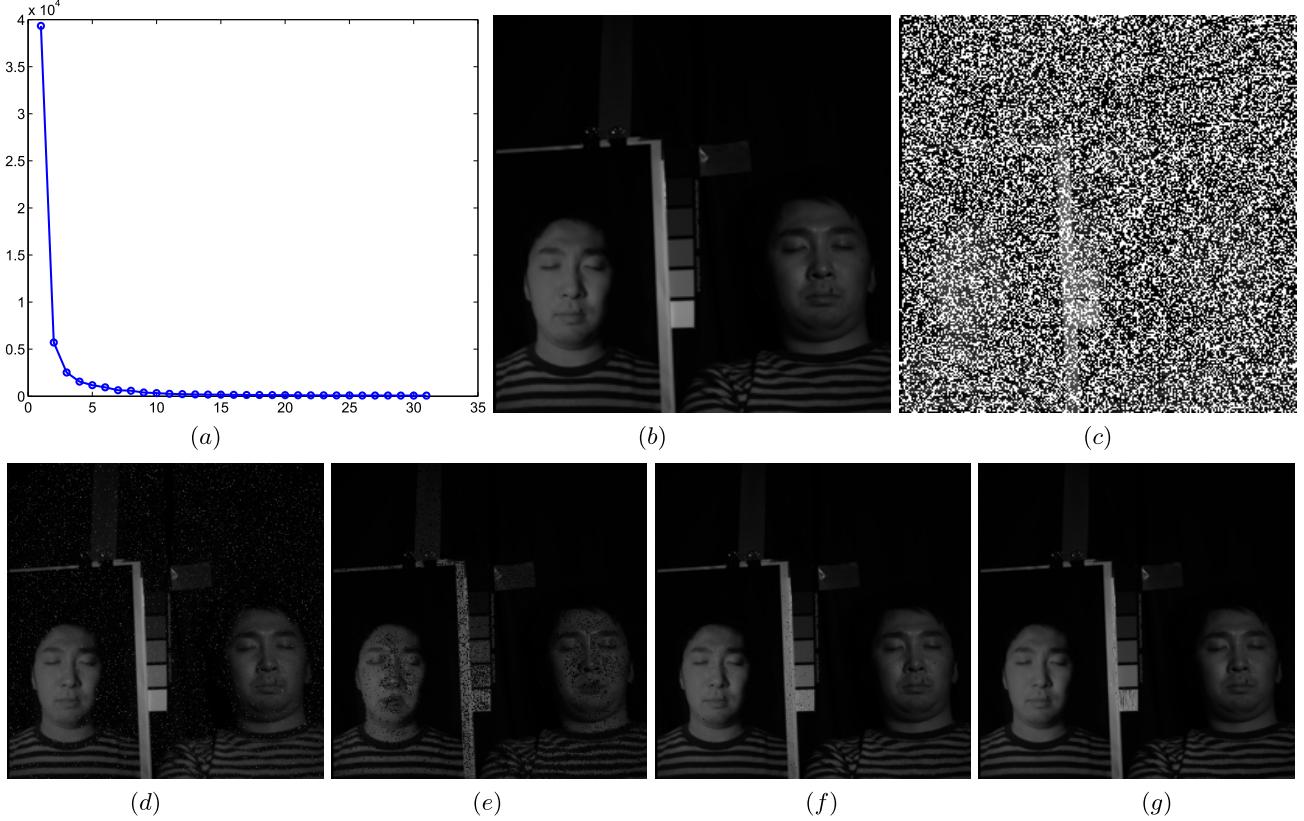


Fig. 14. Restoration results comparison of the hyperspectral image *Photo and Face* of the PCP, Reweighted l_1 , NSVT and SRLRMR. 40% pixels of each band are corrupted by large sparse noise. (a) Singular values distribution of the new large matrix, which is constructed by all bands of the hyperspectral image *Photo and Face*. (b) The 21th band of the original hyperspectral image, its size is 256×256 . (c) The corrupted band with large sparse noise. (d) Restoration result by PCP, the PSNR and SSIM of the 21th band is (29.11dB/0.558). (e) Restoration result by Reweighted l_1 norm minimization, PSNR and SSIM are (26.87dB/0.891). (f) Restoration result by NSVT, PSNR and SSIM are (36.99dB/0.982). (g) Restoration result by SRLRMR, PSNR and SSIM are (42.38dB/0.993).

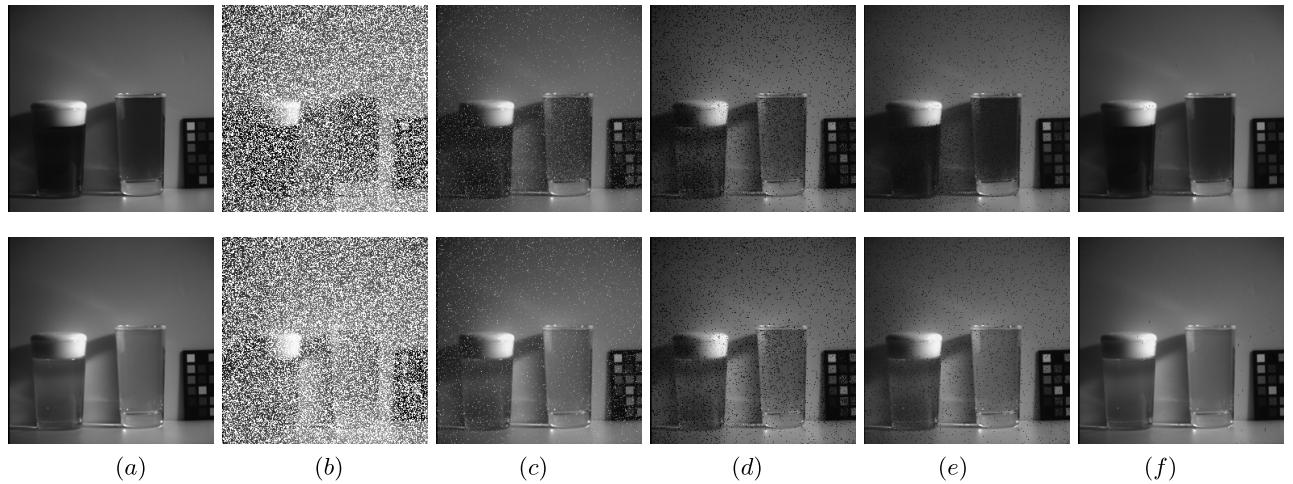


Fig. 15. Restoration results comparison of the hyperspectral image *beers* of the PCP, Reweighted l_1 , NSVT and SRLRMR. 40% pixels of each image are corrupted by large sparse noise. There are 31 bands in total. (a) The column shows two bands of the original image. The size of each band is 256×256 . (b) The corrupted bands with large sparse noise. (c) Restoration result by PCP, average PSNR and SSIM of total 31 frequency bands are (28.38dB/0.745). (d) Restoration result by Reweighted l_1 norm minimization, average PSNR and SSIM are (28.79dB/0.814). (e) Restoration result by NSVT, average PSNR and SSIM are (33.41dB/0.858). (f) Restoration result by SRLRMR, average PSNR and SSIM are (39.95dB/0.977). Here, bands 10 and 20 are shown in rows 1 and 2, respectively.

corrupted by large random sparse noise. It can be observed that the first few singular values are considerably larger than the others in Fig. 14(a). Thus, the new large matrix, constructed by

the all bands of the hyperspectral image *Photo and Face*, has a low-rank structure. Fig. 14 shows the restoration result of the single 21st band of the hyperspectral image. Another example

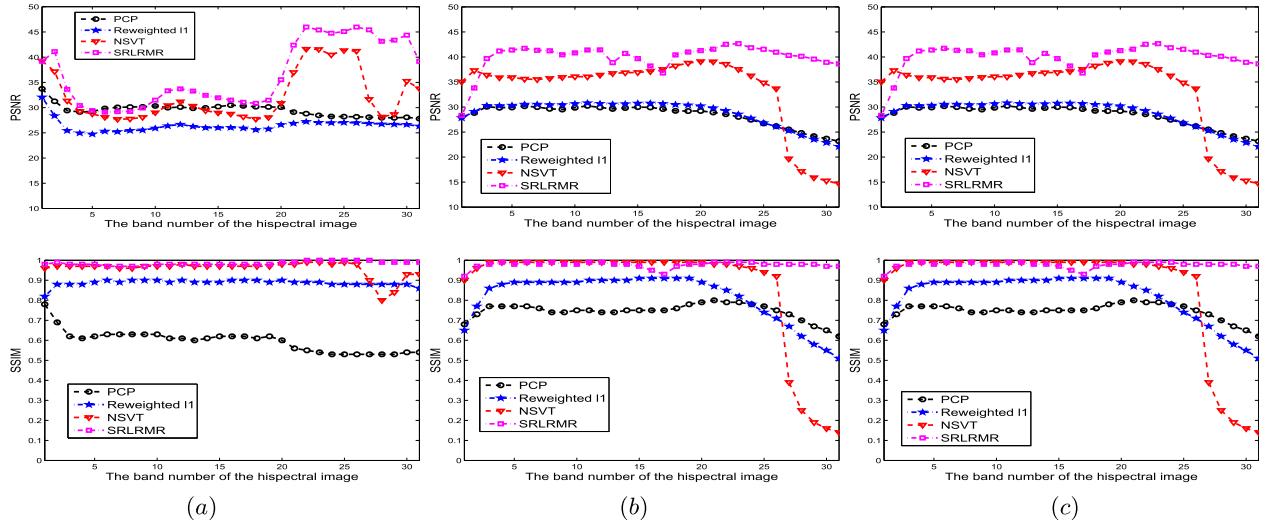


Fig. 16. Restoration results comparison of the hyperspectral image *Photo and Face*, *Clothe*, and *beers* of the PCP, Reweighted l_1 , NSVT, and SRLRMR. 40% pixels of the bands are corrupted by large sparse noise. There are 31 bands in total and the size of each frame is 256×256 . In this figure, the top row is the PSNR comparison, and the bottom row is the SSIM comparison. (a) The column of comparison for *Photo and Face* image. (b) The column of comparison for *Clothe* image. (c) The column of comparison for *beers* image.

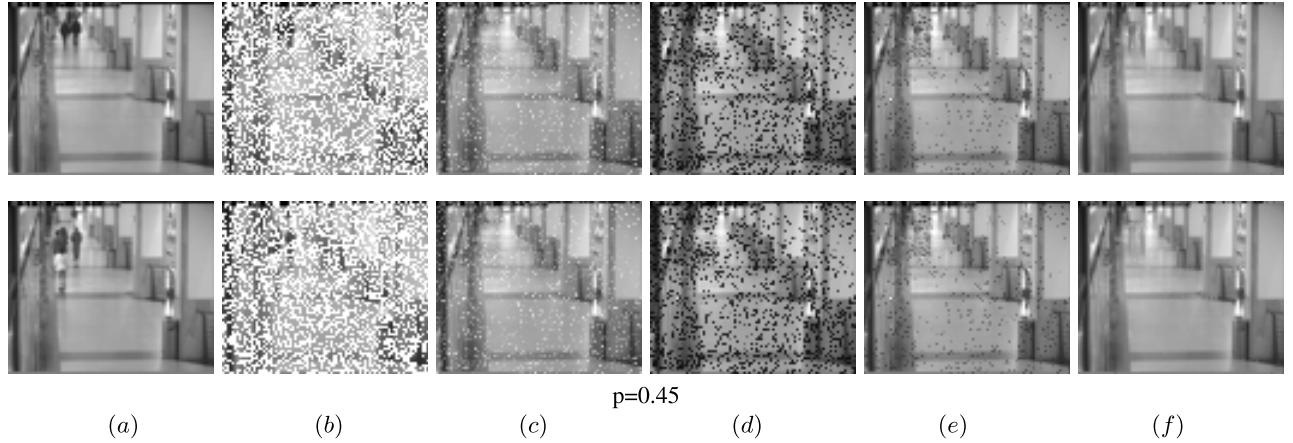


Fig. 17. Comparison of the video background modeling results of the PCP, Reweighted l_1 , NSVT and SRLRMR. 45% pixels of the frames are corrupted by large sparse noise. There are 118 frames in total and the size of each frame is 72×86 . Here, the frames 1 and 45 are shown in the rows 1 and 2, respectively. The noise density is $p = 0.45$. (a) Images in this column are the original video frames. (b) The corrupted frames with large sparse noise. (c) Restored by PCP. (d) Restored by Reweighted l_1 norm minimization. (e) Restored by NSVT. (f) Restored by SRLRMR.

is presented in Fig. 15, which shows the hyperspectral image of two beer cups.

The PSNR and SSIM are calculated on the large matrix, which is computed from all bands of the hyperspectral image *beers*. In other words, the PSNR and SSIM are the mean values of the total 31 bands in Fig. 15. To conduct the performance comparison in more detail, we show the restoration results of all frequency bands in Fig. 16. As shown in these figures, our proposed method has more stable performance than NSVT when the band ranges from 1 to 31. The performance of NSVT decreases significantly with increasing band number for the images *clothes* and *beers*. In particular, the results of our proposed approach are considerably better than those of the other three methods when the number of bands is larger than 20. The PCP and reweighted l_1 always have lower results than our proposed method. Finally, it can be concluded that our proposed method achieves the best performance among these four methods.

D. Video Background Modeling With Noise

Finally, we evaluate the performance of our SRLRMR algorithm on videos.³ There are 118 frames in total, and each original frame is downsampled to 72×86 . We reshape each frame into a vector and arrange all vectors into one large matrix. Here, we consider an even more difficult problem: the videos are corrupted by large random pixel noise. Low-rank matrix analysis has become a powerful approach for background modeling, particularly for videos with large noise. The restored low-rank matrix is considered to be the static background that exists in every frame.

The max iteration number *maxiter* is set to be 1 in Algorithm 1. μ is set as $\mu_0 = 0.50/\|M\|$. In Fig. 17, the background of the input video is reconstructed using low-rank matrix decomposition. In rows 1 and 2, the original

³The video is downloaded from <http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/>.

video is corrupted by gross errors randomly on 45% of pixels. As shown in Fig. 17, our proposed method achieves the best performance among the four mentioned methods.

VII. CONCLUSIONS

In this paper, we have developed a smoothed low-rank matrix recovery algorithm based on a reweighted nuclear norm. The proposed method integrates the nuclear norm, TV norm, and l_1 norm together. We have demonstrated the superior performance of the proposed method on single-image denoising, hyperspectral image denoising and video background modeling from corrupted observations. The experimental results show that our method significantly outperforms the original PCP optimization, reweighted l_1 norm minimization and NSVT methods.

Despite the promising performance of our proposed method, there is still room for further improvement. In the experiments, we used the same values of parameters for all of the hyperspectral images or video. However, the parameters should be changed for different problems. Thus, how to choose noise-adaptive parameters will be studied in the future. Moreover, theoretical analysis of the proposed method, such as the convergence, is one of the important issues for future work.

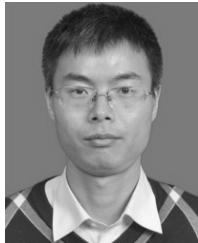
ACKNOWLEDGMENTS

The authors thank Y. G. Peng for sharing the implementation of the NSVT algorithm as well as for guidance.

REFERENCES

- [1] X. Li and Y. Pang, "Deterministic column-based matrix decomposition," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 1, pp. 145–149, Jan. 2010.
- [2] R.-P. Wen and X.-H. Yan, "A new gradient projection method for matrix completion," *Appl. Math. Comput.*, vol. 258, pp. 537–544, May 2015.
- [3] J. Tanner and K. Wei, "Low rank matrix completion by alternating steepest descent methods," *Appl. Comput. Harmon. Anal.*, vol. 40, no. 2, pp. 417–429, Mar. 2016.
- [4] T.-Y. Ji, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, and G. Liu, "Tensor completion using total variation and low-rank matrix factorization," *Inf. Sci.*, vol. 326, pp. 243–257, Jan. 2016.
- [5] Y. Yuan, X. Li, Y. Pang, X. Lu, and D. Tao, "Binary sparse nonnegative matrix factorization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 5, pp. 772–777, May 2009.
- [6] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.
- [7] M. Malek-Mohammadi, M. Babaie-Zadeh, A. Amini, and C. Jutten, "Recovery of low-rank matrices under affine constraints via a smoothed rank function," *IEEE Trans. Signal Process.*, vol. 62, no. 4, pp. 981–992, Feb. 2014.
- [8] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2980–2998, Jun. 2010.
- [9] Z. Lai, W. K. Wong, Y. Xu, J. Yang, and D. Zhang, "Approximate orthogonal sparse embedding for dimensionality reduction," *IEEE Trans. Neural Netw.*, vol. 27, no. 4, pp. 723–735, Apr. 2016.
- [10] C. Hong, J. Yu, J. Wan, D. Tao, and M. Wang, "Multimodal deep autoencoder for human pose recovery," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5659–5670, Dec. 2015.
- [11] F. Shi, J. Cheng, L. Wang, P.-T. Yap, and D. Shen, "Low-rank total variation for image super-resolution," *Med. Image Comput. Comput. Assist. Intervent.*, vol. 16, no. 1, pp. 155–162, 2013.
- [12] F. Shi, J. Cheng, L. Wang, P.-T. Yap, and D. Shen, "LRTV: MR image super-resolution with low-rank and total variation regularizations," *IEEE Trans. Med. Imag.*, vol. 34, no. 12, pp. 2459–2466, Dec. 2015.
- [13] H. Zhang, W. He, L. Zhang, H. Shen, and Q. Yuan, "Hyperspectral image restoration using low-rank matrix recovery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 4729–4743, Aug. 2014.
- [14] X. Liu, G. Zhao, J. Yao, and C. Qi, "Background subtraction based on low-rank and structured sparse decomposition," *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2502–2514, Aug. 2015.
- [15] N. Komodakis, "Image completion using global optimization," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2006, pp. 442–452.
- [16] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma, "Robust principal component analysis: Exact recovery of corrupted low-rank matrices," in *Proc. Neural Inf. Process. Syst.*, 2009, pp. 1–9.
- [17] S. Jonna, V. S. Voltei, R. R. Sahay, and M. S. Kankanhalli, "A multimodal approach for image de-fencing and depth inpainting," in *Proc. Int. Conf. Adv. Pattern Recognit.*, Jan. 2015, pp. 1–6.
- [18] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
- [19] S. Hu and J. Wang, "Absolute exponential stability of a class of continuous-time recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 35–45, Jan. 2003.
- [20] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.
- [21] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 1, pp. 1–37, 2009.
- [22] Y. Deng, Y. Liu, Q. Dai, Z. Zhang, and Y. Wang, "Noisy depth maps fusion for multiview stereo via matrix completion," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 5, pp. 566–582, Sep. 2012.
- [23] E. J. Candès and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, no. 6, pp. 925–936, Jun. 2010.
- [24] S. Oymak, K. Mohan, M. Fazel, and B. Hassibi, "A simplified approach to recovery conditions for low rank matrices," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul./Aug. 2011, pp. 2318–2322.
- [25] F. Zhang, J. Yang, Y. Tai, and J. Tang, "Double nuclear norm-based matrix decomposition for occluded image recovery and background modeling," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1956–1966, Jun. 2015.
- [26] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.
- [27] M. Jaggi and M. Sulovský, "A simple algorithm for nuclear norm regularized problems," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 471–478.
- [28] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific J. Optim.*, vol. 6, no. 15, pp. 615–640, 2010.
- [29] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [30] Z. Lin, M. Chen, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," Dept. Elect. Comput. Eng., Univ. Illinois Urbana-Champaign, Champaign, IL, USA, Tech. Rep. UIUC-ENG-09-2215, 2009.
- [31] C. Lu, Z. Lin, and S. Yan, "Smoothed low rank and sparse matrix recovery by iteratively reweighted least squares minimization," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 646–654, Feb. 2015.
- [32] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, nos. 5–6, pp. 877–905, 2008.
- [33] Y. Peng, J. Suo, Q. Dai, and W. Xu, "Reweighted low-rank matrix recovery and its application in image restoration," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2418–2430, Feb. 2014.
- [34] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 460–489, 2005.
- [35] M.-D. Iordache, J. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, Jun. 2011.
- [36] H. Zhang, H. Shen, and L. Zhang, "A super-resolution reconstruction algorithm for hyperspectral images," *Signal Process.*, vol. 92, no. 9, pp. 2082–2096, Sep. 2012.
- [37] D. Needell and R. Ward, "Stable image reconstruction using total variation minimization," *SIAM J. Imag. Sci.*, vol. 6, no. 2, pp. 1035–1058, Mar. 2013.
- [38] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2419–2434, Nov. 2009.

- [39] X. M. Yuan and J. F. Yang. (2009). *Sparse and Low-Rank Matrix Decomposition via Alternating Direction Methods*. [Online]. Available: <http://perception.csl.illinois.edu/matrix-rank/sample-code.html>
- [40] Y. Deng, Q. Dai, R. Liu, Z. Zhang, and S. Hu, "Low-rank structure learning via nonconvex heuristic recovery," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 3, pp. 383–396, Mar. 2013.
- [41] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [42] Y. Hu, C. Zhao, D. Cai, X. He, and X. Liu, "Atom decomposition with adaptive basis selection strategy for matrix completion," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 12, no. 3, Jun. 2016, Art. no. 43.



Hengyou Wang received the Ph.D. degree from Beijing Jiaotong University, Beijing, China, in 2017. He is currently an Associate Professor with the School of Science, Beijing University of Civil Engineering and Architecture, Beijing. His research interests include sparse representation, low-rank matrix theory, and image classification and recognition.



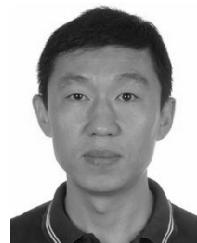
Yigang Cen received the Ph.D. degree in control science and engineering from the Huazhong University of Science and Technology, in 2006. In 2006, he joined the Signal Processing Centre, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, as a Research Fellow. From 2014 to 2015, he was a Visiting Scholar of the Department of Computer Science, University of Missouri, USA. He is currently a Professor and a Supervisor of doctor student with the School of Computer and Information Technology, Beijing Jiaotong University. His research interests include compressed sensing, sparse representation, low-rank matrix reconstruction, and wavelet construction theory.



Zhiqian He received the Ph.D. degree from the Department of Computer Science, University of Missouri, Columbia, MO, USA, in 2014. He was a Research and Development Software Engineer with Math Works, USA, from 2014 to 2017. He is currently an Assistant Professor with the College of Information Engineering, Shenzhen University, China. His research interest is focused on computer vision, machine learning, and deep learning.



Zhihai He received the B.S. degree in mathematics from Beijing Normal University, Beijing, China, in 1994, the M.S. degree in mathematics from the Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, in 1997, and the Ph.D. degree in electrical engineering from the University of California at Santa Barbara, Santa Barbara, CA, USA, in 2001. In 2001, he joined Sarnoff Corporation, Princeton, NJ, USA, as a member of Technical Staff. In 2003, he joined the Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO, USA, where he is currently a tenured Full Professor. His current research interests include image/video processing and compression, wireless sensor network, computer vision, and cyber-physical systems. He is a member of the Visual Signal Processing and Communication Technical Committee of the IEEE Circuits and Systems Society, and has served as a technical program committee member or the session chair for a number of international conferences. He is the Co-Chair of the 2007 International Symposium on Multimedia over Wireless, Hawaii. He received the 2002 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Best Paper Award and the SPIE VCIP Young Investigator Award in 2004. He has served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, and the *Journal of Visual Communication and Image Representation*. He is also a Guest Editor for the IEEE TCSVT special issue on video surveillance.



Ruizhen Zhao received the Ph.D. degree from Xidian University, Xian, China, in 2001. After that, he was a Post-Doctoral Fellow with the Institute of Automation, Chinese Academy of Sciences. He is currently a Professor and a Supervisor of doctor student with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. His research interests include wavelet transform and its applications, and algorithms of image and signal processing compressive and sparse representation.



Fengzhen Zhang is currently pursuing the Ph.D. degree with the Institution of Information Science, Beijing Jiaotong University. Her research interests include compressed sensing, sparse coding, and dictionary learning.