Reweighted Low-Rank Matrix Recovery and its Application in Image Restoration

Yigang Peng, Jinli Suo, Qionghai Dai, *Senior Member, IEEE,* and Wenli Xu

***Abstract*—In this paper, we propose a reweighted low-rank matrix recovery method and demonstrate its application for robust image restoration. In the literature, principal component pursuit solves low-rank matrix recovery problem via a convex program of mixed nuclear norm and ℓ_1 norm. Inspired by reweighted ℓ_1 minimization for sparsity enhancement, we propose reweighting singular values to enhance low rank of a matrix. An efficient iterative reweighting scheme is proposed for enhancing low rank and sparsity simultaneously and the performance of low-rank matrix recovery is prompted greatly. We demonstrate the utility of the proposed method both on numerical simulations and real images/videos restoration, including single image restoration, hyperspectral image restoration, and background modeling from corrupted observations. All of these experiments give empirical evidence on significant improvements of the proposed algorithm over previous work on low-rank matrix recovery.**

***Index Terms*—Image restoration, iterative reweighting, low-rank matrix recovery, nonuniform singular value thresholding.**

I. Introduction

LOW-RANK MATRIX recovery from corrupted matrix is a highly developing research topic with close relationship to both compressed sensing [1] and matrix rank minimization [2], [3], and has wide applications in computer vision and image processing [4]–[7], signal processing [8], and many other fields. It considers recovering a low-rank matrix from gross sparse errors, also known as robust principle component analysis (RPCA) [4]

$$\min_{\mathbf{L},\mathbf{S}} \quad \text{rank}(\mathbf{L}) + \gamma \|\mathbf{S}\|_0, \quad \text{s.t.} \quad \mathbf{M} = \mathbf{L} + \mathbf{S} \tag{1}$$

where parameter $\gamma > 0$, rank($\mathbf{L}$) is the rank of matrix $\mathbf{L}$, and $\|\mathbf{S}\|_0$ is the pseudo-ℓ_0 norm of matrix $\mathbf{S}$ (i.e., the number of non-zero items in matrix $\mathbf{S}$). However, both minimizing the ℓ_0 norm and matrix rank function are not directly tractable due to the highly nonlinear, highly nonconvex properties of the objective functions. A surprising result given by Candès *et al*. [4] shows that under rather weak assumptions, solving the following convex optimization problem:

$$\min_{\mathbf{L},\mathbf{S}} \quad \|\mathbf{L}\|_* + \lambda \|\mathbf{S}\|_1, \quad \text{s.t.} \quad \mathbf{M} = \mathbf{L} + \mathbf{S} \tag{2}$$

which is dubbed as principal component pursuit (PCP), can exactly recover the groundtruth low-rank matrix $\mathbf{L}^0$ and the sparse matrix $\mathbf{S}^0$ (the observation matrix $\mathbf{M}$ is generated by $\mathbf{M} = \mathbf{L}^0 + \mathbf{S}^0$), where $\|\mathbf{L}\|_*$ is the nuclear norm of matrix $\mathbf{L}$ (i.e. the sum of all its singular values), $\|\mathbf{S}\|_1$ is the ℓ_1 norm of matrix $\mathbf{S}$ (i.e. the sum of absolute value of all its entries), $\mathbf{M}, \mathbf{L}, \mathbf{S}, \mathbf{L}^0, \mathbf{S}^0 \in \mathbb{R}^{m\times n}$, and $\lambda > 0$ is a weighting parameter.

Due to the inherent sparse or/and low rank property in images and videos, such a low-rank matrix recovery model and its variants have been successfully applied in many image processing and computer vision problems. A patch based robust video restoration method, which groups similar patches in spatial-temporal domain and then formulates the restoration problem as joint sparse and low-rank matrix approximation, is proposed by Ji *et al*. [9]. In [4], by modeling the stable background as low-rank part and foreground variations as sparse part, PCP demonstrates its successful application in surveillance video background modeling. A robust alignment method by sparse and low rank decomposition (RASL) is proposed in [10], which simultaneously aligns a batch of linearly correlated images in spite of gross corruption. Low-rank structure and sparse modeling is harnessed for holistic symmetry detection and rectification by a method called transform invariant low-rank texture (TILT) [11]. These techniques have been extended for camera calibration [12], generalized cylindrical surfaces unwrapping [13], urban structuresreconstruction [14], text detection [15], etc. In a recent work [16] on image completion, they correctly repaired the global structure of a corrupted texture by harnessing both low-rank and sparse structures in regular or near regular textures. More applications also lie in dynamic MRI reconstruction [17] and many others.

A lot of practical algorithms are developed to solve ℓ_1 minimization or/and nuclear norm minimization, such as singular value thresholding (SVT) [18], accelerated proximal gradient (APG) algorithm [19], augmented lagrangian multiplier (ALM) method [20], [21] etc. In [4], they empirically showed the algorithms' ability of matrix recovery with varying rank from errors of varying sparsity.

Although the convex optimization algorithm PCP (2) works well in many low-rank matrix recovery scenarios, it has limited

Manuscript received September 6, 2012; revised November 6, 2013; accepted February 6, 2014. Date of publication March 19, 2014; date of current version November 13, 2014. This work was supported in part by the National Nature Science Foundation of China under Grant 61327902, Grant 61035002, and Grant 61271450 and in part by the Beijing Key Laboratory of Multidimension & Multi-scale Computational Photography, Tsinghua University. This paper was recommended by Associate Editor E. Santos, Jr.

Y. Peng is with the National Computer Network Emergency Response Technical Team Coordination Center of China, Beijing, 100029, China (e-mail: pengyigang@cert.org.cn).

J. Suo, Q. Dai, and W. Xu are with the Department of Automation, Tsinghua University, Beijing, 100084, China (e-mail: jlsuo@tsinghua.edu.cn; qionghaidai@tsinghua.edu.cn; xuwl@tsinghua.edu.cn).

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/TCYB.2014.2307854

performance especially when the matrix becomes complicated. If the intrinsic rank of data matrix is high, or the corrupted errors become dense, the convex PCP algorithm may not be able to successfully recover the data matrix. In order to promote the performance of low-rank matrix recovery, we propose a reweighting scheme for handling this tough task.

A. Related Work

In the low-rank matrix recovery problem (1), the main difficulties lie in the combined ℓ_0 norm and matrix rank minimization, which are closely related to compressed sensing [1] and matrix rank minimization [2]. Except for convex relaxations, nonconvex optimization techniques are also adopted to solve ℓ_0 norm minimization and matrix rank minimization. For instance, Daubechies *et al*. [22] investigated iteratively reweighted least squares (IRLS) minimization for compressed sensing. Similar ideas are extended for matrix rank minimization by different researchers.

Fornasier *et al*. [23] investigated an iteratively reweighted least squares algorithm for affine matrix rank minimization (IRLS-M) by observing

$$\|\mathbf{X}\|_* = \text{trace}\left[(\mathbf{X}\mathbf{X}^T)^{-1/2}(\mathbf{X}\mathbf{X}^T)\right] = \|\mathbf{W}^{1/2}\mathbf{X}\|_F^2 \tag{3}$$

where $\mathbf{W} = (\mathbf{X}\mathbf{X}^T)^{-1/2}$, and $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. They recursively solve

$$\mathbf{X}^{k+1} = \arg\min_{\phi(\mathbf{X})=\mathcal{M}} \|(\mathbf{W}^k)^{1/2}\mathbf{X}\|_F^2 \tag{4}$$

where $\mathbf{W}^k = \left(\mathbf{X}^k(\mathbf{X}^k)^T\right)^{-1/2}$, the sampling operator $\phi : \mathbb{R}^{m\times n} \to \mathbb{R}^q$ maps the matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$ to a measured data vector $\mathcal{M} \in \mathbb{R}^q$, and $\mathbf{X}^k$ is obtained in the kth iteration.

Mohan and Fazel [24] extended a family of iterative reweighted least squares algorithms, including IRLS-p and sIRLS-p, for matrix rank minimization problem. They consider nonconvex approximations to the rank function and define the smooth Schatten-p function as

$$f_p(\mathbf{X}) = \text{trace}(\mathbf{X}^T\mathbf{X} + \delta\mathbf{I})^{p/2} \tag{5}$$

which is differentiable for $p > 0$ and convex for $p \geq 1$, and where $\delta > 0$ and $\mathbf{I}$ is an identity matrix. Notice that, when $\delta = 0$, $f_1(\mathbf{X}) = \|\mathbf{X}\|_*$.

Alternatively, one of the most popular nonconvex surrogate functions for matrix rank may be log-det heuristic [25]. For positive semidefinite matrix $\mathbf{X}$, they use a smooth surrogate for rank($\mathbf{X}$) and solve

$$\min_{\mathbf{X}} \quad \log\det(\mathbf{X} + \delta\mathbf{I}) \quad \text{s.t.} \quad \mathbf{X} \in \mathcal{C} \tag{6}$$

where $\delta > 0$, $\mathbf{I}$ is an identity matrix, and $\mathcal{C}$ is a convex set. For general case, Mohan and Fazel showed each local iterative linearization of log-det objective function reduces to a reweighted trace heuristic (RTH), and further proposed reweighted nuclear norm minimization

$$\min_{\mathbf{X}} \quad \|\mathbf{W}_1^k\mathbf{X}\mathbf{W}_2^k\|_*, \quad \text{s.t.} \quad \mathbf{X} \in \mathcal{C} \tag{7}$$

where $\mathbf{W}_1^k$ and $\mathbf{W}_2^k$ are weights in kth iteration [26], [27].

A majorize-minimize algorithm for recovery of sparse and low-rank matrices from noisy and undersampled measurements is proposed by Hu *et al*. [28] recently. They combined data consistency error, nonconvex spectral penalty, and nonconvex sparsity penalty. To solve the problem, they relax the problem by using the convex Schatten p-norm $\sum_{i=1}^r \sigma_i(\mathbf{X})^p$ ($p > 1$) for relaxation of the rank constraint rank($\mathbf{X}$) and the convex ℓ_p norm $\|\mathbf{X}\|_p^p = \sum_{i,j} |X_{i,j}|^p$ ($p > 1$) for sparsity penalty $\|\mathbf{X}\|_0$, where $\sigma_i(\mathbf{X})$ is the ith singular value of matrix $\mathbf{X}$ and $X_{i,j}$ is the i,jth element of it.

B. Motivation

This paper is inspired by reweighted ℓ_1 minimization for enhancing sparsity [29]. By minimizing a sequence of weighted ℓ_1 norm, Candès *et al*. [29] showed significant performance improvement on sparse recovery or/and estimation. The reweighted ℓ_1 minimization is further analyzed in [30] via the Grassmann angle framework.

Reweighted ℓ_1 norm is defined as $\|\mathbf{w} \odot \mathbf{x}\|_1$ for some appropriate weights $\mathbf{w}$, where $\mathbf{x} = \{x_i\} \in \mathbb{R}^n$, $\mathbf{w} = \{w_i\} \in \mathbb{R}_{++}^n$. Note that, if for each item x_i, the weights w_i are exactly inversely proportional to the absolute value of x_i, i.e., $w_i = \frac{1}{|x_i|}$ (for $x_i \neq 0$), we have $\|\mathbf{w} \odot \mathbf{x}\|_1 = \|\mathbf{x}\|_0$, where $\odot$ denotes the component-wise product (Hadamard product) of two variables, and $\|\cdot\|_0$ denotes the pseudo-ℓ_0 norm (the number of nonzero entries).

If reweighted ℓ_1 norm is applied for sparsity enhancement in low-rank matrix recovery problem, we obtain the following convex optimization problem [31], that is

$$\min_{\mathbf{L},\mathbf{S}} \quad \|\mathbf{L}\|_* + \lambda\|\mathbf{W}_\mathbf{S} \odot \mathbf{S}\|_1 \quad \text{s.t.} \quad \mathbf{M} = \mathbf{L} + \mathbf{S} \tag{8}$$

where $\mathbf{W}_\mathbf{S} \in \mathbb{R}^{m\times n}$ are weights for entries of $\mathbf{S}$.

While, to minimize rank($\mathbf{X}$), it is equivalent to minimizing $\|\sigma(\mathbf{X})\|_0$, where $\sigma(\mathbf{X}) = \{\sigma_i(\mathbf{X})\}$ is the singular value vector constructed from $\mathbf{X}$. Inspired by reweighted ℓ_1 norm minimization, we propose reweighted nuclear norm which is defined as $\sum_i w_i\sigma_i(\mathbf{X})$ for some appropriate weights w_i. Moreover, if $w_i = \frac{1}{\sigma_i(\mathbf{X})}$ (for $\sigma_i(\mathbf{X}) \neq 0$), we have $\sum_i w_i\sigma_i(\mathbf{X}) = \text{rank}(\mathbf{X})$. In practice, we estimate the weights w_i^{k+1} from the previous iteration. For instance, in $(k+1)$th iteration, $w_i^{k+1} = \frac{1}{\sigma_i(\mathbf{X}^k)+\epsilon}$ (where, $\epsilon > 0$ is a preset constant).

C. Contribution

In this paper, we propose a nonuniform singular value thresholding (NSVT) operator to enhance low rank for matrix rank minimization, which shares a similar philosophy with nonuniform soft-thresholding operator for reweighted ℓ_1 norm minimization for sparsity enhancement. We apply NSVT accompanying with reweighted ℓ_1 norm minimization to RPCA problem so as to increase the correct recovery region. We show that by properly reweighting singular values for low-rank matrix and reweighting ℓ_1 norm for sparse matrix, better matrix recovery performance can be obtained. As a summary, our contributions lie in the following.

1) We propose an iterative reweighting algorithm for low-rank matrix recovery and apply the NSVT operator for matrix low rank enhancement.

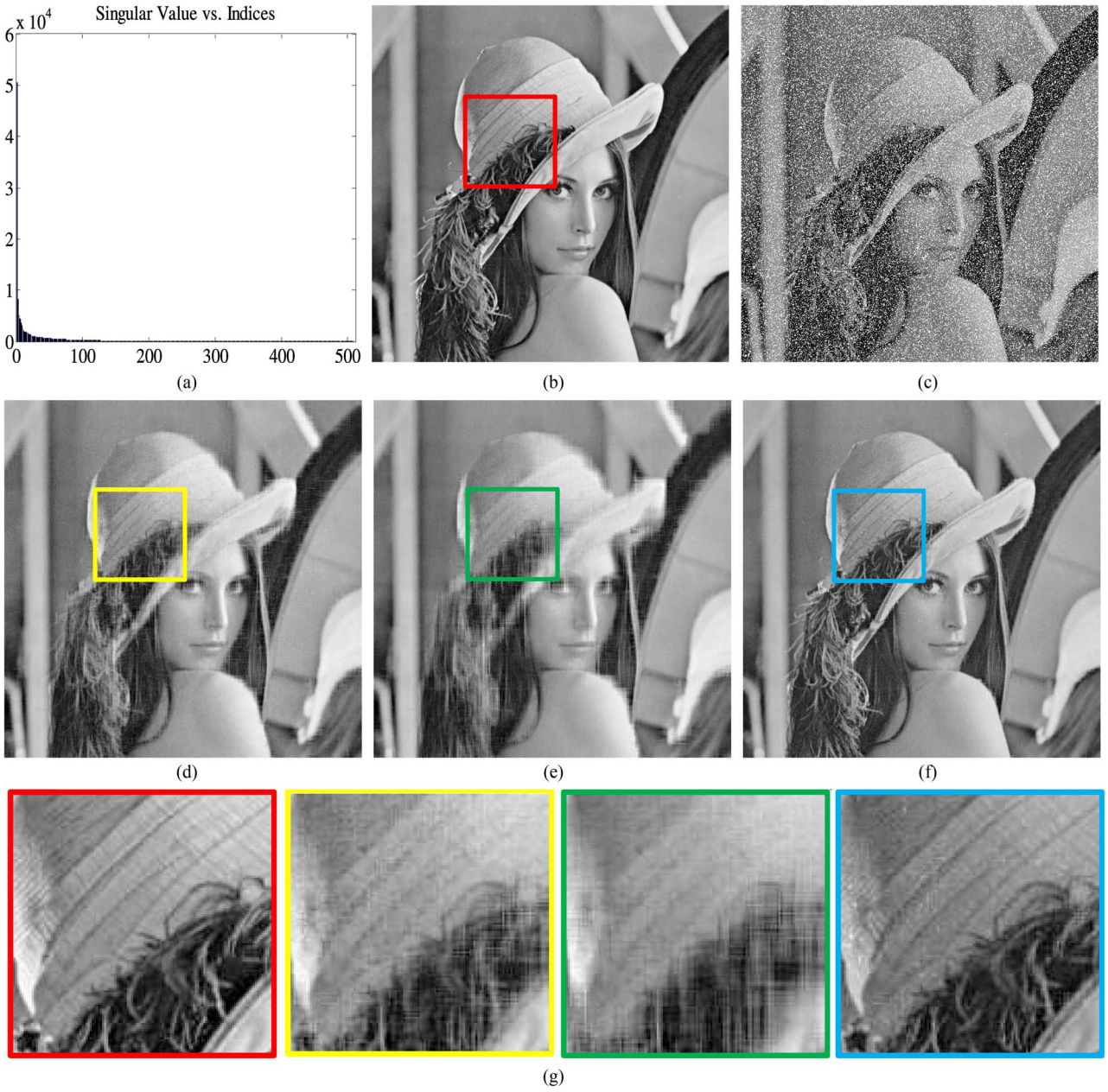


Fig. 1. Restoration of a single image *Lena*, 20% of whose pixels are corrupted by large noise. (a) Singular value distribution. (b) Original. (c) Input (10.90 dB/0.054). (d) PCP (28.60 dB/0.844). (e) Reweighted ℓ_1 (25.96 dB/0.808). (f) NSVT (34.62 dB/0.931). (g) Close-up comparison. (a) Singular value distribution of the original image shown in (b). (c) Corrupted image to be restored. (d)–(f) Restored images obtained by PCP, reweighted ℓ_1, and NSVT, respectively. (g) Close-up patches of (b) and (d)–(f), respectively.

2) Proposed method has significant improvement on correct recovery of low-rank matrix and demonstrates successful applications in different image restoration problems.

As the first glance of our proposed algorithm, we show a successful application of the NSVT algorithm in restoring images from gross large errors here. Considering an image as a matrix, Fig. 1(a) shows the singular value distribution obtained by singular value decomposition of the original image shown in Fig. 1(b). From the distribution of singular values, we observe that the singular values decay very fast, which indicates the low-rank structure of the image. The input image to be restored is corrupted by gross sparse errors on 20% pixels randomly as shown in Fig. 1(c). We hope to restore it from the corrupted images by utilizing the low-rank property of the image. We formulate the image restoration as recovering a low-rank matrix (representing the uncorrupted image) from the corrupted observation. We show the recovered images obtained by PCP, reweighted ℓ_1, and NSVT in Figs. 1(d)–(f), respectively (please see below for details). We could see the visual appealing restoration result obtained by our proposed NSVT algorithm from Fig. 1(f). To have a close-up observation, we also show a 128×128 patch of Figs. 1 and 3(b) and (d)–(f) in Fig. 1(g).

As a reminder, the paper is organized as follows. In Section II, we introduce our proposed non-uniform singular

value thresholding operator. The iterative low rank and sparsity enhancement algorithm for low-rank matrix recovery problem is proposed in details in Section III, followed by extensive numerical simulations and different image restoration experiments in Section IV. Finally, we conclude our work and discuss future work in Section V.

II. Nonuniform Singular Value Thresholding Operator

In this section, we first introduce the non-uniform soft thresholding operator used for reweighted ℓ_1 norm minimization [29], and then propose the NSVT operator for low rank enhancement.

A. Nonuniform Soft Thresholding Operator

For a vector $\mathbf{y} \in \mathbb{R}^n$ and thresholding weights vector $\mathbf{w} \in \mathbb{R}^n_{++}$, the non-uniform soft thresholding operator $\mathcal{S}_\mathbf{w}$ is introduced as follows:

$$\mathcal{S}_\mathbf{w}[\mathbf{y}] = \{\text{sign}(y_i)(|y_i| - w_i)_+\} \tag{9}$$

where $t_+ = \max(t, 0)$ is the positive part of t.

Proposition 1: For a vector $\mathbf{y} \in \mathbb{R}^n$ and thresholding weights vector $\mathbf{w} \in \mathbb{R}^n_{++}$, the non-uniform soft thresholding operator (9) obeys

$$\mathcal{S}_\mathbf{w}[\mathbf{y}] = \arg\min_\mathbf{x} \left(\frac{1}{2}\|\mathbf{x} - \mathbf{y}\|_2^2 + \|\mathbf{w} \odot \mathbf{x}\|_1 \right) \tag{10}$$

where $\odot$ denotes the component-wise product (Hadamard product) of two variables, and $\|\cdot\|_2$ and $\|\cdot\|_1$ denote the ℓ_2 norm and ℓ_1 norm of a vector, respectively. Equivalently, the nonuniform soft thresholding operator obeys

$$\mathcal{S}_\mathbf{w}[\mathbf{y}] = \mathcal{S}_{\mathbf{w}_0}[\mathbf{z}] = \arg\min_\mathbf{x} \left(\frac{1}{2}\|\mathbf{x} - \mathbf{z}\|_2^2 + w_0\|\mathbf{x}\|_1 \right) \tag{11}$$

where, $\mathbf{z}$ denotes $\mathbf{z} = \text{sign}(\mathbf{y}) \odot (|\mathbf{y}| + w_0\mathbf{1} - \mathbf{w})$, $w_0 = \max(\{w_i\})$, $\mathbf{w}_0 = w_0\mathbf{1}$, $|\mathbf{y}| = \text{sign}(\mathbf{y}) \odot \mathbf{y}$, and $\mathbf{1} \in \mathbb{R}^n$ with all its entries 1.

B. Nonuniform Singular Value Thresholding Operator

For a matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$ of rank r, we denote its Singular Value Decomposition (SVD) as

$$\mathbf{Y} = \mathbf{U}\Sigma\mathbf{V}^\mathbf{T}, \quad \Sigma = \text{diag}(\{\sigma_i\}_{1\le i\le r}) \tag{12}$$

where $\mathbf{U} \in \mathbb{R}^{m\times r}$ and $\mathbf{V} \in \mathbb{R}^{r\times n}$ are with orthonormal columns, respectively, the singular values $\sigma_i > 0$, and $\Sigma \in \mathbb{R}^{r\times r}$ is the diagonal matrix with $\sigma_1, \cdots, \sigma_r$ on the diagonal and zeros elsewhere. Given the thresholding weights vector $\mathbf{w} \in \mathbb{R}^r_{++}$, we define NSVT operator as follows:

$$\mathcal{D}_\mathbf{w}\big[\mathbf{Y}\big] = \mathbf{U}\mathcal{S}_\mathbf{w}\big[\Sigma\big]\mathbf{V}^\mathbf{T} \tag{13}$$

where $\mathcal{S}_\mathbf{w}\big[\Sigma\big] = \text{diag}(\{(\sigma_i - w_i)_+\})$.

Proposition 2: For a matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$ of rank r and thresholding weights vector $\mathbf{w} \in \mathbb{R}^r_{++}$, the NSVT operator (13) obeys

$$\mathcal{D}_\mathbf{w}\big[\mathbf{Y}\big] = \mathcal{D}_{\mathbf{w}_0}\big[\mathbf{Z}\big] = \arg\min_\mathbf{X} \left(\frac{1}{2}\|\mathbf{X} - \mathbf{Z}\|_F^2 + w_0\|\mathbf{X}\|_* \right) \tag{14}$$

where $\mathbf{Z}$ denotes $\mathbf{Z} = \mathbf{U}(\Sigma + w_0\mathbf{I} - \mathbf{W})\mathbf{V}^\mathbf{T}$, $\mathbf{Y} = \mathbf{U}\Sigma\mathbf{V}^\mathbf{T}$ is defined as (12), $w_0 = \max(\{w_i\})$, $\mathbf{I} \in \mathbb{R}^{r\times r}$ is the identity matrix, $\mathbf{w}_0 \in \mathbb{R}^r_{++}$ with all its entries w_0, and $\mathbf{W} = \text{diag}(\mathbf{w})$ is a diagonal matrix.

As we can see from (10) and (11), the nonuniform soft thresholding of a vector $\mathbf{y}$ is equivalent to the uniform soft thresholding of a modified vector $\mathbf{z}$. Similarly, the nonuniform singular value thresholding operator applied to a matrix $\mathbf{Y}$ equals to uniform singular value thresholding operator applied to a modified matrix $\mathbf{Z}$, as shown in (13) and (14).

III. Iterative Algorithm for Reweighted Low-Rank Matrix Recovery

We describe an iterative algorithm for reweighted low-rank matrix recovery by enhancing low rank and sparsity simultaneously below.

A. Problem Description: Minimization of Mixed Weighted Nuclear Norm and ℓ_1 Norm

The idea of reweighted ℓ_1 norm minimization is to use large weights so as to discourage nonzero entries and small weights so as to encourage nonzero entries. Our proposed NSVT is motivated by such a similar idea for singular values. Notice that the rank of a matrix is actually the number of nonzero singular values of the matrix. Thus, similar to the philosophy of reweighted ℓ_1 norm minimization, we explore large weights to discourage nonzero singular values and small weights to encourage nonzero singular values, which leads us to consider the following weighted nuclear norm and ℓ_1 norm mixed minimization problem:

$$\min_{\mathbf{L},\mathbf{S}} \quad \sum_{j=1}^{n} w_{\mathbf{L},j} \cdot \sigma_j + \lambda\|\mathbf{W}_\mathbf{S} \odot \mathbf{S}\|_1 \quad \text{s.t.} \quad \mathbf{M} = \mathbf{L} + \mathbf{S} \tag{15}$$

where $\mathbf{w}_\mathbf{L} = \{w_{\mathbf{L},j}\}$ and $\mathbf{W}_\mathbf{S}$ are weights for singular values of $\mathbf{L}$ and entries of $\mathbf{S}$, respectively, and $\{\sigma_j\}$ are singular values of matrix $\mathbf{L}$. Note that, the 'weighted' nuclear norm, i.e., weighted sum of singular values, is not a convex function in general and thus it is not a norm any more.

B. Iterative Algorithm for Enhancing Low Rank and Sparsity

One immediate question is how to choose weights $\mathbf{w}_\mathbf{L}$ and $\mathbf{W}_\mathbf{S}$ wisely to improve matrix recovery. As suggested in [29], the weights $\mathbf{W}_\mathbf{S}$ are inversely proportional to signal magnitude for reweighted ℓ_1 norm minimization. Similarly, the weights $\mathbf{w}_\mathbf{L}$ are chosen to be inversely proportional to the magnitudes of singular values. In practice, the favorable weights $\mathbf{w}_\mathbf{L}$ and $\mathbf{W}_\mathbf{S}$ are updated based on estimations of $(\mathbf{L}^{(k)}, \mathbf{S}^{(k)})$ in the previous iteration according to (16).

Thus, we propose the following iterative low rank and sparsity enhancement scheme for matrix recovery by alternatively estimating $(\mathbf{L}, \mathbf{S})$ and redefining weights $(\mathbf{w}_\mathbf{L}, \mathbf{W}_\mathbf{S})$. The whole algorithm is summarized in Algorithm 1.

C. Solving Inner Optimization Via Inexact Augmented Lagrangian Multiplier Method

Next, we show how an Inexact Augmented Lagrangian Multiplier (IALM) method [20] can be adopted to solve

Algorithm 1 Enhancing Low Rank and Sparsity for Low-Rank Matrix Recovery

Step 1: Set the iteration counter $k = 0$, and $\mathbf{w_L}^{(0)} = \mathbf{1} \in \mathbb{R}^n$, $\mathbf{W_S}^{(0)} = \mathbf{1} \cdot \mathbf{1}^T \in \mathbb{R}^{m \times n}$ (supposing $m \geq n$).

Step 2: Enhance low rank and sparsity using **Algorithm 2** by weights $\mathbf{w_L}^{(k)}$ and $\mathbf{W_S}^{(k)}$, and get solutions $(\mathbf{L}^*, \mathbf{S}^*) \rightarrow (\mathbf{L}^{(k)}, \mathbf{S}^{(k)})$.

Step 3: Update weights: the weights for each $i = 1, ..., m$ and $j = 1, ..., n$ is updated by

$$w_{\mathbf{L},j}^{(k+1)} = \frac{1}{\sigma_j^{(k)} + \epsilon_{\mathbf{L}}}, \quad w_{\mathbf{S},ij}^{(k+1)} = \frac{1}{|s_{ij}^{(k)}| + \epsilon_{\mathbf{S}}}, \tag{16}$$

where $\epsilon_{\mathbf{L}}$ and $\epsilon_{\mathbf{S}}$ are predetermined positive constants, and the singular value matrix $\Sigma^{(k)} = \text{diag}\left(\left[\sigma_1^{(k)}, \cdots, \sigma_n^{(k)}\right]\right) \in \mathbb{R}^{n \times n}$ with

$$\left[\mathbf{U}^{(k)}, \ \Sigma^{(k)}, \ \mathbf{V}^{(k)}\right] = \text{svd}(\mathbf{L}^{(k)}), \tag{17}$$

where $\mathbf{U}^{(k)} \in \mathbb{R}^{m \times n}$, $\mathbf{V}^{(k)} \in \mathbb{R}^{n \times n}$.

Step 4: Terminate on convergence or when k attains a specified maximum number of iterations k_{max}, and output solutions $(\tilde{\mathbf{L}}, \tilde{\mathbf{S}})$. Otherwise, increase k and go to **Step 2**.

problem (15). The augmented Lagrangian function is defined as

$$l(\mathbf{L}, \mathbf{S}, \mathbf{Y}) = \sum_{j=1}^{n} w_{\mathbf{L},j} \cdot \sigma_j + \lambda \|\mathbf{W_S} \odot \mathbf{S}\|_1 + \langle \mathbf{Y}, \mathbf{M} - \mathbf{L} - \mathbf{S} \rangle + \frac{\mu}{2} \|\mathbf{M} - \mathbf{L} - \mathbf{S}\|_F^2 \tag{18}$$

where $\langle , \rangle$ denotes the inner product of two matrices.

The IALM takes advantage of the separable forms with respect to $\mathbf{L}$ and $\mathbf{S}$, and solves a sequence of programs alternatively between

$$\min_{\mathbf{L}} \ l(\mathbf{L}, \mathbf{S}, \mathbf{Y}), \quad \text{(fixing } \mathbf{S}\text{)}$$

and

$$\min_{\mathbf{S}} \ l(\mathbf{L}, \mathbf{S}, \mathbf{Y}), \quad \text{(fixing } \mathbf{L}\text{)}$$

and then updates the Lagrange multiplier matrix via

$$\mathbf{Y} \leftarrow \mathbf{Y} + \mu(\mathbf{M} - \mathbf{L} - \mathbf{S}) \tag{19}$$

where, the matrix $\mathbf{L}$ is updated by NSVT operator

$$\mathbf{L} \leftarrow \mathcal{D}_{\mu^{-1}\mathbf{w_L}}\left[\mathbf{M} - \mathbf{S} + \mu^{-1}\mathbf{Y}\right] \tag{20}$$

with the thresholding value $\mu^{-1}\mathbf{w_L}$, and the matrix $\mathbf{S}$ is updated by the non-uniform soft thresholding operator

$$\mathbf{S} \leftarrow \mathcal{S}_{\lambda\mu^{-1}\mathbf{W_S}}\left[\mathbf{M} - \mathbf{L} + \mu^{-1}\mathbf{Y}\right] \tag{21}$$

with the thresholding value $\lambda\mu^{-1}\mathbf{W_S}$. Specically, such a method is also known as the alternating direction method of multipliers (ADMM), which is popular in and useful for optimizing problems with separable variables, also see [21] and [32].

The IALM algorithm for solving the mixed 'weighted' nuclear norm and ℓ_1 norm minimization problem (15) is summarized in Algorithm 2. Note that the objective function in (15) is not a convex optimization problem and the convergence of the algorithm is still under investigation. However, the numerical simulation and image restoration experiments give empirical evidence on the performance of the algorithm.

Algorithm 2 IALM for Solving The Mixed Weighted Nuclear Norm and ℓ_1 Norm Minimization Problem (15)

INPUT: Initialize $(\mathbf{L}_0, \mathbf{S}_0) \in \mathbb{R}^{m \times n} \times \mathbb{R}^{m \times n}$, $\mathbf{Y}_0 \in \mathbb{R}^{m \times n}$, $\mu_0 > 0$, $\rho > 1$, $t = 0$.

WHILE not converged **DO**

$\mathbf{L}_{t+1} = \mathcal{D}_{\mu_t^{-1}\mathbf{w_L}}\left[\mathbf{M} - \mathbf{S}_t + \mu_t^{-1}\mathbf{Y}_t\right]$;

$\mathbf{S}_{t+1} = \mathcal{S}_{\lambda\mu_t^{-1}\mathbf{W_S}}\left[\mathbf{M} - \mathbf{L}_{t+1} + \mu_t^{-1}\mathbf{Y}_t\right]$;

$\mathbf{Y}_{t+1} = \mathbf{Y}_t + \mu_t (\mathbf{M} - \mathbf{L}_{t+1} - \mathbf{S}_{t+1})$;

$\mu_{t+1} = \rho\mu_t; \quad t \leftarrow t + 1$.

OUTPUT: solutions $(\mathbf{L}^*, \mathbf{S}^*)$.

IV. Experiments

We test our proposed algorithm on a variety of data models to verify its ability in enhancing low rank and sparsity for low-rank matrix recovery problem in this section. We also show its performance on single image restoration, hyperspectral image restoration, and video background modeling from corrupted observations problems.

In the experiments, the weighting parameter λ is set to be $1/\sqrt{\max(m, n)}$ the same as in PCP problem [4]. In Algorithm 1, we fix the constant $\epsilon_{\mathbf{L}} = 0.01$ and $\epsilon_{\mathbf{S}} = 0.01$ empirically. In Algorithm 2, we set $(\mathbf{L}_0, \mathbf{S}_0) = (\mathbf{0}, \mathbf{0})$, $\mathbf{Y}_0 = \mathbf{M}/\max\left(\|\mathbf{M}\|, \lambda^{-1}\|\mathbf{M}\|_\infty\right)$, where $\|\mathbf{M}\|$ denotes the spectral norm of matrix $\mathbf{M}$ and $\|\mathbf{M}\|_\infty$ is the maximum absolute value of entries in matrix $\mathbf{M}$, $\mu_0 = 1.25/\|\mathbf{M}\|$, $\rho = 1.5$, and the stopping criterion for the iteration is $\|\mathbf{M} - \mathbf{L}_t - \mathbf{S}_t\|_F / \|\mathbf{M}\|_F < 10^{-7}$. We denote our proposed rank and sparsity enhancement scheme as NSVT for simplicity. We compare our proposed NSVT method with the original convex PCP problem (2) and the optimization problem (8), which we call as reweighted ℓ_1 for short. For all the three problems of PCP, reweighted ℓ_1 and NSVT, we use IALM to solve them.[1]

A. Phase Transition in Rank and Sparsity

We empirically investigate the ability of the proposed algorithm to recover matrices of varying rank from errors of varying sparsity on different data models. The random matrix pair $(\mathbf{L}^0, \mathbf{S}^0)$ satisfy $\text{rank}(\mathbf{L}^0) = \rho_r n$ and $\|\mathbf{S}^0\|_0 = \rho_s n^2$ where both ρ_r and ρ_s vary between 0 and 1. We test three typical types of random data matrices which are commonly used in the literatures. In the three data models, the low-rank matrices $\mathbf{L}^0$ are all set as $\mathbf{L}^0 = \mathbf{R}_1\mathbf{R}_2^T$ with $\mathbf{R}_1, \mathbf{R}_2 \in \mathbb{R}^{n \times r}$ independently chosen with i.i.d. Gaussian entries sampled from an $\mathcal{N}(0, 1/n)$ distribution, and the sparse matrices $\mathbf{S}^0$ are chosen according to the following models.

1) Data model 1: Uniform sparse matrices $\mathbf{S}^0$: with support chosen uniformly at random and non-zero entries

[1]The code for solving PCP problem using IALM is downloaded from http://perception.csl.illinois.edu/matrix-rank/home.html.

TABLE I

COMPARISONS BETWEEN PCP, REWEIGHTED ℓ_1, AND NSVT FOR LOW-RANK MATRIX RECOVERY ON DATA MODEL 1. HERE, ρ_r = 0.2 AND ρ_s = 0.08

Algorithm	k_{max}	$\frac{\|\tilde{\mathbf{L}}-\mathbf{L}^0\|_F}{\|\mathbf{L}^0\|_F}$	$\text{rank}(\tilde{\mathbf{L}})$	$\|\tilde{\mathbf{S}}\|_0$	# iter	Recovered?
$n = 400$, $\text{rank}(\mathbf{L}^0) = 80$, $\|\mathbf{S}^0\|_0 = 12,800$						
PCP		0.0154	218	101091	34	×
Reweighted ℓ_1	1	$\mathbf{4.45 \times 10^{-5}}$	86	12,828	67	√
Reweighted ℓ_1	2	$\mathbf{4.44 \times 10^{-5}}$	86	12,795	100	√
NSVT	1	$\mathbf{5.87 \times 10^{-5}}$	86	12,795	65	√
NSVT	2	$\mathbf{4.84 \times 10^{-7}}$	**80**	**12,800**	101	√
$n = 800$, $\text{rank}(\mathbf{L}^0) = 160$, $\|\mathbf{S}^0\|_0 = 51,200$						
PCP		0.0061	432	403,779	34	×
Reweighted ℓ_1	1	$\mathbf{7.64 \times 10^{-5}}$	179	51,263	67	√
Reweighted ℓ_1	2	$\mathbf{7.56 \times 10^{-5}}$	176	51,186	100	√
NSVT	1	$\mathbf{7.53 \times 10^{-5}}$	178	51,186	66	√
NSVT	2	$\mathbf{6.84 \times 10^{-5}}$	162	51,199	103	√
$n = 1200$, $\text{rank}(\mathbf{L}^0) = 240$, $\|\mathbf{S}^0\|_0 = 115,200$						
PCP		0.0097	631	912,675	35	×
Reweighted ℓ_1	1	$\mathbf{4.06 \times 10^{-5}}$	280	115,682	69	√
Reweighted ℓ_1	2	$\mathbf{3.95 \times 10^{-5}}$	278	115,173	103	√
NSVT	1	$\mathbf{7.91 \times 10^{-5}}$	304	115,160	68	√
NSVT	2	$\mathbf{5.27 \times 10^{-5}}$	252	115,193	105	√

TABLE II

COMPARISONS BETWEEN PCP, REWEIGHTED ℓ_1, AND NSVT FOR LOW-RANK MATRIX RECOVERY ON DATA MODEL 2. HERE, ρ_r = 0.3 AND ρ_s = 0.2

Algorithm	k_{max}	$\frac{\|\tilde{\mathbf{L}}-\mathbf{L}^0\|_F}{\|\mathbf{L}^0\|_F}$	$\text{rank}(\tilde{\mathbf{L}})$	$\|\tilde{\mathbf{S}}\|_0$	# iter	Recovered?
$n = 400$, $\text{rank}(\mathbf{L}^0) = 120$, $\|\mathbf{S}^0\|_0 = 32,000$						
PCP		0.4756	226	102,391	34	×
Reweighted ℓ_1	1	0.2742	382	32,000	70	×
Reweighted ℓ_1	2	0.2711	383	32,000	106	×
NSVT	1	$\mathbf{4.43 \times 10^{-5}}$	151	32,002	100	√
NSVT	2	$\mathbf{4.85 \times 10^{-6}}$	**120**	**32,000**	178	√
$n = 800$, $\text{rank}(\mathbf{L}^0) = 240$, $\|\mathbf{S}^0\|_0 = 128,000$						
PCP		0.4680	456	405,036	34	×
Reweighted ℓ_1	1	0.2439	760	128,000	70	×
Reweighted ℓ_1	2	0.2413	761	128,000	106	×
NSVT	1	$\mathbf{8.44 \times 10^{-6}}$	277	128,000	93	√
NSVT	2	$\mathbf{6.55 \times 10^{-6}}$	**240**	**128,000**	164	√
$n = 1200$, $\text{rank}(\mathbf{L}^0) = 360$, $\|\mathbf{S}^0\|_0 = 288,000$						
PCP		0.4688	679	922,440	34	×
Reweighted ℓ_1	1	0.2196	1138	288,000	70	×
Reweighted ℓ_1	2	0.2174	1137	288,000	106	×
NSVT	1	$\mathbf{5.68 \times 10^{-6}}$	406	288,000	90	√
NSVT	2	$\mathbf{7.97 \times 10^{-6}}$	**360**	**288,000**	160	√

independently and uniformly distributed in the range $[-500, 500]$ [20].

2) Data model 2: Bernoulli sparse matrices $\mathbf{S}^0$ with random signs: with entries taking value 0 with probability $1-\rho_s$ and taking value ± 1 with probability $\rho_s/2$ on a random support [4].
3) Data model 3: Bernoulli sparse matrices $\mathbf{S}^0$ with coherent signs: with support Ω chosen uniformly at random and nonzero entries taking value $\mathbf{S}^0 = \mathcal{P}_\Omega\left(\text{sign}(\mathbf{L}^0)\right)$ [4].

In the experiments, only if the recovered $\tilde{\mathbf{L}}$ satisfies $\frac{\|\tilde{\mathbf{L}}-\mathbf{L}^0\|_F}{\|\mathbf{L}^0\|_F} \leq 10^{-4}$ (10^{-3} in [4]), the matrix is considered to be recovered. The experiments are repeated 10 times independently.

We list the recovery results using different methods on different data models in Tables I–III. The data parameters and typical total iteration numbers (adding together all the number of SVDs) are also listed in the tables. As we can see from Tables I–III, NSVT can still recover matrices even when both PCP and reweighted ℓ_1 fail. Only one or two reweighting iterations can already give great improvements on matrix recovery. Concerning the iteration number, we check the iterations for each loop of reweighting (Step 2 in Algorithm 1), i.e. the number of SVDs for Algorithm 2 to converge. We observe that the Reweighed ℓ_1 method uses almost the same iterations (around 35 iterations) as the PCP method to converge on all the three kinds of data models. While, the iteration numbers of the NSVT method vary on different types of data models. On data model 1, the NSVT method needs similar iterations as the PCP and reweighted ℓ_1 methods. On data model 2 and 3, the NSVT method needs almost as many as twice iterations of the PCP and reweighted ℓ_1 methods.

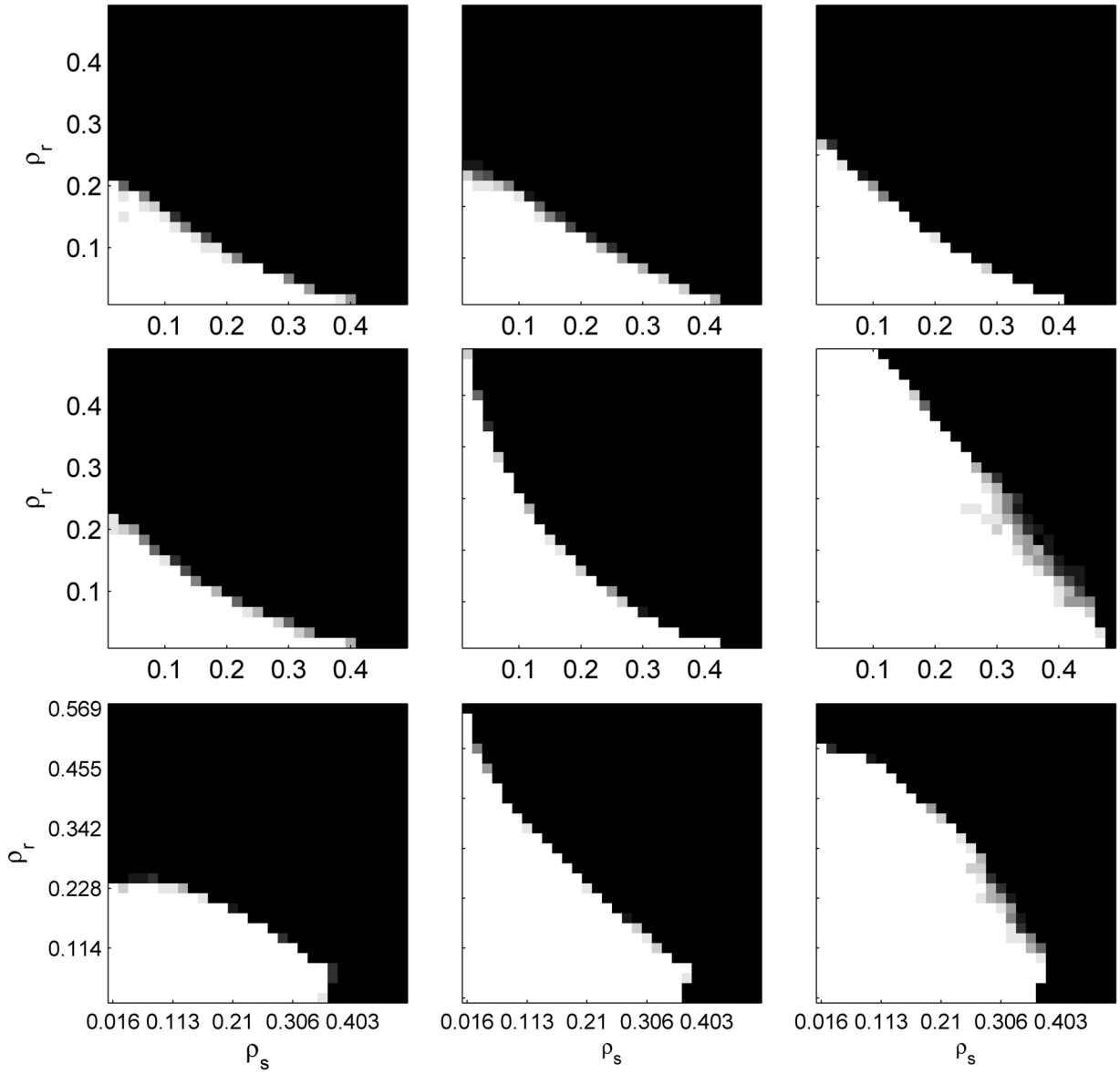


Fig. 2. Correct recovery in rank and sparsity on data model 1 (top row), 2 (middle row), and 3 (bottom row), by PCP (left column), reweighted ℓ_1 (middle column), and NSVT (right column), respectively.

Fig. 2 plots the fraction of correct recoveries for varying (ρ_r, ρ_s) on data model 1, 2, and 3, respectively. Square matrices of dimension $m = n = 400$ are considered in the experiments. We set $k_{max} = 1$ for data model 1 and 3, and set $k_{max} = 2$ for data model 2 for both reweighted ℓ_1 method and our proposed NSVT method. As we can see from Fig. 2, our proposed NSVT algorithm improves the correct recovery region a lot (the white region represents the correct recovery region). This is because the weighted nuclear norm and ℓ_1 norm prompt low rank and sparsity respectively. Although both reweighted ℓ_1 and NSVT help to improve performance of recovering matrices, the NSVT always has the significantly largest region of correct recoveries, especially on data model 2 and 3. For instance, NSVT can still recover a badly corrupted matrix even when $\rho_r + \rho_s$ is up to 0.9 on data model 2. We also observe the phenomenon that the ability of improving recovery performance is different across different data models.

B. Single Image Restoration

We show the application of proposed method on single image restoration. Most natural images are approximately low-rank, as can bee seen from Figs. 1(a) and 3(a), for example. Notice that, these two images contain different scenes: Fig. 1(a) shows a portrait of a woman, and Fig. 3(a) shows a scene of bridge and lighthouse. Considering an image as a matrix, Figs. 1(a) and 3(a) show the singular value distribution obtained by SVD of the images, which are shown in Figs. 1(b) and 3(b), respectively. From the distribution of singular values, we observe that the singular values decay very fast, which indicates the low-rank structure of an image. The low-rank prior is used for restoring images from corruptions. In the experiment, we set $k_{max} = 1$ for both reweighted ℓ_1 method and our proposed NSVT method. The original images of 515×512 pixels are shown in Figs. 1(b) and 3(b). The input images to be restored are corrupted by gross

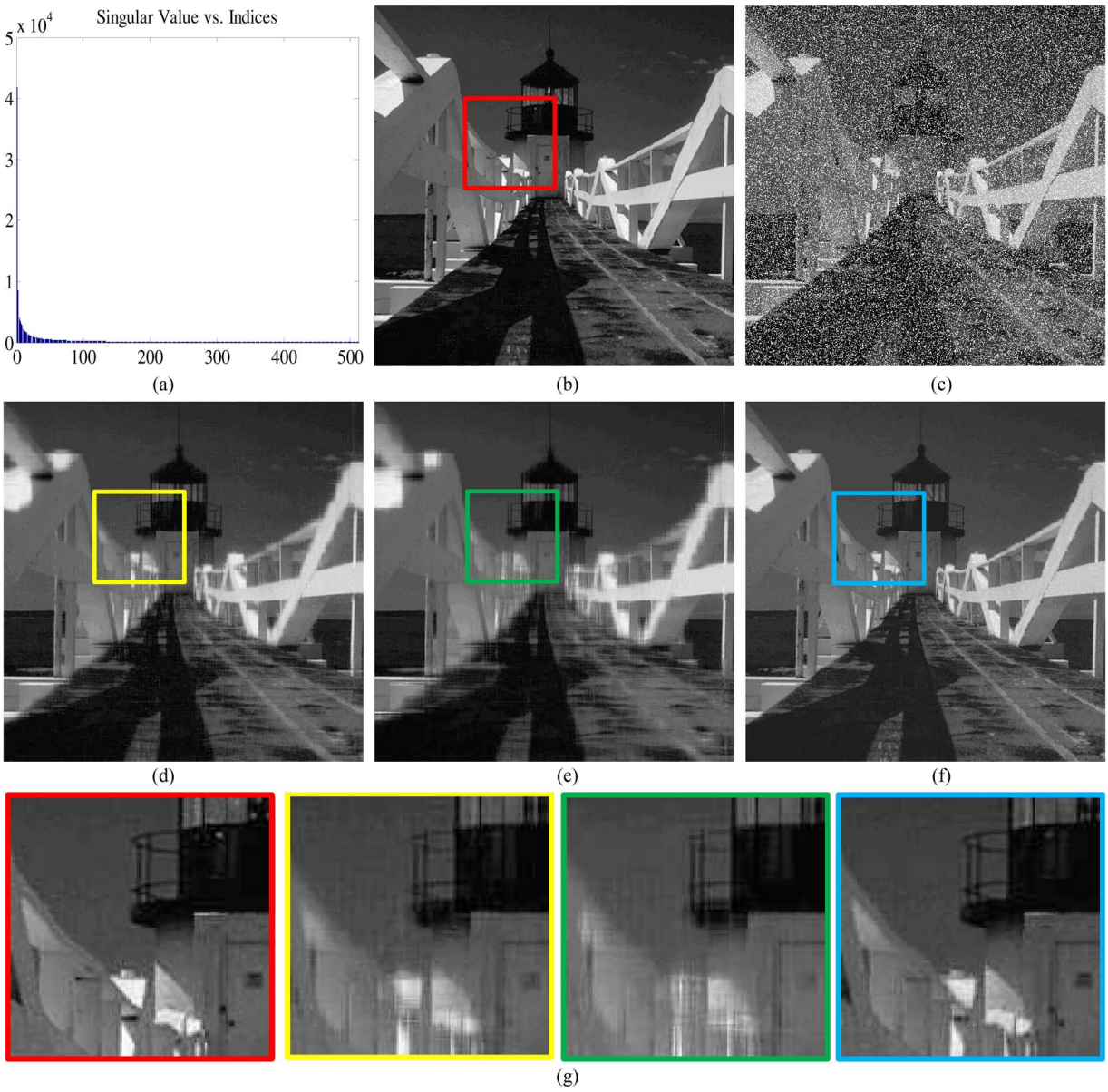


Fig. 3. Restoration of a single image, 20% of whose pixels are corrupted by large noise. (a) Singular value distribution of the original image shown in (b). (c) Corrupted image to be restored. 3(d)–(f) Restored images obtained by PCP, reweighted ℓ_1, and NSVT, respectively. (g) Close-up patches of (b) and (d)–(f), respectively. (a) Singular value distribution. (b) Original. (c) Input (9.99 dB/0.057). (d) PCP (29.76 dB/0.869). (e) Reweighted ℓ_1 (26.35 dB/0.837). (f) NSVT (34.84 dB/0.938). (g) Close-up comparison.

sparse error on 20% pixels randomly as shown in Figs. 1(c) and 3(c). We hope to restore images from those corrupted images. Note that, this is different from low-rank matrix completion or tensor completion for restoring visual data from missing values [33], [34]. Therein, the locations of missing values are known. While, in our problem, the observation is corrupted by gross errors with unknown positions, which is generally an even more challenging problem. Since we do not know positions of the corrupted pixels in advance, we formulate the image restoration as recovering a low-rank matrix (representing the uncorrupted image) from corrupted observations. We show the recovered images obtained by PCP, reweighted ℓ_1, and NSVT in Figs. 1 and 3(d)–(f), respectively. To have a close-up observation, we also show a 128×128 patch of Figs. 1 and 3(b), and (d)–(f), in Figs. 1 and 3(g). We can observe that much more appealing results are obtained by NSVT. This is because low rank and sparsity are enhanced simultaneously and in a balanced manner in our algorithm, which helps to capture data structure.

To achieve statistical significance, more experiments are performed on various natural images.[2] We use both peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [35] to evaluate performances of different methods. The results are summarized in Fig. 4. Our proposed method consistently achieves best performance among the three methods both in terms of PSNR and SSIM index. Notice that the restoration performance differs among the test images. For instance, the

[2]The original images are downloaded from: http://decsai.ugr.es/cvg/CG/base.htm .

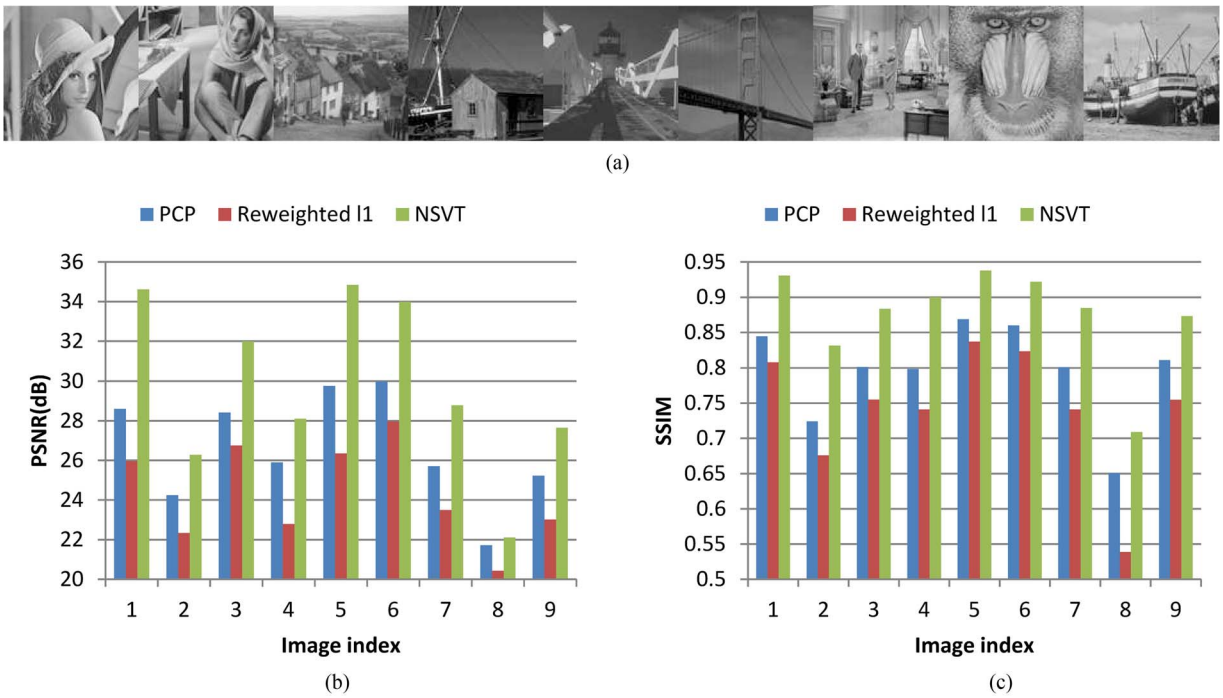


Fig. 4. Statistical evaluation of robust image restoration performance using PCP, reweighted ℓ_1, and NSVT, respectively. (a) Test images. (b) and (c) PSNR and SSIM of the restored images with respect to the original images.

TABLE III

COMPARISONS BETWEEN PCP, REWEIGHTED ℓ_1, AND NSVT FOR LOW-RANK MATRIX RECOVERY ON DATA MODEL 3. HERE, $\rho_r = 0.3$ AND $\rho_s = 0.2$

Algorithm	k_{max}	$\frac{\|\tilde{\mathbf{L}}-\mathbf{L}^0\|_F}{\|\mathbf{L}^0\|_F}$	$\text{rank}(\tilde{\mathbf{L}})$	$\|\tilde{\mathbf{S}}\|_0$	# iter	Recovered?
$n = 400$, $\text{rank}(\mathbf{L}^0) = 120$, $\|\mathbf{S}^0\|_0 = 32,000$						
PCP		0.1391	221	97,674	35	×
Reweighted ℓ_1	1	0.0451	365	32,000	71	×
Reweighted ℓ_1	2	0.0437	364	32,000	107	×
NSVT	1	$\mathbf{4.89 \times 10^{-6}}$	**120**	**32,000**	106	√
NSVT	2	$\mathbf{4.84 \times 10^{-6}}$	**120**	**32,000**	177	√
$n = 800$, $\text{rank}(\mathbf{L}^0) = 240$, $\|\mathbf{S}^0\|_0 = 128,000$						
PCP		0.1338	441	389,076	35	×
Reweighted ℓ_1	1	0.0419	721	128,000	71	×
Reweighted ℓ_1	2	0.0409	719	128,000	107	×
NSVT	1	$\mathbf{6.50 \times 10^{-6}}$	**240**	**128,000**	104	√
NSVT	2	$\mathbf{6.30 \times 10^{-6}}$	**240**	**128,000**	173	√
$n = 1200$, $\text{rank}(\mathbf{L}^0) = 360$, $\|\mathbf{S}^0\|_0 = 288,000$						
PCP		0.1350	659	874,315	35	×
Reweighted ℓ_1	1	0.0409	1076	288,000	71	×
Reweighted ℓ_1	2	0.0393	1075	288,000	107	×
NSVT	1	$\mathbf{8.04 \times 10^{-6}}$	**360**	**288,000**	103	√
NSVT	2	$\mathbf{7.83 \times 10^{-6}}$	**360**	**288,000**	165	√

performance on image baboon (index 8) is lower, because its image structure is quite special due to the furry fur and thus the rank of original image matrix is high. In such a case challenging all of the three methods, we still get apparently superior performance.

Note that, our proposed method is different from the video restoration method via joint sparse and low-rank matrix approximation in [9] or the robust video denoising method using low-rank matrix completion in [36]. In their work, the low-rank structure of similar image patches is harnessed, while, our method utilizes the holistic low-rank structure of images.

C. Hyperspectral Image Restoration

Hyperspectral images are well-localized in wavelength and they represent the scene radiance much more faithfully comparing with traditional RGB images [37]. By stacking image of each frequency band into a vector and arranging all the image vectors into a matrix, a hyperspectral image is represented as a data matrix $\mathbf{H} \in \mathbb{R}^{m \times n}$ with m pixels and n frequency bands. Low-rank assumption of a hyperspectral image assumes that a hyperspectral image can be well explained by nonnegative linear combination of an endmember matrix $\mathbf{B} \in \mathbb{R}^{m \times k}_{++}$, where $k < \min(m, n)$, i.e., $\mathbf{H} = \mathbf{BC}$,

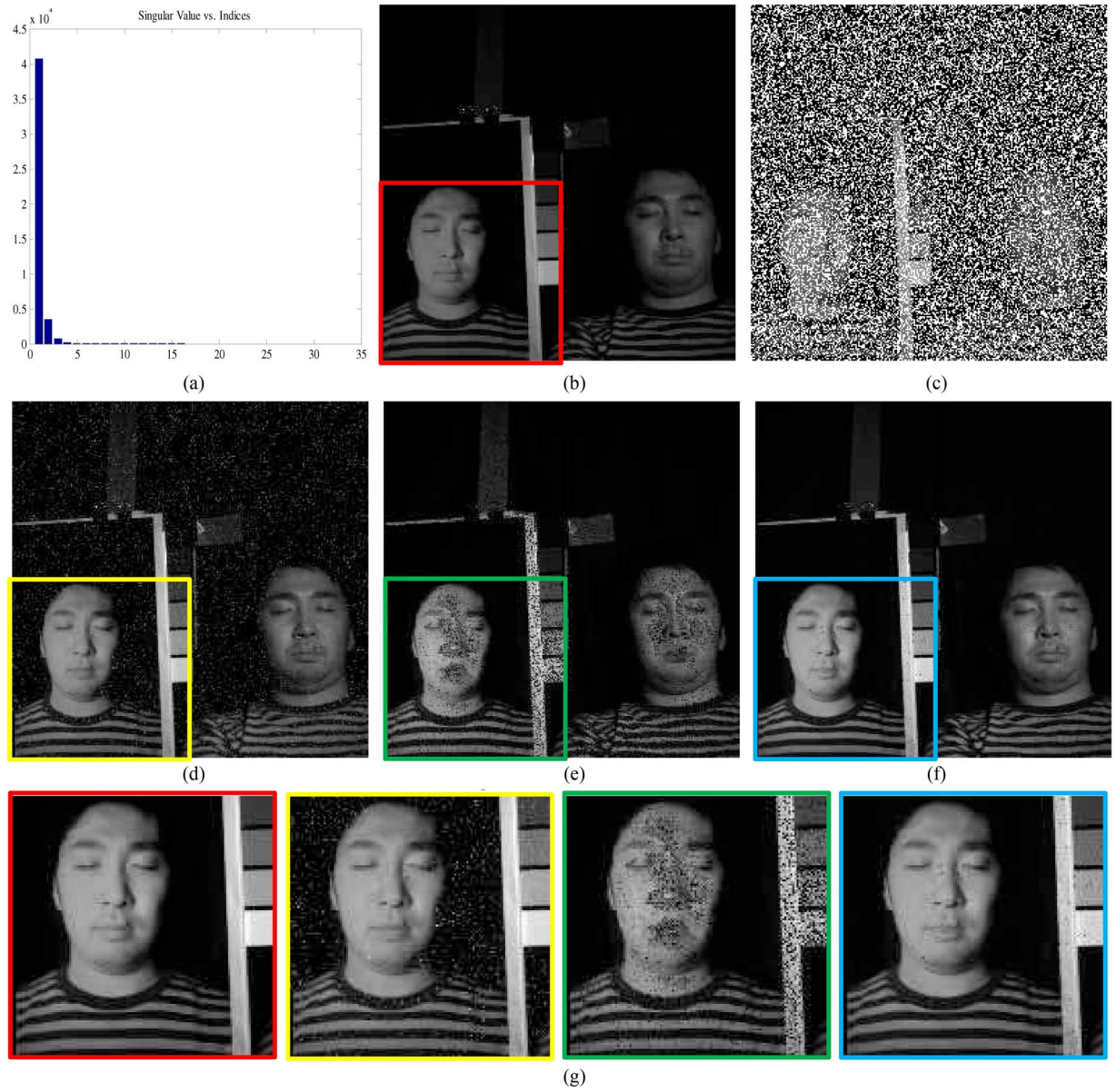


Fig. 5. Restoration of a hyperspectral image *Photo and Face*, 40% of whose pixels are corrupted by large noise. (a) Singular value distribution of the original hyperspectral image, an image from one of whose frequency bands is shown in (b). (c) Corresponding corrupted image to be restored. (d)–(f) Restored images obtained by PCP, reweighted ℓ_1, and NSVT, respectively. (g) Close-up patches of (b) and (d)–(f), respectively. (a) Singular value distribution. (b) Original. (c) Input. (d) PCP (28.95 dB/0.553). (e) Reweighted ℓ_1 (26.48 dB/0.886). (f) NSVT (38.13 dB/0.985). (g) Close-up comparison.

where $\mathbf{C} \in \mathbb{R}_+^{k\times n}$ is a nonnegative abundance matrix [38]. For instance, Fig. 5(a) shows the singular value distribution obtained by SVD of a hyperspectral image, one image of whose frequency bands is shown in Fig. 5(b). The image contains a face image of a man and his photo. There are 31 frequency bands in total and an image of each frequency band is of 256×256 pixels, i.e., $\mathbf{H} \in \mathbb{R}^{65536\times 31}$. One can clearly observe the low-rank structure as the first few singular values are much larger than the remains. We show in this subsection that how to restore hyperspectral image from corrupted observations using the low-rank prior. The 40% pixels of a hyperspectral image to be restored are corrupted by large errors randomly. Fig. 5(c) shows such a corrupted image of its one frequency band, which is very difficult for a human to tell what is in the image. We use the low-rank prior to restore a hyperspectral image from severe corruptions. We show the recovered results obtained by PCP, reweighted ℓ_1, and NSVT in Fig. 5 (d)–(f), respectively. We also show a 128×128 patch of Fig. 5(b), and (d)–(f) in Fig. 5(g) to have a close-up observation. We set $k_{max} = 1$ for both reweighted ℓ_1 and our proposed NSVT. The restored image obtained by PCP still contains a lot of large noise (the white dots), as can be seen from Fig. 5(d). Although reweighted ℓ_1 relieves the errors in some part of the image comparing to PCP, it brings in annoying noise in other part. Again, the NSVT method gives the best restoration result among the three methods. Another example is given in Fig. 6, which shows the hyperspectral image of a piece of clothes with rich texture. These experimental results demonstrate the good performance of the NSVT algorithm.

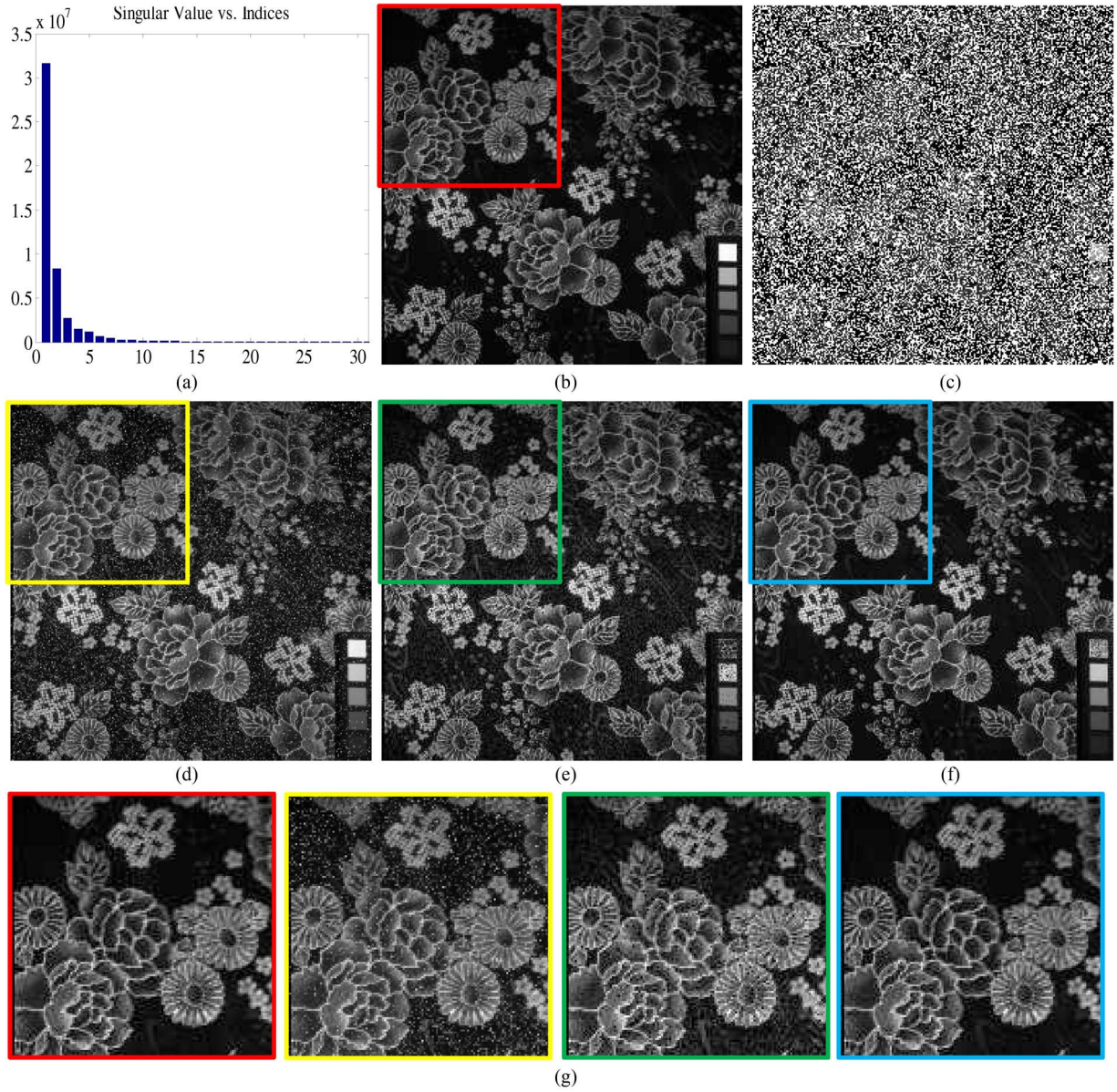


Fig. 6. Restoration of a hyperspectral image *Cloth*, 40% of whose pixels are corrupted by large noise. (a) Singular value distribution of the original hyperspectral image, an image from one of whose frequency bands is shown (b). (c) Corresponding corrupted image to be restored. (d)–(f) are restored images obtained by PCP, reweighted ℓ_1, and NSVT, respectively. (g) Close-up patches of (b) and (d)–(f), respectively. (a) Singular value distribution. (b) Original. (c) Input. (d) PCP (29.50 dB/0.744). (e) Reweighted ℓ_1 (30.17 dB/0.892). (f) NSVT (36.29 dB/0.994). (g) Close-up comparison.

D. Video Background Modeling From Corrupted Observations

The correlation between frames makes low-rank modeling very natural for videos. In surveillance video, by modeling background variations as approximately low-rank matrix and foreground objects as sparse errors, the PCP algorithm has been successfully applied in background modeling [4]. Here, we consider an even more challenging situation: the video observations are corrupted by gross errors, as shown in the second column of Fig. 7. To be more specific, the first column of Fig. 7 shows three original frames from a surveillance video.[3] Instead of observing the original video, the observations shown in the second column are corrupted by gross errors randomly on 40% pixels. We stack each frame into a vector and form the entire video as a matrix. Again, we model the background as approximately low-rank matrix and use sparse matrix to model both the foreground objects and gross errors. In the third, fourth, and fifth columns of Fig. 7, we show the corresponding columns of the recovered low-rank matrix, i.e., the recovered background, obtained by PCP, reweighted ℓ_1, and NSVT, respectively. Due to the severe corruptions in the observations, although the PCP helps to recover the background, there are still apparent errors (random white dots in the images). The reweighted ℓ_1 introduces extra errors (random black dots in the images) as it does not consider low-rank component and sparse component appreciatively. The NSVT gives the most appealing background modeling results among the three methods despite of severe gross errors.

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

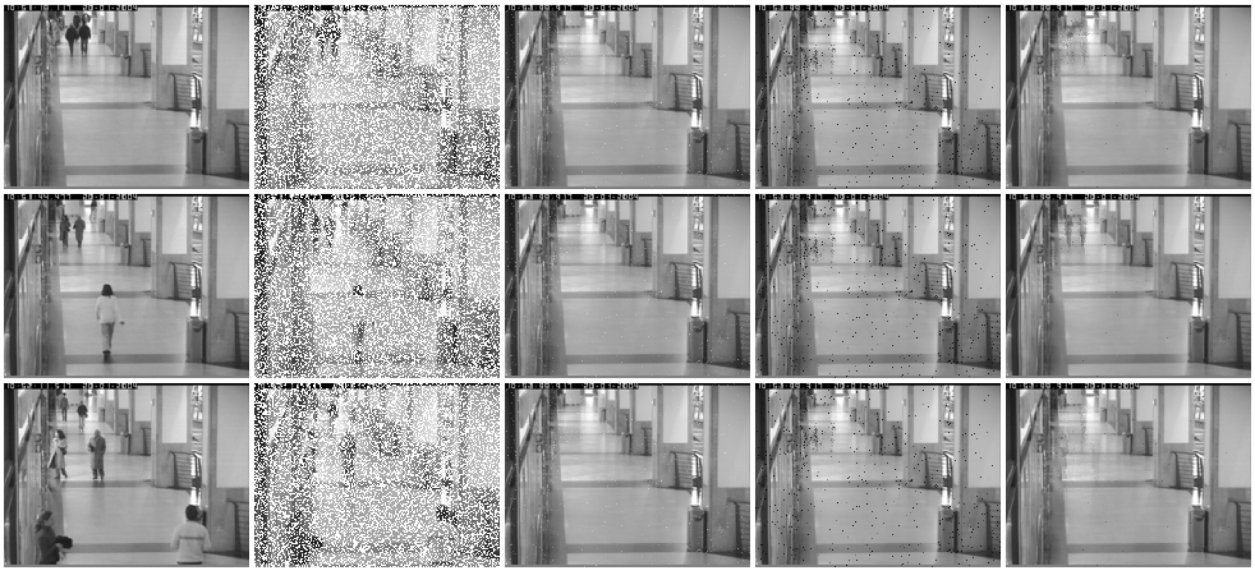


Fig. 7. Video background modeling from corrupted observations, 40% of whose pixels are corrupted by large noise. There are 158 frames in total and the size of each frame is 144 × 172-pixel. Column 1 shows three frames from the original video, and Column 2 shows the corrupted observations. Columns 3, 4, and 5 are the corresponding background frames restored by PCP, reweighted ℓ_1, and NSVT, respectively. Here, frames 1, 45, and 90 are shown in rows 1, 2, and 3, respectively.

V. Conclusion

In this paper, we proposed a reweighted low-rank matrix recovery method and showed its application in different robust image restoration problems. An NSVT operator is used for enhancing low rank property of a matrix. Significant improvements on recovery of corrupted low-rank matrix and image/video restoration experiment are observed, outperforming the original PCP optimization and reweighted ℓ_1 minimization to a board extend. In light of promise of NSVT, deeper investigations of the algorithm remain: How to adaptively choose the parameters such as $\epsilon_{\mathbf{L}}$ and $\epsilon_{\mathbf{S}}$ to further improve the performance is considered as one of the future work. Theoretical analysis of the proposed method, such as the global convergence, is one of the important future research directions.

Acknowledgment

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

References

[1] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[2] E. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[3] B. Recht, M. Fazel, and P. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.

[4] E. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, pp. 1–37, 2011.

[5] K. Zhao and Z. Zhang, "Successively alternate least square for low-rank matrix factorization with bounded missing data," *Comput. Vision Image Understand.*, vol. 114, no. 10, pp. 1084–1096, Oct. 2010.

[6] K. Li, Q. Dai, W. Xu, J. Yang, and J. Jiang, "Three-dimensional motion estimation via matrix completion," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 539–551, Apr. 2012.

[7] X. Lu, T. Gong, P. Yan, Y. Yuan, and X. Li, "Robust alternative minimization for matrix completion," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 3, pp. 939–949, Jun. 2012.

[8] K. Konishi and T. Furukawa, "A nuclear norm heuristic approach to fractionally spaced blind channel equalization," *IEEE Signal Process. Lett.*, vol. 18, no. 1, pp. 59–63, Jan. 2011.

[9] H. Ji, S. Huang, Z. Shen, and Y. Xu, "Robust video restoration by joint sparse and low rank matrix approximation," *SIAM J. Imag. Sci.*, vol. 4, no. 4, pp. 1122–1142, 2011.

[10] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma., "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2233–2246, Nov. 2012.

[11] Z. Zhang, A. Ganesh, X. Liang, and Y. Ma, "TILT: Transform-invariant low-rank textures," *Int. J. Comput. Vision*, vol. 99, no. 1, pp. 1–24, Aug. 2012.

[12] Z. Zhang, Y. Matsushita, and Y. Ma, "Camera calibration with lens distortion from low-rank textures," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2011, pp. 2321–2328.

[13] Z. Zhang, X. Liang, and Y. Ma, "Unwrapping low-rank textures on generalized cylindrical surfaces," in *Proc. Int. Conf. Comput. Vision*, 2011, pp. 1347–1354.

[14] H. Mobahi, Z. Zhou, A. Y. Yang, and Y. Ma, "Holistic reconstruction of urban structures from low-rank textures," in *Proc. ICCV workshop*, 2011, pp. 593–600.

[15] C. Yao, Z. Tu, and Y. Ma, "Detecting texts of arbitrary orientations in natural images," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2012, pp. 1083–1090.

[16] X. Liang, X. Ren, Z. Zhang, and Y. Ma, "Repairing sparse low-rank texture," in *Proc. Eur. Conf. Comput. Vision*, 2012, pp. 482–495.

[17] S. G. Lingala, Y. Hu, E. DiBella, and M. Jacob, "Accelerated dynamic MRI exploiting sparsity and low rank structure: k-t SLR," *IEEE Trans. Med. Imag.*, vol. 30, no. 5, pp. 1042–1054, May 2011.

[18] J. Cai, E. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optimiz.*, vol. 20, no. 4, pp. 1956–1982, 2010.

[19] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems," *Pacific J. Optimiz.*, vol. 6, no. 15, pp. 615–640, 2010.

[20] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," Mathematical Programming, UIUC Tech. Rep. UILU-ENG-09-2215, 2009.

[21] X. Yuan and J. Yang, "Sparse and low-rank matrix decomposition via alternating direction methods," *Pacific J Optimiz.*, vol. 9, no. 1, pp. 167–180, 2013.

[22] I. Daubechies, R. DeVore, M. Fornasier, and S. Gunturk, "Iteratively re-weighted least squares minimization for sparse recovery," *Commun. Pure Appl. Math.*, vol. 63, no. 1, pp. 1–38, 2010.
[23] M. Fornasier, H. Rauhut, and R. Ward, "Low rank matrix recovery via iteratively reweighted least squares minimization," *SIAM J. Optimiz.*, vol. 21, no. 4, pp. 1614–1640, 2011.
[24] K. Mohan and M. Fazel, "Iterative reweighted least squares for matrix rank minimization," in *Proc. Allerton Conf. Commun., Control, Comput.*, 2010, pp. 653–661.
[25] M. Fazel, H. Hindi, and S. Boyd, "Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices," in *Proc. ACC*, 2003, pp. 2156–2162.
[26] K. Mohan and M. Fazel, "Reweighted nuclear norm minimization with application to system identification," in *Proc. ACC*, 2010, pp. 2953–2959.
[27] Y. Deng, Q. Dai, R. Liu, Z. Zhanga, and S. Hu, "Low-rank structure learning via nonconvex heuristic recovery," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 24, no. 3, pp. 383–396, Mar. 2013.
[28] Y. Hu, S. Goud, and M. Jacob, "A fast majorize-minimize algorithm for the recovery of sparse and low rank matrices," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 742–753, Feb. 2012.
[29] E. Candès, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Applicat.*, vol. 14, no. 5, pp. 877–905, 2007.
[30] W. Xu, M. A. Khajehnejad, S. Avestimehr, and B. Hassibi, "Breaking through the thresholds: An analysis for iterative reweighted ℓ_1 minimization via the grassmann angle framework," in *Proc. ICASSP*, 2010, pp. 5498–5501.
[31] Y. Deng, R. Liu, Q. Dai, and Z. Zhang, "Reweighted scheme for low-rank structure learning via log-sum heuristic recovery," *arXiv:1012.1919*, 2011, pp. 1–13.
[32] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, "An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–695, Mar. 2011.
[33] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," in *Proc. IEEE Int. Conf. Comput. Vision*, 2009, pp. 2114–2121.
[34] J. Huang, S. Zhang, H. Li, and D. Metaxas, "Composite splitting algorithms for convex optimization," *Comput. Vision Image Understand.*, vol. 115, no. 12, p. 1610–1622, Dec. 2011.
[35] 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.
[36] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 1791–1798.
[37] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar, "Generalized assorted pixel camera: Postcapture control of resolution, dynamic range, and spectrum," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2241–2253, Sep. 2010.
[38] A. Zymnis, S.-J. Kim, J. Skaf, M. Parente, and S. Boyd, "Hyperspectral image unmixing via alternating projected subgradients," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2007, pp. 1164–1168.

Yigang Peng received the bachelor's degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2007, and the Ph.D. degree from Tsinghua University, Beijing, in 2012.

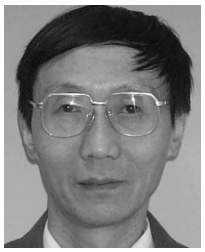
He is currently an Engineer with the National Computer Network Emergency Response Technical Team Coordination Center of China, Beijing. His current research interests include computer vision, image processing, and computer network.

Jinli Suo received the bachelor's degree from Shandong University, Shandong, China, in 2004, and the Ph.D. degree from the Graduate University of Chinese Academy of Sciences, Beijing, China, in 2010.

She is currently a Lecturer with the Department of Automation, Tsinghua University, Beijing. Her current research interests include computational photography and computer vision.

Qionghai Dai (SM'05) received the Ph.D. degree from Northeastern University, Boston, MA, USA.

He is currently a Professor with the Department of Automation, Tsinghua University, Beijing, China. His current research interests include signal processing, computer vision and graphics, video processing and communications, and computational photography.



Wenli Xu received the bachelor's and master's degrees from Tsinghua University, Beijing, China, in 1970 and 1980, respectively, and the Ph.D. degree from the University of Colorado, Boulder, CO, USA, in 1990.

He is currently a Professor with the Department of Automation, Tsinghua University. His current research interests include automatic control and computer vision.