Int J Comput Vis (2017) 121:183–208
DOI 10.1007/s11263-016-0930-5

Weighted Nuclear Norm Minimization and Its Applications to Low Level Vision

Shuhang Gu[1] · Qi Xie[2] · Deyu Meng[2] · Wangmeng Zuo[3] · Xiangchu Feng[4] · Lei Zhang[1]

Received: 15 April 2015 / Accepted: 5 July 2016 / Published online: 18 July 2016
© Springer Science+Business Media New York 2016

Abstract As a convex relaxation of the rank minimization model, the nuclear norm minimization (NNM) problem has been attracting significant research interest in recent years. The standard NNM regularizes each singular value equally, composing an easily calculated convex norm. However, this restricts its capability and flexibility in dealing with many practical problems, where the singular values have clear physical meanings and should be treated differently. In this paper we study the weighted nuclear norm minimization (WNNM) problem, which adaptively assigns weights on different singular values. As the key step of solving general WNNM models, the theoretical properties of the weighted nuclear norm proximal (WNNP) operator are investigated. Albeit nonconvex, we prove that WNNP is equivalent to a standard quadratic programming problem with linear constrains, which facilitates solving the original problem with off-the-shelf convex optimization solvers. In particular, when the weights are sorted in a non-descending order, its optimal solution can be easily obtained in closed-form. With WNNP, the solving strategies for multiple extensions of WNNM, including robust PCA and matrix completion, can be readily constructed under the alternating direction method of multipliers paradigm. Furthermore, inspired by the reweighted sparse coding scheme, we present an automatic weight setting method, which greatly facilitates the practical implementation of WNNM. The proposed WNNM methods achieve state-of-the-art performance in typical low level vision tasks, including image denoising, background subtraction and image inpainting.

Keywords Low rank analysis · Nuclear norm minimization · Low level vision

Communicated by Jean-Michel Morel.

✉ Lei Zhang
cslzhang@comp.polyu.edu.hk

Shuhang Gu
cssgu@comp.polyu.edu.hk

Qi Xie
xq.liwu@stu.xjtu.edu.cn

Deyu Meng
dymeng@mail.xjtu.edu.cn

Wangmeng Zuo
wmzuo@hit.edu.cn

Xiangchu Feng
xcfeng@mail.xidian.edu.cn

1 Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China

2 School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an, China

3 School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

4 Department of Applied Mathematics, Xidian University, Xi'an, China

1 Introduction

Low rank matrix approximation (LRMA), which aims to recover the underlying low rank matrix from its degraded observation, has a wide range of applications in computer vision and machine learning. For instance, human facial images can be modeled as reflections of a Lambertian object and approximated by a low dimensional linear subspace; this low rank nature leads to a proper reconstruction of a face model from occluded/corrupted face images (De La Torre and Black 2003; Liu et al. 2010; Zheng et al. 2012). In recommendation system, the LRMA approach achieves outstanding performance on the celebrated Netflix compe-

tition, whose low-rank insight is based on the fact that the customers′ choices are mostly affected by only a few common factors (Ruslan and Srebro 2010). In background subtraction, the video clip captured by a surveillance camera is generally taken under a static scenario in background and with relatively small moving objects in foreground over a period, naturally resulting in its low-rank property; this inspires various effective techniques on background modeling and foreground object detection in recent years (Wright et al. 2009; Mu et al. 2011). In image processing, it has also been shown that the matrix formed by nonlocal similar patches in a natural image is of low rank; such a prior knowledge benefits the image restoration tasks (Wang et al. 2012). Owing to the rapid development of convex and nonconvex optimization techniques in past decades, there are a flurry of studies in LRMA, and many important models and algorithms have been reported (Srebro et al. 2003; Buchanan and Fitzgibbon 2005; Ke and Kanade 2005; Eriksson and Van Den Hengel 2010; Fazel et al. 2001; Wright et al. 2009; Candès and Recht 2009; Cai et al. 2010; Candès et al. 2011; Lin et al. 2011).

The current development of LRMA can be categorized into two categories: the low rank matrix factorization (LRMF) approaches and the rank minimization approaches. Given a matrix $\boldsymbol{Y} \in \Re^{m\times n}$, LRMF aims to factorize it into two smaller ones, $\boldsymbol{A} \in \Re^{m\times k}$ and $\boldsymbol{B} \in \Re^{n\times k}$, such that their product $\boldsymbol{AB}^T$ can reconstruct $\boldsymbol{Y}$ under certain fidelity loss functions. Here $k < min(m, n)$ ensures the low-rank property of the reconstructed matrix $\boldsymbol{AB}^T$. A variety of LRMF methods have been proposed, including the classical singular value decomposition (SVD) under ℓ_2-norm loss, robust LRMF methods under ℓ_1-norm loss, and many probabilistic methods (Srebro et al. 2003; Buchanan and Fitzgibbon 2005; Ke and Kanade 2005; Mnih and Salakhutdinov 2007; Kwak 2008; Eriksson and Van Den Hengel 2010).

Rank minimization methods represent another main branch along this line of research. These methods reconstruct the data matrix through imposing an additional rank constraint upon the estimated matrix. Since direct rank minimization is NP hard and is difficult to solve, the problem is generally relaxed by substitutively minimizing the nuclear norm of the estimated matrix, which is a convex relaxation of minimizing the matrix rank (Fazel 2002). This methodology is called as nuclear norm minimization (NNM). The nuclear norm of a matrix $\boldsymbol{X}$, denoted by $\|\boldsymbol{X}\|_*$, is defined as the sum of its singular values, i.e., $\|\boldsymbol{X}\|_* = \sum_i \sigma_i(\boldsymbol{X})$, where $\sigma_i(\boldsymbol{X})$ denotes the i-th singular value of $\boldsymbol{X}$. The NNM approach has been attracting significant attention due to its rapid development in both theory and implementation. On one hand, (Candès et al. 2011) proved that from the noisy input, its intrinsic low-rank reconstruction can be exactly achieved with a high probability through solving an NNM problem. On the other hand, (Cai et al. 2010) proved that the nuclear norm proximal (NNP) problem

$$\hat{\boldsymbol{X}} = \mathbf{prox}_{\lambda\|\cdot\|_*}(\boldsymbol{Y}) = \arg\min_{\boldsymbol{X}} \|\boldsymbol{Y} - \boldsymbol{X}\|_F^2 + \lambda\|\boldsymbol{X}\|_* \tag{1}$$

can be easily solved in closed-form by imposing a soft-thresholding operation on the singular values of the observation matrix:

$$\hat{\boldsymbol{X}} = \boldsymbol{U}\mathcal{S}_{\frac{\lambda}{2}}(\boldsymbol{\Sigma})\boldsymbol{V}^T, \tag{2}$$

where $\boldsymbol{Y} = \boldsymbol{U\Sigma V}^T$ is the SVD of $\boldsymbol{Y}$ and $\mathcal{S}_{\frac{\lambda}{2}}(\boldsymbol{\Sigma})$ is the soft-thresholding function on diagonal matrix $\boldsymbol{\Sigma}$ with parameter $\frac{\lambda}{2}$. For each diagonal element $\boldsymbol{\Sigma}_{ii}$ in $\boldsymbol{\Sigma}$, there is

$$\mathcal{S}_{\frac{\lambda}{2}}(\boldsymbol{\Sigma})_{ii} = \max\left(\boldsymbol{\Sigma}_{ii} - \frac{\lambda}{2}, 0\right). \tag{3}$$

By utilizing NNP as the key proximal technique (Moreau 1965), many NNM-based models have been proposed in recent years (Lin et al. 2009; Ji and Ye 2009; Cai et al. 2010; Lin et al. 2011).

Albeit its success as aforementioned, NNM still has certain limitations. In traditional NNM, all singular values are treated equally and shrunk with the same threshold $\frac{\lambda}{2}$ as defined in (3). This, however, ignores the prior knowledge we often have on singular values of a practical data matrix. More specifically, larger singular values of an input data matrix quantify the information of its underlying principal directions. For example, the large singular values of a matrix of image similar patches deliver the major edge and texture information. This implies that to recover an image from its corrupted one, we should shrink less the larger singular values while shrink more the smaller ones. Clearly, traditional NNM model, as well as its corresponding soft-thresholding solvers, are not flexible enough to handle such issues.

To improve the flexibility of NNM, we propose the weighted nuclear norm and study its minimization strategy in this work. The weighted nuclear norm of a matrix $\boldsymbol{X}$ is defined as

$$\|\boldsymbol{X}\|_{\boldsymbol{w},*} = \sum_i w_i\sigma_i(\boldsymbol{X}), \tag{4}$$

where $\boldsymbol{w} = [w_1, \ldots, w_n]^T$ and $w_i \geq 0$ is a non-negative weight assigned to $\sigma_i(\boldsymbol{X})$. The weight vector will enhance the representation capability of the original nuclear norm. Rational weights specified based on the prior knowledge and understanding of the problem will benefit the corresponding weighted nuclear norm minimization (WNNM) model for achieving a better estimation of the latent data from the corrupted input. The difficulty of solving a WNNM model, however, lies in that it is non-convex in general cases, and the sub-gradient method (Cai et al. 2010) used for achieving the closed-form solution of an NNP problem is no longer

applicable. In this paper, we investigate in detail how to properly and efficiently solve such non-convex WNNM problem.

As the NNP operator to the NNM problem, the following weighted nulcear norm proximal (WNNP)[1] operator determines the general solving regime of the WNNM problem:

$$\hat{\boldsymbol{X}} = \mathbf{prox}_{\|\cdot\|_{\boldsymbol{w},*}}(\boldsymbol{Y}) = \arg\min_{\boldsymbol{X}} \|\boldsymbol{Y} - \boldsymbol{X}\|_F^2 + \|\boldsymbol{X}\|_{\boldsymbol{w},*}. \quad (5)$$

We prove that the WNNP problem can be equivalently transformed to a quadratic programming (QP) problem with linear constraints. This allows us to easily reach the global optimum of the original problem by using off-the-shelf convex optimization solvers. We further show that when the weights are non-descending, the global optimum of WNNP can be easily achieved in closed-form, i.e., by a so-called weighted soft-thresholding operator. Such an efficient solver makes it possible to utilize weighted nuclear norm in more complex applications. Particularly, we propose the WNNM-based robust principle component analysis (WNNM-RPCA) model and WNNM-based matrix completion (WNNM-MC) model, and solve these models by virtue of the WNNP solver. Furthermore, inspired by the previous developments of reweighted sparse coding, we present a rational scheme to automatically set the weights for the given data.

To validate the effectiveness of the proposed WNNM models, we test them on several typical low level vision tasks. Specifically, we first test the performance of the proposed WNNP model on image denoising. By utilizing the nonlocal self-similarity prior of images (Buades et al. 2008), the WNNP model achieves superior performance to state-of-the-art denoising algorithms. We perform the WNNM-RPCA model on the background subtraction application. Both the quantitative results and visual examples demonstrate the superiority of the proposed model beyond previous low-rank learning methods. We further apply WNNM-MC to the image inpainting task, and it also shows competitive results with state-of-the-art methods.

The contribution of this paper is three-fold. First, we prove that the non-convex WNNP problem can be equivalently transformed into a QP problem with liner constraints. This results in that the global optimum of the non-convex WNNP problem can be readily achieved by off-the-shelf convex optimization solvers. Furthermore, for non-descendingly ordered weights, we can get the optimal solution in closed-form by the weighted soft-thresholding operator, which greatly eases the computation for the problem. Second, we extend WNNP to WNNM-RPCA and WNNM-MC models to handle sparse noise/outlier and missing data, respectively. Attributed to the proposed WNNP, both WNNM-RPCA and WNNM-MC can be solved without introducing extra computation burden of traditional RPCA/MC algorithms. Third, we adapt the proposed WNNM models to different computer vision tasks. In image denoising, the WNNP model outperforms state-of-the-art methods in both PSNR index and visual perception. In background subtraction and image inpainting, the proposed WNNM-RPCA and WNNM-MC models also achieve better performance than traditional algorithms, especially, the low-rank learning methods designed for these tasks.

The rest of this paper is organized as follows. Section 2 briefly introduces some related work. Section 3 analyzes the optimization of the WNNM models, including the basic WNNP operator and more complex WNNM-RPCA and WNNM-MC ones. Section 4 applies the WNNM model to image denoising and compares the proposed algorithm with state-of-the-art denoising algorithms. Sections 5 and 6 apply the WNNM-RPCA and WNNM-MC to background subtraction and image inpainting, respectively. Section 7 discusses the proposed WNNM models and other singular value regularization methods from the viewpoint of weight setting. Section 8 concludes the paper.

2 Related Work

The main goal of LRMA is to recover the underlying low rank matrix from its degraded/corrupted observation. As the data from many practical problems possess intrinsic low-rank structure, the research on LRMA has achieved a great success in various applications of computer vision. Here we provide a brief review for the current advancement on this topic. A more comprehensive review can be found in a recent survey paper (Zhou et al. 2014).

The LRMA approach mainly includes two categories: the LRMF methods and the rank minimization methods. Given an input data matrix $\boldsymbol{Y} \in \Re^{m\times n}$, LRMF intends to find out the output $\boldsymbol{X}$ as a product of two smaller matrices $\boldsymbol{X} = \boldsymbol{A}\boldsymbol{B}^T$, where $\boldsymbol{A} \in \Re^{m\times k}$, $\boldsymbol{B} \in \Re^{n\times k}$, $k < m, n$. The most classical LRMF method is the well known SVD technique, which is capable of attaining the optimal rank-k approximation of $\boldsymbol{Y}$ by a truncation operator on its singular value matrix in terms of F-norm fidelity loss. To handle outliers/heavy noises mixed in data, multiple advanced robust LRMF methods have been proposed. (De La Torre and Black 2003) initiated a robust principal component analysis (RPCA) framework by utilizing the Geman-McClure function instead of the conventional ℓ_2-norm loss function. Beyond this non-convex loss function, a more commonly used one for robust LRMF is the ℓ_1-norm. In the seminal work of (Ke and Kanade 2005), Ke and Kanade designed an alternative convex programming algorithm to solve this ℓ_1 norm factorization model. More strategies have also been attempted for solving this model, e.g., PCAL1

[1] A general proximal operator is defined on a convex problem to guarantee an accurate projection. Although the problem here is nonconvex, we can strictly prove that it is equivalent to a convex quadratic programing problem in Sect. 3. We thus also call it a proximal operator throughout the paper for convenience.

(Kwak 2008) and L_1-Wiberg (Eriksson and Van Den Hengel 2010). Furthermore, to deal with the LRMF problem in the presence of missing entries, a series of matrix completion techniques have been proposed to estimate the low-rank matrix from only partial observations. The related investigations range from studies on optimization algorithms (Jain et et al. 2013) to new regularization strategies for highly ill-posed matrix completion tasks (Srebro et al. 2004). Besides these deterministic LRMF methods, there have been several probabilistic extensions of matrix factorizations. These probabilistic methods are capable of adapting different kinds of noises, such as dense noise with Gaussian distribution (Tipping and Bishop 1999; Mnih and Salakhutdinov 2007), sparse noise distribution (Wang and Yeung 2013; Babacan et al. 2012; Ding et al. 2011) and more complex mixture-of-Gaussian noise (Meng and Torre 2013).

The main idea of the rank minimization methods is to estimate the low-rank reconstruction by minimizing the rank of its relaxations upon it. Since directly minimizing the matrix rank corresponds to an NP-hard problem (Fazel 2002), the relaxation methods are more commonly utilized. The most popular choice is to replace the rank function with its tightest convex relaxation, the nuclear norm, leading to NNM-based methods. The superiority of these methods is two-fold. Firstly, the exact recovery property of some low rank recovery models have been theoretically proved (Candès and Recht 2009; Candès et al. 2011). Secondly, in implementation, the typical NNP problem along this line can be easily solved in closed-form by soft-thresholding the singular values of the input matrix (Cai et al. 2010). The recent advancement of NNM in applications such as RPCA and MC (Lin et al. 2009; Ji and Ye 2009; Cai et al. 2010; Lin et al. 2011) has been successfully utilized in many computer vision tasks (Ji et al. 2010; Liu et al. 2010; Zhang et al. 2012b; Liu et al. 2012; Peng et al. 2012).

As a convex surrogate of the rank function, the nuclear norm corresponds to the ℓ_1-norm on the singular values. While convexity simplifies the computation, some previous works have shown that nonconvex models may have better performance in applications such as compressive sensing (Chartrand 2007; Candès et al. 2008). Recently, researchers have attempted to substitute nuclear norm by certain nonconvex ameliorations to make it more adaptable to real scenarios. For instance, (Nie et al. 2012) and (Mohan and Fazel 2012) used the Schatten-p norm, which is defined as the ℓ_p norm of the singular values $(\sum_i \sigma_i^p)^{1/p}$, to regularize the rank of the reconstructed matrix. When $p = 0$, the Schatten-0 norm degenerates to the matrix rank, and when $p = 1$, the Schatten-1 norm complies with the standard nuclear norm. (Chartrand 2012) defined a generalized Huber penalty function to regularize the singular values of matrix. (Zhang et al. 2012a) proposed a truncated nuclear norm regularization method which only regularizes the last k singular values of matrix. Similar strategy was utilized in (Oh et al. 2013) to deal with different low level vision problems. However, these low-rank regularization terms are all designed for a specific category of penalty functions for LRMA, while they are not flexible enough to adapt various real cases. Recently, (Dong et al. 2014) utilized nonconvex $logdet(\boldsymbol{X})$ as a surrogate of the rank function and achieved state-of-the-art performance on compressive sensing. (Lu et al. 2014a, b) studied the generalized nonconvex nonsmooth rank minimization problem and validated that if the penalty function is monotonic and lower bounded, a generalized singular value soft-thresholding operation is able to get the optimal solution. Their optimization framework can be utilized to a wide range of regularization penalty functions.

The WNNM problem has been formally proposed and analyzed in (Gu et al. 2014), where Gu et al presented some preliminary theoretical results on this problem and evaluated its effectiveness in image denoising. In this paper, we thoroughly polish the theory and provide the general solving scheme for this problem. The automatic weight setting scheme is also rationally designed by referring to the reweighting technique in sparse coding (Candès et al. 2008). We also substantiate our theoretical analysis by conducting extensive experiments in image denoising, background subtraction and image inpainting.

3 Weighted Nuclear Norm Minimization for Low Rank Modeling

In this section, we first introduce the general solving scheme for the WNNP problem (5), and then introduce its WNNM-RPCA and WNNM-MC extensions. Note that these WNNM models are more difficult to optimize than conventional NNM ones due to the non-convexity of the involved weighted nuclear norm. Furthermore, the subgradient method proposed in (Cai et al. 2010) to solve NNP is not applicable to WNNP. We thus construct new solving regime for this problem. Obviously, NNM is a special case of WNNM when all the weights w_i are set the same. Our solution thus covers that of the traditional NNP.

3.1 Weighted Nuclear Norm Proximal for WNNM

In (Gu et al. 2014), a lemma is presented to analyze the WNNP problem. This Lemma is actually a special case of the following Lemma 1, i.e., the classical von Neumanns trace inequality (Mirsky 1975; Rhea 2011):

Lemma 1 (von Neumanns trace inequality (Mirsky 1975; Rhea 2011)) For any $m \times n$ matrices $\mathbf{A}$ and $\mathbf{B}$, $tr\left(\mathbf{A}^T\mathbf{B}\right) \leq \sum_i \sigma_i(\mathbf{A})\sigma_i(\mathbf{B})$, where $\sigma_1(\mathbf{A}) \geq \sigma_2(\mathbf{A}) \geq \cdots \geq 0$ and $\sigma_1(\mathbf{B}) \geq \sigma_2(\mathbf{B}) \geq \cdots \geq 0$ are the descending singular

values of $\mathbf{A}$ and $\mathbf{B}$, respectively. The case of equality occurs if and only if it is possible to find unitaries $\mathbf{U}$ and $\mathbf{V}$ that simultaneously singular value decompose $\mathbf{A}$ and $\mathbf{B}$ in the sense that

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}_A\mathbf{V}^T, \textit{and} \;\; \mathbf{B} = \mathbf{U}\boldsymbol{\Sigma}_B\mathbf{V}^T,$$

where $\boldsymbol{\Sigma}_A$ and $\boldsymbol{\Sigma}_B$ denote the ordered eigenvalue matrices with singular value $\sigma(\mathbf{A})$ and $\sigma(\mathbf{B})$ along the diagonal with the same order, respectively.

Based on the result of Lemma 1, we can deduce the following important theorem.

Theorem 1 *Given $\boldsymbol{Y} \in \Re^{m\times n}$, without loss of generality, we assume that $m \geq n$, and let $\boldsymbol{Y} = \boldsymbol{U\Sigma V}^T$ be the SVD of $\boldsymbol{Y}$, where $\boldsymbol{\Sigma} = \begin{pmatrix} diag(\sigma_1, \sigma_2, ..., \sigma_n) \\ \boldsymbol{0} \end{pmatrix} \in \Re^{m\times n}$. The global optimum of the WNNP problem in (5) can be expressed as $\hat{\boldsymbol{X}} = \boldsymbol{U}\hat{\boldsymbol{D}}\boldsymbol{V}^T$, where $\boldsymbol{D} = \begin{pmatrix} diag(d_1, d_2, ..., d_n) \\ \boldsymbol{0} \end{pmatrix}$ is a diagonal non-negative matrix and $(d_1, d_1, ..., d_n)$ is the solution to the following convex optimization problem:*

$$\begin{aligned} &\min_{d_1,d_2,...,d_n} \sum\nolimits_{i=1}^{n} (\sigma_i - d_i)^2 + w_i d_i, \\ &\quad s.t. \;\; d_1 \geq d_2 \geq ... \geq d_n \geq 0. \end{aligned} \tag{6}$$

The proof of Theorem 1 can be found in the Appendix 9.1. Theorem 1 shows that the WNNP problem can be transformed into a new optimization problem (6). It is interesting that (6) is a quadratic optimization problem with linear constraints, and its global optimum can be easily calculated by off-the-shelf convex optimization solvers. This means that for the non-convex WNNP problem, we can always get its global solution through (6). In the following corollary, we further show that the global solution of (6) can be achieved in closed-form when the weights are sorted in a non-descending order.

Corollary 1 *If $\sigma_1 \geq \cdots \geq \sigma_n \geq 0$ and the weights satisfy $0 \leq w_1 \leq \cdots \leq w_n$, then the global optimum of (6) is $\hat{d}_i = max\left(\sigma - \frac{w_i}{2}, 0\right)$.*

The proof of Corollary 1 is given in Appendix 9.2. The conclusion in Corollary 1 is very useful. The singular values of a matrix are sorted in a non-ascending order, and the larger singular values usually correspond to the subspaces of more important components of the data matrix. We thus always expect to shrink less the larger singular values to keep the major and faithful information of the underneath data. In this sense, Corollary 1 guarantees that we have a closed-form optimal solution for the WNNP problem by the weighted singular value soft-thresholding operation:

$$\mathbf{prox}_{\lambda\|\cdot\|_{\boldsymbol{w},*}}(\boldsymbol{Y}) = \boldsymbol{U}\mathcal{S}_{\frac{\boldsymbol{w}}{2}}(\boldsymbol{\Sigma})\boldsymbol{V}^{\mathbf{T}},$$

where $\boldsymbol{Y} = \boldsymbol{U\Sigma V}^T$ is the SVD of $\boldsymbol{Y}$, and $\mathcal{S}_{\frac{\boldsymbol{w}}{2}}(\boldsymbol{\Sigma})$ is the generalized soft-thresholding operator with weight vector $\boldsymbol{w}$

$$\mathcal{S}_{\frac{\boldsymbol{w}}{2}}(\boldsymbol{\Sigma})_{ii} = \max\left(\boldsymbol{\Sigma}_{ii} - \frac{w_i}{2}, 0\right).$$

Note that when all the weights w_i are set the same, the above WNNP solver exactly degenerates to the NNP solver for the traditional NNM problem.

A recent work by (Lu et al. 2014b) has proved a similar conclusion to our Corollary 1. As Lu et al. analyzed the generalized singular value regularization model with different penalty functions for the singular values, the condition in their paper is the monotonicity property of the proximal operator which is determined by the penalty function. While our work attains the WNNP solver in general weighting cases rather than only in the case of nonascendingly ordered weights. Interested readers may refer to the proof of our paper and (Lu et al. 2014b) for details.

3.2 WNNM for Robust PCA

In last section, we analyzed the optimal solution to the WNNP operator for the WNNM problem. Based on our definition of WNNP in (5), $\mathbf{prox}_{\|\cdot\|_{w,*}}(\boldsymbol{Y})$ is the low rank approximation to the observation matrix $\boldsymbol{Y}$ under the F-norm data fidelity term. However, in real applications, the observation data may be corrupted by outliers or sparse noise with large magnitude. In such cases, the large magnitude noise, even with small amount, tends to greatly affect the F-norm data fidelity and lead to a biased low rank estimation. The recently proposed NNM-based RPCA (NNM-RPCA) method (Candès et al. 2011) alleviates this problem by optimizing the following problem:

$$\min_{\boldsymbol{E},\boldsymbol{X}} \|\boldsymbol{E}\|_1 + \|\boldsymbol{X}\|_*, \quad s.t. \;\; \boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{E}. \tag{7}$$

Using the ℓ_1-norm to model the error $\boldsymbol{E}$, model (7) guarantees a more robust matrix approximation in the presence of outliers/sparse noise. In particular, (Candès et al. 2011) proved that if the low rank matrix $\boldsymbol{X}$ and the sparse component $\boldsymbol{E}$

Algorithm 1 WNNM-RPCA

Input: Observation data $\boldsymbol{Y}$, weight vector $\boldsymbol{w}$
Initialize $\mu_0 > 0, \rho > 1, \theta > 0, k = 0, \boldsymbol{X}_0 = \boldsymbol{Y}, \boldsymbol{L}_0 = \boldsymbol{0}$;
do
  $\boldsymbol{E}_{k+1} = \arg\min_{\boldsymbol{E}} \|\boldsymbol{E}\|_1 + \frac{\mu_k}{2}\|\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{X}_k - \boldsymbol{E}\|_F^2$;
  $\boldsymbol{X}_{k+1} = \arg\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_{w,*} + \frac{\mu_k}{2}\|\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{E}_{k+1} - \boldsymbol{X}\|_F^2$;
  $\boldsymbol{L}_{k+1} = \boldsymbol{L}_k + \mu_k(\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1})$;
  Update $\mu_{k+1} = \rho * \mu_k$;
  $k = k + 1$;
while $\|\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1}\|_F / \|\boldsymbol{Y}\|_F > \theta$
Output: Matrix $\boldsymbol{X} = \boldsymbol{X}_{k+1}$ and $\boldsymbol{E} = \boldsymbol{E}_{k+1}$;

satisfy certain conditions, the NNM-RPCA model (7) can exactly recover $\boldsymbol{X}$ with a probability close to 1.

In this section, we propose to use the weighted nuclear norm to reformulate (7), leading to the following WNNM-based RPCA (WNNM-RPCA) model:

$$\min_{\boldsymbol{E},\boldsymbol{X}} \|\boldsymbol{E}\|_1 + \|\boldsymbol{X}\|_{w,*}, \quad s.t. \quad \boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{E}. \tag{8}$$

As in NNM-RPCA, we also employ the alternating direction method of multipliers (ADMM) to solve the WNNM-RPCA problem. Its augmented Lagrange function is

$$\begin{aligned}&\Gamma(\boldsymbol{X}, \boldsymbol{E}, \boldsymbol{L}, \mu)\\ &= \|\boldsymbol{E}\|_1 + \|\boldsymbol{X}\|_{w,*} + \langle \boldsymbol{L}, \boldsymbol{Y} - \boldsymbol{X} - \boldsymbol{E}\rangle + \frac{\mu}{2}\|\boldsymbol{Y}-\boldsymbol{X}-\boldsymbol{E}\|_F^2,\end{aligned} \tag{9}$$

where $\boldsymbol{L}$ is the Lagrange multiplier and μ is a positive scaler. The optimization procedure is described in Algorithm 1. Note that the convergence of this ADMM algorithm is more difficult to analyze than the convex NNM-RPCA model due to the non-convexity of the WNNM-RPCA model. We give the following weak convergence result to facilitate the construction of a rational termination condition for Algorithm 1.

Theorem 2 *If the weights are sorted in a non-descending order, the sequences $\{\boldsymbol{E}_k\}$ and $\{\boldsymbol{X}_k\}$ generated by Algorithm 1 satisfy:*

(1) $\lim_{k\to\infty} \|\boldsymbol{X}_{k+1} - \boldsymbol{X}_k\|_F = 0,$

(2) $\lim_{k\to\infty} \|\boldsymbol{E}_{k+1} - \boldsymbol{E}_k\|_F = 0.$

(3) $\lim_{k\to\infty} \|\boldsymbol{Y} - \boldsymbol{E}_{k+1} - \boldsymbol{X}_{k+1}\|_F = 0.$

The proof of Theorem 2 can be found in Appendix 9.3. Please note that the proof of Theorem 2 relies on the unboundedness of the parameter μ_k. In most of previous ADMM based methods, an upper bound of μ_k is needed to ensure the optimal solution for convex objective functions (Lin et al. 2015). However, since our WNNM-RPCA model is non-convex for general weight conditions, we use an unbounded μ_k to guarantee the convergence of Algorithm 1. If μ_k increases too fast, the iteration may stop quickly and we might not get a good solution. Thus in both Algorithm 1 and the following Algorithm 2, a small ρ is adopted to prevent μ_k increase too fast. Please refer to the experiments in Sects. 5 and 6 for more details.

3.3 WNNM for Matrix Completion

In Sect. 3.2, we introduced the WNNM-RPCA model and provided the ADMM algorithm to solve it. In this section, we further use WNNM to deal with the matrix completion problem, and propose the following WNNM-based matrix completion (WNNM-MC) model:

Algorithm 2 WNNM-MC

Input: Observation data $\boldsymbol{Y}$, indicator matrix Ω, weight vector $\boldsymbol{w}$,
1: Initialize $\mu_0 > 0, \rho > 1, \theta > 0, k = 0, k = 0, \boldsymbol{X}_0 = \boldsymbol{Y}, \boldsymbol{L}_0 = \boldsymbol{0}$;
2: **do**
3: $\boldsymbol{E}_{k+1} = \arg\min_{\boldsymbol{E}} \|\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{X}_k - \boldsymbol{E}\|_F^2$ $s.t. \|\mathcal{P}_\Omega(\boldsymbol{E})\|_F^2 = 0$;
4: $\boldsymbol{X}_{k+1} = \arg\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_{w,*} + \frac{\mu_k}{2}\|\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{E}_k - \boldsymbol{X}\|_F^2$;
5: $\boldsymbol{L}_{k+1} = \boldsymbol{L}_k + \mu_k(\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1})$;
6: Update $\mu_{k+1} = \rho * \mu_k$;
7: $k = k + 1$;
8: **while** $\|\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1}\|_F / \|\boldsymbol{Y}\|_F > \theta$
Output: $\boldsymbol{X} = \boldsymbol{X}_{k+1}$;

$$\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_{\boldsymbol{w},*} \quad s.t. \quad \mathcal{P}_\Omega(\boldsymbol{X}) = \mathcal{P}_\Omega(\boldsymbol{Y}), \tag{10}$$

where Ω is a binary support indicator matrix of the same size as $\boldsymbol{Y}$, and zeros in Ω indicate the missing entries in the observation matrix. $\mathcal{P}_\Omega(\boldsymbol{Y}) = \Omega \odot \boldsymbol{Y}$ is the element-wise matrix multiplication (Hardamard product) between the support matrix Ω and the variable $\boldsymbol{Y}$. The constraint implies that the estimated matrix $\boldsymbol{X}$ agrees with $\boldsymbol{Y}$ in the observed entries.

By introducing a variable $\boldsymbol{E}$, we reformulate (10) as

$$\min_{\boldsymbol{X}} \|\boldsymbol{X}\|_{w,*}, \quad s.t. \quad \boldsymbol{X} + \boldsymbol{E} = \boldsymbol{Y}, \quad \mathcal{P}_\Omega(\boldsymbol{E}) = 0. \tag{11}$$

The ADMM algorithm for solving WNNM-MC can then be constructed in Algorithm 2. For non-descending weights, both the subproblems in steps 3 and 4 of Algorithm 2 have closed-form optimal solutions. However, as the weighted nuclear norm is not convex, it is difficult to accurately analyze the convergence of the algorithm. Like in Theorem 2, in the following Theorem 3, we also present a weak convergence result.

Theorem 3 *If the weights are sorted in a non-descending order, the sequence $\{\boldsymbol{X}_k\}$ generated by Algorithm 2 satisfies*

(1) $\lim_{k\to\infty} \|\boldsymbol{X}_{k+1} - \boldsymbol{X}_k\|_F = 0,$

(2) $\lim_{k\to\infty} \|\boldsymbol{Y} - \boldsymbol{E}_{k+1} - \boldsymbol{X}_{k+1}\|_F = 0.$

The proof of Theorem 3 is similar to the proof of Theorem 2, and thus we omit it here.

3.4 The Setting of Weighting Vector

In previous sections, we proposed to utilize the WNNM model to solve different problems. By introducing the weight vector, the WNNM model improves the flexibility of the original NNM model. However, the weight vector itself also

brings more parameters in the model. Appropriate setting of the weights plays a crucial role in the success of the proposed WNNM model.

In (Candès et al. 2008), an effective reweighted mechanism is proposed to enhance the sparsity of sparse coding solutions by adaptively tuning weights through the following formula:

$$w_i^{\ell+1} = \frac{C}{|x_i^\ell| + \varepsilon}, \tag{12}$$

where x_i^ℓ is the i-th sparse coding coefficient in the ℓ-th iteration and $w_i^{\ell+1}$ is its corresponding regularization parameter in the $(\ell+1)$-th iteration, ε is a small positive number to avoid dividing by zero and C is a compromising constant. Such a reweighted procedure has proved to be capable of getting a good resemble of the ℓ_0 norm and the model achieves superior performance in compressive sensing.

Inspired by the success of reweighted sparse coding, we can adopt a similar reweighted strategy in WNNM by replacing x_i^ℓ in (12) with the singular value $\sigma_i(\boldsymbol{X}_\ell)$ in the ℓth iteration. Because (12) is monotonically decreasing with the singular values, the non-descending order of weights with respect to singular values will be kept throughout the reweighting process. Very interestingly, the following remark indicates that we can directly get the closed-form solution of WNNP operator with such a reweighting strategy.

Remark 1 Let $\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ be the SVD of $\boldsymbol{Y}$, where $\boldsymbol{\Sigma} = \begin{pmatrix} diag(\sigma_1(\boldsymbol{Y}), \sigma_2(\boldsymbol{Y}), ..., \sigma_n(\boldsymbol{Y})) \\ \mathbf{0}, \end{pmatrix}$, and $\sigma_i(\boldsymbol{Y})$ denotes the i-th singular value of $\boldsymbol{Y}$. If the regularization parameter C is positive and the positive value ε is small enough to make the inequality $\varepsilon < min\left(\sqrt{C}, \frac{C}{\sigma_1(\boldsymbol{Y})}\right)$ hold, by using the reweighting formula $w_i^\ell = \frac{C}{\sigma_i(\mathbf{X}_\ell)+\varepsilon}$ with initial estimation $\boldsymbol{X}_0 = \boldsymbol{Y}$, the reweighted WNNP problem $\{\min_{\boldsymbol{X}} \|\boldsymbol{Y} - \boldsymbol{X}\|_F^2 + \|\boldsymbol{X}\|_{\boldsymbol{w},*}\}$ has the closed-form solution: $\boldsymbol{X}^* = \boldsymbol{U}\widetilde{\boldsymbol{\Sigma}}\boldsymbol{V}^T$, where

$$\widetilde{\boldsymbol{\Sigma}} = \begin{pmatrix} diag(\sigma_1(\boldsymbol{X}^*), \sigma_2(\boldsymbol{X}^*), ..., \sigma_n(\boldsymbol{X}^*)) \\ \mathbf{0}, \end{pmatrix},$$

and

$$\sigma_i(\boldsymbol{X}^*) = \begin{cases} 0 & \text{if } c_2 < 0 \\ \frac{c_1+\sqrt{c_2}}{2} & \text{if } c_2 \geq 0 \end{cases} \tag{13}$$

where

$$c_1 = \sigma_i(\boldsymbol{Y}) - \varepsilon, \quad c_2 = (\sigma_i(\boldsymbol{Y}) + \varepsilon)^2 - 4C.$$

The proof of Remark 1 can be found in Appendix 9.4. Remark 1 shows that although a reweighting strategy $w_i^\ell = \frac{C}{\sigma_i(\mathbf{X}_\ell)+\varepsilon}$ is used, we do not need to iteratively perform the thresholding and weight calculation operations. Based on the relationship between the singular value of observation matrix $\mathbf{X}$ and the regularization parameter C, the convergence of the sigular value of estimated matrix after the reweighting process can be directly obtained. In each iteration of both the WNNM-RPCA and WNNM-MC algorithms, such a reweighting strategy is performed on the WNNP subproblem (step 4 in Algorithms 1 and 2) to adjust weights based on current $\boldsymbol{X}_k$. Thanks to Remark 1, utilizing reweighting strategy in step 4 of Algorithm 1 and Algorithm 2 will increase little the computation burden. We are able to use an operation to directly shrank the original singular value $\sigma_i(\mathbf{Y})$ to 0 or $\frac{c_1+\sqrt{c_2}}{2}$, just like the soft-thresholding operation in the NNM method.

In implementation, we initialize $\boldsymbol{X}_0$ as the observation $\boldsymbol{Y}$. The above weight setting strategy greatly facilitates the WNNM calculations. Note that there remains one parameter C in the WNNM implementation. In all of our experiments, we set it by experience for certain tasks. Please see the following sections for details.

4 Image Denoising by WNNM

In this section, we validate the proposed WNNM model in application of image denoising. Image denoising is one of the fundamental problems in low level vision, and is an ideal test bed to investigate and evaluate the statistical image modeling techniques and optimization methods. Image denoising aims to reconstruct the original image $\boldsymbol{x}$ from its noisy observation $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n}$, where $\boldsymbol{n}$ is generally assumed to be additive white Gaussian noise (AWGN) with zero mean and variance σ_n^2.

The seminal work of nonlocal means (Buades et al. 2005; Buades et al. 2008) triggers the study of nonlocal self-similarity (NSS) based methods for image denoising. NSS refers to the fact that there are many repeated local patterns across a natural image, and those nonlocal similar patches to a given patch offer helpful remedy for its better reconstruction. The NSS-based image denoising algorithms such as BM3D (Dabov et al. 2007), LSSC (Mairal et al. 2009), NCSR (Dong et al. 2011) and SAIST (Dong et al. 2013) have achieved state-of-the-art denoising results. In this section, we utilize the NSS prior to develop the following WNNM-based denoising algorithm.

4.1 Denoising Algorithm

For a local patch $\boldsymbol{y}_j$ in image $\boldsymbol{y}$, we can search for its nonlocal similar patches across a relatively large area around it by methods such as block matching (Dabov et al. 2007). By stacking those nonlocal similar patches into a matrix, denoted

Algorithm 3 Image Denoising by WNNM

Input: Noisy image $\boldsymbol{y}$
Initialize $\hat{\boldsymbol{x}}^{(0)} = \boldsymbol{y}, \boldsymbol{y}^{(0)} = \boldsymbol{y}$
for k=1:K **do**
  Iterative regularization $\boldsymbol{y}^{(k)} = \hat{\boldsymbol{x}}^{(k-1)} + \delta(\boldsymbol{y} - \hat{\boldsymbol{y}}^{(k-1)})$
  for each patch $\boldsymbol{y}_j$ in $\boldsymbol{y}^{(k)}$ **do**
    Find similar patch group $\boldsymbol{Y}_j$
    Apply the WNNP operator to $\boldsymbol{Y}_j$ to estimate $\boldsymbol{X}_j$
  end for
  Aggregate $\boldsymbol{X}_j$ to form the clean image $\hat{\boldsymbol{x}}^{(k)}$
end for
Output: Denoised image $\hat{\boldsymbol{x}}^{(K)}$

by $\boldsymbol{Y}_j$, we have $\boldsymbol{Y}_j = \boldsymbol{X}_j + \boldsymbol{N}_j$, where $\boldsymbol{X}_j$ and $\boldsymbol{N}_j$ are the original clean patch matrix and the corresponding corruption matrix, respectively. Intuitively, $\boldsymbol{X}_j$ should be a low rank matrix, and the LRMA methods can be used to estimate $\boldsymbol{X}_j$ from $\boldsymbol{Y}_j$. By aggregating all the estimated patches $\boldsymbol{Y}_j$, the whole image can be reconstructed. Indeed, the NNM method has been adopted in (Ji et al. 2010) for video denoising, and we apply the proposed WNNP operator to each $\boldsymbol{Y}_j$ to estimate $\boldsymbol{X}_j$ for image denoising. By using the noise variance σ_n^2 to normalize the F-norm data fidelity term $\|\boldsymbol{Y}_j - \boldsymbol{X}_j\|_F^2$, we have the following energy function:

$$\hat{\boldsymbol{X}}_j = \arg\min_{\boldsymbol{X}_j} \frac{1}{\sigma_n^2}\|\boldsymbol{Y}_j - \boldsymbol{X}_j\|_F^2 + \|\boldsymbol{X}_j\|_{\boldsymbol{w},*}. \tag{14}$$

Throughout our experiments, we set the parameter C as $\sqrt{2n}$ by experience, where n is the number of similar patches.

By applying the above procedures to each patch and aggregating all patches together, the image $\boldsymbol{x}$ can be reconstructed. In practice, we can run several iterations of this reconstruction process across all image patches to enhance the denoising outputs. The whole denoising algorithm is summarized in Algorithm 3.

4.2 Experimental Setting

We compare the proposed WNNM-based image denoising algorithm with several state-of-the-art denoising methods, including BM3D[2] (Dabov et al. 2007), EPLL[3] (Zoran and Weiss 2011), LSSC[4] (Mairal et al. 2009), NCSR[5] (Dong et al. 2011) and SAIST[6] (Dong et al. 2013). The baseline NNM algorithm is also compared. All the competing methods exploit the image nonlocal redundancies.

There are several other parameters (δ, the iteration number K and the patch size) in the proposed algorithm. For all noise levels, the iterative regularization parameter δ is fixed to 0.1. The iteration number K and the patch size are set based on noise level. For higher noise level, we choose bigger patches and run more times the iteration. By experience, we set patch sizes to 6×6, 7×7, 8×8 and 9×9 for $\sigma_n \leq 20$, $20 < \sigma_n \leq 40$, $40 < \sigma_n \leq 60$ and $60 < \sigma_n$, respectively. The iteration times K is set to 8, 12, 14, and 14, and the number of selected non-local similar patches is to 70, 90, 120 and 140,respectively, on these noise levels.

For NNM, we use the same parameters as WNNM except for the uniform weight $\sqrt{n}\sigma_n$. The source codes of the comparison methods are obtained directly from the original authors, and we use the default parameters. Our code can be downloaded from http://www4.comp.polyu.edu.hk/cslzhang/code/WNNM_code.zip.

Fig. 1 The 20 test images used in image denoising experiments

4.3 Experimental Results on 20 Test Images

We evaluate the competing methods on 20 widely used test images, whose thumbnails are shown in Fig. 1. The first 12 images are of size 256×256, and the other 8 images are of size 512×512. AWGN with zero mean and variance σ_n^2 are added to those test images to generate the noisy observations. We present the denoising results on four noise levels, ranging from low noise level $\sigma_n = 10$, to medium noise levels $\sigma_n = 30$ and 50, and to strong noise level $\sigma_n = 100$. The PSNR results under these noise levels for all competing denoising methods are shown in Table 1. For a more comprehensive comparison, we also list the average denoising results on more noise levels by all competing methods in Tabel 2.

From Tables 1 and 2, we can see that the proposed WNNM method achieves the highest PSNR in almost all cases. It achieves 1.3dB–2dB improvement over the NNM method on average and outperforms the benchmark BM3D method by

[2] http://www.cs.tut.fi/foi/GCF-BM3D/BM3D.zip
[3] http://people.csail.mit.edu/danielzoran/noiseestimation.zip
[4] http://lear.inrialpes.fr/people/mairal/software.php
[5] http://www4.comp.polyu.edu.hk/~cslzhang/code/NCSR.rar
[6] http://www.csee.wvu.edu/xinl/demo/saist.html

Table 1 Denoising results (PSNR, dB) by competing methods on the 20 test images

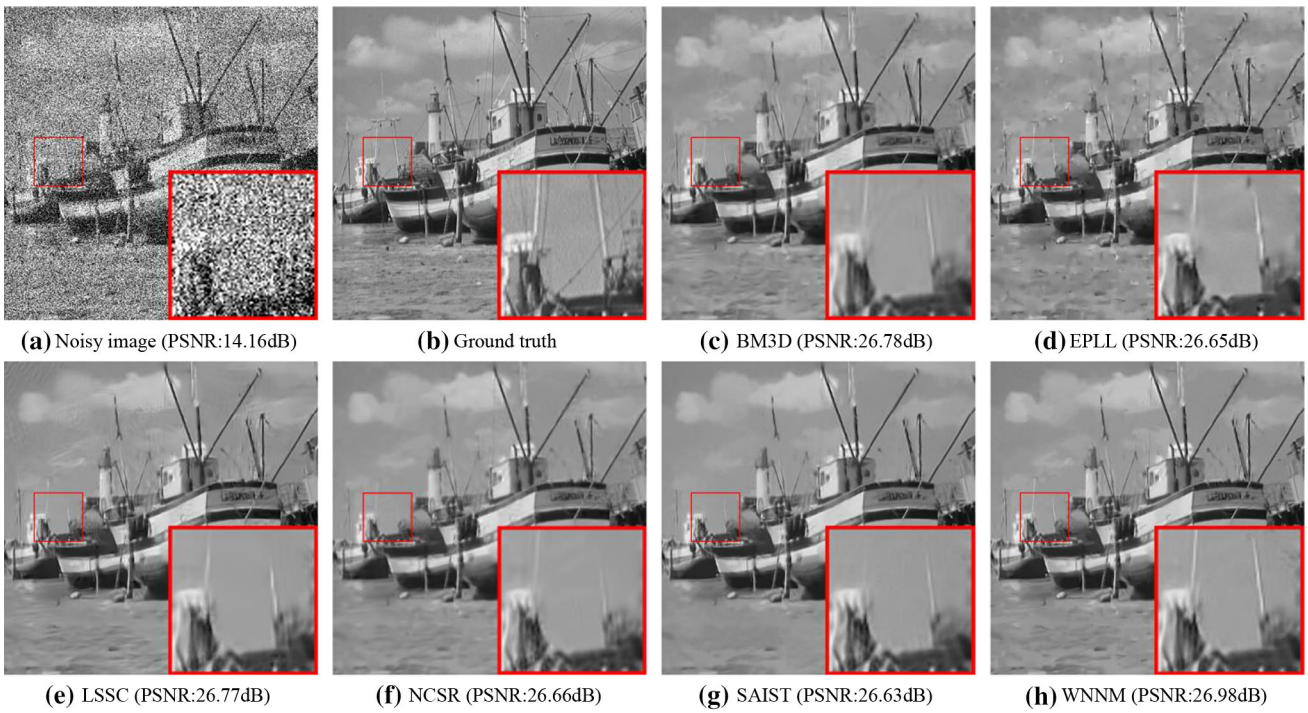
	σ_n=10							σ_n=30						
	NNM	BM3D	EPLL	LSSC	NCSR	SAIST	WNNM	NNM	BM3D	EPLL	LSSC	NCSR	SAIST	WNNM
C.Man	32.87	34.18	34.02	34.24	34.18	34.30	**34.44**	27.43	28.64	28.36	28.63	28.59	28.36	**28.80**
House	35.97	36.71	35.75	36.95	36.80	36.66	**36.95**	30.99	32.09	31.23	32.41	32.07	32.30	**32.52**
Peppers	33.77	34.68	34.54	34.80	34.68	34.82	**34.95**	28.11	29.28	29.16	29.25	29.10	29.24	**29.49**
Montage	36.09	37.35	36.49	37.26	37.17	37.46	**37.84**	29.28	31.38	30.17	31.10	30.92	31.06	**31.65**
Leaves	33.55	34.04	33.29	34.52	34.53	34.92	**35.20**	26.81	27.81	27.18	27.65	28.14	28.29	**28.60**
StarFish	32.62	33.30	33.29	33.74	33.65	33.72	**33.99**	26.62	27.65	27.52	27.70	27.78	27.92	**28.08**
Monarch	33.54	34.12	34.27	34.44	34.51	34.76	**35.03**	27.44	28.36	28.35	28.20	28.46	28.65	**28.92**
Airplane	32.19	33.33	33.39	33.51	33.40	33.43	**33.64**	26.53	27.56	27.67	27.53	27.53	27.66	**27.83**
Paint	33.13	34.00	34.01	34.35	34.15	34.28	**34.50**	27.02	28.29	28.33	28.29	28.10	28.44	**28.58**
J.Bean	37.52	37.91	37.63	38.69	38.31	38.37	**38.93**	31.03	31.97	31.56	32.39	32.13	32.14	**32.46**
Fence	32.62	33.50	32.89	33.60	33.65	33.76	**33.93**	27.19	28.19	27.23	28.16	28.23	28.26	**28.56**
Parrot	32.54	33.57	33.58	33.62	33.56	33.66	**33.81**	27.26	28.12	28.07	27.99	28.07	28.12	**28.33**
Lena	35.19	35.93	35.58	35.83	35.85	35.90	**36.03**	30.15	31.26	30.79	31.18	31.06	31.27	**31.43**
Barbara	34.40	34.98	33.61	34.98	35.00	35.24	**35.51**	28.59	29.81	27.57	29.60	29.62	30.14	**30.31**
Boat	33.05	33.92	33.66	34.01	33.91	33.91	**34.09**	27.82	29.12	28.89	29.06	28.94	28.98	**29.24**
Hill	32.89	33.62	33.48	33.66	33.69	33.65	**33.79**	28.11	29.16	28.90	29.09	28.97	29.06	**29.25**
F.print	31.38	32.46	32.12	32.57	32.68	32.69	**32.82**	25.84	26.83	26.19	26.68	26.92	26.95	**26.99**
Man	32.99	33.98	33.97	34.10	34.05	34.12	**34.23**	27.87	28.86	28.83	28.87	28.78	28.81	**29.00**
Couple	32.97	34.04	33.85	34.01	34.00	33.96	**34.14**	27.36	28.87	28.62	28.77	8.57	28.72	**28.98**
Straw	29.84	30.89	30.74	31.25	31.35	31.49	**31.62**	23.52	24.84	24.64	24.99	25.00	25.23	**25.27**
AVE.	33.462	34.326	34.008	34.507	34.456	34.555	**34.772**	27.753	28.905	28.463	28.877	28.849	28.980	**29.214**
	σ_n=50							σ_n=100						
	NNM	BM3D	EPLL	LSSC	NCSR	SAIST	WNNM	NNM	BM3D	EPLL	LSSC	NCSR	SAIST	WNNM
C-Man	24.88	26.12	26.02	26.35	26.14	26.15	**26.42**	21.49	23.07	22.86	23.15	22.93	23.09	**23.36**
House	27.84	29.69	28.76	29.99	29.62	30.17	**30.32**	23.65	25.87	25.19	25.71	25.56	26.53	**26.68**
Peppers	25.29	26.68	26.63	26.79	26.82	26.73	**26.91**	21.24	23.39	23.08	23.20	22.84	23.32	**23.46**
Montage	26.04	27.9	27.17	28.10	27.84	28.0	**28.27**	21.70	23.89	23.42	23.77	23.74	23.98	**24.16**
Leaves	23.36	24.68	24.38	24.81	25.04	25.25	**25.47**	18.73	20.91	20.25	20.58	20.86	21.40	**21.57**
Starfish	23.83	25.04	25.04	25.12	25.07	25.29	**25.44**	20.58	22.10	21.92	21.77	21.91	22.10	**22.22**
Mornar.	24.46	25.82	25.78	25.88	25.73	26.10	**26.32**	20.22	22.52	22.23	22.24	22.11	22.61	**22.95**
Plane	23.97	25.10	25.24	25.25	24.93	25.34	**25.43**	20.73	22.11	22.02	21.69	21.83	22.27	**22.55**
Paint	24.19	25.67	25.77	25.59	25.37	25.77	**25.98**	21.02	22.51	22.50	22.14	22.11	22.42	**22.74**
J.Bean	27.96	29.26	28.75	29.42	29.29	29.32	**29.62**	23.79	25.80	25.17	25.64	25.66	25.82	**26.04**
Fence	24.59	25.92	24.58	25.87	25.78	26.00	**26.43**	21.23	22.92	21.11	22.71	22.23	22.98	**23.37**
Parrot	24.87	25.90	25.84	25.82	25.71	25.95	**26.09**	21.38	22.96	22.71	22.79	22.53	23.04	**23.19**
Lena	27.74	29.05	28.42	28.95	28.90	29.01	**29.24**	24.41	25.95	25.30	25.96	25.71	25.93	**26.20**
Barbara	25.75	27.23	24.82	27.03	26.99	27.51	**27.79**	22.14	23.62	22.14	23.54	23.20	24.07	**24.37**
Boat	25.39	26.78	26.65	26.77	26.66	26.63	**26.97**	22.48	23.97	23.71	23.87	23.68	23.80	**24.10**
Hill	25.94	27.19	26.96	27.14	26.99	27.04	**27.34**	23.32	24.58	24.43	24.47	24.36	24.29	**24.75**
F.print	23.37	24.53	23.59	24.26	24.48	24.52	**24.67**	20.01	21.61	19.85	21.30	21.39	21.62	**21.81**
Man	25.66	26.81	26.72	26.72	26.67	26.68	**26.94**	22.88	24.22	24.07	23.98	24.02	24.01	**24.36**
Couple	24.84	26.46	26.24	26.35	26.19	26.30	**26.65**	22.07	23.51	23.32	23.27	23.15	23.21	**23.55**
Straw	20.99	22.29	21.93	22.51	22.30	22.65	**22.74**	18.33	19.43	18.84	19.43	19.10	19.42	**19.67**
AVE.	25.048	26.406	25.965	26.436	26.326	26.521	**26.752**	21.570	23.247	22.706	23.061	22.996	23.296	**23.555**

The best results are highlighted in bold

Table 2 The average PSNR (dB) values by competing methods on the 20 test images

	σ_n=10	σ_n=20	σ_n=30	σ_n=40	σ_n=50	σ_n=75	σ_n=100
NNM	33.462	30.040	27.753	26.422	25.048	22.204	21.570
BM3D	34.326	30.840	28.905	27.360	26.406	24.560	23.247
EPLL	34.008	30.470	28.463	27.060	25.965	24.020	22.706
LSSC	34.507	30.950	28.877	27.500	26.436	24.450	23.061
NCSR	34.456	30.890	28.849	27.390	26.326	24.340	22.996
SAIST	34.555	30.970	28.980	27.590	26.521	24.620	23.296
WNNM	**34.772**	**31.200**	**29.214**	**27.780**	**26.752**	**24.860**	**23.555**

The best results are highlighted in bold



(a) Noisy image (PSNR:14.16dB) **(b)** Ground truth **(c)** BM3D (PSNR:26.78dB) **(d)** EPLL (PSNR:26.65dB)

(e) LSSC (PSNR:26.77dB) **(f)** NCSR (PSNR:26.66dB) **(g)** SAIST (PSNR:26.63dB) **(h)** WNNM (PSNR:26.98dB)

Fig. 2 Denoising results on image *Boats* by competing methods (noise level $\sigma_n = 50$). The demarcated area is enlarged in the right bottom corner for better visualization. The figure is better seen by zooming on a computer screen

0.3dB–0.45dB consistently on all the four noise levels. Such an improvement is notable since few methods can surpass BM3D more than 0.3dB on average (Levin et al. 2012; Levin and Nadler 2011).

In Figs. 2 and 3, we compare the visual quality of the denoised images by the competing algorithms. More visual comparison results can be found in the supplementary file. Fig. 2 demonstrates that the proposed WNNM reconstructs more image details from the noisy observation. Compared with WNNM, the LSSC, NCSR and SAIST methods over-smooth more textures in the sands area of image *Boats*, and the BM3D and EPLL methods generate more artifacts. More interestingly, as can be seen in the demarcated window, the proposed WNNM is capable of well reconstructing the tiny masts of the boat, while the masts are almost unrecognizable in the reconstructed images by other methods. Fig. 3 shows an example with strong noise. It is easy to see that WNNM generates less artifacts and preserves better the image edge structures compared with other competing methods. In summary, WNNM shows strong denoising capability, producing more pleasant denoising outputs in visualization and higher PSNR indices in quantity.

4.4 Experimental Results on Real Noisy Images

We also evaluate different denoising algorithms on two real images, including a gray level SAR image[7] and a color image[8], as shown in Figs. 4 and 5, respectively. Since the

[7] The SAR image was downloaded at http://aess.cs.unh.edu/radar%20se%20Lecture%2018%20B.html.

[8] The color image was used in previous work (Portilla 2004).

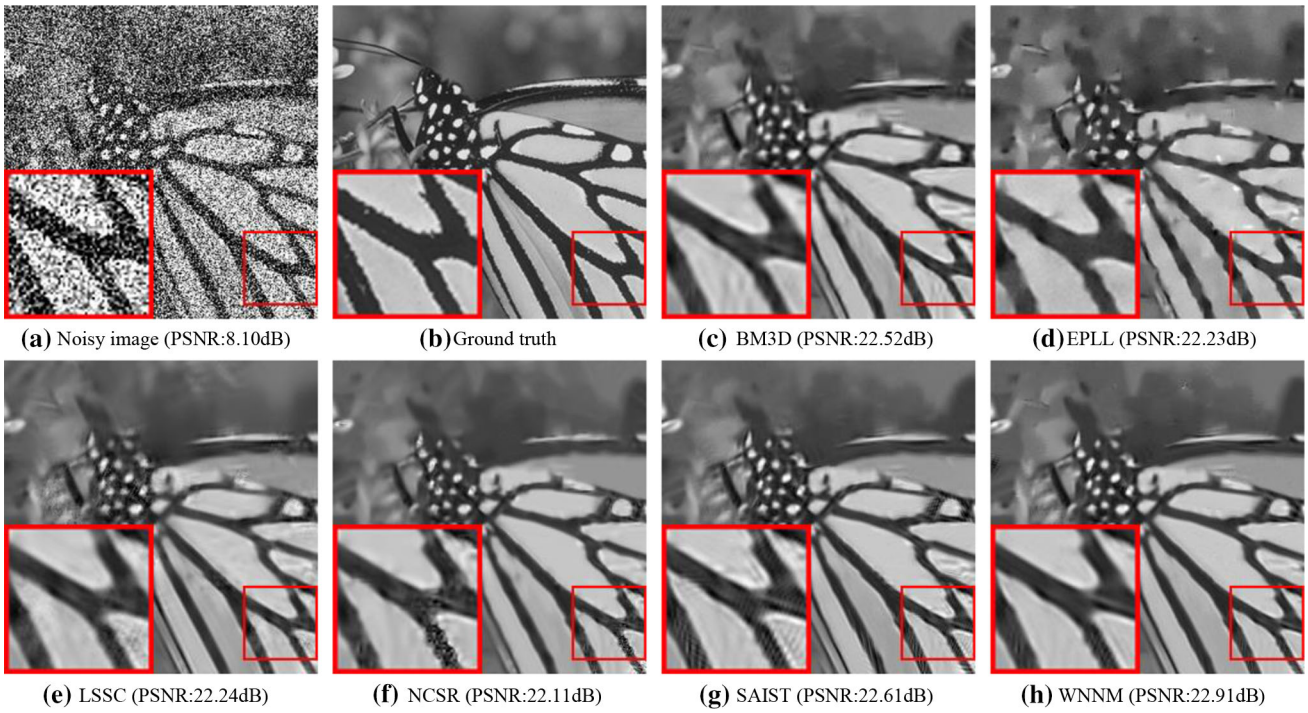


Fig. 3 Denoising results on image *Monarch* by competing methods (noise level $\sigma_n = 100$). The demarcated area is enlarged in the left bottom corner for better visualization. The figure is better seen by zooming on a computer screen

noise level is unknown for real noisy images, some noise estimation method is needed to estimate the noise level in the image. We adopted the classical method in (Donoho 1995) to estimate the noise level, and the estimated noise level was used for all the competing methods. Figs. 4 and 5 compare the denoising results by the competing methods on the two images. One can see that the proposed WNNM method keeps well the local structures in the images and generates the least visual artifacts among the competing methods.

5 WNNM-RPCA for Background Subtraction

SVD/PCA aims to capture the principal (affine) subspace along which the data variance can be maximally captured. It has been widely used in the area of data modeling, compression, and visualization. In the conventional PCA model, the error is measured under the ℓ_2-norm fidelity, which is optimal to suppress additive Gaussian noise. However, there are occasions that outliers or sparse noise are corrupted in data, which may disable SVD/PCA in estimating the ground truth subspace. To address this problem, multiple RPCA models have been proposed to robustify PCA, and have been attempted in different applications such as structure from motion, ranking and collaborative filtering, face reconstruction and background subtraction (Zhou et al. 2014).

Recently, (Candès et al. 2011) proposed the NNM-RPCA model which not only can be efficiently solved by ADMM, but also can guarantee the exact reconstruction of the original data under certain conditions. In this paper, we propose WNNM-RPCA to further enhance the flexibility of NNM-RPCA. In the following, we first design synthetic simulations to comprehensively compare the performance between WNNM-RPCA and NNM-RPCA, and then show the superiority of the proposed method in background subtraction by comparing with more typical low-rank learning methods designed for this task.

5.1 Experimental Results on Synthetic Data

To quantitatively evaluate the performance of the proposed WNNM-RPCA model, we generate synthetic low rank matrix recovering simulations for testing. The ground truth low rank data matrix $\boldsymbol{X} \in \Re^{m\times m}$ is obtained by the multiplication of two low rank matrices: $\boldsymbol{X} = \boldsymbol{A}\boldsymbol{B}^T$, where $\boldsymbol{A}$ and $\boldsymbol{B}$ are both of size $m \times r$. Here $r = p_r \times m$ constrains the upper bound of $Rank(\boldsymbol{X})$. In all experiments, each element of $\boldsymbol{A}$ and $\boldsymbol{B}$ is generated from a Gaussian distribution $\mathcal{N}(0, 1)$. The ground truth matrix $\boldsymbol{X}$ is corrupted by sparse noise $\boldsymbol{E}$ which has $p_e \times m^2$ non-zero entries. The non-zero entries in $\boldsymbol{E}$ are located in random positions and the value of each non-zero element is generated from a uniform distribution between $[-5, 5]$. We set $m = 400$, and let both p_r and p_e vary from 0.01 to 0.5 with step length 0.01. For each parameter setting $\{p_r, p_e\}$, we generate the synthetic low-rank

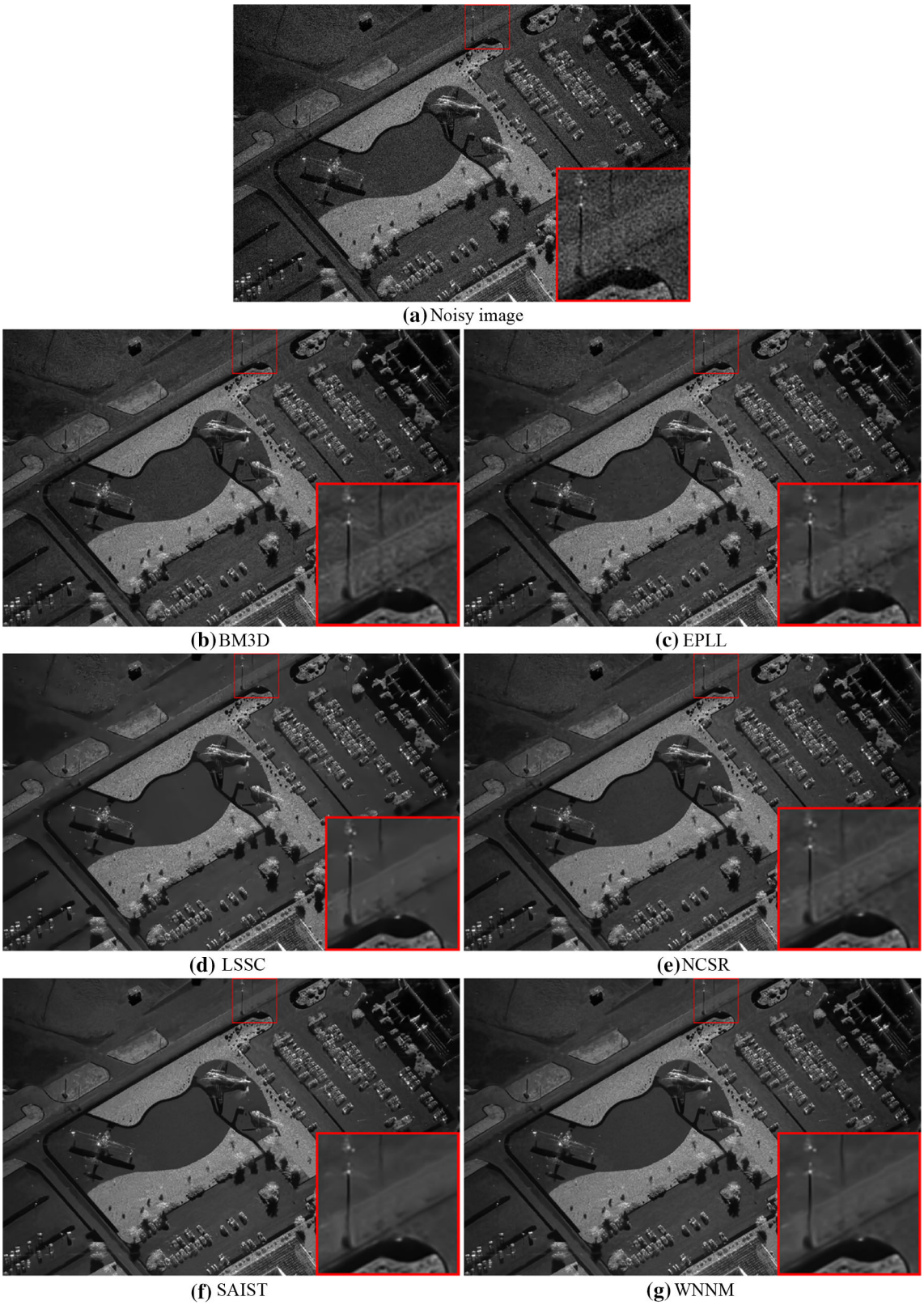


Fig. 4 Denoising results on a real SAR image by all competing methods

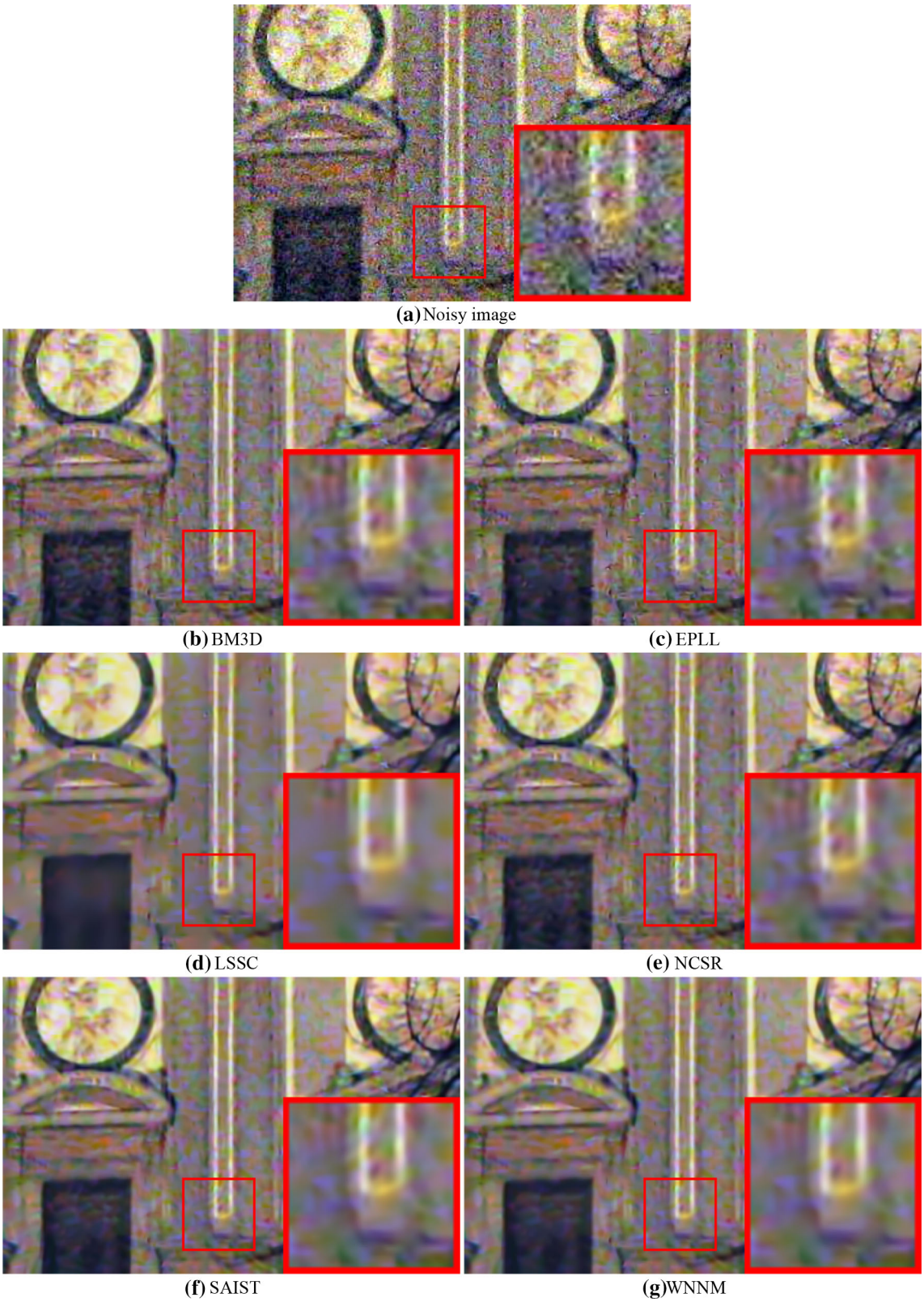


Fig. 5 Denoising results on a real *color* image by all competing methods

Table 3 Relative error of low rank matrix recovery results by NNM-RPCA and WNNM-RPCA, with p_e fixed as 0.05, and p_r varying from 0.05 to 0.45 with step length 0.05

$Rank(\boldsymbol{X})$	20	40	60	80	100	120	140	160	180
NNM-RPCA	2.41e−8	3.91e−8	5.32e−8	7.91e−8	2.90e−4	1.72e−2	6.49e−2	0.13	0.21
WNNM-RPCA	1.79e−8	3.49e−8	5.83e−8	6.53e−8	9.28e−8	1.30e−7	1.68e−7	2.02e−7	2.43e−7

Table 4 Relative error of low rank matrix recovery results by NNM-RPCA and WNNM-RPCA, with p_e fixed as 0.1, and p_r varying from 0.05 to 0.45 with step length 0.05

$Rank(\boldsymbol{X})$	20	40	60	80	100	120	140	160	180
NNM-RPCA	2.26e−8	4.58e−8	7.44e−8	2.50e−4	2.31e−2	6.16e−2	9.96e−2	0.15	0.22
WNNM-RPCA	2.34e−8	3.71e−8	6.03e−8	8.87e−8	1.37e−7	1.82e−7	2.24e−7	4.80e−3	2.41e−2

Table 5 Relative error of low rank matrix recovery results by NNM-RPCA and WNNM-RPCA, with p_e fixed as 0.2, and p_r varying from 0.05 to 0.45 with step length 0.05

$Rank(\boldsymbol{X})$	20	40	60	80	100	120	140	160	180
NNM-RPCA	4.22e−8	6.84e−8	8.89e−3	5.80e−2	9.29e−2	0.12	0.14	0.18	0.24
WNNM-RPCA	3.68e−8	6.09e−8	1.18e−7	1.72e−7	3.76e−4	2.94e−2	5.42E−2	6.82e−2	7.53e−2

matrix 10 times and the final results are measured by the average of these 10 runs.

For the NNM-RPCA model, there is an important parameter λ. We set it as $1/\sqrt{m}$ following the suggested setting of (Candès et al. 2011). For our WNNM-RPCA model, the parameter C is empirically set as the square root of matrix size, i.e., $C = \sqrt{m \times m} = m$, in all experiments. The parameters in ADMM for both methods are set as $\rho = 1.05$. Typical experimental results are listed in Tables 3, 4, and 5 for easy comparison.

It is easy to see that when the rank of matrix is low or the number of corrupted entries is small, both NNM-RPCA and WNNM-RPCA models are able to deliver accurate estimation of the ground truth matrix. However, with the rank of matrix or the number of corrupted entries getting larger, NNM-RPCA fails to deliver an accurate estimation of the ground truth matrix. Yet the error of the results by WNNM-RPCA is much smaller than NNM-RPCA in these cases. In Fig. 6, we show the log-scale relative error map of the recovered matrices by NNM-RPCA and WNNM-RPCA with different settings of $\{p_r, p_e\}$. It is clear that the success area of WNNM-RPCA is much larger than NNM-RPCA, which means that WNNM-RPCA has much better low-rank matrix reconstruction capability in the presence of outliers/sparse noise.

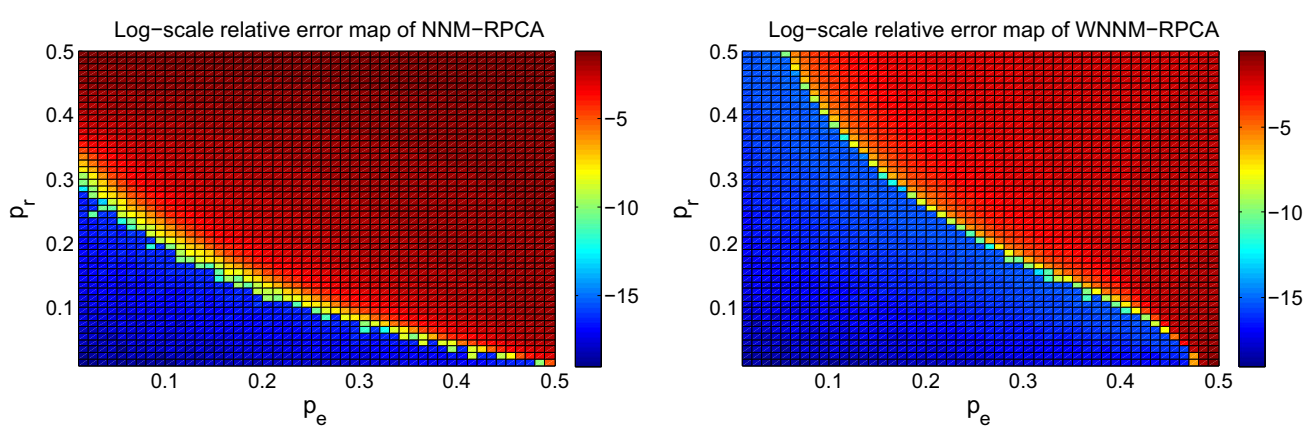


Fig. 6 The log-scale relative error $log\frac{\|\hat{\boldsymbol{X}}-\boldsymbol{X}\|_F^2}{\|\boldsymbol{X}\|_F^2}$ of NNM-RPCA and WNNM-RPCA with different rank and outlier rate settings $\{p_r, p_e\}$

5.2 Experimental Results on Background Subtraction

As an important application in video surveillance, background sbutraction refers to the problem of separating the moving objects in foreground and the stable scene in background. The matrix $\boldsymbol{Y}$ obtained by stacking the video frames as columns corresponds to a low-rank matrix with stationary background corrupted by the sparse moving objects in the foreground. Thus, RPCA model is appropriate to deal with this problem. We compare WNNM-RPCA with NNM-RPCA and several representative low-rank learning models, including the classic iteratively reweighted least squares (IRLS) based RPCA model[9] (De La Torre and Black 2003), and the ℓ_2-norm and ℓ_1-norm based matrix factorization models: singular value decomposition (SVD), Bayesian robust matrix factorization (BRMF)[10] (Wang and Yeung 2013) and RegL1ALM[11] (Zheng et al. 2012). The results of a recently proposed Mixture of Gaussian (MoG) model[12] (Meng and Torre 2013; Zhao et al. 2014) are also included. These comparison methods range over state-of-the-art ℓ_2 norm, ℓ_1 norm and probabilistic subspace learning methods, including both categories of rank minimization and LRMF based approaches. We downloaded the codes of these algorithms from the corresponding authors' websites and keep their initialization and stopping criteria unchanged. The code of the proposed WNNM-RPCA model can be downloaded at http://www4.comp.polyu.edu.hk/cslzhang/code/WNNM_RPCA_code.zip.

Four benchmark video sequences provided by Li et al. (2004) were adopted in our experiments, including two outdoor scenes (*Fountain* and *Watersurface*) and two indoor scenes (*Curtain* and *Airport*). In each sequence, 20 frames of ground truth foreground regions were provided by Li et al. for quantitative comparison. For all the comparison methods, parameters are fixed on the four sequences. We follow the experimental setting in Zhao et al. (2014) and constrain the maximum rank to 6 for all the factorization based methods. The regularization parameter λ for the ℓ_1-norm sparse term in the NNM-RPCA model is set to $\frac{1}{2\sqrt{max(m,n)}}$, since we empirically found that it can perform better than the recommended parameter $\frac{1}{\sqrt{max(m,n)}}$ in the original paper (Candès et al. 2011) in this series of experiments. For the proposed WNNM-RPCA model, we set $C = \sqrt{2max(m^3, n^3)}$ in all experiments.

To quantitatively compare the performance of competing methods, we use $S(A, B) = \frac{A \cap B}{A \cup B}$ to measure the similarity between the estimated foreground regions and the ground truth ones. To generate the binary foreground map, we applied the Markov random field (MRF) model to label the absolute value of the estimated sparse error. The MRF labeling problem was solved by the multi-label optimization tool box (Boykov et al. 2001). The quantitative results of S by different methods are shown in Table 6. One can see that on all the four utilized sequences, the proposed WNNM-RPCA model outperforms all other competing methods.

Table 6 Quantitative performance (S) comparison of background subtraction results obtained by different methods

Method	Watersurface	Fountain	Airport	Curtain
SVD	0.0995	0.2840	0.4022	0.1615
IRLS	0.4917	0.4894	0.4128	0.3524
BRMF	0.5786	0.5840	0.4694	0.5998
RegL1ALM	0.1346	0.4248	0.4420	0.2983
MoG	0.2782	0.4342	0.4921	0.3332
NNM-RPCA	0.7703	0.5859	0.3782	0.3191
WNNM-RPCA	**0.7884**	**0.6043**	**0.5144**	**0.7863**

The best results are highlighted in bold

The visual results of representative frames in the *Watersurface* and *Curtain* sequences are shown in Figs. 7 and 8. The visual result of the other two sequences can be found in the supplementary file. From these figures, we can see that WNNM-RPCA method is able to deliver clear background estimation even under prominently embedded foreground moving objects. This on the other hand facilitates a more accurate foreground estimation. Comparatively, in the results estimated by the other methods, there are some ghost shadows in the background, leading to relatively less complete foreground detection results.

6 WNNM-MC for Image Inpainting

Matrix completion refers to the problem of recovering a matrix from only partial observation of its entries. It is a well known ill-posed problem which needs prior of the ground truth matrix as supplementary information for reconstruction. Fortunately, in many practical instance, the matrix to be recovered has a low-rank structure. Such a prior knowledge has been utilized in many low-rank matrix completion (LRMC) methods, such as ranking and collaborative filtering (Ruslan and Srebro 2010) and image inpainting (Zhang et al. 2012a). Matrix completion can be solved by both the matrix factorization or the rank minimization approaches. As the exact recovery property of the NNM-based methods has been proved by (Candès and Recht 2009), this methodology has received great research interest, and many algorithms have been proposed to solve the NNM-MC problem (Cai

[9] http://www.cs.cmu.edu/ftorre/codedata.html

[10] http://winsty.net/brmf.html

[11] http://sites.google.com/site/yinqiangzheng/

[12] http://www.cs.cmu.edu/~deyum/Publications.htm

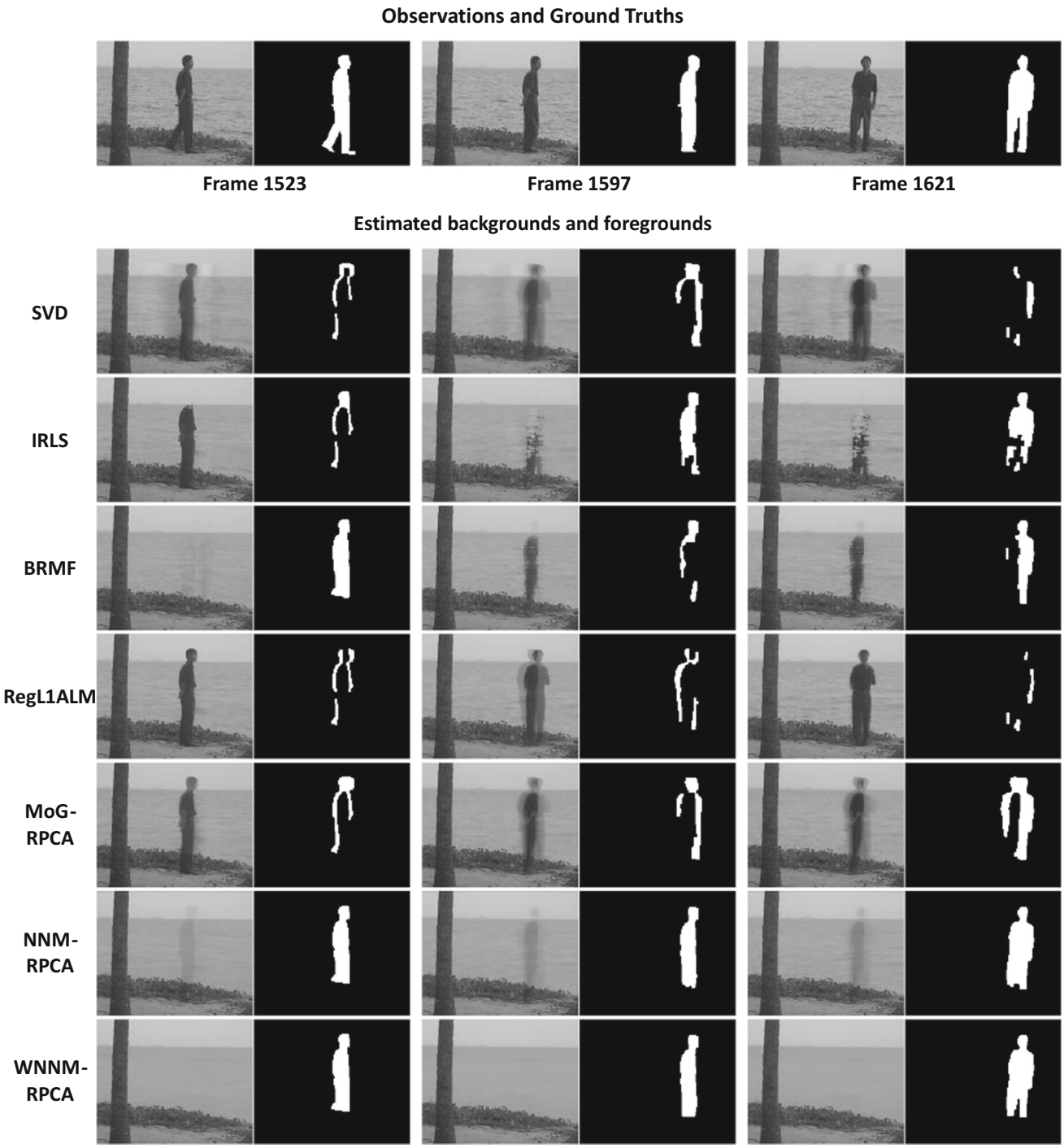


Fig. 7 Performance comparison in visualization of competing methods on the *Watersurface* sequence. *First row* the original frames and annotated ground truth foregrounds. *Second row* to the *last row* estimated backgrounds and foregrounds by SVD, IRLS, BRMF, RegL1ALM, MoGRPCA, NNM-RPCA and WNNM-RPCA, respectively

et al. 2010; Lin et al. 2011). In the following, we provide experimental results on synthetic data and image inpainting to show the superiority of the proposed WNNM-MC model to the traditional NNM-MC technology.

6.1 Experimental Results on Synthetic Data

We first compare NNM-MC with WNNM-MC using synthetic low-rank matrices. Similar to our experimental setting

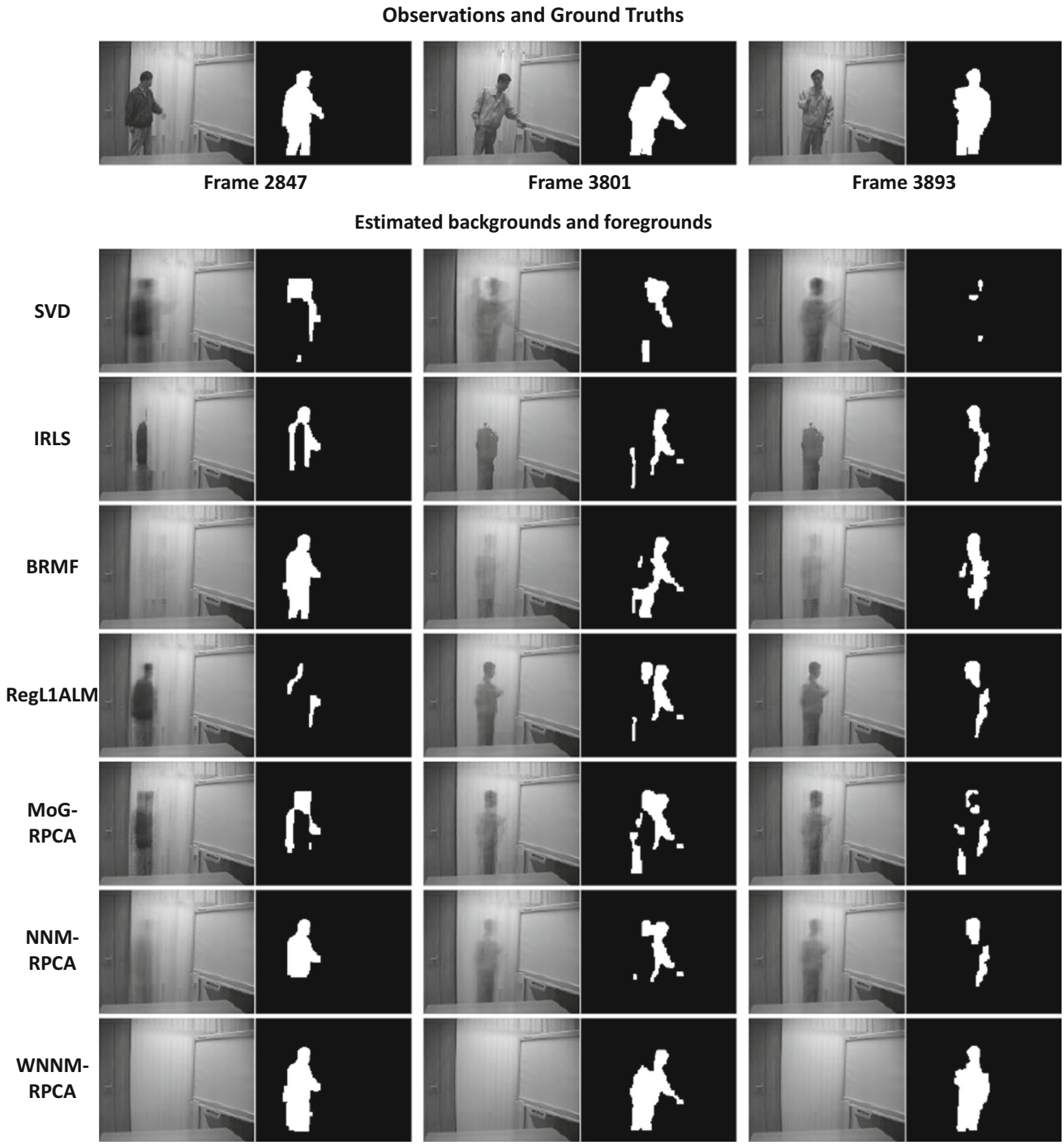


Fig. 8 Performance comparison in visualization of competing methods on the *Curtain* sequence. *First row* the original frames and annotated ground truth foregrounds. *Second row* to the *last row* estimated backgrounds and foregrounds by SVD, IRLS, BRMF, RegL1ALM, MoGRPCA, NNM-RPCA and WNNM-RPCA, respectively

in the RPCA problem, we generate the ground truth low-rank matrix by a multiplication between two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ of size $m \times r$. Here $r = p_r \times m$ constrains the upper bound of $Rank(\boldsymbol{X})$. All of their elements are generated from the Gaussian distribution $\mathcal{N}(0, 1)$. In the observed matrix $\boldsymbol{Y}$, $p_e \times m^2$ entries in the ground truth matrix $\boldsymbol{X}$ are missing. We set $m = 400$, and let p_r and p_e vary from 0.01 to 0.5 with step length 0.01. For each parameter setting of $\{p_r, p_e\}$, 10 groups of synthetic data are generated for testing and the performance of each method is assessed by the average of the 10 runs on these groups.

Table 7 Relative error of low rank matrix recovery results by NNM-MC and WNNM-MC, with p_e fixed as 0.1, and p_r varying from 0.05 to 0.45 with step length 0.05

$Rank(\boldsymbol{X})$	20	40	60	80	100	120	140	160	180
NNM-MC	5.51e−8	7.25e−8	9.51e−8	1.12e−7	1.43e−7	1.76e−7	2.10e−7	2.58e−7	9.97e−5
WNNM-MC	4.40e−8	7.72e−8	9.44e−8	1.18e−7	1.41e−7	1.77e−7	2.09e−7	2.53e−7	3.25e−7

Table 8 TRelative error of low rank matrix recovery results by NNM-MC and WNNM-MC, with p_e fixed as 0.2, and p_r varying from 0.05 to 0.45 with step length 0.05

$Rank(\boldsymbol{X})$	20	40	60	80	100	120	140	160	180
NNM-MC	7.33e−8	1.03e−7	1.22e−7	1.65e−7	2.01e−7	2.76e−7	1.91e−2	8.52e−2	0.14
WNNM-MC	6.35e−8	9.21e−8	1.30e−7	1.60e−7	1.94e−7	2.48e−7	3.32e−7	4.66e−7	7.21e−7

Table 9 Relative error of low rank matrix recovery results by NNM-MC and WNNM-MC, with p_e fixed as 0.3, and p_r varying from 0.05 to 0.45 with step length 0.05

$Rank(\boldsymbol{X})$	20	40	60	80	100	120	140	160	180
NNM-MC	9.20e−8	1.21e−7	1.61e−7	2.06e−7	0.53e−5	8.94e−2	0.18	0.25	0.30
WNNM-MC	9.31e−8	1.21e−7	1.60e−7	2.13e−7	2.81e−7	4.00e−7	6.15e−7	1.71e−2	0.22

In all the experiments we fix parameters $\lambda = 1/\sqrt{m}$ and $C = m$ in NNM-MC and WNNM-MC, respectively. The parameter ρ in the ALM algorithm is set to 1.05 for both methods. Typical experimental results are listed in Tables 7, 8, and 9.

It can be easily observed that when the rank of latent ground truth matrix $\boldsymbol{X}$ is relatively low, both NNM-RPCA and WNNM-RPCA can successfully recover it with high accuracy. The advantage of WNNM-MC over NNM-MC is reflected when dealing with more challenging cases. Table 8 shows that when 20 % of entries in the matrix are missing, NNM-MC will not have good recovery accuracy once the rank is higher than 120, while WNNM-MC can still have very high accuracy. Similar observations can be made in Table 9.

The log-scale relative error map with different settings of $\{p_r, p_e\}$ are shown in Fig. 9. From this figure, it is clear to see that WNNM-MC has a much larger success area than NNM-MC.

6.2 Experimental Results on Image Inpainting

We then test the proposed WNNM-MC model on image inpainting. In some previous works, the whole image is assumed to be a low rank matrix and matrix completion is directly performed on the image to get the inpainting result. However, a natural image is only approximately low rank, and the small singular values in the long tail distribution include many details. Simply using the low rank prior on the whole image may fail to recover the missing pixels or lose too much detailed information in the image. As in the image denoising experiments, we utilize the NSS prior and perform WNNM-MC on each group of non-local similar patches for this task. We initialize the inpainting by the field of experts (FOE) (Roth and Black 2009) method to search the non-local patches, and then for each patch group we perform WNNM-MC to get an updated low-rank reconstruction. After the first round estimation of the missing values in all patch groups, all reconstructed patches are aggregated to get the recovered image. We then perform a new stage of similar patch searching based on the first round estimation, and iteratively implement the similar process to converge to a final inpainting output.

The first 12 test images[13] with size 256×256 in Fig. 1 are used to evaluate WNNM-MC. Random masks with 25, 50 and 75 % missing pixels and a text mask are used to test the inpainting performance, respectively. We compare WNNM-MC with NNM-MC and several representative and state-of-the-art inpainting methods, including the TV method (Chan and Shen 2005), FOE method[14] (Roth and Black 2009), variational nonlocal example-based (VNL) method[15] (Arias et al. 2011) and the beta process dictionary learning (BPDL) method[16] (Zhou et al. 2009). The setting of the TV inpainting method follows the implementation of (Dahl et al. 2010)[17] and the codes for other comparison

[13] The color versions of images #3, #5, #6, #7, #9, #11 are used in this MC experiment.

[14] http://www.gris.informatik.tu-darmstadt.de/sroth/research/foe

[15] http://gpi.upf.edu/static/vnli/interp/interp.html

[16] http://people.ee.duke.edu/mz1/Softwares

[17] http://www.imm.dtu.dk/pcha/mxTV/,

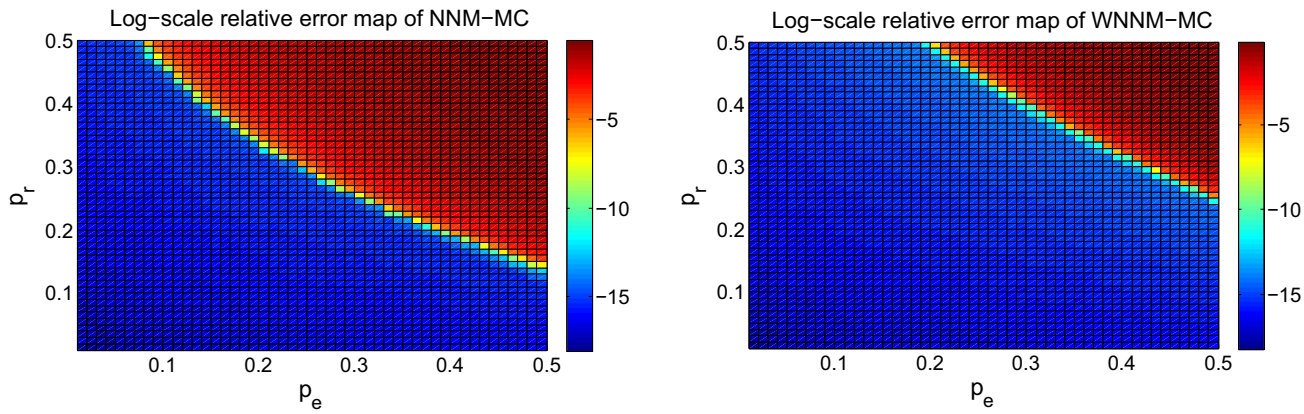


Fig. 9 The log-scale relative error $log\frac{\|\hat{X}-X\|_F^2}{\|X\|_F^2}$ of NNM-MC and WNNM-MC with different rank and outlier rate settings $\{p_r, p_e\}$

methods are provided by the original authors. The source code of the proposed WNNM-MC model can be downloaded at http://www4.comp.polyu.edu.hk/~cslzhang/code/WNNM_MC_code.zip.

The PSNR results by different methods are shown in Table 10. It is easy to see that WNNM-MC achieves much better results than the other methods. Visual examples on a random mask and a text mask are shown in Figs. 10 and 11, respectively. More visual examples can be found in the supplementary file. From the enlarged demarcated windows, we can see that inpainting methods based on image local prior (e.g., TV, FOE and BPDL) are able to recover the image smooth areas, while they have difficulties in extracting the details in edge and texture areas. The VNL, NNM and WNNM methods utilized the rational NSS prior, and thus the results are more visually plausible. However, in some challenging cases when the percentage of missing entries is high, it can be observed that VNL and NNM more or less generate artifacts across the recovered image. As a comparison, the proposed WNNM-MC model has much better visual quality of the inpainting results.

7 Discussions

To improve the flexibility of the original nuclear norm, we proposed the weighted nuclear norm and studied its minimization strategy in this work. Based on the observed data and the specific application, different weight setting strategies can be utilized to achieve better performance. Inspired by (Candès et al. 2008), in this work we utilized the reweighting strategy $w_i^\ell = \frac{C}{|\sigma_i(\mathbf{X}_\ell)|+\varepsilon}$ to approximate the ℓ_0 norm on the singular values of the data matrix. Other than this setting, there also exist other weight setting strategies for certain types of data and applications. For instance, the truncated nuclear norm regularization (TNNR) (Zhang et al. 2012a)

Table 10 Inpainting results (PSNR, dB) by different methods

	Random mask with 25 % missing entries						Random mask with 50 % missing entries					
	TV	FOE	VNL	BPDL	NNM	WNNM	TV	FOE	VNL	BPDL	NNM	WNNM
C.Man	32.20	30.23	26.98	33.39	34.12	**35.21**	27.41	27.42	25.71	28.59	29.42	**30.58**
House	39.37	41.90	33.69	42.03	42.90	**44.59**	34.25	36.84	32.35	37.63	37.45	**38.83**
Peppers	37.44	38.46	31.00	39.66	39.65	**41.53**	32.16	34.51	28.80	34.80	34.02	**35.85**
Montage	32.28	28.00	28.45	35.86	37.48	**39.63**	26.47	24.53	26.41	29.68	29.91	**31.02**
Leaves	32.10	30.52	27.57	36.77	36.27	**38.95**	26.07	27.22	25.32	30.35	29.87	**32.32**
StarFish	34.20	35.34	27.44	36.94	36.79	**38.93**	29.05	31.18	26.13	31.84	31.36	**33.03**
Monarch	34.27	32.92	28.55	36.74	36.51	**38.14**	28.84	29.16	26.84	31.17	31.02	**32.75**
Airplane	30.80	30.34	26.05	33.19	32.94	**33.76**	26.61	28.11	24.62	29.00	28.72	**29.30**
Paint	34.75	35.87	28.87	37.27	36.84	**38.79**	29.29	31.45	27.22	32.62	31.52	**33.02**
J.Bean	41.56	44.57	34.97	45.15	44.49	**48.04**	35.61	38.88	32.46	40.07	37.47	**40.93**
Fence	30.24	31.96	28.98	35.85	36.55	**37.91**	25.06	29.97	27.43	31.47	31.77	**32.85**
Parrot	32.88	30.69	27.50	33.37	33.93	**35.09**	27.77	28.24	25.80	28.73	29.20	**30.52**
AVE.	34.340	34.234	29.170	37.185	37.372	**39.215**	29.049	30.625	27.425	32.162	31.812	**33.416**

Table 10 continued

	Random mask with 75 % missing entries						Text mask					
	TV	FOE	VNL	BPDL	NNM	WNNM	TV	FOE	VNL	BPDL	NNM	WNNM
C-Man	23.65	23.35	23.65	24.17	23.55	**25.69**	28.49	27.27	26.75	26.24	29.74	**31.68**
House	29.59	31.63	31.07	31.61	28.62	**34.12**	34.75	37.41	34.46	32.01	37.69	**39.60**
Peppers	27.16	28.86	26.72	28.97	27.22	**30.04**	33.90	34.82	31.72	29.57	34.55	**37.04**
Montage	22.47	21.92	23.62	24.89	23.35	**25.65**	27.20	26.06	27.54	25.62	29.77	**31.47**
Leaves	20.15	20.69	22.30	23.11	22.08	**25.01**	25.48	24.21	26.39	23.07	27.32	**30.69**
Starfish	24.44	26.43	24.36	26.57	25.45	**27.11**	29.39	30.97	27.53	27.24	31.66	**33.44**
Mornar.	23.97	24.10	24.77	26.08	24.84	**27.45**	28.35	27.51	27.56	26.10	29.89	**32.85**
Plane	23.16	24.20	22.03	24.87	24.05	**25.47**	28.70	28.79	25.95	26.61	29.66	**30.57**
Paint	24.05	25.81	24.73	26.71	25.02	**26.93**	29.54	30.50	28.47	31.86	31.32	**32.96**
J.Bean	29.95	32.89	29.48	32.46	28.00	**33.40**	34.50	37.12	32.59	32.62	35.71	**38.91**
Fence	21.03	23.56	25.49	26.12	25.65	**28.38**	25.44	27.55	29.55	27.01	32.44	**34.62**
Parrot	22.85	23.33	23.54	24.53	24.02	**25.61**	28.07	27.54	26.58	25.83	29.54	**30.61**
AVE.	24.372	25.564	25.146	26.674	25.155	**27.905**	29.484	29.979	28.757	27.814	31.606	**33.702**

The best results are highlighted in bold

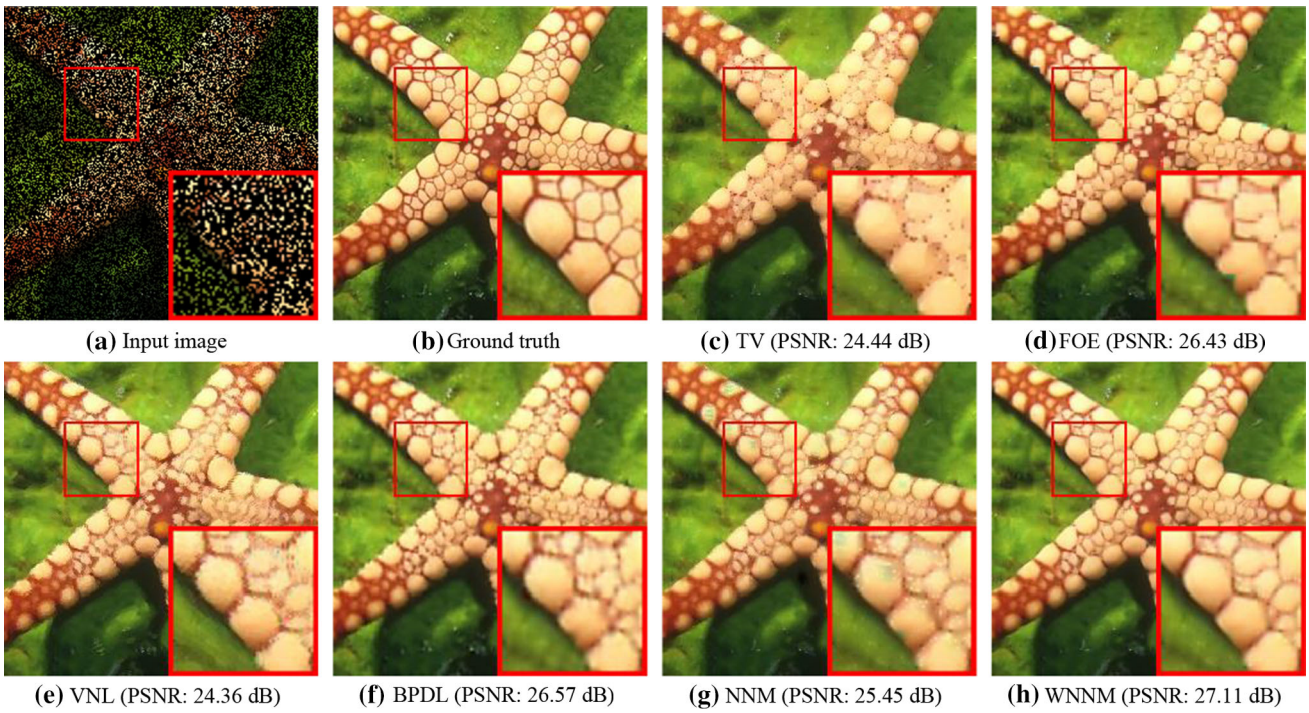


Fig. 10 Inpainting results on image *Starfish* by different methods (Random mask with 75 % missing values)

and the partial sum minimization (PSM) (Oh et al. 2013) were proposed to regularize only the smallest $N - r$ singular values. Actually, they can be viewed as a special case of WNNM, where weight vector $[w_{1\cdots r} = 0, w_{r+1\cdots N} = \lambda]$ is used to approximate the rank function of matrix. The Schatten-p norm minimization (SPNM) methods (Nie et al. 2012,Mohan and Fazel 2012) can also be understood as special cases of WNNM since the ℓ_p norm proximal problem can be solved by the iteratively thresholding method (She 2012). The same strategy as in our work can be used to set weights for each singular values.

In Tables 11, 12 and 13, we compare the proposed WNNM with the TNNR and SPNM methods on image denosing, background subtraction and image inpainting, respectively. The results obtained by the NNM method are also shown in the tables as baseline comparison. The experimental settings on the three applications are exactly the same as the experiments in Sects. 4, 5 and 6. For the TNNR method, there

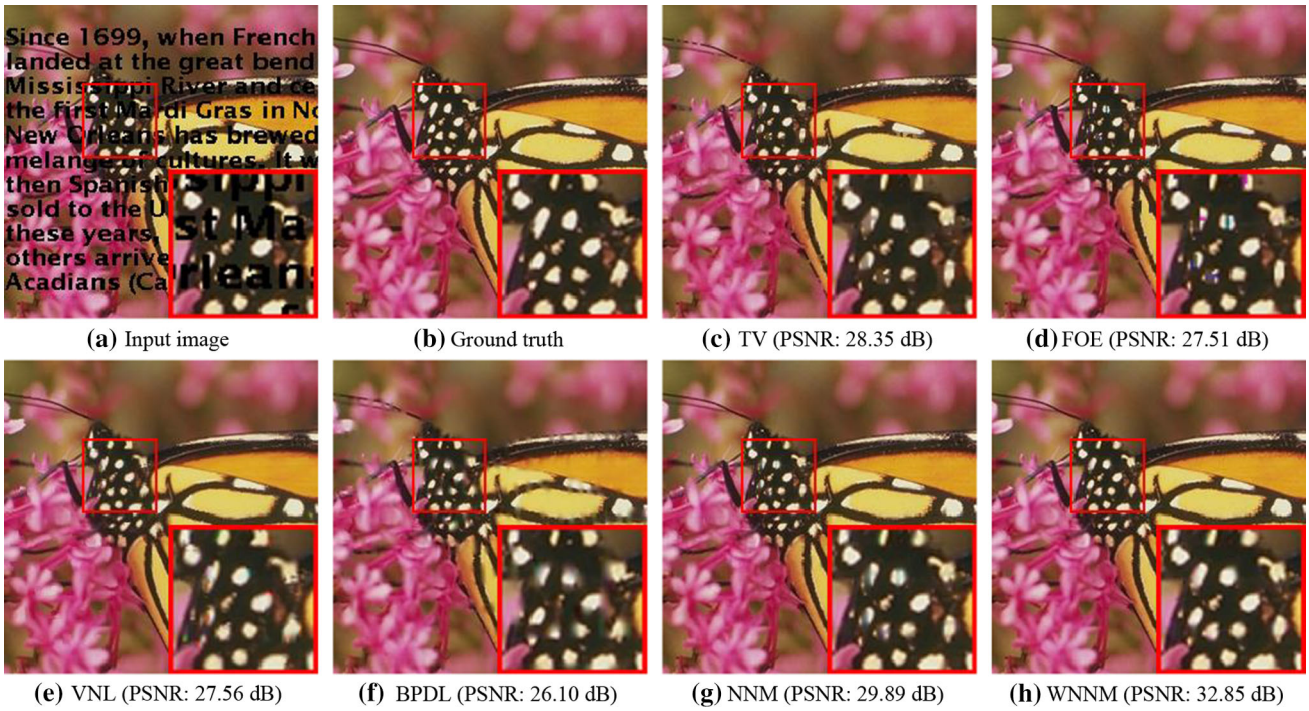


Fig. 11 Inpainting results on image *Monarch* by different methods (*Text mask*)

Table 11 The average PSNR (dB) values of denoising results by competing methods on the 20 test images

	σ_n=10	σ_n=20	σ_n=30	σ_n=40	σ_n=50	σ_n=75	σ_n=100
NNM	33.462	30.040	27.753	26.422	25.048	22.204	21.570
TNNM	34.125	30.558	28.627	27.250	26.256	24.284	23.109
SPNM	34.739	31.048	29.075	27.488	26.582	24.757	23.174
WNNM	**34.772**	**31.200**	**29.214**	**27.780**	**26.752**	**24.860**	**23.555**

The best results are highlighted in bold

Table 12 The average PSNR (dB) values of inpainting results by competing methods on the 12 test images

	25 % missing entries	50 % missing entries	75 % missing entries	Text mask
NNM	37.372	31.812	25.155	31.606
TNNM	37.789	32.378	27.201	32.738
SPNM	39.203	33.196	27.800	33.668
WNNM	**39.215**	**33.416**	**27.905**	**33.702**

The best results are highlighted in bold

Table 13 Quantitative performance (S) comparison of background subtraction results obtained by different methods

Method	Watersurface	Fountain	Airport	Curtain
NNM-RPCA	0.7703	0.5859	0.3782	0.3191
TNNM-RPCA	0.7772	0.5926	0.3829	0.3310
SPNM-RPCA	**0.7906**	0.6033	0.3714	0.3233
WNNM-RPCA	0.7884	**0.6043**	**0.5144**	**0.7863**

The best results are highlighted in bold

are two parameters r and λ, which represent the truncation position and the regularization parameter for the remaining $N - r$ singular values. For the SPNM model, we need to set the p value for the ℓ_p norm and the regularization parameter γ. We tried our best to adjust these parameters for the two models for different applications. The remaining parameters are the same as the WNNM model on each task.

From the tables, we can find that regularizing more flexibly the singular values is beneficial for low rank models. Besides, WNNM outperforms TNNR and SPNM in the testing applications, which substitutes the superiority of the proposed method along this line of research. Nonetheless, it is worth to note that there may exist other weighting mechanisms which might achieve better performance on certain computer vision

tasks. Actually, since we have proved in Theorem 1 that the WNNP problem is equivalent to an element wise proximal problem with order constrains, a wide range of weighting and reweighting strategies can be used to regularize the singular values of a data matrix. One important research problem of our future work is thus to investigate more sophisticated weight setting mechanisms for different types of data and applications.

8 Conclusion

We studied the weighted nuclear norm minimization problem (WNNM) in this paper. We first presented the solving strategy for the weighted nuclear norm proximal (WNNP) operator under ℓ_2-norm fidelity loss to facilitate the solving of different WNNM paradigms. We then extended the WNNM model to robust PCA (RPCA) and matrix completion (MC) tasks and constructed efficient algorithms to solve them based on the derived WNNP operator. Inspired by previous results on reweighted sparse coding, we further designed a rational scheme for automatic weight setting, which offers closed form solutions of the WNNP operator and ease the utilization of WNNM in different applications.

We validated the effectiveness of the proposed WNNM models on multiple low level vision tasks. For baseline WNNM, we applied it to image denoising, and achieved state-of-the-art performance in both quantitative PSNR measures and visual qualities. For WNNM-RPCA, we applied it to background subtraction, and validated its superiority among up-to-date low-rank learning methods. For WNNM-MC, we applied it to image inpainting, and demonstrate its superior performance among state-of-the-art inpainting technologies.

Acknowledgements This work is supported by the Hong Kong RGC GRF grant (PolyU 5313/13E).

9 Appendix

In this appendix, we provide the proof details of the theoretical results in the main text.

9.1 Proof of Theorem 1

Proof For any $\boldsymbol{X}, \boldsymbol{Y} \in \Re^{m\times n}(m > n)$, denote by $\bar{\boldsymbol{U}}\boldsymbol{D}\bar{\boldsymbol{V}}^T$ and $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ the singular value decomposition of matrix $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively, where $\boldsymbol{\Sigma} = \begin{pmatrix} diag(\sigma_1, \sigma_2, ..., \sigma_n) \\ \mathbf{0} \end{pmatrix} \in \Re^{m\times n}$, and $\boldsymbol{D} = \begin{pmatrix} diag(d_1, d_2, ..., d_n) \\ \mathbf{0} \end{pmatrix}$ are the diagonal singular value matrices. Based on the property of Frobenius norm, the following derivations hold:

$$\begin{aligned}
&\|\boldsymbol{Y} - \boldsymbol{X}\|_F^2 + \|\boldsymbol{X}\|_{w,*} \\
&= Tr\left(\boldsymbol{Y}^T\boldsymbol{Y}\right) - 2Tr\left(\boldsymbol{Y}^T\boldsymbol{X}\right) + Tr\left(\boldsymbol{X}^T\boldsymbol{X}\right) + \sum_i^n w_i d_i \\
&= \sum_i^n \sigma_i^2 - 2Tr\left(\boldsymbol{Y}^T\boldsymbol{X}\right) + \sum_i^n d_i^2 + \sum_i^n w_i d_i.
\end{aligned}$$

Based on the von Neumann trace inequality in Lemma 1, we know that $Tr\left(\boldsymbol{Y}^T\boldsymbol{X}\right)$ achieves its upper bound $\sum_i^n \sigma_i d_i$ if $\boldsymbol{U} = \bar{\boldsymbol{U}}$ and $\boldsymbol{V} = \bar{\boldsymbol{V}}$. Then, we have

$$\begin{aligned}
&\min_{\boldsymbol{X}} \|\boldsymbol{Y} - \boldsymbol{X}\|_F^2 + \|\boldsymbol{X}\|_{w,*} \\
&\Leftrightarrow \min_{\boldsymbol{D}} \sum_i^n \sigma_i^2 - 2\sum_i^n \sigma_i d_i + \sum_i^n d_i^2 + \sum_i^n w_i d_i \\
&s.t. d_1 \geq d_2 \geq ... \geq d_n \geq 0 \\
&\Leftrightarrow \min_{\boldsymbol{D}} \sum_i (d_i - \sigma_i)^2 + w_i d_i \\
&s.t.\ d_1 \geq d_2 \geq ... \geq d_n \geq 0.
\end{aligned}$$

From the above derivation, we can see that the optimal solution of the WNNP problem in (5) is

$$\boldsymbol{X}^* = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T,$$

where $\boldsymbol{D}$ is the optimum of the constrained quadratic optimization problem in (6).

End of proof. □

9.2 Proof of Corollary 1

Proof Without considering the constraint, the optimization problem (6) degenerates to the following unconstrained formula:

$$\begin{aligned}
&\min_{d_i \geq 0} (d_i - \sigma_i)^2 + w_i d_i \\
&\Leftrightarrow \min_{d_i \geq 0} \left(d_i - (\sigma_i - \frac{w_i}{2})\right)^2.
\end{aligned}$$

It is not difficult to derive its global optimum as:

$$\bar{d}_i = max\left(\sigma_i - \frac{w_i}{2}, 0\right),\ i = 1, 2, ..., n. \tag{15}$$

Since we have $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n$ and the weight vector has a non-descending order $w_1 \leq w_2 \leq ... \leq w_n$, it is easy to see that $\bar{d}_1 \geq \bar{d}_2 \geq ... \geq \bar{d}_n$. Thus, $\bar{d}_{i=1,2,...,n}$ satisfy the constraint of (6), and the solution in (15) is then the globally optimal solution of the original constrained problem in (6).

End of proof. □

9.3 Proof of Theorem 2

Proof Denote by $\boldsymbol{U}_k\boldsymbol{\Lambda}_k\boldsymbol{V}_k^T$ the SVD of the matrix $\{\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{E}_{k+1}\}$ in the $(k+1)$-th iteration, where $\boldsymbol{\Lambda}_k = \{diag(\sigma_k^1, \sigma_k^2, ..., \sigma_k^n)\}$ is the diagonal singular value matrix. Based on the conclusion of Corollary 1, we have

$$\boldsymbol{X}_{k+1} = \boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^T, \tag{16}$$

where $\boldsymbol{\Sigma}_k = \mathcal{S}_{\mathbf{w}/\mu_k}(\boldsymbol{\Lambda}_k)$ is the singular value matrix after weighted shrinkage. Based on the Lagrange multiplier updating method in step 5 of Algorithm 1, we have

$$\begin{aligned}
\|\boldsymbol{L}_{k+1}\|_F &= \|\boldsymbol{L}_k + \mu_k(\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1})\|_F \\
&= \mu_k\|\mu_k^{-1}\boldsymbol{L}_k + \boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1}\|_F \\
&= \mu_k\|\boldsymbol{U}_k\boldsymbol{\Lambda}_k\boldsymbol{V}_k^T - \boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^T\|_F \\
&= \mu_k\|\boldsymbol{\Lambda}_k - \boldsymbol{\Sigma}_k\|_F \\
&= \mu_k\|\boldsymbol{\Lambda}_k - \mathcal{S}_{\mathbf{w}/\mu_k}(\boldsymbol{\Lambda}_k)\|_F \\
&\le \mu_k\sqrt{\sum_i\left(\frac{w_i}{\mu_k}\right)^2} \\
&= \sqrt{\sum_i w_i^2}.
\end{aligned} \tag{17}$$

Thus, $\{\boldsymbol{L}_k\}$ is bounded.

To analyze the boundedness of $\Gamma(\boldsymbol{X}^{k+1}, \boldsymbol{E}^{k+1}, \boldsymbol{L}^k, \mu^k)$, first we can see the following inequality holds because in step 3 and step 4 we have achieved the globally optimal solutions of the $\boldsymbol{X}$ and $\boldsymbol{E}$ subproblems:

$$\Gamma(\boldsymbol{X}_{k+1}, \boldsymbol{E}_{k+1}, \boldsymbol{L}_k, \mu_k) \le \Gamma(\boldsymbol{X}_k, \boldsymbol{E}_k, \boldsymbol{L}_k, \mu_k).$$

Then, based on the way we update $\boldsymbol{L}$:

$$\boldsymbol{L}_{k+1} = \boldsymbol{L}_k + \mu_k(\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1}),$$

there is

$$\begin{aligned}
&\Gamma(X_k, E_k, L_k, \mu_k) \\
&\quad= \Gamma(X_k, E_k, L_{k-1}, \mu_{k-1}) \\
&\qquad+\frac{\mu_k - \mu_{k-1}}{2}\|Y - X_k - E_k\|_F^2 \\
&\qquad+\langle L_k - L_{k-1}, Y - X_k - E_k\rangle \\
&\quad= \Gamma(X_k, E_k, L_{k-1}, \mu_{k-1}) \\
&\qquad+\frac{\mu_k - \mu_{k-1}}{2}\left\|\mu_{k-1}^{-1}(L_k - L_{k-1})\right\|_F^2 \\
&\qquad+\left\langle L_k - L_{k-1}, \mu_{k-1}^{-1}(L_k - L_{k-1})\right\rangle \\
&\quad= \Gamma(X_k, E_k, L_{k-1}, \mu_{k-1}) \\
&\qquad+\frac{\mu_k + \mu_{k-1}}{2\mu_{k-1}^2}\|L_k - L_{k-1}\|_F^2.
\end{aligned}$$

Denote by Θ the upper bound of $\|\boldsymbol{L}_k - \boldsymbol{L}_{k-1}\|_F^2$ for all $\{k = 1, \ldots, \infty\}$. We have

$$\begin{aligned}
\Gamma(\boldsymbol{X}_{k+1}, \boldsymbol{E}_{k+1}, \boldsymbol{L}_k, \mu_k) \le& \Gamma(\boldsymbol{X}_1, \boldsymbol{E}_1, \boldsymbol{L}_0, \mu_0) \\
&+ \Theta\sum_{k=1}^{\infty}\frac{\mu_k + \mu_{k-1}}{2\mu_{k-1}^2}.
\end{aligned}$$

Since the penalty parameter $\{\mu_k\}$ satisfies $\sum_{k=1}^{\infty}\mu_k^{-2}\mu_{k+1} < +\infty$, we have

$$\sum_{k=1}^{\infty}\frac{\mu_k + \mu_{k-1}}{2\mu_{k-1}^2} \le \sum_{k=1}^{\infty}\mu_{k-1}^{-2}\mu_k < +\infty.$$

Thus, we know that $\Gamma(\boldsymbol{X}^{k+1}, \boldsymbol{E}^{k+1}, \boldsymbol{L}^k, \mu^k)$ is also upper bounded.

The boundedness of $\{\boldsymbol{X}^k\}$ and $\{\boldsymbol{E}^k\}$ can be easily deduced as follows:

$$\begin{aligned}
&\|\boldsymbol{E}_k\|_1 + \|\boldsymbol{X}_k\|_{w,*} \\
&\quad= \Gamma(\boldsymbol{X}_k, \boldsymbol{E}_k, \boldsymbol{L}_{k-1}, \mu_{k-1}) + \frac{\mu_{k-1}}{2}\Big(\frac{1}{\mu_{k-1}^2}\|\boldsymbol{L}_{k-1}\|_F^2 \\
&\qquad- \|\boldsymbol{Y} - \boldsymbol{X}_k - \boldsymbol{E}_k + \frac{1}{\mu_{k-1}}\boldsymbol{L}_{k-1}\|_F^2\Big) \\
&\quad= \Gamma(\boldsymbol{X}_k, \boldsymbol{E}_k, \boldsymbol{L}_{k-1}, \mu_{k-1}) - \frac{1}{2\mu_{k-1}}(\|\boldsymbol{L}_k\|_F^2 - \|\boldsymbol{L}_{k-1}\|_F^2).
\end{aligned}$$

Thus, $\{\boldsymbol{X}_k\}$, $\{\boldsymbol{E}_k\}$ and $\{\boldsymbol{L}_k\}$ generated by the proposed algorithm are all bounded. There exists at least one accumulation point for $\{\boldsymbol{X}_k, \boldsymbol{E}_k, \boldsymbol{L}_k\}$. Specifically, we have

$$\lim_{k\to\infty}\|\boldsymbol{Y} - \boldsymbol{X}_{k+1} - \boldsymbol{E}_{k+1}\|_F = \lim_{k\to\infty}\frac{1}{\mu_k}\|\boldsymbol{L}_{k+1} - \boldsymbol{L}_k\|_F = 0,$$

and the accumulation point is a feasible solution to the objective function.

We then prove that the change of the variables in adjacent iterations tends to be zero. For the $\boldsymbol{E}$ subproblem in step 3, we have

$$\begin{aligned}
&\lim_{k\to\infty}\|\boldsymbol{E}_{k+1} - \boldsymbol{E}_k\|_F \\
&\quad= \lim_{k\to\infty}\|\mathcal{S}_{\frac{1}{\mu_k}}\left(\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{X}_k\right) - \left(\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{X}_k\right) \\
&\qquad- 2\mu_k^{-1}\boldsymbol{L}_k - \mu_{k-1}^{-1}\boldsymbol{L}_{k-1}\|_F \\
&\quad\le \lim_{k\to\infty}\frac{mn}{\mu_k} + \|2\mu_k^{-1}\boldsymbol{L}_k + \mu_{k-1}^{-1}\boldsymbol{L}_{k-1}\|_F = 0,
\end{aligned}$$

in which $\mathcal{S}_{\frac{1}{\mu_k}}(\cdot)$ is the soft-thresholding operation with parameter $\frac{1}{\mu_k}$, and m and n are the size of matrix $\boldsymbol{Y}$.

To prove $\lim_{k\to\infty}\|\boldsymbol{X}_{k+1} - \boldsymbol{X}_k\|_F = 0$, we recall the updating strategy in Algorithm 1 which makes the following inequalities hold:

$$\boldsymbol{X}_k = \boldsymbol{U}_{k-1}\mathcal{S}_{\mathbf{w}/\mu_{k-1}}(\boldsymbol{\Lambda}_{k-1})\boldsymbol{V}_{k-1}^T,$$
$$\boldsymbol{X}_{k+1} = \boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{E}_{k+1} - \mu_k^{-1}\boldsymbol{L}_{k+1},$$

where $\boldsymbol{U}_{k-1}\boldsymbol{\Lambda}_{k-1}\boldsymbol{V}_{k-1}^T$ is the SVD of the matrix $\{\boldsymbol{Y} + \mu_{k-1}^{-1}\boldsymbol{L}_{k-1} - \boldsymbol{E}_k\}$ in the k-th iteration. We then have

$$\begin{aligned}
&\lim_{k\to\infty} \|\boldsymbol{X}_{k+1} - \boldsymbol{X}_k\|_F \\
&= \lim_{k\to\infty} \|(\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{E}_{k+1} - \mu_k^{-1}\boldsymbol{L}_{k+1}) - \boldsymbol{X}_k\|_F \\
&= \lim_{k\to\infty} \|(\boldsymbol{Y} + \mu_k^{-1}\boldsymbol{L}_k - \boldsymbol{E}_{k+1} - \mu_k^{-1}\boldsymbol{L}_{k+1}) - \boldsymbol{X}_k \\
&\quad + (\boldsymbol{E}_k + \mu_{k-1}^{-1}\boldsymbol{L}_{k-1}) - (\boldsymbol{E}_k + \mu_{k-1}^{-1}\boldsymbol{L}_{k-1})\|_F \\
&\le \lim_{k\to\infty} \|\boldsymbol{Y} + \mu_{k-1}^{-1}\boldsymbol{L}_{k-1} - \boldsymbol{E}_k - \boldsymbol{X}_k\|_F + \|\boldsymbol{E}_k \\
&\quad - \boldsymbol{E}_{k+1} + \mu_k^{-1}\boldsymbol{L}_k - \mu_k^{-1}\boldsymbol{L}_{k+1} - \mu_{k-1}^{-1}\boldsymbol{L}_{k-1}\|_F \\
&\le \lim_{k\to\infty} \|\boldsymbol{\Lambda}_{k-1} - \mathcal{S}_{\mathbf{w}/\mu_{k-1}}(\boldsymbol{\Lambda}_{k-1})\|_F + \|\boldsymbol{E}_k - \boldsymbol{E}_{k+1}\|_F \\
&\quad + \|\mu_k^{-1}\boldsymbol{L}_k - \mu_k^{-1}\boldsymbol{L}_{k+1} - \mu_{k-1}^{-1}\boldsymbol{L}_{k-1}\|_F \\
&= 0.
\end{aligned}$$

End of proof. □

9.4 Proof of Remark 1

Proof Based on the conclusion of Theorem 1, the WNNM problem can be equivalently transformed to a constrained singular value optimization problem. Furthermore, when utilizing the reweighting strategy $w_i^{\ell+1} = \frac{C}{\sigma_i^\ell(\boldsymbol{X})+\varepsilon}$, the singular values of $\boldsymbol{X}$ are consistently sorted in a non-ascending order. The weight vector thus follows the non-descending order. It is then easy to deduce that the sorted orders of the sequences $\{\sigma_i(\boldsymbol{Y}), \sigma_i(\boldsymbol{X}_\ell), w_i^\ell; i = 1, 2, \cdots, n\}$ keep unchanged during the iteration. Thus, the optimization for each singular value $\sigma_i(\boldsymbol{X})$ can be analyzed independently. For the purpose of simplicity, in the following development we omit the subscript i and denote by y a singular value of matrix $\boldsymbol{Y}$, and denote by x and w the corresponding singular value of $\boldsymbol{X}$ and its weight.

For the weighting strategy $w^\ell = \frac{C}{x^{\ell-1}+\varepsilon}$, we have

$$x^\ell = max\left(y - \frac{C}{x^{\ell-1}+\varepsilon}, 0\right).$$

Since we initialize x^0 as the singular value of matrix $\boldsymbol{X}_0 = \boldsymbol{Y}$, and each x^ℓ is a result of soft-thresholding operation on positive value $y = \sigma_i(\boldsymbol{Y})$, $\{x^\ell\}$ is a non-negative sequence. The convergence value $\lim_{\ell\to\infty} x^\ell$ for different conditions are analyzed as follows.

(1) $c_2 < 0$
From the definition of c_1 and c_2, we have $(y+\varepsilon)^2 - 4C < 0$. In such case, the quadratic system $x^2 + (\varepsilon - y)x + C - y\varepsilon = 0$ does not have a real solution and function $f(x) = x^2 + (\varepsilon - y)x + C - y\varepsilon$ gets its positive minimum value $C - y\varepsilon - \frac{(y-\varepsilon)^2}{4}$ at $x = \frac{y-\varepsilon}{2}$. $\forall \tilde{x} \ge 0$, the following inequalities hold

$$\begin{aligned}
&f(\tilde{x}) \ge f\left(\frac{y-\varepsilon}{2}\right) \\
&\tilde{x}^2 + (\varepsilon - y)\tilde{x} \ge -\frac{(y-\varepsilon)^2}{4} \\
&\tilde{x} - \frac{C - y\varepsilon - \frac{(y-\varepsilon)^2}{4}}{\tilde{x}+\varepsilon} \ge y - \frac{C}{\tilde{x}+\varepsilon}.
\end{aligned}$$

The sequence $x^{\ell+1} = max\left(y - \frac{C}{x^\ell+\varepsilon}, 0\right)$ with initialization $x^0 = y$ is a monotonically decreasing sequence for any $x^\ell \ge 0$. We have $x^\ell < y$, and

$$x^\ell - \left(y - \frac{C}{x^\ell+\varepsilon}\right) > \frac{C - y\varepsilon - \frac{(y-\varepsilon)^2}{4}}{y+\varepsilon}.$$

If $x^\ell \le \frac{C-y\varepsilon}{y}$, we have $y - \frac{C}{x^\ell+\varepsilon} \le 0$ and $x^{\ell+1} = max\left(y - \frac{C}{x^\ell+\varepsilon}, 0\right) = 0$. If $x^\ell > \frac{C-y\varepsilon}{y}$, $\exists N \in \mathbb{N}$ makes $x^{\ell+N} < x^\ell - N \cdot \frac{C-y\varepsilon-\frac{(y-\varepsilon)^2}{4}}{y+\varepsilon}$ less than $\frac{C-y\varepsilon}{y}$. The sequence $\{x^\ell\}$ will shrink to 0 monotonically.

(2) $c_2 \ge 0$
In such case, we can know that $y > 0$, because if $y = 0$, we will have $c_2 = (y+\varepsilon)^2 - 4C = \varepsilon^2 - 4C < 0$. For positive C and sufficiently small value ε, we can know that c_1 is also non-negative:

$$\begin{aligned}
&c_2 = (y+\varepsilon)^2 - 4C \ge 0 \\
&(y+\varepsilon)^2 \ge 4C \\
&y - \varepsilon \ge 2(\sqrt{C} - \varepsilon) \\
&c_1 = y - \varepsilon \ge 0.
\end{aligned}$$

Having $c_2 \ge 0$, $c_1 \ge 0$, we have

$$\bar{x}_2 = \frac{y - \varepsilon + \sqrt{(y-\varepsilon)^2 - 4(C - \varepsilon y)}}{2} > 0.$$

For any $x > \bar{x}_2 > 0$, the following inequalities hold:

$$\begin{aligned}
&f(x) = x^2 + (\varepsilon - y)x + C - y\varepsilon > 0 \\
&\left[x - \left(y - \frac{C}{x+\varepsilon}\right)\right](x+\varepsilon) > 0 \\
&x > y - \frac{C}{x+\varepsilon}.
\end{aligned}$$

Furthermore, we have

$$x > y - \frac{C}{x+\varepsilon} > y - \frac{C}{\bar{x}_2+\varepsilon} = \bar{x}_2.$$

Thus, for $x^0 = y > \bar{x}_2$, we always have $x^\ell > x^{\ell+1} > \bar{x}_2$, the sequence is monotonically decreasing and has lower bound $\bar{x}_2$. The sequence will converge to $\bar{x}_2$, as we prove below.

We propose a proof by contradiction. If x^ℓ converges to $\hat{x} \neq \bar{x}_2$, then we have $\hat{x} > \bar{x}_2$ and $f(\hat{x}) > 0$. By the definition of convergence, we can obtain that $\forall \epsilon > 0$, $\exists N \in \mathbb{N}$ s.t. $\forall \ell \geq N$, the following inequality must be satisfied

$$|x^\ell - \hat{x}| < \epsilon. \tag{18}$$

We can also have the following inequalies

$$\begin{aligned}
&f(x^N) \geq f(\hat{x}) \\
&\left[x^N - \left(y - \frac{C}{x^N+\varepsilon}\right)\right](x^N+\varepsilon) \geq f(\hat{x}) \\
&\left[x^N - \left(y - \frac{C}{x^N+\varepsilon}\right)\right](y+\varepsilon) \geq f(\hat{x}) \\
&x^N - \left(y - \frac{C}{x^N+\varepsilon}\right) \geq \frac{f(\hat{x})}{y+\varepsilon} \\
&x^N - x^{N+1} > \frac{f(\hat{x})}{y+\varepsilon}
\end{aligned}$$

If we take $\epsilon = \frac{f(\hat{x})}{2(y+\varepsilon)}$, then $x^N - x^{N+1} > 2\epsilon$, and we can thus obtain

$$\begin{aligned}
&|x^{N+1} - \hat{x}| \\
&= |x^{N+1} - x^N + x^N - \hat{x}| \\
&\geq \left| |x^{N+1} - x^N| - |x^N - \hat{x}| \right| \\
&\leq |2\epsilon - \epsilon| = \epsilon
\end{aligned}$$

This is however a contradiction to (18), and thus x^ℓ converges to $\bar{x}_2$.

End of proof. □

References

Arias, P., Facciolo, G., Caselles, V., & Sapiro, G. (2011). A variational framework for exemplar-based image inpainting. *International Journal of computer Vision*, *93*(3), 319–347.

Babacan, S. D., Luessi, M., Molina, R., & Katsaggelos, A. K. (2012). Sparse bayesian methods for low-rank matrix estimation. *IEEE Transactions on Signal Processing*, *60*(8), 3964–3977.

Boykov, Y., Veksler, O., & Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, *23*(11), 1222–1239.

Buades, A., Coll, B., & Morel, J. M. (2005). A non-local algorithm for image denoising. In *CVPR*.

Buades, A., Coll, B., & Morel, J. M. (2008). Nonlocal image and movie denoising. *International Journal of Computer Vision*, *76*(2), 123–139.

Buchanan, A.M., & Fitzgibbon, A.W, (2005). Damped newton algorithms for matrix factorization with missing data. In *CVPR*.

Cai, J. F., Candès, E. J., & Shen, Z. (2010). A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, *20*(4), 1956–1982.

Candès, E. J., & Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, *9*(6), 717–772.

Candès, E. J., Wakin, M. B., & Boyd, S. P. (2008). Enhancing sparsity by reweighted l_1 minimization. *Journal of Fourier Analysis and Applications*, *14*(5–6), 877–905.

Candès, E. J., Li, X., Ma, Y., & Wright, J. (2011). Robust principal component analysis? *Journal of the ACM*, *58*(3), 11.

Chan, T. F., & Shen, J. J. (2005). *Image processing and analysis: Variational, PDE, wavelet, and stochastic methods*. Philadelphia: SIAM Press.

Chartrand, R. (2007). Exact reconstruction of sparse signals via nonconvex minimization. *IEEE Signal Processing Letters*, *14*(10), 707–710.

Chartrand, R. (2012). Nonconvex splitting for regularized low-rank+ sparse decomposition. *IEEE Transaction on Signal Processing*, *60*(11), 5810–5819.

Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2007). Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transaction on Image Processing*, *16*(8), 2080–2095.

Dahl, J., Hansen, P. C., Jensen, S. H., & Jensen, T. L. (2010). Algorithms and software for total variation image reconstruction via first-order methods. *Numerical Algorithms*, *53*(1), 67–92.

De La Torre, F., & Black, M. J. (2003). A framework for robust subspace learning. *International Journal of Computer Vision*, *54*(1–3), 117–142.

Ding, X., He, L., & Carin, L. (2011). Bayesian robust principal component analysis. *IEEE Transactions on Image Processing*, *20*(12), 3419–3430.

Dong, W., Zhang, L., & Shi, G. (2011). Centralized sparse representation for image restoration. In *ICCV*.

Dong, W., Shi, G., & Li, X. (2013). Nonlocal image restoration with bilateral variance estimation: A low-rank approach. *IEEE Transaction on Image Processing*, *22*(2), 700–711.

Dong, W., Shi, G., Li, X., Ma, Y., & Huang, F. (2014). Compressive sensing via nonlocal low-rank regularization. *IEEE Transaction on Image Processing*, *23*(8), 3618–3632.

Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transaction on Info Theory*, *41*(3), 613–627.

Eriksson, A., & Van Den Hengel, A. (2010). Efficient computation of robust low-rank matrix approximations in the presence of missing data using the l_1 norm. In *CVPR*.

Fazel, M. (2002). Matrix rank minimization with applications. PhD thesis, PhD thesis, Stanford University.

Fazel, M., Hindi, H., & Boyd, S.P. (2001). A rank minimization heuristic with application to minimum order system approximation. In *American Control Conference. (ACC)*.

Gu, S., Zhang, L., Zuo, W., & Feng, X. (2014). Weighted nuclear norm minimization with application to image denoising. In *CVPR*.

Jain, P., Netrapalli, P., & Sanghavi, S. (2013). Low-rank matrix completion using alternating minimization. In *ACM symposium on theory of computing*.

Ji, H., Liu, C., Shen, Z., & Xu, Y. (2010). Robust video denoising using low rank matrix completion. In *CVPR*.

Ji, S., & Ye, J. (2009). An accelerated gradient method for trace norm minimization. In *ICML* (pp. 457–464).

Ke, Q., & Kanade, T. (2005). Robust l_1 norm factorization in the presence of outliers and missing data by alternative convex programming. In *CVPR*.

Kwak, N. (2008). Principal component analysis based on l1-norm maximization. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, *30*(9), 1672–1680.

Levin, A., & Nadler, B. (2011). Natural image denoising: Optimality and inherent bounds. In *CVPR*.

Levin, A., Nadler, B., Durand, F., & Freeman, W.T. (2012). Patch complexity, finite pixel correlations and optimal denoising. In *ECCV*.

Li, L., Huang, W., Gu, I. H., & Tian, Q. (2004). Statistical modeling of complex backgrounds for foreground object detection. *IEEE Transaction on Image Processing*, *13*(11), 1459–1472.

Lin, Z., Ganesh, A., Wright, J., Wu, L., Chen, M., & Ma, Y. (2009). Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. In *International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*.

Lin, Z., Liu, R., & Su, Z. (2011). Linearized alternating direction method with adaptive penalty for low-rank representation. In *NIPS*.

Lin, Z., Liu, R., & Li, H. (2015). Linearized alternating direction method with parallel splitting and adaptive penalty for separable convex programs in machine learning. *Machine Learning*, *99*(2), 287–325.

Liu, G., Lin, Z., Yan, S., Sun, J., Yu, & Y., Ma, Y. (2010). Robust subspace segmentation by low-rank representation. In *ICML*.

Liu, R., Lin, Z., De la, Torre, F., & Su, Z. (2012). Fixed-rank representation for unsupervised visual learning. In *CVPR*.

Lu, C., Tang, J., Yan, S., & Lin, Z. (2014a). Generalized nonconvex nonsmooth low-rank minimization. In *CVPR*.

Lu, C., Zhu, C., Xu, C., Yan, S., & Lin, Z. (2014b). Generalized singular value thresholding. arXiv preprint arXiv:1412.2231.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., & Zisserman, A. (2009). Non-local sparse models for image restoration. In *ICCV*.

Meng, D., & Torre, F.D.L. (2013). Robust matrix factorization with unknown noise. In *ICCV*.

Mirsky, L. (1975). A trace inequality of john von neumann. *Monatshefte für Mathematik*, *79*(4), 303–306.

Mnih, A., &Salakhutdinov, R. (2007). Probabilistic matrix factorization. In *NIPS*.

Mohan, K., & Fazel, M. (2012). Iterative reweighted algorithms for matrix rank minimization. *The Journal of Machine Learning Research*, *13*(1), 3441–3473.

Moreau, J. J. (1965). Proximité et dualité dans un espace hilbertien. *Bulletin de la Société mathématique de France*, *93*, 273–299.

Mu, Y., Dong, J., Yuan, X., & Yan, S. (2011). Accelerated low-rank visual recovery by random projection. In *CVPR*.

Nie, F., Huang, H., & Ding, C.H. (2012). Low-rank matrix recovery via efficient schatten p-norm minimization. In *AAAI*.

Oh, T.H., Kim, H., Tai, Y.W., Bazin, J.C., & Kweon, I.S. (2013). Partial sum minimization of singular values in rpca for low-level vision. In *ICCV*.

Peng, Y., Ganesh, A., Wright, J., Xu, W., & Ma, Y. (2012). Rasl: Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, *34*(11), 2233–2246.

Portilla, J. (2004). *Blind non-white noise removal in images using gaussian scale*. Citeseer: In *Proceedings of the IEEE benelux signal processing symposium*.

Rhea, D. (2011). The case of equality in the von Neumann trace inequality. *Preprint*.

Roth, S., & Black, M. J. (2009). Fields of experts. *International Journal of Computer Vision*, *82*(2), 205–229.

Ruslan, S., & Srebro, N. (2010). Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. In *NIPS*.

She, Y. (2012). An iterative algorithm for fitting nonconvex penalized generalized linear models with grouped predictors. *Computational Statistics & Data Analysis*, *56*(10), 2976–2990.

Srebro, N., & Jaakkola, T., et al. (2003). Weighted low-rank approximations. In *ICML*.

Srebro, N., Rennie, J., & Jaakkola, T.S. (2004). Maximum-margin matrix factorization. In *NIPS*.

Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *61*(3), 611–622.

Wang, N., & Yeung, D.Y. (2013). Bayesian robust matrix factorization for image and video processing. In *ICCV*.

Wang, S., Zhang, L., &Y, L. (2012). Nonlocal spectral prior model for low-level vision. In *ACCV*.

Wright, J., Peng, Y., Ma, Y., Ganesh, A., & Rao, S. (2009). Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization. In *NIPS*.

Zhang, D., Hu, Y., Ye, J., Li, X., & He X (2012a). Matrix completion by truncated nuclear norm regularization. In *CVPR*.

Zhang, Z., Ganesh, A., Liang, X., & Ma, Y. (2012b). Tilt: transform invariant low-rank textures. *International Journal of Computer Vision*, *99*(1), 1–24.

Zhao, Q., Meng, D., Xu, Z., Zuo, W., & Zhang, L. (2014) Robust principal component analysis with complex noise. In *ICML*.

Zheng, Y., Liu, G., Sugimoto, S., Yan, S., & Okutomi, M. (2012). Practical low-rank matrix approximation under robust l_1 norm. In *CVPR*.

Zhou M, Chen, H., Ren, L., Sapiro, G., Carin, L., & Paisley, J.W. (2009). Non-parametric bayesian dictionary learning for sparse image representations. In *NIPS*.

Zhou, X., Yang, C., Zhao, H., & Yu, W. (2014). Low-rank modeling and its applications in image analysis. arXiv preprint arXiv:1401.3409.

Zoran, D., & Weiss, Y. (2011). From learning models of natural image patches to whole image restoration. In *ICCV*.