

Truncated Quadratic Norm Minimization for Bilinear Factorization based Matrix Completion

Xiang Yu Wang^a, Xiao Peng Li^{b,*}, Hing Cheung So^{a,1}

^a*Department of Electrical Engineering, City University of Hong Kong, Hong Kong SAR, China*

^b*College of Electronics and Information Engineering, Shenzhen University, Shenzhen, 518060, China.*

Abstract

Low-rank matrix completion is an important research topic with a wide range of applications. One prevailing way for matrix recovery is based on rank minimization. Directly solving this problem is NP hard. Therefore, various rank surrogates are developed, like nuclear norm. However, nuclear norm regularization minimizes the sum of all the singular values, and hence the rank is not well approximated. We propose a new rank substitution named truncated quadratic norm that performs the corresponding truncated quadratic operation on the singular values. This function takes the square of the minor singular values and maps large singular values to one. In order to reduce computational complexity, the original target matrix is factorized into two small matrices on which the truncated quadratic norm is imposed. The resultant problem is then solved by alternating minimization. We also prove that the solution sequence is able to converge to a critical point. Experimental results on synthetic data and real-world images demonstrate the excellent performance of our method in terms of recovery accuracy.

Keywords: matrix completion, rank surrogate, bilinear factorization, alternating minimization, convergence.

1. Introduction

Matrix completion (MC) aims at restoring a matrix given only partially observed and possibly noisy entries, which is ill-posed without any assumption. Thereby, low-rank assumption is usually imposed on the target matrix, resulting in low-rank MC problem. It has numerous applications in computer vision [1–4], pattern recognition [5–9], recommender system [10, 11], and radar signal processing [12, 13]. In these scenarios, the rows/columns of the target matrix are highly correlated, and thus the matrix has approximately low rank. For example, only a few large singular values of natural scene images contain the major information, and the remaining singular values are insignificant [14, 15].

One mainstream approach to tackle low-rank MC is based on rank minimization. Due to the noncontinuity and nonconvexity of the rank function, many rank surrogates have been proposed. The most well-known

* Corresponding author

Email addresses: xwang2286-c@my.cityu.edu.hk (Xiang Yu Wang), x.p.li@szu.edu.cn (Xiao Peng Li), hcs0@ee.cityu.edu.hk (Hing Cheung So)

¹EURASIP Member

one is the nuclear norm that is equivalent to the sum of all singular values. The nuclear norm as a convex lower bound of the rank function has been applied to matrix recovery [16, 17], of which the resultant optimization problem is widely handled by singular value thresholding operation [18]. However, minimizing the nuclear norm causes much penalty on large singular values [15, 19], which makes the solution deviate a lot from the ground truth. To deal with this issue, truncated nuclear norm (TNN) is developed [20, 21], which is the nuclear norm subtracted by a few largest singular values. That is, TNN minimization only penalizes the small singular values and outperforms the nuclear norm approach in MC. Unfortunately, TNN requires predetermining the number of largest singular values, which is vital for the completion performance. Weighted nuclear norm (WNN) [22, 23] is another improvement strategy, which assigns different weights to the singular values and then adds them together. Specifically, large singular values are allocated with small weights and vice versa. To determine the weights, Lu *et al.* [22] use the gradients of the surrogate functions of the l_0 -norm, and Gu *et al.* [23] adopt a sparse coding technique [24].

To further reduce the gap between the rank function and variants of nuclear norm, other rank substitutions have been considered. Schatten p -norm [25] calculates the l_p -norm of the singular value vector. When $0 < p < 1$, it can better approximate the rank function than the nuclear norm. However, the solution of l_p -norm minimization may converge to local optimum due to the nonconvexity. Furthermore, the corresponding algorithms are complicated because the l_p -norm is not differentiable at zero point [26, 27]. The log-determinant heuristic [28–30] also shows superior performance over the nuclear norm. It can penalize small singular values more than the large ones [30] and thus improves the performance of the singular value thresholding operator [31, 32]. Nevertheless, the Schatten p -norm and logarithmic norm cannot well depict the rank function for large singular values.

Employing the aforementioned rank surrogates inevitably needs implementing singular value decomposition (SVD) on the target matrix. The computational cost for performing SVD on a large matrix is high. Apart from rank minimization, another approach for MC is based on matrix factorization which avoids computing SVD. The objective low-rank matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ is decomposed into two small factor matrices in the form $\mathbf{X} = \mathbf{U}\mathbf{V}$ with $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{r \times n}$, where r is the assumed rank of \mathbf{X} . Factorization based MC minimizes the gap between the observed matrix and $\mathbf{U}\mathbf{V}$ on the observation set. The low-rank property of \mathbf{X} is ensured by setting $r \ll \min(m, n)$. This model can be formulated as a least squares problem [26], which can be solved efficiently by subspace evolution and transfer [33], alternating minimization [34], and proximal alternating minimization (PAM) [35]. Under the matrix factorization framework, the selection of r is important for the completion performance, especially when the observed matrix contains noise. Rank-one matrix pursuit strategy aims to select the best r , which includes orthogonal rank-one matrix pursuit [36, 37], l_1 -norm and l_p -norm regularized rank-one matrix completion [38, 39]. Furthermore, if the reconstruction error is assumed to be a unimodal function of r , adaptive bisection search [40] is considered to determine r . These algorithms try to estimate the objective matrix by a linear combination of rank-one matrices wherein the estimation error is still controlled by a manually-set accuracy threshold.

In this paper, we design a nonconvex rank substitution, called truncated quadratic norm, to solve the low-rank MC problem. We utilize a piecewise function to process singular values. The singular values are divided into large and small value groups by a threshold. As for large inputs, the function adopts constant value one in order to faithfully follow the rank function, while the quadratic function is used to penalize small singular values. The sum of the function values for all singular values is defined as the truncated quadratic norm. In doing so, over penalization on large singular values is avoided. This piecewise function, or rather truncated quadratic function, is easy to handle, and the corresponding truncated quadratic norm MC problem has a closed-form solution. Furthermore, combining the advantages of the rank minimization-based and factorization-based methods, \mathbf{X} is decomposed into two small matrices $\mathbf{X}_1 \in \mathbb{R}^{m \times d}$ and $\mathbf{X}_2 \in \mathbb{R}^{d \times n}$ in the form $\mathbf{X} = \mathbf{X}_1 \mathbf{X}_2$, where the decomposition parameter d is slightly larger than the rank of \mathbf{X} . When the true rank is unknown, we suggest $d \approx 0.1 * \min(m, n)$ for exactly low-rank data, and $d \approx 0.4 * \min(m, n)$ for images which have approximately low rank. Then the proposed truncated quadratic norm regularization is imposed on \mathbf{X}_1 and \mathbf{X}_2 individually. The main contributions of this paper are summarized as

- (1) We design a new MC model utilizing the truncated quadratic norm, which is defined based on a truncated quadratic function. This norm can better approximate the rank function, and the resultant MC problem is easy to handle. To lessen the computational burden for SVD, the objective matrix is factorized into two small matrices on which the truncated quadratic norm regularization is imposed.
- (2) We employ alternating minimization based on proximal linear method to solve the resultant problem with closed-form expressions. The solution sequence is proved to converge to a critical point, while the sequence convergence of many existing MC schemes using nonconvex surrogates cannot be guaranteed.
- (3) Experimental results on synthetic data and real-world images demonstrate the effectiveness of our method in terms of recovery accuracy. Moreover, we show that the completion results are not sensitive to the decomposition parameter d .

The organization of this paper is as follows. In Section 2, related work on representative matrix rank substitutions and bilinear MC models are reviewed. The proposed algorithm is developed in Section 3. Evaluation results of our method are included in Section 4. Section 5 concludes our work.

2. Related Work

In this paper, scalars and matrices are written in lowercase letters and boldface capital letters, respectively. The j th singular value of matrix \mathbf{X} in descending order is represented by $\sigma_j(\mathbf{X})$. Operator $\text{diag}(\cdot)$ establishes a diagonal matrix whose diagonal elements come from the input entries. Matrix trace operator is denoted as $\text{tr}(\cdot)$.

Let $\mathbf{Y}_{\mathbf{O}} \in \mathbb{R}^{m \times n}$ be an incomplete matrix with missing elements, and $\mathbf{O} \in \mathbb{R}^{m \times n}$ is a binary matrix. The indices of 1 and 0 in \mathbf{O} indicate the locations of the observed and unobserved elements in $\mathbf{Y}_{\mathbf{O}}$, respectively.

The task for MC is to find the unobserved entries in \mathbf{Y}_O using the observed information. Low-rank property is usually imposed on the recovered matrix, which can be represented as [30]

$$\min_{\mathbf{X}} F(\mathbf{X}) = \lambda \|\mathbf{X} \odot \mathbf{O} - \mathbf{Y}_O\|_F^2 + \mathcal{U}(\mathbf{X}). \quad (1)$$

Here $\mathbf{X} \in \mathbb{R}^{m \times n}$ is the estimated target matrix, \odot denotes the element-wise Hadamard product, $\lambda > 0$ is a weight parameter, and $\mathcal{U}(\cdot)$ is a spectral regularization operator [41] related to the matrix rank. The first term in (1) is the fidelity, which measures the difference between \mathbf{X} and \mathbf{Y}_O with regard to the observed elements. The second term $\mathcal{U}(\cdot)$ can be written as $\mathcal{U}(\mathbf{X}) = \sum_{j=1}^{\min(m,n)} S(\sigma_j(\mathbf{X}))$ with $S(\cdot)$ being the penalty function on singular values. For rank function, the penalty function is the l_0 -norm. Intractable rank function is usually replaced by surrogates based on different penalty functions on singular values. The nuclear norm is defined as $\|\mathbf{X}\|_* = \sum_{j=1}^{\min(m,n)} \sigma_j(\mathbf{X})$, that is, $S(\sigma) = \sigma$. To better approximate the rank, other substitutions have been introduced. For example, Schatten p -norm [25]

$$\|\mathbf{X}\|_{S_p} = \left(\sum_{j=1}^{\min(m,n)} \sigma_j^p(\mathbf{X}) \right)^{1/p} \quad (2)$$

is induced by using penalty function $S(\sigma) = \sigma^p$ with $0 < p < 2$, while the logarithmic norm [30] corresponds to $S(\sigma) = \log(\sigma^p + \epsilon)$ with $\epsilon > 0$ and $0 < p \leq 1$.

Solving (1) requires SVD, which is computationally demanding. To lessen the computational burden, \mathbf{X} can be decomposed into two small matrices, and bilinear factor matrix regularization is proposed. For example, double nuclear norm [15] is suggested:

$$\|\mathbf{X}\|_{D-N} = \min_{\mathbf{X}_1, \mathbf{X}_2: \mathbf{X} = \mathbf{X}_1 \mathbf{X}_2} \frac{1}{4} (\|\mathbf{X}_1\|_* + \|\mathbf{X}_2\|_*)^2, \quad (3)$$

where $\mathbf{X}_1 \in \mathbb{R}^{m \times d}$ and $\mathbf{X}_2 \in \mathbb{R}^{d \times n}$ are two small factor matrices. Addressing (3) can be divided into solving two subproblems for \mathbf{X}_1 and \mathbf{X}_2 , which only requires SVD calculation of two small matrices, and thus the computational cost is reduced. Moreover, the Frobenius norm of two factor matrices $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{r \times n}$ is proved to be an upper bound of the nuclear norm [42], namely

$$\|\mathbf{X}\|_* = \min_{\mathbf{U}, \mathbf{V}: \mathbf{X} = \mathbf{U}\mathbf{V}} \frac{1}{2} (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2), \quad (4)$$

resulting in a nonconvex optimization problem, which even avoids SVD. PAM utilizes (4) to solve rank constrained problems and converges to the second-order stationary point with mild initial conditions [35].

3. Proposed Algorithm

In this section, we formulate the truncated quadratic norm-based MC (TQNMCMC). Then the problem is solved by alternating minimization. We adopt the proximal linear technique to simplify the objective function with respect to each variable. Besides, the convergence and computational complexity of the algorithm are analyzed.

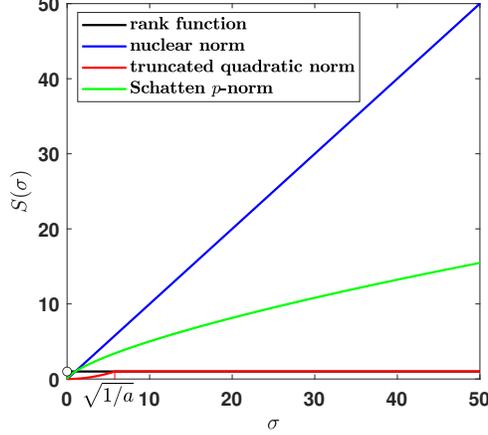


Fig. 1: Penalty functions on singular values of nuclear norm, Schatten p -norm ($p = 0.7$), rank, and truncated quadratic norm ($a = 3e-2$).

3.1. Factor Matrix Rank Minimization via Truncated Quadratic Norm

We introduce truncated quadratic norm to better approximate the rank function. First, a truncated quadratic function is designed to tackle the small and large singular values differently:

$$S_a(\sigma) = \begin{cases} 1, & \text{for } \sigma \geq \sqrt{1/a}, \\ a\sigma^2, & \text{for } 0 \leq \sigma < \sqrt{1/a}, \end{cases} \quad (5)$$

where $a > 0$ is a threshold differentiating the small and large singular values. Next, we give the definition of the truncated quadratic norm based on $S_a(\sigma)$.

Definition 1. (*Matrix Truncated Quadratic Norm*²): Given a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, its truncated quadratic norm is

$$\|\mathbf{X}\|_{TQ} = \sqrt{\sum_{j=1}^{\min(m,n)} S_a(\sigma_j(\mathbf{X}))}. \quad (6)$$

We illustrate the penalty functions on singular values of the nuclear norm, Schatten p -norm, rank operator, and truncated quadratic norm in Fig. 1. The quadratic function for small singular values is continuous at the zero point comparing with the rank function. For large singular values, the truncated quadratic norm is the same as the rank function whereas the nuclear norm regularization may cause over minimization of large singular values [15]. Besides, there is deviation between the Schatten p -norm and the rank. Thus, the truncated quadratic norm is a better rank surrogate.

MC based on truncated quadratic norm minimization is formulated as:

$$\min_{\mathbf{X}} F(\mathbf{X}) = \lambda \|\mathbf{O} \odot \mathbf{X} - \mathbf{Y}_\mathbf{O}\|_F^2 + \|\mathbf{X}\|_{TQ}^2. \quad (7)$$

²It is a quasi-norm.

To reduce the computational cost, similar to (3), we decompose $\mathbf{X} \in \mathbb{R}^{m \times n}$ into $\mathbf{X}_1 \in \mathbb{R}^{m \times d}$ and $\mathbf{X}_2 \in \mathbb{R}^{d \times n}$. When $d \geq \text{rank}(\mathbf{X})$, according to matrix rank property, we have

$$\text{rank}(\mathbf{X}) = \text{rank}(\mathbf{X}_1 \mathbf{X}_2) \leq \min(\text{rank}(\mathbf{X}_1), \text{rank}(\mathbf{X}_2)), \quad (8)$$

which gives an upper bound of the rank of \mathbf{X} . Based on (8), minimizing the rank of \mathbf{X} can be approximated by

$$\min_{\mathbf{X}_1, \mathbf{X}_2} \text{rank}(\mathbf{X}_1) + \text{rank}(\mathbf{X}_2). \quad (9)$$

We substitute $\bar{U}(\mathbf{X})$ in (1) with the rank function and then combine (9) to attain

$$\min_{\mathbf{X}_1, \mathbf{X}_2} F(\mathbf{X}_1, \mathbf{X}_2) = \lambda \|\mathbf{O} \odot (\mathbf{X}_1 \mathbf{X}_2) - \mathbf{Y}_O\|_F^2 + \text{rank}(\mathbf{X}_1) + \text{rank}(\mathbf{X}_2). \quad (10)$$

Replacing the rank function in (10) with the truncated quadratic norm produces the model of TQNMCM:

$$\min_{\mathbf{X}_1, \mathbf{X}_2} F(\mathbf{X}_1, \mathbf{X}_2) = \lambda \|\mathbf{O} \odot (\mathbf{X}_1 \mathbf{X}_2) - \mathbf{Y}_O\|_F^2 + \|\mathbf{X}_1\|_{TQ}^2 + \|\mathbf{X}_2\|_{TQ}^2. \quad (11)$$

As SVD is performed on two small matrices in (11), its computational requirement is less than (7).

3.2. Alternating Minimization via Proximal Linear Method

We adopt alternating minimization to solve (11), where the objective function $F(\mathbf{X}_1, \mathbf{X}_2)$ is minimized alternately with respect to \mathbf{X}_1 and \mathbf{X}_2 . In the k th iteration, the problem to update \mathbf{X}_1^k is

$$\mathbf{X}_1^k = \arg \min_{\mathbf{X}_1} \|\mathbf{X}_1\|_{TQ}^2 + \lambda \|\mathbf{O} \odot (\mathbf{X}_1 \mathbf{X}_2^{k-1}) - \mathbf{Y}_O\|_F^2, \quad (12)$$

For the ease of representation, we denote $f(\mathbf{X}_1, \mathbf{X}_2) = \|\mathbf{O} \odot (\mathbf{X}_1 \mathbf{X}_2) - \mathbf{Y}_O\|_F^2$. It is easy to verify that $f(\mathbf{X}_1, \mathbf{X}_2)$ has a continuous Lipschitz gradient with respect to each coordinate.

Problem (12) is a regularized least squares problem and can be efficiently solved by proximal linear method [43, 44]. We rewrite (12) by linearizing $f(\mathbf{X}_1, \mathbf{X}_2^{k-1})$ with respect to \mathbf{X}_1 :

$$\begin{aligned} \mathbf{X}_1^k &= \arg \min_{\mathbf{X}_1} \frac{1}{\lambda} \|\mathbf{X}_1\|_{TQ}^2 + \frac{\mu_1^k}{2} \|\mathbf{X}_1 - \mathbf{X}_1^{k-1}\|_F^2 + \langle \nabla_{\mathbf{X}_1} f(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1}), \mathbf{X}_1 - \mathbf{X}_1^{k-1} \rangle \\ &= \arg \min_{\mathbf{X}_1} \frac{1}{\lambda} \|\mathbf{X}_1\|_{TQ}^2 + \frac{\mu_1^k}{2} \|\mathbf{X}_1 - \mathbf{X}_1^{k-1}\|_F^2 + \langle \nabla_{\mathbf{X}_1} f(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1}), \mathbf{X}_1 \rangle - \langle \nabla_{\mathbf{X}_1} f(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1}), \mathbf{X}_1^{k-1} \rangle. \end{aligned} \quad (13)$$

Here μ_1^k is the proximal parameter in the k th iteration. We set $\mu_1^k = \gamma L_1^k$, where $1 < \gamma < \infty$ and L_1^k is the Lipschitz constant of $\nabla_{\mathbf{X}_1} f(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1})$ with respect to \mathbf{X}_1 . Neglecting the irrelevant term which is independent of \mathbf{X}_1 , namely $\langle \nabla_{\mathbf{X}_1} f(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1}), \mathbf{X}_1^{k-1} \rangle$, we obtain

$$\mathbf{X}_1^k = \arg \min_{\mathbf{X}_1} \frac{1}{\lambda} \|\mathbf{X}_1\|_{TQ}^2 + \frac{\mu_1^k}{2} \|\mathbf{X}_1 - \mathbf{X}_1^{k-1}\|_F^2 + \langle \nabla_{\mathbf{X}_1} f(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1}), \mathbf{X}_1 \rangle. \quad (14)$$

To accelerate the update rate, we extrapolate a point $\hat{\mathbf{X}}_1^{k-1}$ between \mathbf{X}_1^k and \mathbf{X}_1^{k-1} [43], viz.

$$\hat{\mathbf{X}}_1^{k-1} = \mathbf{X}_1^{k-1} + \omega_1^k (\mathbf{X}_1^{k-1} - \mathbf{X}_1^{k-2}), \quad (15)$$

where ω_1^k is the extrapolation weight. The computation of ω_1^k follows the strategy in [45, 46], which will be detailed later in this subsection. Then \mathbf{X}_1^k is updated based on $\hat{\mathbf{X}}_1^{k-1}$, and (14) can be rewritten as

$$\mathbf{X}_1^k = \arg \min_{\mathbf{X}_1} \frac{1}{\lambda} \|\mathbf{X}_1\|_{TQ}^2 + \frac{\mu_1^k}{2} \|\mathbf{X}_1 - \hat{\mathbf{X}}_1^{k-1}\|_F^2 + \langle \nabla_{\mathbf{X}_1} f(\hat{\mathbf{X}}_1^{k-1}, \mathbf{X}_2^{k-1}), \mathbf{X}_1 \rangle. \quad (16)$$

If the solution \mathbf{X}_1^k of (16) causes $F(\mathbf{X}_1^k, \mathbf{X}_2^{k-1}) > F(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1})$, we adopt $\hat{\mathbf{X}}_1^{k-1} = \mathbf{X}_1^{k-1}$ and redo the update.

We further represent (16) in the proximal optimization format:

$$\mathbf{X}_1^k = \arg \min_{\mathbf{X}_1} \frac{1}{\lambda} \|\mathbf{X}_1\|_{TQ}^2 + \frac{\mu_1^k}{2} \|\mathbf{X}_1 - \mathbf{M}_1^k\|_F^2, \quad (17)$$

where $\mathbf{M}_1^k = \hat{\mathbf{X}}_1^{k-1} - \frac{1}{\mu_1^k} \nabla_{\mathbf{X}_1} f(\hat{\mathbf{X}}_1^{k-1}, \mathbf{X}_2^{k-1})$.

In the following, we devise a theorem to acquire the solution of (17).

Theorem 1. (*Truncated Quadratic Singular Value Thresholding*): Let the SVD of $\mathbf{C} \in \mathbb{R}^{m \times n}$ be $\mathbf{C} = \mathbf{Q}_C \mathbf{\Lambda}_C \mathbf{R}_C^T$ with $\mathbf{\Lambda}_C = \text{diag}\{\sigma_1(\mathbf{C}), \dots, \sigma_{\min(m,n)}(\mathbf{C})\}$. The solution of problem

$$\min_{\mathbf{X}} \frac{1}{2} \|\mathbf{X} - \mathbf{C}\|_F^2 + \alpha \|\mathbf{X}\|_{TQ}^2 \quad (18)$$

is given by $\mathbf{X} = \mathbf{Q}_C \mathcal{T} \mathcal{Q}_{\alpha, a}(\mathbf{\Lambda}_C) \mathbf{R}_C^T$. The element-wise truncated quadratic singular value thresholding (TQSVT) operator $\mathcal{T} \mathcal{Q}_{\alpha, a}(\cdot)$ is defined as

$$\mathcal{T} \mathcal{Q}_{\alpha, a}(c) = \arg \min_{\sigma} h(\sigma), \quad (19)$$

where $h(\sigma) = \frac{1}{2} (\sigma - c)^2 + \alpha S_a(\sigma)$ and $\sigma \in \left\{ \max(c, \sqrt{1/a}), \min(\sqrt{1/a}, c/(1+2\alpha a)) \right\}$.

Proof: Let the SVD of \mathbf{X} be $\mathbf{X} = \mathbf{Q}_X \mathbf{\Lambda}_X \mathbf{R}_X^T$, then the function in (18) is rewritten as

$$\begin{aligned} & \frac{1}{2} \|\mathbf{X} - \mathbf{C}\|_F^2 + \alpha \|\mathbf{X}\|_{TQ}^2 \\ &= \frac{1}{2} \|\mathbf{Q}_X \mathbf{\Lambda}_X \mathbf{R}_X^T - \mathbf{Q}_C \mathbf{\Lambda}_C \mathbf{R}_C^T\|_F^2 + \alpha \text{tr}(S_a(\mathbf{\Lambda}_X)) \\ &= \frac{1}{2} (\text{tr}(\mathbf{\Lambda}_X^T \mathbf{\Lambda}_X) + \text{tr}(\mathbf{\Lambda}_C^T \mathbf{\Lambda}_C) - 2\text{tr}(\mathbf{X}^T \mathbf{C})) + \alpha \text{tr}(S_a(\mathbf{\Lambda}_X)) \\ &\geq \frac{1}{2} (\text{tr}(\mathbf{\Lambda}_X^T \mathbf{\Lambda}_X) + \text{tr}(\mathbf{\Lambda}_C^T \mathbf{\Lambda}_C) - 2\text{tr}(\mathbf{\Lambda}_X^T \mathbf{\Lambda}_C)) + \alpha \text{tr}(S_a(\mathbf{\Lambda}_X)) \\ &= \sum_{j=1}^{\min(m,n)} \frac{1}{2} (\sigma_j(\mathbf{X}) - \sigma_j(\mathbf{C}))^2 + \alpha S_a(\sigma_j(\mathbf{X})). \end{aligned} \quad (20)$$

Here the inequality utilizes the Von Neumann's trace inequality [47], and the equality holds when the left and right singular vector matrices of \mathbf{X} achieve \mathbf{Q}_C and \mathbf{R}_C , respectively. Based on (20), the minimizer of (18), denoted by \mathbf{X}^* , can be obtained by solving the j th decoupled singular value problem:

$$\sigma_j^*(\mathbf{X}) = \arg \min_{\sigma_j(\mathbf{X}) \geq 0} h(\sigma_j(\mathbf{X})) \triangleq \frac{1}{2} (\sigma_j(\mathbf{X}) - \sigma_j(\mathbf{C}))^2 + \alpha S_a(\sigma_j(\mathbf{X})). \quad (21)$$

Then

$$\mathbf{X}^* = \mathbf{Q}_C \text{diag}(\sigma_1^*(\mathbf{X}), \dots, \sigma_{\min(m,n)}^*(\mathbf{X})) \mathbf{R}_C^T. \quad (22)$$

There are two cases for (21) due to the piecewise function $S_a(\sigma_j(\mathbf{X}))$:

Case 1: For $\sigma_j(\mathbf{X}) \geq \sqrt{1/a}$,

$$h(\sigma_j(\mathbf{X})) = \frac{1}{2}(\sigma_j(\mathbf{X}) - \sigma_j(\mathbf{C}))^2 + \alpha, \quad (23)$$

and the minimizer is

$$\sigma_j^*(\mathbf{X}) = \max\left(\sigma_j(\mathbf{C}), \sqrt{1/a}\right). \quad (24)$$

Case 2: For $0 \leq \sigma_j(\mathbf{X}) < \sqrt{1/a}$,

$$h(\sigma_j(\mathbf{X})) = \frac{1}{2}(\sigma_j(\mathbf{X}) - \sigma_j(\mathbf{C}))^2 + \alpha a \sigma_j(\mathbf{X})^2, \quad (25)$$

of which the first-order derivative is

$$h'(\sigma_j(\mathbf{X})) = \sigma_j(\mathbf{X}) - \sigma_j(\mathbf{C}) + 2\alpha a \sigma_j(\mathbf{X}). \quad (26)$$

Setting $h'(\sigma_j(\mathbf{X})) = 0$ yields $\sigma_j(\mathbf{X}) = \sigma_j(\mathbf{C}) / (1 + 2\alpha a)$. The minimizer of (25) is

$$\sigma_j^*(\mathbf{X}) = \min\left(\sigma_j(\mathbf{C}) / (1 + 2\alpha a), \sqrt{1/a}\right). \quad (27)$$

Combining Cases 1 and 2, we conclude that (21) equals

$$\sigma_j^*(\mathbf{X}) = \mathcal{T}_{\alpha, a}(\sigma_j(\mathbf{C})) = \arg \min_{\sigma_j(\mathbf{X})} h(\sigma_j(\mathbf{X})), \quad (28)$$

where

$$\sigma_j(\mathbf{X}) \in \left\{ \min\left(\sigma_j(\mathbf{C}) / (1 + 2\alpha a), \sqrt{1/a}\right), \max\left(\sigma_j(\mathbf{C}), \sqrt{1/a}\right) \right\}. \quad (29)$$

The proof is complete.

The TQSVT operation in Theorem 1 can be used to update \mathbf{X}_1 in (17). Alternately updating \mathbf{X}_1 and \mathbf{X}_2 produces the solution of (11). The update can end after enough iterations or without further improvement of the solution. The procedure of TQNM is summarized in Algorithm 1, where the design of the extrapolation weight ω_1^k is given.

3.3. Convergence Analysis

For ease of representation, we denote $F(\mathbf{X}_1, \mathbf{X}_2)$ and $f(\mathbf{X}_1, \mathbf{X}_2)$ as $F(\mathbf{X})$ and $f(\mathbf{X})$, respectively, while $F(\mathbf{X}_i)$ and $f(\mathbf{X}_i)$ for $i = 1$ or 2 mean the other factor matrix is fixed. We first discuss the subsequence convergence.

Theorem 2. (Subsequence Convergence): *Assuming $F(\mathbf{X}^k) \rightarrow \infty$ if and only if $\|\mathbf{X}^k\|_F \rightarrow \infty$, the sequence $\{\mathbf{X}_i^k\}_{i=1}^2$ generated by Algorithm 1 with parameters μ_i^k and ω_i^k is bounded if $f(\mathbf{X})$ is gradient Lipschitz continuous on any bounded set and the Lipschitz constant L_i^k of $\nabla_{\mathbf{X}_i} f(\hat{\mathbf{X}}_i^{k-1})$ is lower and upper bounded, that is, there are a lower bound ℓ and upper bound L for all L_i^k with $0 < \ell \leq L_i^k \leq L < \infty$.*

Let $\bar{\mathbf{X}}_i$ be a limit point of $\{\mathbf{X}_i^k\}$, then there exists a subsequence of $\{\mathbf{X}_i^k\}$ converging to $\bar{\mathbf{X}}_i$, which is also a critical point of (11).

Algorithm 1 Alternating Minimization based on Proximal Linear Method for Solving (11)

Input: Incomplete matrix $\mathbf{Y}_O \in \mathbb{R}^{m \times n}$, $\mathbf{O} \in \mathbb{R}^{m \times n}$, decomposition parameter d , iteration number N , a , λ ,

μ_{\min} , randomly initialized \mathbf{X}_1^0 and \mathbf{X}_2^0 , $\mathbf{X}_1^{-1} = \mathbf{X}_1^0$, $\mathbf{X}_2^{-1} = \mathbf{X}_2^0$, $\mu_1^0 = 1$, $\mu_2^0 = 1$, $t_0 = 1$.

Output: \mathbf{X}

1: **for** $k = 1$ to N **do**

2: Update $\mu_1^k = \max\left(\left(\sigma_1(\mathbf{X}_2^{k-1})\right)^2, \mu_{\min}\right)$.

3: Update $\omega_1^k = \min\left((t_{k-1}-1)/t_k, 0.9999\sqrt{\mu_1^{k-1}/\mu_1^k}\right)$, where $t_k = (1 + \sqrt{1 + 4t_{k-1}^2})/2$.

4: Update $\hat{\mathbf{X}}_1^{k-1} = \mathbf{X}_1^{k-1} + \omega_1^k (\mathbf{X}_1^{k-1} - \mathbf{X}_1^{k-2})$ and $\mathbf{M}_1^k = \hat{\mathbf{X}}_1^{k-1} - \frac{1}{\mu_1^k} \nabla_{\mathbf{X}_1} f(\hat{\mathbf{X}}_1^{k-1}, \mathbf{X}_2^{k-1})$.

5: Update $\mathbf{X}_1^k = \mathbf{Q}_{\mathbf{M}_1^k} \mathcal{T} \mathcal{Q}_{\frac{1}{\lambda \mu_1^k}, a}(\Lambda_{\mathbf{M}_1^k}) \mathbf{R}_{\mathbf{M}_1^k}^T$, where $\mathbf{M}_1^k = \mathbf{Q}_{\mathbf{M}_1^k} \Lambda_{\mathbf{M}_1^k} \mathbf{R}_{\mathbf{M}_1^k}^T$.

6: **if** $F(\mathbf{X}_1^k, \mathbf{X}_2^{k-1}) > F(\mathbf{X}_1^{k-1}, \mathbf{X}_2^{k-1})$ **then**

7: Update $\hat{\mathbf{X}}_1^{k-1} = \mathbf{X}_1^{k-1}$ and $\mathbf{M}_1^k = \hat{\mathbf{X}}_1^{k-1} - \frac{1}{\mu_1^k} \nabla_{\mathbf{X}_1} f(\hat{\mathbf{X}}_1^{k-1}, \mathbf{X}_2^{k-1})$.

8: Update $\mathbf{X}_1^k = \mathbf{Q}_{\mathbf{M}_1^k} \mathcal{T} \mathcal{Q}_{\frac{1}{\lambda \mu_1^k}, a}(\Lambda_{\mathbf{M}_1^k}) \mathbf{R}_{\mathbf{M}_1^k}^T$, where $\mathbf{M}_1^k = \mathbf{Q}_{\mathbf{M}_1^k} \Lambda_{\mathbf{M}_1^k} \mathbf{R}_{\mathbf{M}_1^k}^T$.

9: **end if**

10: Update $\mu_2^k = \max\left(\left(\sigma_1(\mathbf{X}_1^k)\right)^2, \mu_{\min}\right)$.

11: Update $\omega_2^k = \min\left((t_{k-1}-1)/t_k, 0.9999\sqrt{\mu_2^{k-1}/\mu_2^k}\right)$.

12: Update $\hat{\mathbf{X}}_2^{k-1} = \mathbf{X}_2^{k-1} + \omega_2^k (\mathbf{X}_2^{k-1} - \mathbf{X}_2^{k-2})$ and $\mathbf{M}_2^k = \hat{\mathbf{X}}_2^{k-1} - \frac{1}{\mu_2^k} \nabla_{\mathbf{X}_2} f(\mathbf{X}_1^k, \hat{\mathbf{X}}_2^{k-1})$.

13: Update $\mathbf{X}_2^k = \mathbf{Q}_{\mathbf{M}_2^k} \mathcal{T} \mathcal{Q}_{\frac{1}{\lambda \mu_2^k}, a}(\Lambda_{\mathbf{M}_2^k}) \mathbf{R}_{\mathbf{M}_2^k}^T$, where $\mathbf{M}_2^k = \mathbf{Q}_{\mathbf{M}_2^k} \Lambda_{\mathbf{M}_2^k} \mathbf{R}_{\mathbf{M}_2^k}^T$.

14: **if** $F(\mathbf{X}_1^k, \mathbf{X}_2^k) > F(\mathbf{X}_1^k, \mathbf{X}_2^{k-1})$ **then**

15: Update $\hat{\mathbf{X}}_2^{k-1} = \mathbf{X}_2^{k-1}$ and $\mathbf{M}_2^k = \hat{\mathbf{X}}_2^{k-1} - \frac{1}{\mu_2^k} \nabla_{\mathbf{X}_2} f(\mathbf{X}_1^k, \hat{\mathbf{X}}_2^{k-1})$.

16: Update $\mathbf{X}_2^k = \mathbf{Q}_{\mathbf{M}_2^k} \mathcal{T} \mathcal{Q}_{\frac{1}{\lambda \mu_2^k}, a}(\Lambda_{\mathbf{M}_2^k}) \mathbf{R}_{\mathbf{M}_2^k}^T$, where $\mathbf{M}_2^k = \mathbf{Q}_{\mathbf{M}_2^k} \Lambda_{\mathbf{M}_2^k} \mathbf{R}_{\mathbf{M}_2^k}^T$.

17: **end if**

18: **if** $\sum_{i=1}^2 \frac{\|\mathbf{X}_i^k - \mathbf{X}_i^{k-1}\|_F}{\|\mathbf{X}_i^{k-1}\|_F} \leq 10^{-3}$ **then**

19: **break**

20: **end if**

21: **end for**

22: Calculate $\mathbf{X} = \mathbf{X}_1^k \mathbf{X}_2^k$.

The proof of Theorem 2 is given in Appendix A.

We then introduce the definition of Kurdyka-Łojasiewicz (KL) property [43] which is useful to prove the whole sequence convergence. After that, a theorem on whole sequence convergence is provided.

Definition 2. (KL Property): A function ψ satisfies the KL property at point $\bar{\mathbf{x}} \in \text{dom}(\partial\psi)$ if there exist $\eta > 0$, a neighborhood $\mathcal{B}(\bar{\mathbf{x}}, \rho) \triangleq \{\mathbf{x} : \|\mathbf{x} - \bar{\mathbf{x}}\| < \rho\}$, and a concave function $\phi(a) = c \cdot a^{1-\theta}$ for some $c > 0$ and $\theta \in [0, 1)$, such that for any $\mathbf{x} \in \mathcal{B}(\bar{\mathbf{x}}, \rho) \cap \text{dom}(\partial\psi)$ and $\psi(\bar{\mathbf{x}}) < \psi(\mathbf{x}) < \psi(\bar{\mathbf{x}}) + \eta$, it holds

$$\phi'(|\psi(\mathbf{x}) - \psi(\bar{\mathbf{x}})|) \text{dist}(\mathbf{0}, \partial\psi(\mathbf{x})) \geq 1. \quad (30)$$

Here $\text{dom}(\partial\psi) = \{\mathbf{x} : \partial\psi \neq \emptyset\}$ and $\text{dist}(\mathbf{0}, \partial\psi) = \min\{\|\mathbf{y}\| : \mathbf{y} \in \partial\psi\}$.

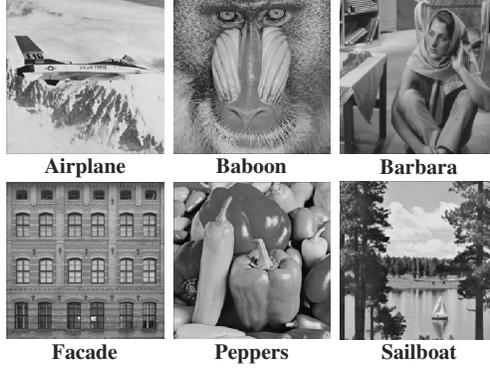


Fig. 2: Six benchmark grayscale images with dimensions 256×256 .

Theorem 3. (*Whole Sequence Convergence*): Based on Theorem 2, assuming that \mathbf{X}_i^0 is sufficiently close to $\bar{\mathbf{X}}_i$, the sequence $\{\mathbf{X}_i^k\}_{i=1}^2$ generated by Algorithm 1 converges to $\bar{\mathbf{X}}_i$, if the following conditions are satisfied:

1. The objective function $F(\mathbf{X}_i)$ conforms KL property at point $\bar{\mathbf{X}}_i$ within neighborhood $\mathcal{B}(\bar{\mathbf{X}}_i, \rho)$ for certain ρ ;
2. The sequence $\{F(\mathbf{X}_i^k)\}$ is non-increasing.

The proof of Theorem 3 is provided in Appendix B. Note that all appendices are presented in supplementary material.

In the following, we verify Algorithm 1 satisfying the conditions required in Theorems 2 and 3.

As in Theorem 2, we have already mentioned that $f(\mathbf{X})$ is gradient Lipschitz continuous. Moreover, in Algorithm 1, $\mu_1^k = \gamma L_1^k = \max(\|\mathbf{X}_2^{k-1}\|_F^2, \mu_{\min})$ is lower and upper bounded. Similarly, μ_2^k is lower and upper bounded. Since $1 < \gamma < \infty$, L_i^k for $i = 1, 2$ is also lower and upper bounded. Next we turn to the requirements in Theorem 3. KL property has been introduced for the real analytic function. Piecewise polynomial function has been proved to satisfy this property [48]. The truncated quadratic norm is defined based on a piecewise polynomial function, and $f(\mathbf{X})$ is a real polynomial function. Hence the objective function $F(\mathbf{X})$ is a real analytic function and satisfies KL inequality. For Condition 2, whenever $F(\mathbf{X}_i^k) > F(\mathbf{X}_i^{k-1})$, we set $\hat{\mathbf{X}}_i^{k-1} = \mathbf{X}_i^{k-1}$ and redo the iteration. Then the solution \mathbf{X}_i^k is acquired by minimizing (14). The difference between (14) and the objective function (12) is that (14) utilizes the first-order approximation of the quadratic term $f(\mathbf{X})$ in (12). Thus, minimizing (14) equals minimizing (12). Therefore, $F(\mathbf{X}_i^k) \leq F(\mathbf{X}_i^{k-1})$ is guaranteed.

Above all, we conclude that $\{\mathbf{X}_i^k\}_{i=1}^2$ converges to a critical point of (11).

3.4. Computational Complexity Analysis

In the k th iteration, we need to compute $\mathbf{X}_1^k \in \mathbb{R}^{m \times d}$ and $\mathbf{X}_2^k \in \mathbb{R}^{d \times n}$ given $\mathbf{Y}_O \in \mathbb{R}^{m \times n}$ and the decomposition parameter d . Without loss of generality, we assume $m \geq n \gg d$. For \mathbf{X}_1^k , according to Algorithm 1, we need to calculate $\|\mathbf{X}_2^{k-1}\|_F^2$, $\nabla_{\mathbf{X}_1} f(\hat{\mathbf{X}}_1^{k-1}, \mathbf{X}_2^{k-1})$, the SVD of \mathbf{M}_1^k , and $\mathbf{X}_1^k = \mathbf{Q}_{\mathbf{M}_1^k} \mathcal{T} \mathcal{Q}_{\frac{1}{\lambda \mu_1^k}, a}(\boldsymbol{\Lambda}_{\mathbf{M}_1^k}) \mathbf{R}_{\mathbf{M}_1^k}^T$. Calculating $\|\mathbf{X}_2^{k-1}\|_F^2$ and $\nabla_{\mathbf{X}_1} f(\hat{\mathbf{X}}_1^{k-1}, \mathbf{X}_2^{k-1})$ consumes at most $\mathcal{O}(md)$ and $\mathcal{O}(mnd)$, respectively. The

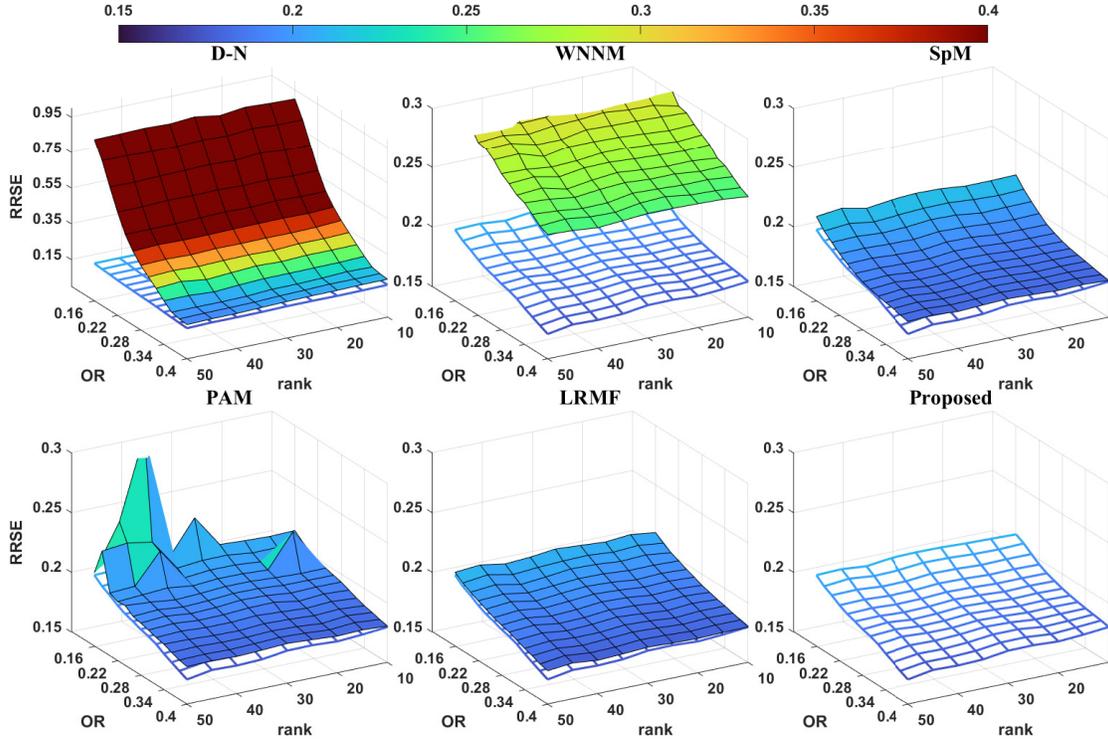


Fig. 3: Phase transition results for synthetic data completion by D-N, WNNM, SpM, PAM, LRMF, and TQNM. The x , y , and z axes represent rank, OR, and RRSE, respectively. For ease of comparison, results of TQNM are included in all cases.

truncated SVD of \mathbf{M}_1^k needs at most $\mathcal{O}(md^2)$ [49]. Calculating $\mathbf{X}_1^k = \mathbf{Q}_{\mathbf{M}_1^k} \mathcal{T} \mathcal{Q}_{\frac{1}{\lambda \mu_1^k}, a} (\boldsymbol{\Lambda}_{\mathbf{M}_1^k}) \mathbf{R}_{\mathbf{M}_1^k}^T$ costs at most $\mathcal{O}((m+d)d^2)$. Overall, the computational complexity of TQNM is $\mathcal{O}(mnd)$.

4. Experimental Results

In this section, we conduct experiments to evaluate the performance of TQNM using synthetic data and benchmark grayscale images in Fig. 2. Moreover, TQNM is compared with D-N [15], WNNM [23], SpM [25], PAM [35], and LRMF [30], which adopt different rank surrogates. We tune the parameters of the selected algorithms according to authors' suggestions to get their best performances. For TQNM, a is chosen in the interval $[2 \times 10^{-3}, 5 \times 10^{-2}]$, which does not have severe impact on the results. All these algorithms are run in MATLAB R2021a on a computer with 2.9GHz CPU and 16 GB memory.

4.1. Synthetic Data Completion

For the generation of synthetic matrices, we first create two random matrices $\mathbf{U} \in \mathbb{R}^{m \times r^*}$ and $\mathbf{V} \in \mathbb{R}^{r^* \times n}$ using MATLAB command `randn()` with $m = 200$ and $n = 200$. Multiplying \mathbf{U} and \mathbf{V} produces matrix $\hat{\mathbf{A}} \in \mathbb{R}^{200 \times 200}$. To further approximate the singular values of a practical low-rank matrix, we modify the j th singular value of $\hat{\mathbf{A}}$ as 2^{10-j} for $j \in [1, r^*]$ [37]. Finally, the synthesized matrix \mathbf{A} is acquired by adding white Gaussian noise to $\hat{\mathbf{A}}$ with signal-to-noise ratio being 15dB.

We generate nine matrices using the above strategy by setting r^* from 10 to 50 in step of 5. Then we randomly sample the matrix entities as the observation set and utilize different algorithms to recover the

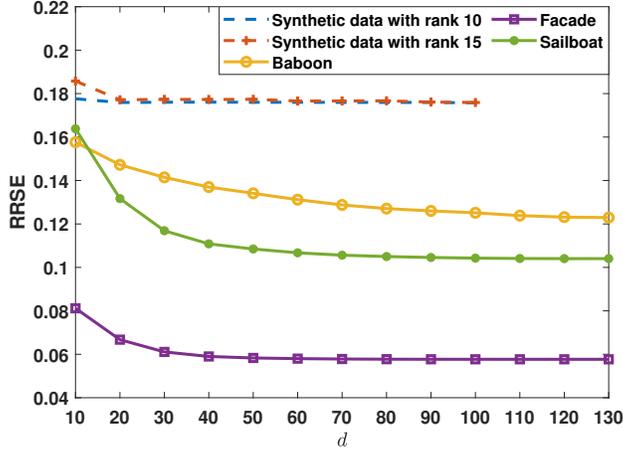


Fig. 4: RRSE versus d for synthetic data of $r^* = 10, 15$ and images *Baboon*, *Facade*, *Sailboat* at $OR = 0.5$.

unobserved elements. For each matrix, the observation ratio (OR) is set from 0.16 to 0.4 in step of 0.02. We compare the results of D-N, WNNM, SpM, PAM, LRMF, and TQNMCM in terms of the relative root square error (RRSE) defined as

$$RRSE = \|\mathbf{X}^* - \mathbf{X}_{\text{ref}}\|_F / \|\mathbf{X}_{\text{ref}}\|_F. \quad (31)$$

Here \mathbf{X}_{ref} is the ground-truth matrix, and \mathbf{X}^* is the restored matrix using the observed entities of \mathbf{X}_{ref} . For a fair comparison, the decomposition parameter d of D-N, LRMF as well as TQNMCM is chosen to be $\lceil 1.25r^* \rceil$. For PAM, we try different values of r to get its best performance.

The average phase transition results are displayed in Fig 3. For a better visualization, we show the RRSE versus OR and rank. When the ORs are low, the RRSEs obtained by D-N are very large. For PAM, under low OR with large r^* , the reconstruction error is also large. The RRSEs of TQNMCM are included in all cases for the purpose of easy comparison. It is clear that the TQNMCM provides the minimum RRSE, indicating its superiority over other competing schemes.

4.2. Choice of d

One key parameter in our algorithm is d , which should be slightly larger than the rank of the target matrix. For synthetic data in Section 4.1, the true rank is known, and we set $d = \lceil 1.25r^* \rceil$. While in some applications, the rank is unknown. In this subsection, we provide a rule of thumb to choose d for unknown rank situations.

The low-rank assumption in matrix completion requires the rank to be far smaller than $\min(m, n)$. Generally, “far smaller” means that the rank and $\min(m, n)$ are at least not in the same order of magnitude. Therefore, we suggest $d \approx 0.1 * \min(m, n)$. We perform matrix completion using synthetic matrices of $r^* = 10, 15$ with dimensions 200×200 . The RRSE versus d is plotted in Fig. 4 at $OR = 0.5$. We see that the RRSEs are already very small for $d = 20$, which corroborates our suggestion.

On the other hand, image data are of approximately low rank [20], and thus the suggestion $d \approx 0.1 * \min(m, n)$ for exact low-rank matrix might not attain excellent recovery performance. Then, we investigate

Image	OR	0.3				0.5				0.7			
		Method	RRSE	PSNR (dB)	SSIM	Time	RRSE	PSNR (dB)	SSIM	Time	RRSE	PSNR (dB)	SSIM
<i>Airplane</i>	D-N	0.1074	22.15	0.6032	0.52	0.0678	26.15	0.7923	0.55	0.0425	30.19	0.8982	0.56
	WNNM	0.1369	20.04	0.4070	11.81	0.0855	24.13	0.6230	8.13	0.0536	28.19	0.7866	8.65
	SpM	0.1009	22.69	0.5905	5.98	0.0659	26.39	0.7778	8.31	0.0426	30.18	0.8793	10.50
	PAM	0.1318	20.37	0.5036	1.37	0.0906	23.63	0.6456	3.75	0.0602	27.18	0.7784	7.48
	LRMF	0.1075	22.14	0.5552	1.88	0.0670	26.25	0.7527	2.82	0.0431	30.08	0.8717	7.69
	Proposed	0.1028	22.53	0.6382	2.35	0.0652	26.49	0.7970	4.57	0.0410	30.51	0.8863	5.09
<i>Baboon</i>	D-N	0.1592	21.44	0.4749	0.51	0.1256	23.50	0.6450	0.47	0.0990	25.56	0.7625	0.56
	WNNM	0.2351	18.05	0.2904	11.89	0.1849	20.14	0.4595	8.21	0.1383	22.66	0.6487	9.21
	SpM	0.1703	20.86	0.4238	5.95	0.1329	23.01	0.6068	8.19	0.0978	25.67	0.7689	10.75
	PAM	0.1770	20.52	0.3179	1.30	0.1575	21.53	0.4124	1.58	0.1374	22.72	0.5488	5.29
	LRMF	0.1777	20.48	0.3933	2.02	0.1378	22.70	0.5633	2.48	0.1041	25.13	0.7487	7.48
	Proposed	0.1553	21.65	0.4754	1.97	0.1231	23.67	0.6471	2.58	0.0935	26.06	0.7773	4.65
<i>Facade</i>	D-N	0.0818	27.65	0.8492	0.62	0.0567	30.84	0.9190	1.02	0.0399	33.89	0.9562	0.68
	WNNM	0.1071	25.31	0.7545	11.64	0.0739	28.54	0.8645	8.25	0.0517	31.64	0.9272	9.02
	SpM	0.0791	27.94	0.8515	6.06	0.0571	30.77	0.9118	8.52	0.0453	32.78	0.9411	10.91
	PAM	0.0996	25.95	0.7910	1.24	0.0780	28.06	0.8651	2.73	0.0561	30.93	0.9157	6.05
	LRMF	0.0790	27.95	0.8454	4.15	0.0565	30.87	0.9129	3.75	0.0397	33.92	0.9528	4.87
	Proposed	0.0767	28.21	0.8599	3.34	0.0536	31.32	0.9245	2.58	0.0373	34.47	0.9610	4.28
<i>Sailboat</i>	D-N	0.1593	21.08	0.5516	0.52	0.1050	24.70	0.7398	0.48	0.0670	28.59	0.8626	0.55
	WNNM	0.2082	18.75	0.3769	11.49	0.1379	22.33	0.5709	8.28	0.0874	26.29	0.7511	8.62
	SpM	0.1543	21.35	0.5259	6.19	0.1008	25.05	0.7282	8.71	0.0649	28.87	0.8625	11.13
	PAM	0.1873	19.67	0.4341	1.36	0.1377	22.34	0.5868	3.04	0.0926	25.79	0.7399	8.33
	LRMF	0.1648	20.78	0.4885	1.90	0.1041	24.77	0.7126	2.85	0.0661	28.72	0.8472	7.83
	Proposed	0.1510	21.54	0.5749	2.03	0.0994	25.17	0.7518	3.31	0.0634	29.08	0.8672	4.61

Table 1: Reconstruction results of grayscale images *Airplane*, *Baboon*, *Facade*, and *Sailboat* under different ORs.

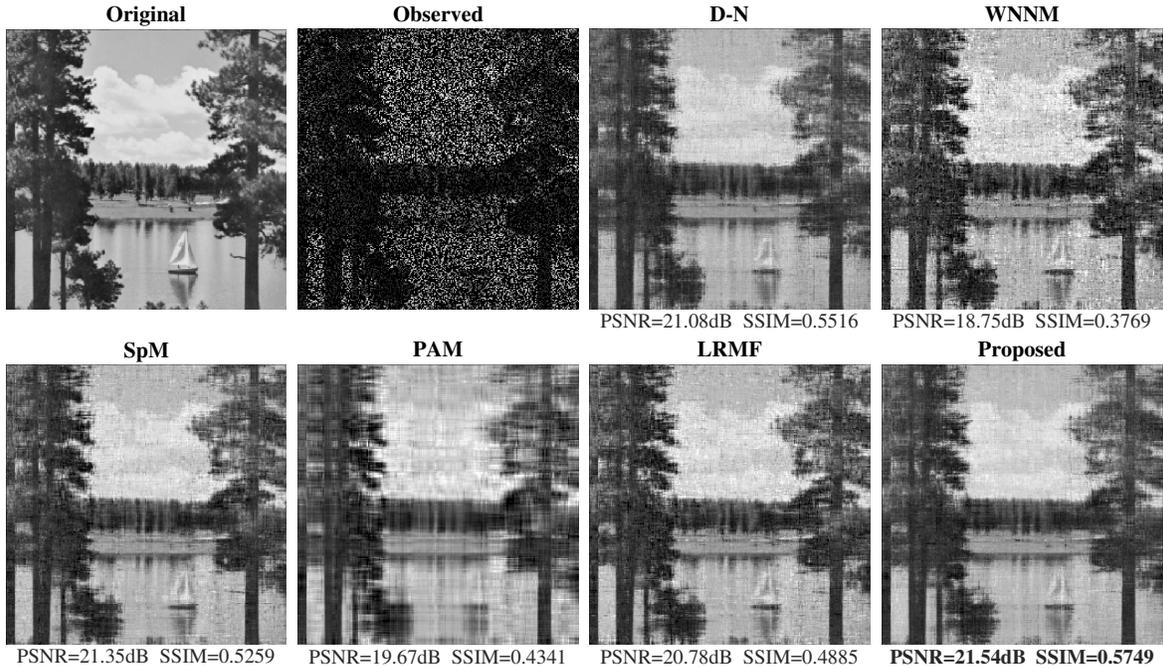


Fig. 5: Reconstruction results for image *Sailboat* under random loss at OR = 0.3. Top to bottom, left to right correspond to original image, observed image, recovered results by D-N, WNNM, SpM, PAM, LRMF, and TQNMCM.

the impact of d on the recovery accuracy for grayscale images with dimensions 256×256 at OR = 0.5. As shown in Fig. 4, a larger d produces a smaller RRSE. As enlarging d increases the computational cost, $d = 100$, viz. $d \approx 0.4 * \min(m, n)$, is a suggested choice. As for the effectiveness of this choice for image data,

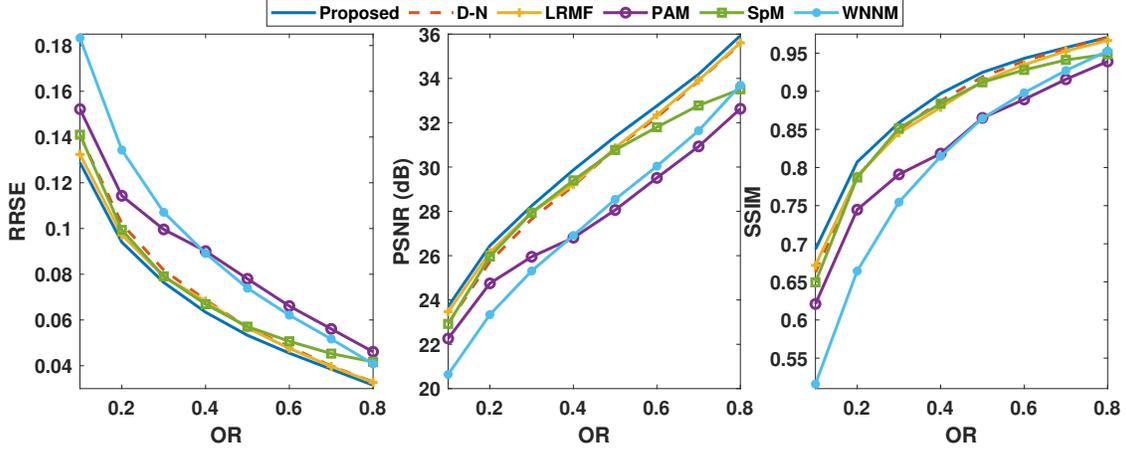


Fig. 6: RRSEs, PSNRs, and SSIMs of reconstruction results of *Facade* under different ORs by D-N, WNNM, SpM, PAM, LRMF, and TQNMNC.

	Image						
	Method	<i>Airplane</i>	<i>Baboon</i>	<i>Barbara</i>	<i>Facade</i>	<i>Peppers</i>	<i>Sailboat</i>
RRSE	D-N	0.0545	0.1000	0.0759	0.0439	0.0721	0.0848
	WNNM	0.0658	0.1352	0.0952	0.0545	0.0832	0.1076
	SpM	0.0541	0.0992	0.0752	0.0474	0.0684	0.0824
	PAM	0.0879	0.1514	0.1141	0.0825	0.1031	0.1346
	LRMF	0.0532	0.1040	0.0740	0.0436	0.0670	0.0846
	Proposed	0.0518	0.0992	0.0716	0.0408	0.0649	0.0818
PSNR (dB)	D-N	28.04	25.47	28.94	33.06	28.57	26.55
	WNNM	26.40	22.86	26.97	31.18	27.32	24.49
	SpM	28.10	25.54	29.01	32.38	29.02	26.80
	PAM	23.89	21.88	25.39	27.57	25.46	22.54
	LRMF	28.26	25.13	29.16	33.12	29.20	26.57
	Proposed	28.48	25.55	29.45	33.69	29.48	26.87
SSIM	D-N	0.8758	0.7819	0.8668	0.9487	0.8670	0.8403
	WNNM	0.8113	0.7325	0.8134	0.9267	0.8012	0.7772
	SpM	0.8596	0.7870	0.8512	0.9354	0.8509	0.8336
	PAM	0.7094	0.4577	0.7216	0.8472	0.7304	0.6535
	LRMF	0.8605	0.7743	0.8590	0.9461	0.8581	0.8271
	Proposed	0.8804	0.7836	0.8735	0.9543	0.8727	0.8446

Table 2: Reconstruction results of grayscale images in Fig. 2 under text masked observations.

more are detailed in Appendix C.

In summary, when the rank is unknown, we suggest $d \approx 0.1 * \min(m, n)$ for exactly low-rank data, and $d \approx 0.4 * \min(m, n)$ for images which have approximately low rank.

4.3. Grayscale Image Inpainting

To further investigate the effectiveness of TQNMNC, in this subsection, we conduct image inpainting experiments on the benchmark images in Fig. 2 using different algorithms. In our algorithm, d is chosen as 100. For fairness, d for D-N and LRMF is also set as 100. As for PAM, different ranks are tried, and we select the one with the best performance. Apart from RRSE, we also include the peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) [50] as evaluation metrics. These two measures are determined using the MATLAB built-in commands `psnr()` and `ssim()`.

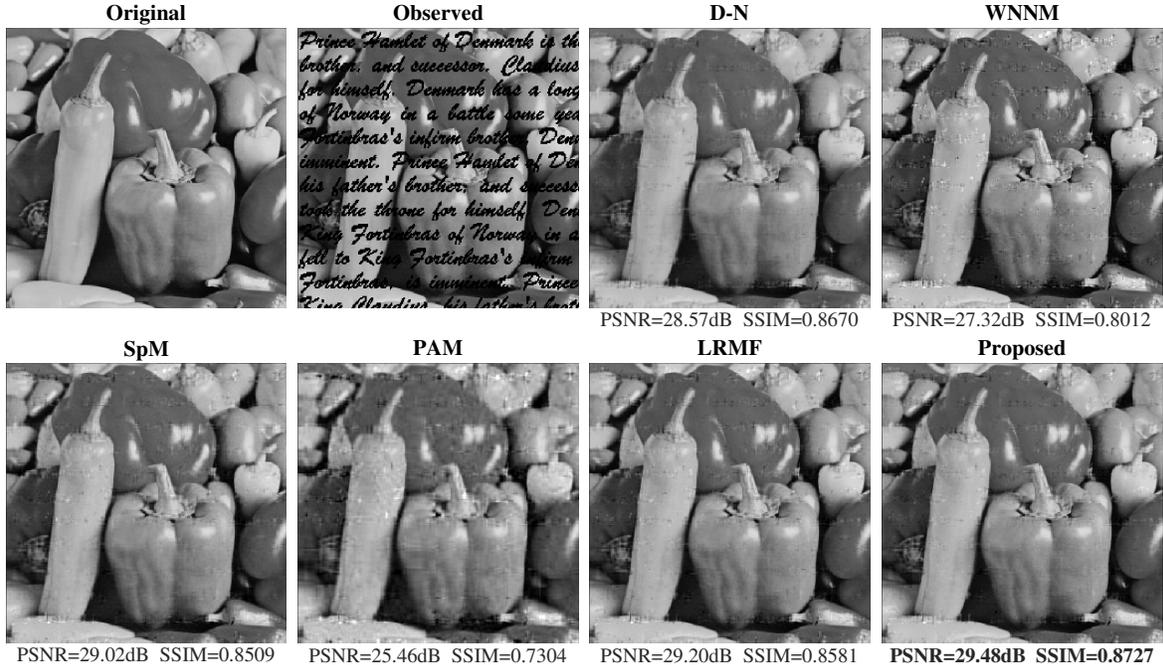


Fig. 7: Inpainting results for *Peppers* under text mask loss. The original image, text-masked image, recovered results by D-N, WNNM, SpM, PAM, LRMF, and the proposed method are shown successively.

First, we randomly pick certain number of pixels as the observation set. When ORs are 0.3, 0.5, and 0.7, the RRSEs, PSNRs, and SSIMs of inpainting results for *Airplane*, *Baboon*, *Facade*, and *Sailboat* utilizing different methods are listed in Table 1. Under most of the situations, TQNMCM realizes the best reconstruction results. Furthermore, we show the inpainting results by D-N, WNNM, SpM, PAM, LRMF, and TQNMCM for *Sailboat* at $OR = 0.3$ in Fig. 5. We can see the recovered image by PAM is a little blurred. Besides, comparing with TQNMCM, the inpainting images acquired by D-N, WNNM, SpM, and LRMF contain more noise. By contrast, TQNMCM can produce more vivid outcome.

Besides, algorithm runtimes are also listed in Table 1. Among them, D-N is the fastest, followed by PAM, our method and LRMF, while SpM and WNNM are relatively slower. D-N utilizes alternating direction method of multipliers to solve the matrix completion problem and converges in just a few tens of iterations. However, it does not consider the presence of dense noise, like Gaussian noise, which is commonly encountered in a wide range of applications. It is observed from the experimental results that the D-N exhibits poor performance for synthetic data with Gaussian noise. Both PAM and our method use bilinear factorization, which is able to decrease the computational cost. However, the performance of PAM relies heavily on the prior rank information. Tuning the rank for each application scenario is time-consuming. In contrast, our method only needs the upper bound of rank, viz. the decomposition parameter d . SpM and WNNM perform SVD on the original target matrix, whereas the proposed method computes SVD on two small factor matrices. Therefore, our method is faster than SpM and WNNM. This finding further verifies the effectiveness of our motivation to impose the truncated quadratic norm on two small factor matrices, which leads to reduced computational cost.

To further compare the performances of these algorithms under a wide range of ORs, we randomly sample 10% to 80% elements of image *Facade* and implement MC using D-N, WNNM, SpM, PAM, LRMF, and TQNMC. The RRSEs, PSNRs, and SSIMs of the restored image versus OR are plotted in Fig. 6. The image recovered by TQNMC has the smallest RRSE, highest PSNR and SSIM under all ORs.

Then we evaluate the performance of all the algorithms under text mask scenarios. Six images in Fig. 2 are masked by text, and the masked images are employed for MC. The RRSEs, PSNRs, and SSIMs for the reconstructed images are tabulated in Table 2. TQNMC performs the best among the competing algorithms. In addition, the recovery results of *Peppers* are visualized in Fig. 7. WNNM and PAM fail to fill in some loss pixels and introduce some artefacts. Comparing with D-N, SpM, and LRMF, TQNMC achieves better performance, and the recovered image is more natural.

5. Conclusion

In this paper, we present an MC algorithm based on minimizing a new matrix rank substitution named truncated quadratic norm. Taking advantage of factorization approach, we minimize the truncated quadratic norm of two small factor matrices of the original target matrix. Experimental results on synthetic data and grayscale images show the excellent performance of the proposed method.

As a future research direction, new rank function surrogates are welcomed to handle various low-rank approximation problems, not confined to MC. Besides, in practical situations, the observed matrices have inevitable noise and outliers. Thus, robust TQNMC will be another research task.

References

- [1] J. Wen, Y. Xu, J. Tang, Y. Zhan, Z. Lai, X. Guo, Joint video frame set division and low-rank decomposition for background subtraction, *IEEE Trans. Circuits Syst. Video. Technol.* 24 (12) (2014) 2034–2048.
- [2] F. Cao, M. Cai, Y. Tan, Image interpolation via low-rank matrix completion and recovery, *IEEE Trans. Circuits Syst. Video. Technol.* 25 (8) (2014) 1261–1270.
- [3] S. Iizuka, E. Simo-Serra, H. Ishikawa, Globally and locally consistent image completion, *ACM Trans. Graph.* 36 (4) (2017) 1–14.
- [4] J. Liu, J. Lin, W. Zhang, L. Nong, J. Peng, J. Wang, GCN-based proximal unrolling matrix completion for piecewise smooth signal recovery, *Signal Process.* 207 (2023) 108932.
- [5] M. Wan, Y. Yao, T. Zhan, G. Yang, Supervised low-rank embedded regression (SLRER) for robust subspace learning, *IEEE Trans. Circuits Syst. Video. Technol.* 32 (4) (2021) 1917–1927.
- [6] T. K. Pong, P. Tseng, S. Ji, J. Ye, Trace norm regularization: Reformulations, algorithms, and multi-task learning, *SIAM J. Optim* 20 (6) (2010) 3465–3489.

- [7] Y. Luo, T. Liu, D. Tao, C. Xu, Multiview matrix completion for multilabel image classification, *IEEE Trans. Image Process.* 24 (8) (2015) 2355–2368.
- [8] X. Ma, S. Zhang, K. Pena-Pena, G. R. Arce, Fast spectral clustering method based on graph similarity matrix completion, *Signal Process.* 189 (2021) 108301.
- [9] P. V. Giampouras, A. A. Rontogiannis, K. E. Themelis, K. D. Koutroumbas, Online sparse and low-rank subspace learning from incomplete data: A Bayesian view, *Signal Processing* 137 (2017) 199–212.
- [10] A. Ramlatchan, M. Yang, Q. Liu, M. Li, J. Wang, Y. Li, A survey of matrix completion methods for recommendation systems, *Big Data Mining Anal.* 1 (4) (2018) 308–323.
- [11] M. Yang, S. Xu, A novel deep quantile matrix completion model for top-N recommendation, *Knowl.-Based Syst.* 228 (2021) 107302.
- [12] V. Garg, P. Giménez-Febrer, A. Pagès-Zamora, I. Santamaria, DOA estimation via shift-invariant matrix completion, *Signal Process.* 183 (2021) 107993.
- [13] X. Hu, N. Tong, J. Wang, S. Ding, X. Zhao, Matrix completion-based MIMO radar imaging with sparse planar array, *Signal Process.* 131 (2017) 49–57.
- [14] X. Li, H. Zhang, R. Zhang, Matrix completion via non-convex relaxation and adaptive correlation learning, *IEEE Trans. Pattern Anal. Mach. Intell.* 45 (02) (2023) 1981–1991.
- [15] F. Shang, J. Cheng, Y. Liu, Z.-Q. Luo, Z. Lin, Bilinear factor matrix norm minimization for robust PCA: Algorithms and applications, *IEEE Trans. Pattern Anal. Mach. Intell.* 40 (9) (2018) 2066–2080.
- [16] E. Candès, B. Recht, Exact matrix completion via convex optimization, *Commun. ACM* 55 (6) (2012) 111–119.
- [17] E. J. Candès, T. Tao, The power of convex relaxation: Near-optimal matrix completion, *IEEE Trans. Inf. Theory* 56 (5) (2010) 2053–2080.
- [18] J.-F. Cai, E. J. Candès, Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.* 20 (4) (2010) 1956–1982.
- [19] J. Fan, R. Li, Variable selection via nonconcave penalized likelihood and its oracle properties, *J. Amer. Stat. Assoc.* 96 (456) (2001) 1348–1360.
- [20] Y. Hu, D. Zhang, J. Ye, X. Li, X. He, Fast and accurate matrix completion via truncated nuclear norm regularization, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (9) (2013) 2117–2130.
- [21] X. Su, Y. Wang, X. Kang, R. Tao, Nonconvex truncated nuclear norm minimization based on adaptive bisection method, *IEEE Trans. Circuits. Syst. Video. Technol.* 29 (11) (2019) 3159–3172.

- [22] C. Lu, J. Tang, S. Yan, Z. Lin, Generalized nonconvex nonsmooth low-rank minimization, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., Columbus, OH, USA, 2014, pp. 4130–4137.
- [23] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, L. Zhang, Weighted nuclear norm minimization and its applications to low level vision, *Int. J. Comput. Vis.* 121 (2) (2017) 183–208.
- [24] E. J. Candès, M. B. Wakin, S. P. Boyd, Enhancing sparsity by reweighted ℓ_1 minimization, *J. Fourier Anal. Appl.* 14 (5) (2008) 877–905.
- [25] F. Nie, H. Wang, H. Huang, C. Ding, Joint Schatten p -norm and ℓ_p -norm robust matrix completion for missing value recovery, *Knowl. Inf. Syst.* 42 (3) (2015) 525–544.
- [26] W.-J. Zeng, H. C. So, Outlier-robust matrix completion via ℓ_p -minimization, *IEEE Trans. Signal Process.* 66 (5) (2017) 1125–1140.
- [27] W. Zuo, D. Meng, L. Zhang, X. Feng, D. Zhang, A generalized iterated shrinkage algorithm for non-convex sparse coding, in: Proc. IEEE Int. Conf. Comput. Vis., Sydney, Australia, 2013, pp. 217–224.
- [28] L. Chen, Y. Gu, The convergence guarantees of a non-convex approach for sparse recovery, *IEEE Trans. Signal Process.* 62 (15) (2014) 3754–3767.
- [29] C. Yang, X. Shen, H. Ma, B. Chen, Y. Gu, H. C. So, Weakly convex regularized robust sparse recovery methods with theoretical guarantees, *IEEE Trans. Signal Process.* 67 (19) (2019) 5046–5061.
- [30] L. Chen, X. Jiang, X. Liu, Z. Zhou, Logarithmic norm regularized low-rank factorization for matrix and tensor completion, *IEEE Trans. Image Process.* 30 (2021) 3434–3449.
- [31] C. Yang, Y. Gu, B. Chen, H. Ma, H. C. So, Learning proximal operator methods for nonconvex sparse recovery with theoretical guarantee, *IEEE Trans. Signal Process.* 68 (2020) 5244–5259.
- [32] C. Yang, Y. Gu, B. Chen, H. Ma, H. C. So, Two-dimensional learned proximal gradient algorithm for fast sparse matrix recovery, *IEEE Trans. Circuits and Syst. II, Exp. Briefs* 68 (4) (2020) 1492–1496.
- [33] W. Dai, O. Milenkovic, SET: An algorithm for consistent matrix completion, in: Proc. IEEE Int. Conf. Acoust. Speech Signal Process., Dallas, Texas, USA, 2010, pp. 3646–3649.
- [34] P. Jain, P. Netrapalli, S. Sanghavi, Low-rank matrix completion using alternating minimization, in: Proc. 45th Annu. ACM Symp. Theory Comput. (STOC), Palo Alto, California, USA, 2013, pp. 665–674.
- [35] Z. Zhu, Q. Li, G. Tang, M. B. Wakin, The global optimization geometry of low-rank matrix optimization, *IEEE Trans. Inf. Theory* 67 (2) (2021) 1308–1331.
- [36] Z. Wang, L. Sun, C. Wu, W. Zhu, S. Yang, Joint online transcoding and geo-distributed delivery for dynamic adaptive streaming, in: Proc. INFOCOM IEEE Conf. Comput. Commun., Toronto, Canada, 2014, pp. 91–99.

- [37] Z. Wang, M.-J. Lai, Z. Lu, W. Fan, H. Davulcu, J. Ye, Orthogonal rank-one matrix pursuit for low rank matrix completion, *SIAM J. Sci. Comput.* 37 (1) (2015) A488–A514.
- [38] Q. Shi, H. Lu, Y.-M. Cheung, Rank-one matrix completion with automatic rank estimation via ℓ_1 -norm regularization, *IEEE Trans. Neural. Netw. Learn. Syst.* 29 (10) (2017) 4744–4757.
- [39] X. P. Li, Q. Liu, H. C. So, Rank-one matrix approximation with ℓ_p -norm for image inpainting, *IEEE Signal Process. Lett.* 27 (2020) 680–684.
- [40] Y. Hu, C. Zhao, D. Cai, X. He, X. Li, Atom decomposition with adaptive basis selection strategy for matrix completion, *ACM Trans. Multimedia Comput. Commun. Appl.* 12 (3) (2016) 1–25.
- [41] J. Abernethy, F. Bach, T. Evgeniou, J.-P. Vert, A new approach to collaborative filtering: Operator estimation with spectral regularization., *J. Mach. Learn. Res.* 10 (3) (2009) 803–826.
- [42] R. Cabral, F. De la Torre, J. P. Costeira, A. Bernardino, Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition, in: *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, Australia, 2013, pp. 2488–2495.
- [43] Y. Xu, W. Yin, A globally convergent algorithm for nonconvex optimization based on block coordinate update, *J. Sci. Comput.* 72 (2) (2017) 700–734.
- [44] Y. Xu, W. Yin, A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion, *SIAM J. Imaging Sci.* 6 (3) (2013) 1758–1789.
- [45] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sci.* 2 (1) (2009) 183–202.
- [46] Y. Xu, Alternating proximal gradient method for sparse nonnegative Tucker decomposition, *Math. Program. Comput.* 7 (1) (2015) 39–70.
- [47] L. Mirsky, A trace inequality of John von Neumann, *Monatshefte für math.* 79 (4) (1975) 303–306.
- [48] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality, *Math. Oper. Res.* 35 (2) (2010) 438–457.
- [49] X. Li, S. Wang, Y. Cai, Tutorial: Complexity analysis of singular value decomposition and its variants, *arXiv preprint arXiv:1906.12085*.
- [50] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Trans. Image Process.* 13 (4) (2004) 600–612.