



Research article

A fast matrix completion method based on truncated $L_{2,1}$ norm minimization

Zhengyu Liu¹, Yufei Bao¹, Changhai Wang³, Xiaoxiao Chen¹ and Qing Liu^{1,2,*}

¹ School of Electronics and Information Engineering, West Anhui University, Lu'an 237012, China

² School of Mathematics and Big Data, Anhui University of Science and Technology, Huai'nan 232001, China

³ Software Engineering College, Zhengzhou University of Light Industry, No.136 Science Avenue, Zhengzhou 450000, China

* **Correspondence:** Email: clyqig2008@126.com; Tel: +8619395126610.

Abstract: In recent years, a truncated nuclear norm regularization (TNNR) method has obtained much attention from researchers in machine learning and image processing areas, because it is much more accurate on matrices with missing data than other traditional methods based on nuclear norm. However, the TNNR method is reported to be very slow, due to its large number of singular value decomposition (SVD) iterations. In this paper, a truncated $L_{2,1}$ norm minimization method was presented for fast and accurate matrix completion, which is abbreviated as TLNM. In the proposed TLNM method, the truncated nuclear norm minimization model of TNNR was improved to a truncated $L_{2,1}$ norm minimization model that aimed to optimize the truncated $L_{2,1}$ Norm and a weighted noisy matrix simultaneously for improving the accuracy of TLNM. Using Qatar Riyal (QR) decomposition to calculate the orthogonal bases for reconstructing recovery results, the proposed TLNM method is much faster than the TNNR method. Adequate results for color images validate the effectiveness and efficiency of TLNM comparing with TNNR and other competing methods.

Keywords: matrix completion; non-nuclear norm; truncated $L_{2,1}$ norm; low-rank matrix; data recovery; image processing

1. Introduction

Low rank matrix completion, which is a method of estimating the missing values in a matrix, is becoming more and more important for the research areas that includes image processing [1,2], machine learning [3,4], information theory [5,6], wireless communications [7,8], radar target localization [9,10], etc. Since most real-world data matrices may have low-rank structures, the unobserved values in a matrix $M (M \in \mathbb{R}^{m \times n}, 0 < m \leq n)$ can be recovered accurately by optimizing the following rank minimization problem,

$$\min_X \text{rank}(X), \text{ s. t. } X_\Omega = M_\Omega, \quad (1)$$

where $X \in \mathbb{R}^{m \times n}$ is a low rank matrix whose rank is r and $0 < r \ll \min(m, n)$. Ω is the collection of positions of observations in M and X . M_Ω is a matrix whose values of unobserved entries are initialized to zeros. Since the matrix rank is discrete and not convex, the problem in Eq (1) is difficult to optimize [11]. The most widely used matrix completion methods include nuclear norm minimization methods, fast methods based on matrix factorization, and weighted nuclear norm minimization methods, where the nuclear norm of a matrix is the summation of its singular values [12].

The nuclear norm minimization methods include singular value thresholding (SVT) [13], fixed point continuation with approximating singular value decomposition (FPCA) [14], approximated proximal gradient (APG) [15], and feature matrix based nuclear norm minimization (FNNM) [34], etc. Since the nuclear norm is the tightest convex relaxation of matrix rank, these nuclear norm minimization methods can converge accurately on synthetic data-sets that have strict low rank structures with a theoretical guarantee [16]. However, it has been reported that these nuclear norm minimization methods are very slow on large scale matrices, because of the full SVD iterations in their updating steps. Another disadvantage of these methods is that they are not accurate when dealing with matrices of complex structures in some real-world applications [17].

The fast matrix completion methods proposed in recent years include matrix bi-factorization (MBF) [18], fast tri-factorization (FTF) [19], and low rank matrix fitting (LMaFit) [20], whose common purpose is to reduce the computational costs of traditional methods using SVD. These fast methods suppose that the incomplete matrix containing missing values can be decomposed into the multiplication of two or three small scale sub-matrices at first. Then, they recover the missing values in an incomplete matrix using QR decomposition to calculate the variables for reducing their computational costs. Sufficient experimental results show that these fast methods based on matrix factorization are faster than the nuclear norm optimization methods, i.e., SVT [13], FPCA [14], APG [15], and so on. However, these fast methods have been reported to be not accurate especially on test matrices with full rank structures in some real applications [17].

The weighted nuclear norm minimization methods include weighted nuclear norm minimization (WNNM) [21], truncated Schatten-p norm minimization [22], Schatten capped p norm minimization (SCPN) [23], truncated nuclear norm regularization (TNNR) [17], truncated quadratic norm minimization [44], and so on. In the WNNM method [21], different singular values are assigned with different weights by a decreasing function, i.e., a large singular value can be assigned with a small weight. The main reason is that the useful information of a matrix can be reconstructed by the large singular values and singular vectors. It has been reported that WNNM can generate more accurate results than the nuclear norm minimization methods can. The truncated Schatten-p norm minimization method and SCPN method proposed two improved weighted functions for singular values and can

optimize their non-convex minimization model efficiently by an alternating direction method of multipliers. They can converge much faster than the WNNM method.

The TNNR method [17] proposed a novel truncated nuclear norm, which is the summation of the smallest $r - t$ singular values, where t is the number of subtracted singular values and r is the rank of incomplete matrix. In the TNNR method, the largest t singular values are not optimized, which leads to a fact that the main useful information of tested matrix is retained. Hence, TNNR is much more accurate than the SVT, APG, WNNM, SCPN, FTF, and MBF methods. However, it has been reported that the TNNR method converges more slowly than the SVT method does. In order to reduce the number of iterations of TNNR, a modified TNNR method based on weighted residual error (TNNR-WRE) [24] has been proposed. In the TNNR-WRE method, the rows with less missing values are recovered prior to the rest ones by optimizing a weighted residual error matrix. Sufficient experimental results show that the modified TNNR-WRE method is much faster than the TNNR method, because the number of SVD iterations of the former is much smaller than that of the latter. However, the TNNR-WRE method uses SVD decomposition to calculate the singular values of matrices, which leads to a fact that the computation cost of TNNR-WRE is as large as that of TNNR. Consequently, the TNNR-WRE method may become slow when dealing with real world images. In the truncated quadratic norm minimization method [44], a truncated quadratic norm has been proposed as a better relaxation function of matrix rank than nuclear norm. Experimental results show that this method performs well in terms of convergence accuracy. However, it is much slower than the fast methods using QR, such as LMaFit and FTF.

In recent years, the robustness of matrix completion methods and the balance between speed and convergence accuracy have received increasing attention. On one side, some robust matrix completion methods have been proposed, such as the robust matrix completion methods based on l_p norm minimization [25,26], the robust M-estimation based matrix completion method [27], the robust matrix completion based on half-quadratic optimization [28], and some robust methods based on l_0 norm [29] or l_1 norm [30], or l_2 norm [31]. Compared with the traditional matrix completion methods, this class of methods have better performances on parameter robustness and resistance to non-Gauss noises. On the other side, some fast and accurate matrix completion methods based on matrix factorization and $L_{2,1}$ norm minimization (the sum of Frobenius norms of rows in X) have been investigated, such as an iteratively reweighted $L_{2,1}$ norm minimization method (IRLNM) [32], a modified fast matrix bi-factorization (FMBF) [33] method, and so on. Since the IRLNM method replaces the SVD decomposition with a less computationally intensive QR decomposition to calculate the eigenvectors, it is faster than the methods requiring SVD. The recently proposed FMBF method is faster than the IRLNM method, because the former just performing one time of QR decomposition in its each iteration, while the latter performs two times of QR decompositions.

For further improving the speed and convergence accuracy of TNNR, a truncated $L_{2,1}$ norm minimization method is proposed, which can be abbreviated as TLNM. The major contributions of our proposed TLNM can be concluded as follows,

- A novel truncated $L_{2,1}$ Norm is designed for low rank matrix recovery. Since the truncated $L_{2,1}$ Norm is a better relaxation of rank function than the truncated nuclear norm, the proposed TLNM method is much more accurate than the TNNR method and other state-of-arts methods.
- The proposed truncated $L_{2,1}$ Norm minimization model for matrix completion can be solved extremely fast by QR decomposition, which indicates that the proposed TLNM method should be much faster than the TNNR method.

The proposed truncated $L_{2,1}$ norm and the optimization model in this paper will have a significant impact on relevant research areas, such as low rank matrix representation [42], background modeling [43], sparse signal recovery [45], and so on.

2. Related works

TNNR is a widely used matrix completion method, which has received widespread attention in many research fields, such as signal processing [35–37], image processing [38,39], neural networks [40,41], and so on. However, it seems to become slow in some real applications. Hence, a modified TNNR method based on truncated $L_{2,1}$ norm is investigated to increase the speed of TNNR. Before introducing our proposed TLNM method, the optimization model of TNNR, and some related works should be introduced in section.

2.1. The TNNR method [17]

Suppose $X(X \in \mathbb{R}^{m \times n})$ is a low rank matrix corresponding to a partially observed matrix and its SVD decomposition is that $X = USV^T$, where $U(U \in \mathbb{R}^{m \times m})$ and $V(V \in \mathbb{R}^{n \times n})$ are orthogonal matrices, $S(S \in \mathbb{R}^{m \times n})$ is a diagonal matrix whose diagonal entries are the singular values of X . The truncated nuclear norm of X is defined as follows,

$$\|X\|_t = \sum_{i=t+1}^m \sigma_i, \quad (2)$$

where $\sigma_i = S_{i,i}$ and the singular values are arranged in decreasing order, i.e., $\sigma_i \geq \sigma_j$ ($0 < i < j \leq m$). The parameter t ($0 \leq t < m$) stands for the number of subtracted singular values.

In the TNNR method, the truncated nuclear norm is rewritten as follows,

$$\|X\|_t = \|X\|_* - \sum_{i=1}^t \sigma_i, \quad (3)$$

where $\|X\|_*$ is the nuclear norm of X , i.e., $\|X\|_* = \sum_{i=1}^m \sigma_i$. According to the Von Neumann's trace inequality, it is not difficult to see that

$$\|X\|_t = \|X\|_* - \max_{A,B} \text{Tr}(AXB^T), \quad (4)$$

where A ($A \in \mathbb{R}^{t \times m}$) and B ($B \in \mathbb{R}^{t \times n}$) are row orthogonal matrices. $\text{Tr}(X)$ stands for the trace function of matrix X , i.e., the summation of diagonal entries of X .

According to Eq (4), the optimization model of TNNR can be formulated as follows,

$$\min_X \|X\|_* - \max_{A,B} \text{Tr}(AXB^T), \text{ s. t. } X_\Omega = M_\Omega. \quad (5)$$

The variables in Eq (5) can be optimized alternatively using an alternating direction method of multipliers. The readers are suggested to see more details about the optimization process in [17].

Sufficient experimental results show that the TNNR method converges more accurately than the SVT, APG, MBF and FTF methods do. However, it has been reported that the TNNR method is not fast on images with full rank structures, because of its large number of singular value decomposition updating iterations.

2.2. The MBF method [18]

In order to improve the speed of matrix completion, a fast matrix completion method based on MBF has been proposed recently by Liu [18]. In the MBF method, the underlying matrix X that needs recovering is decomposed as follows,

$$X = CD, \quad (6)$$

where the variable $C \in \mathbb{R}^{m \times r}$ is a column orthogonal matrix, $D \in \mathbb{R}^{r \times n}$ is a real matrix. r is a positive integer and $r \in (0, m]$.

Then, a nuclear norm minimization problem is formulated for fast matrix completion as follows,

$$\min_{C,D} \|D\|_*, \text{ s. t. } X = CD, X_\Omega = M_\Omega. \quad (7)$$

The nuclear norm minimization problem in Eq (7) can be solved using an alternation direction method efficiently. Since the size of variable D is $r \times n$ and $r \ll n$, the computation cost of SVD on D is much smaller than that of SVD on X . Hence, the speed of MBF is faster than TNNR, SCPN, FNNM, SVT, etc.

However, the accuracy of the MBF method is not as large as the recently proposed methods, such as TNNR, SCPN, and FNNM. The explanation is that the variable D should be updated by a singular value thresholding operator [13], which results in a convergence accuracy of MBF that is almost equal to those of SVT and APG. Moreover, the MBF method may become not fast when it deals with matrices of full rank. It may be because that the parameter r , i.e., the row number of sub-matrix D , should be arranged with a large value and the computation cost of SVD decomposition on matrix D will become large obviously in such case.

3. Our proposed method

In an attempt to increase the speed and convergence accuracy of TNNR, a truncated $L_{2,1}$ norm minimization method using QR decomposition for fast and accurate matrix completion is proposed in this section.

3.1. The truncated $L_{2,1}$ norm

Suppose that the optimal matrix X for incomplete matrix M is decomposed as that

$$X = CD + E, \quad (8)$$

where the definitions of C and D are the same as those of C and D in Eq (6), respectively. E is a noise matrix in size of $m \times n$. Then, the $L_{2,1}$ norm of D is formulated as follows,

$$\|D\|_{2,1} = \sum_{i=1}^r \|D_{i\cdot}\|_F, \quad (9)$$

where $D_{i\cdot}$ stands for the i^{th} row of D and $\|D_{i\cdot}\|_F$ is the Frobenius norm of $D_{i\cdot}$, i.e.,

$$\|D_{i\cdot}\|_F = \sqrt{\sum_{j=1}^n D_{i,j}^2}. \quad (10)$$

In fact, the $L_{2,1}$ norm of D shown in Eq (9) can be reformulated as follows,

$$\|D\|_{2,1} = \text{Tr}(C^T X H^+), \quad (11)$$

where H^+ is the Moore-Penrose pseudo inverse of H and $H (H \in \mathbb{R}^{m \times n})$ is a row normalized matrix, i.e.,

$$D = GH, \quad (12)$$

where the variable G is a diagonal matrix in size of $r \times r$, $G_{i,i}$ and $H_{i \cdot}$ are respectively defined as follows,

$$G_{i,i} = \|D_{i \cdot}\|_F, \quad (13)$$

and

$$H_{i \cdot} = \frac{D_{i \cdot}}{G_{i,i}}, \quad (14)$$

where $D_{i \cdot}$ stands for the i^{th} row of D and $H_{i \cdot}$ stands for the i^{th} row of H . According to Eq (3), the truncated $L_{2,1}$ norm of matrix D is defined as follows,

$$\|D\|_{t(2,1)} = \sum_{i=t+1}^r \|D_{i \cdot}\|_F, \quad (15)$$

where the parameter t ($0 \leq t < r$) stands for the number of subtracted Frobenius norms of rows in D . In view of Eqs (9) and (11), it is not difficult to see that $\|D\|_{t(2,1)}$ is equal to

$$\|D\|_{t(2,1)} = \text{Tr}(\hat{C}^T X H^+), \quad (16)$$

where $\hat{C} = (\hat{c}_1, \hat{c}_2, \dots, \hat{c}_r)$ is a real matrix and its rows are defined as follows,

$$\hat{c}_i = \begin{cases} (0, 0, \dots, 0)^T, & \text{if } i \leq t; \\ c_i, & \text{if } i > t. \end{cases} \quad (17)$$

where c_i stands for the i^{th} column of C . Since the variable H in Eq (12) is not an orthogonal matrix, it is easy to see that the truncated $L_{2,1}$ norm in Eq (16) can be solved very fast by QR decomposition.

3.2. The proposed truncated $L_{2,1}$ norm minimization based matrix completion method (TLNM)

Suppose the underlying matrix X has been decomposed as in Eq (8). The optimization problem of TLNM is given as follows,

$$\min_{X, C, D, E} \|D\|_{t(2,1)} + \frac{\mu}{2} \|WE\|_F, \text{ s.t. } X = CD + E, X_\Omega = M_\Omega, \quad (18)$$

where W is a diagonal matrix with positive randomized diagonal entries. The remained variables, such as X , C , D , and E , have been defined in Eqs (8) and (12). The parameter μ is a positive real number.

Since $\|D\|_{t(2,1)}$ is equal to $\text{Tr}(\hat{C}^T X H^+)$, the constrained optimization model in Eq (18) can be relaxed as the following model,

$$\min_{X, C, \hat{C}, D, H, E} \text{Tr}(\hat{C}^T X H^+) + \frac{\mu}{2} \|WE\|_F + \frac{\mu}{2} \|X - CD - E\|_F. \quad (19)$$

By letting $X_1 = X_M$, the variables i.e., X , C , D , and E in Eq (19), are updated alternatively one by one.

First, the variable C_{k+1} is updated by solving the following sub-problem,

$$C_{k+1} = \arg \min_C \| X_k - CD_k - E_k \|_F . \quad (20)$$

According to the work by Wen [20], the optimization problem in Eq (20) can be rewritten as the following problem

$$C_{k+1} = \arg \min_C \| C - T \|_F . \quad (21)$$

where

$$T = (X_k - E_k)D_k^T . \quad (22)$$

Because the variable C is an orthonormal matrix, it can be optimized with the help of QR. Suppose the QR decomposition of T is

$$T = QP, \quad (23)$$

where $Q = (q_1, q_2, \dots, q_m)$ is an orthonormal matrix, $P (P \in \mathbb{R}^{m \times r})$ is an upper triangular matrix. Then, C_{k+1} can be updated as follows,

$$C_{k+1} = (q_1, q_2, \dots, q_r), \quad (24)$$

where $q_i (i \in [1, r])$ is a column vector whose length is equal to m . According to Eq (17), the variable \hat{C}_{k+1} can be updated as follows,

$$\hat{C}_{k+1} = C_{k+1} , \quad (25)$$

$$\hat{c}_i = (0, \dots, 0)^T, \quad (26)$$

where \hat{c}_i stands for the i^{th} row of \hat{C}_{k+1} and $i \in [1, t]$.

Second, the variable D_{k+1} is updated by solving the following problem,

$$D_{k+1} = \arg \min_D \| X_k - C_{k+1}D - E_k \|_F . \quad (27)$$

Since C_{k+1} is a column orthogonal matrix, it is not difficult to see that

$$D_{k+1} = C_{k+1}^T (X_k - E_k) . \quad (28)$$

Third, according to the problem in Eq (19), the matrix X_{k+1} can be updated by solving the following minimization problem,

$$\min_X F(X) = Tr(\hat{C}_k^T X H_k^+) + \frac{\mu_k}{2} \| X - C_k D_k - E_k \|_F . \quad (29)$$

Let $\frac{\nabla F}{\nabla X} = 0$, we have,

$$\frac{\nabla Tr(\hat{C}_k^T X H_k^+)}{\nabla X} + \mu_k (X - C_k D_k - E_k) = 0, \quad (30)$$

$$\hat{C}_k H_k^+ + \mu_k (X - C_k D_k - E_k) = 0, \quad (31)$$

In order to reduce the computation cost for solving Eq (31), the matrix $\hat{C}_k H_k^+$ is replaced with

$\hat{C}_k H_k$, because the matrices H^{+T} and H have the same singular vectors. Consequently, we have,

$$\hat{C}_k H_k + \mu_k (X - X_k) = 0, \quad (32)$$

and

$$X_{k+1} = X_k - \frac{1}{\mu_k} \hat{C}_k H_k, \quad (33)$$

$$X_{k+1} = X_{k+1} - (X_{k+1})_{\Omega} + M_{\Omega}. \quad (34)$$

Fourth, the variable E_{k+1} is updated by solving the following problem,

$$\min_E \|WE\|_F + \|X_{k+1} - C_{k+1} D_{k+1} - E\|_F. \quad (35)$$

The problem in Eq (35) can be solved via a gradient descent search, and

$$E_{k+1} = \lambda (X_{k+1} - C_{k+1} D_{k+1}) W^+. \quad (36)$$

where λ ($0 < \lambda < 1$) is a positive parameter that regulates the step size of gradient descent search.

Algorithm 1. The updating steps of our proposed TLNM.

Input: M and Ω ;

Initialization:

$$\lambda > 0, \mu_0 > 0, \rho > 0, \varepsilon > 0; X_0 = M; C_0 = \text{eye}(m, r), \\ D_0 = \text{eye}(r, n); E_0 = \text{zeros}(m, n).$$

Repeat:

$$C_{k+1}: \quad \text{Eq (24);} \quad \hat{C}_{k+1}: \quad \text{Eq (25) and Eq (26);} \\ D_{k+1}: \quad \text{Eq (23);} \quad X_{k+1}: \quad \text{Eq (33) and Eq (34);} \\ E_{k+1}: \quad \text{Eq (36);} \quad \mu_{k+1}: \quad \text{Eq (37);} \\ k = k + 1.$$

Until: $\|X_k - X_{k-1}\|_F \leq \varepsilon$.

Output: $X_{opt} = X_k$. (The recovered image.)

Finally, the parameter μ_{k+1} , which regulates the step size shown in Eq (33), is updated as follows,

$$\mu_{k+1} = \rho \cdot \mu_k, \quad (37)$$

where ρ ($\rho > 1$) and $\mu_0 > 0$.

In general, the proposed truncated $L_{2,1}$ norm minimization method that is abbreviated as TLNM is obviously faster than the TNNR method and other methods proposed in recent years. In order to facilitate the implementation of our proposed TLNM method for the readers, the updating steps of TLNM are summarized in Algorithm 1.

3.3. Complexity analysis of our proposed TLNM method

In this section, the computation complexities of TLNM, TNNR and other popular methods are summarized in Table 1.

Table 1. The computation complexities of TLNM and other methods.

Method	Complexity
TLNM	$O(m \cdot r^2)$
TNNR	$O(m \cdot n^2)$
LMaFit	$O(m \cdot r^2)$
FNNM	$O(m \cdot n^2)$
SCPN	$O(m \cdot n^2)$
MBF	$O(m \cdot r^2 + r^3)$

In Table 1, m is the row number of M , n is the column number of M and $r(0 < r \ll m)$ is the estimated rank of M . Since the most CPU time of TLNM is spent on executing QR for calculating the orthogonal matrix C , the complexity of TLNM is equal to that of QR, i.e., $O(m \cdot r^2)$. The most CPU time of TNNR is spent on using SVD to compute the singular values at its each iteration. Consequently, the complexity of TNNR is equal to that of SVD, i.e., $O(m \cdot n^2)$. Because the LMaFit method is also fast method based on QR, its complexity is equal to that of TLNM. FNNM and SCPN are two kinds of methods based on weighted nuclear norm minimization, in which the SVD decomposition is an essential tool for computing the singular values. Thus, the computation complexities of FNNM and SCPN are all equal to that of SVD. Since the MBF method uses both QR decomposition and SVD decomposition at its each iteration, its computation complexity is $O(m \cdot r^2 + r^3)$.

According to the complexities shown in Table 1, it is clear that the computation complexity of TLNM is much smaller than those of TNNR, FNNM and SCPN.

4. Experimental results

In the experiments, TLNM is tested on a number of real-world color images and its convergence accuracy and speed are compared with the results of other popular methods including LMaFit [20], TNNR [17], MBF [18], SCPN [23], and FNNM [34].

In the experiments for TLNM, the parameter λ is set to 0.001, μ_0 is set to $\frac{100}{\|M\|_F}$ and t is selected from 1~25. The value of parameter ρ should be adjusted according to the tested data. The maximum iteration numbers for TLNM, TNNR, MBF, FNNM, SCPN, and MBF are set to 300, 300, 300, 500, 500, and 500, respectively. The parameters for TNNR, SCPN, FNNM, LMaFit, and MBF are adjusted optimally.

All the experiments are conducted on a laptop computer equipped with i7-1195G7 CPU and 32 GB of RAM memory.

4.1. Images containing random missing values

For evaluating our proposed TLNM method, 16 colorful images are tested. These images shown in Figure 1 are also widely used by some popular matrix completion methods proposed in recent years, such as the TNNR [17] method, a low-rank quaternion matrix completion method [1], and a non-local robust quaternion method [2]. In our proposed TLNM, the channels of these images are recovered separately.

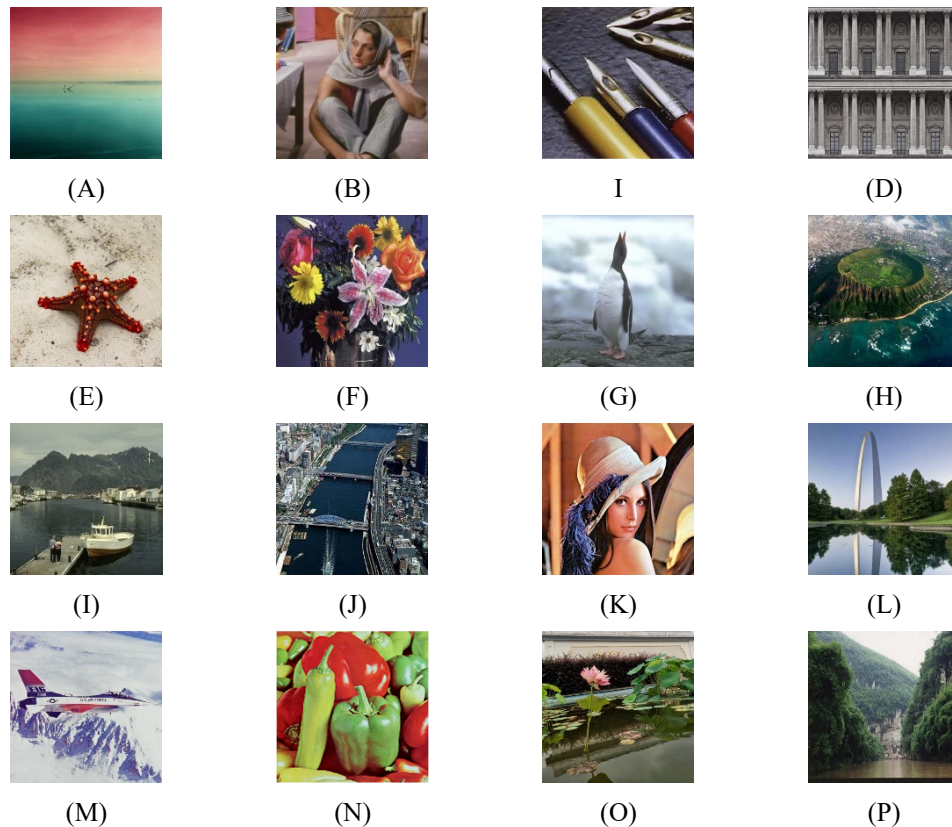


Figure 1. The 16 original images, i.e., A, B, ..., P, used in our experiment are all in size of 500×500 .

In our experimental, 50% entries of the images (A~P) in Figure 1 are randomly selected as missing entries, i.e., their values are initialized to zero, to generate incomplete images ($A_{\Omega} \sim P_{\Omega}$) at first. Then, the TLNM method is tested using these images and its performance is compared with those of other five methods. The peak signal-to-noise ratio (PSNR) is chosen as the standard for evaluating convergence accuracy and is given as that

$$PSNR = 10 \cdot \log_{10} \left(\frac{255^2}{MSE} \right), \quad (38)$$

$$MSE = \frac{1}{mn} \|X_{opt} - I\|_F^2, \quad (39)$$

where X_{opt} is an output of a matrix completion method, I is matrix without missing values.

Let $r = 300$ and $\rho = 1.02$, the TLNM method is tested on the 16 incomplete images, i.e., $A_{\Omega} \sim P_{\Omega}$. The PSNR values of TLNM, TNNR, MBF, LMaFit, SCPN, and FNNM are listed in Table 2.

Table 2. The PSNR values of TLNM using images containing 50% randomly missing entries.

Image	MBF	LMaFit	SCPN	FNNM	TNNR	TLNM
A_{Ω}	44.12	39.91	41.17	<u>45.52</u>	45.42	45.86
B_{Ω}	33.58	33.42	33.49	34.42	<u>34.93</u>	35.17
C_{Ω}	31.46	31.08	32.16	31.48	<u>31.88</u>	31.85
D_{Ω}	30.85	30.70	31.32	32.34	32.69	<u>32.35</u>
E_{Ω}	24.59	24.43	24.44	25.14	<u>25.54</u>	25.68
F_{Ω}	25.63	25.07	<u>26.27</u>	25.95	26.26	26.35
G_{Ω}	33.08	32.31	33.53	34.46	<u>35.21</u>	35.36
H_{Ω}	25.90	24.66	<u>26.63</u>	26.54	26.51	26.65
I_{Ω}	29.58	28.73	30.69	30.27	<u>31.19</u>	31.25
J_{Ω}	20.77	19.28	21.12	21.33	<u>21.30</u>	21.17
K_{Ω}	24.93	23.82	25.14	25.12	<u>25.25</u>	25.35
L_{Ω}	31.38	30.52	32.02	31.84	<u>32.25</u>	32.50
M_{Ω}	26.15	25.36	27.10	26.32	<u>27.24</u>	27.58
N_{Ω}	27.82	27.35	28.35	28.06	<u>28.48</u>	28.91
O_{Ω}	24.87	23.85	24.61	25.11	<u>25.51</u>	25.73
P_{Ω}	28.19	27.87	30.42	<u>30.79</u>	30.84	<u>30.79</u>

The PSNR values listed in Table 2 show that our proposed TLNM is much more accurate than the compared methods. The MBF method is not as accurate as the SCPN and TNNR methods, because the former cannot adaptively regulate the thresholds for the singular values of tested images. The LMaFit method is less accurate than the MBF method, because it cannot obtain the singular values of incomplete images via its updating steps. The PSNR values of TNNR on the 16 incomplete images are much better than those of MBF and LMaFit, because the truncated nuclear norm is a much better approximation of rank function than the nuclear norm. The SCPN method is a bit more accurate than the FNNM method on most tested incomplete images. The main reason is that the Schatten capped p norm used in SCPN can be seen as a special case of weighted nuclear norm that assigns different weights for different singular values, while FNNM is a nuclear norm minimization method. The PSNR values of TLNM on the tested images are much more accurate than the TNNR method and the recently proposed FNNM and SCPN. Hence, the proposed truncated $L_{2,1}$ norm should be a better relaxation of rank function than truncated nuclear norm.

After the comparison of convergence accuracy, it is necessary to compare the speed of TLNM with those of TNNR, SCPN, FNNM, MBF, and LMaFit. The CPU times for these six methods to reach the PSNR values shown in Table 2 are plotted in Figure 2.

The CPU time curves plotted in Figure 2 show that TLNM is obviously faster than the TNNR, SCPN, FNNM, and MBF methods, because TLNM can recover the incomplete images by QR and the computation cost of QR is 7 percent of that of SVD [32]. FNNM is the slowest one among the six tested methods, because its CPU time is much larger than those of other methods. More exactly, it takes about 110~180 seconds to search for the optimal solutions, while other methods take less than 80 seconds. The TNNR method takes about 40~71 seconds to converge on the 16 tested images, which indicates that it converges faster than the FNNM method does. Compared with FNNM and TNNR, the CPU time of SCPN is much smaller, which leads to a fact that the SCPN method can converge with fewer number of iterations. The main reason is that these three methods are almost equal in terms of

computational cost. The MBF method is a bit faster than the SCPN method, because it can improve its speed by performing SVD decomposition on a small size matrix. The CPU time of LMaFit is almost equal to TLNM, because neither of these two methods require the usage of SVD.

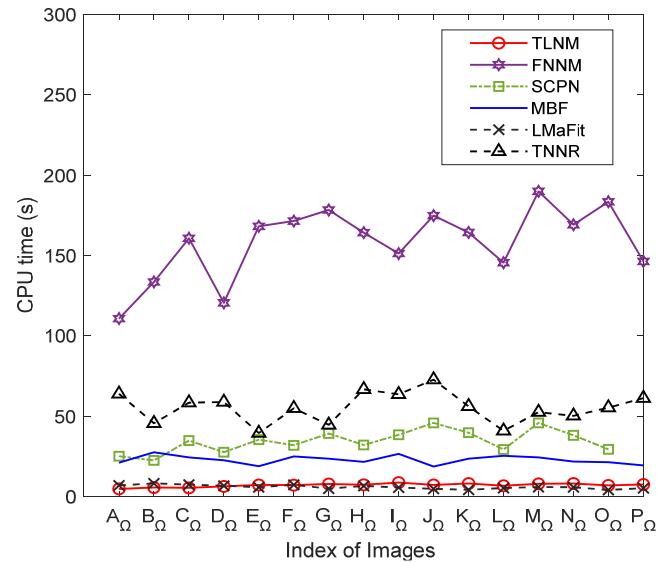


Figure 2. The CPU times of TLNM and other state-of-art methods using the 16 incomplete images with 50% missing values randomly distributed.

In view of the PSNR values in Table 2 and the CPU time curves plotted in Figure 2, it is easy to see that TLNM outperforms its competing methods, i.e., TNNR, SCPN, FNNM, LMaFit, and MBF, in both convergence accuracy and CPU time.

To intuitively show the details of output of TLNM and other state-of-art methods, some of their outputs are shown in Figure 3.

From Figure 3, it is easy to see that the recovered images shown in (A4) and (B4) given by LMaFit are respectively not as clear as the results shown in (A8) and (B8) given by TLNM, because there're many noises remained in the former. The recovery results of MBF shown in (A3) and (B3) are much better than the results given by LMaFit, however they contain a small bit of noisy pixels. The SCPN method performs better on the incomplete image shown in (B2) than on the image shown in (A2), because its output shown in (B5) is much better than the result shown in (A5). The recovered images given by TLNM, FNNM and TNNR, which are almost as clear as the original images shown in (A1) and (B1), are much better than the results given by LMaFit, MBF, and SCPN.

4.2. Experimental results on images with structured missing values

Eliminating structural missing textures that occur in an image is a more difficult work than restoring the images containing random missing values, because the pixels covered by texts are distributed continuously.

In this section, the 16 color images shown in Figure 1 are arranged with some textures to generate the incomplete images for testing our proposed TLNM method. In Figure 4, an example of incomplete image with texture missing entries is plotted for show.

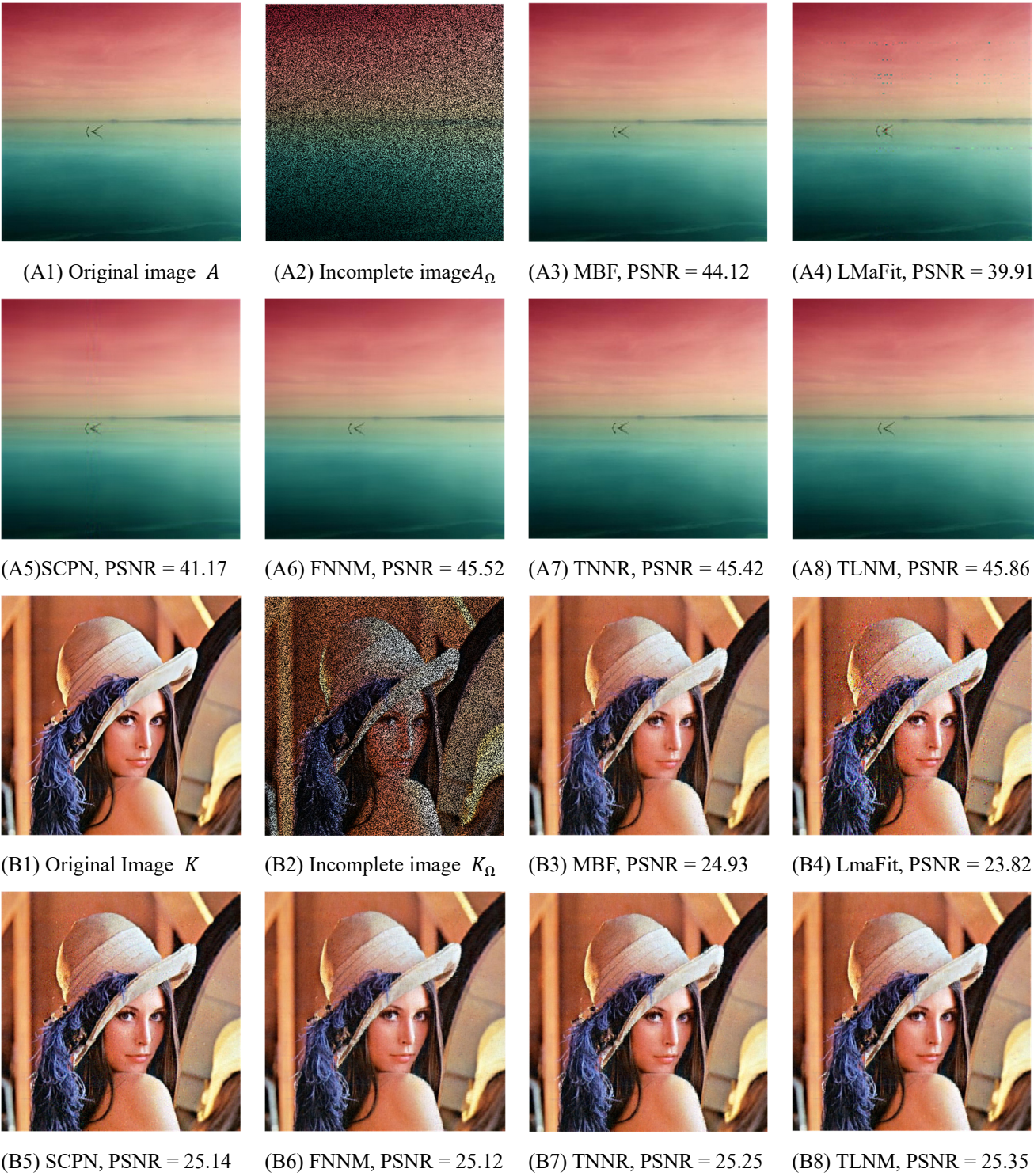


Figure 3. The recovered images given by MBF, LmaFit, SCPN, FNNM, TNNR, and TLNM. The CPU times of these methods are shown in Figure 2. One-half of the entries of the images shown in (A2) and (B2) are randomly missing.



Figure 4. An example of incomplete image with textured missing entries.

All the 16 color images shown in Figure 1 will be masked by the same textures as in Figure 4 for generating the incomplete images in this section. The final PSNR values and CPU times of TLNM on the 16 incomplete images with textures are compared with those of other state-of-art methods.

Table 3. The PSNR values of TLNM, TNNR, MBF, SCPN, FNNM, and LMaFit on the incomplete images with textured missing values.

Image	MBF	LMaFit	SCPN	FNNM	TNNR	TLNM
A_{Ω}	39.12	14.95	38.40	38.81	<u>39.31</u>	39.33
B_{Ω}	24.10	21.85	25.25	24.15	<u>25.78</u>	25.93
C_{Ω}	21.33	19.88	24.28	23.86	23.78	<u>24.03</u>
D_{Ω}	29.51	27.81	29.71	29.66	30.42	<u>30.25</u>
E_{Ω}	19.84	18.67	20.77	20.08	<u>20.19</u>	20.29
F_{Ω}	18.23	16.51	19.09	18.31	18.56	<u>18.51</u>
G_{Ω}	25.31	23.68	25.95	25.33	<u>26.33</u>	27.02
H_{Ω}	21.87	19.93	22.21	22.14	<u>22.23</u>	22.56
I_{Ω}	24.39	21.94	24.58	24.28	<u>24.62</u>	24.71
J_{Ω}	18.34	16.72	18.38	<u>18.57</u>	18.27	18.66
K_{Ω}	17.41	16.31	17.87	17.35	18.11	<u>17.94</u>
L_{Ω}	23.81	19.47	24.57	23.75	25.25	24.86
M_{Ω}	19.37	16.21	20.57	19.95	20.18	<u>20.19</u>
N_{Ω}	21.19	15.49	22.35	21.10	<u>22.19</u>	22.11
O_{Ω}	20.11	19.49	<u>20.63</u>	20.05	20.77	20.51
P_{Ω}	23.46	21.06	24.36	23.39	<u>24.55</u>	24.57

First, the PSNR values of TLNM and other five methods are listed for comparison in Table 3. Although removing the textures is a more difficult task than that in Section 4. 1, our proposed TLNM method has a much better performance in such case than the performances of LmaFit, MBF, SCPN,

FNNM, and TNNR methods. The PSNR values of LMaFit on the 16 tested images are much smaller than those of MBF. Hence, the MBF method is more capable of handling the matrix completion tasks with structured missing values. Compared with the MBF method, the recently proposed SCPN method is much more accurate, because the latter can regulate the weights for singular values adaptively. Since the FNNM method is a nuclear norm minimization method, its convergence accuracies are smaller than those of SCPN on the tested images. The proposed TLNM method is much more accurate than the TNNR method on most tested images, because the PSNR values of TLNM are better than those of TNNR. In view of the PSNR values in Tables 2 and 3, it is easy to see that TLNM can recover the incomplete images with random missing entries and textured missing entries much more accurately than the five compared methods on most tested images.

Second, the CPU times of TLNM on images containing textures are plotted in Figure 5.

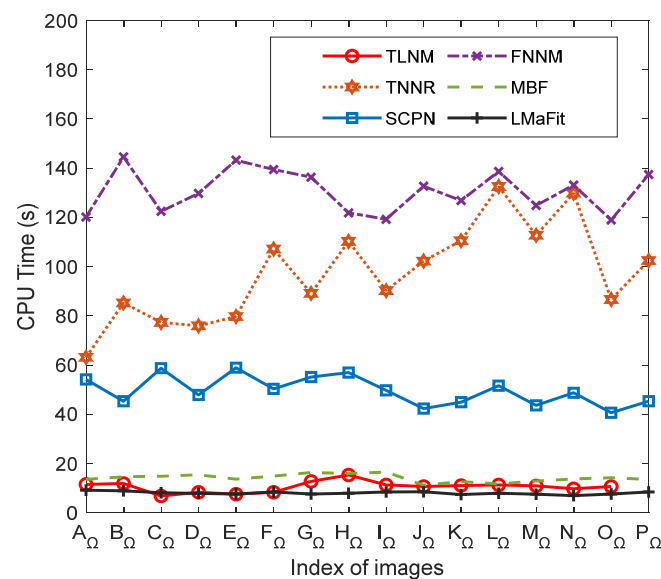


Figure 5. The CPU times of TLNM, TNNR, MBF, LMaFit, SCPN, and FNNM on images containing textures.

The curves in Figure 5 show that the speed of TLNM is equal to that of LMaFit and is much smaller than those of TNNR, SCPN and FNNM. The TLNM method takes about 10 seconds to recover the textured missing entries for each image, while the TNNR method takes about 65~130 seconds. The SCPN methods should have a smaller number of iterations than TNNR and FNNM, its CPU times on the 16 images are much smaller than those of the latter. The FNNM method is the slowest one among the six tested matrix completion methods, because it takes about 120~150 seconds to converge.

By comparing the CPU times shown in Figure 5, we see that TLNM should be 5 times as fast as SCPN, 10 times as fast as TNNR and 13 times as fast as FNNM, respectively.

Third, some outputs given by TLNM and other methods are displayed in Figure 6.

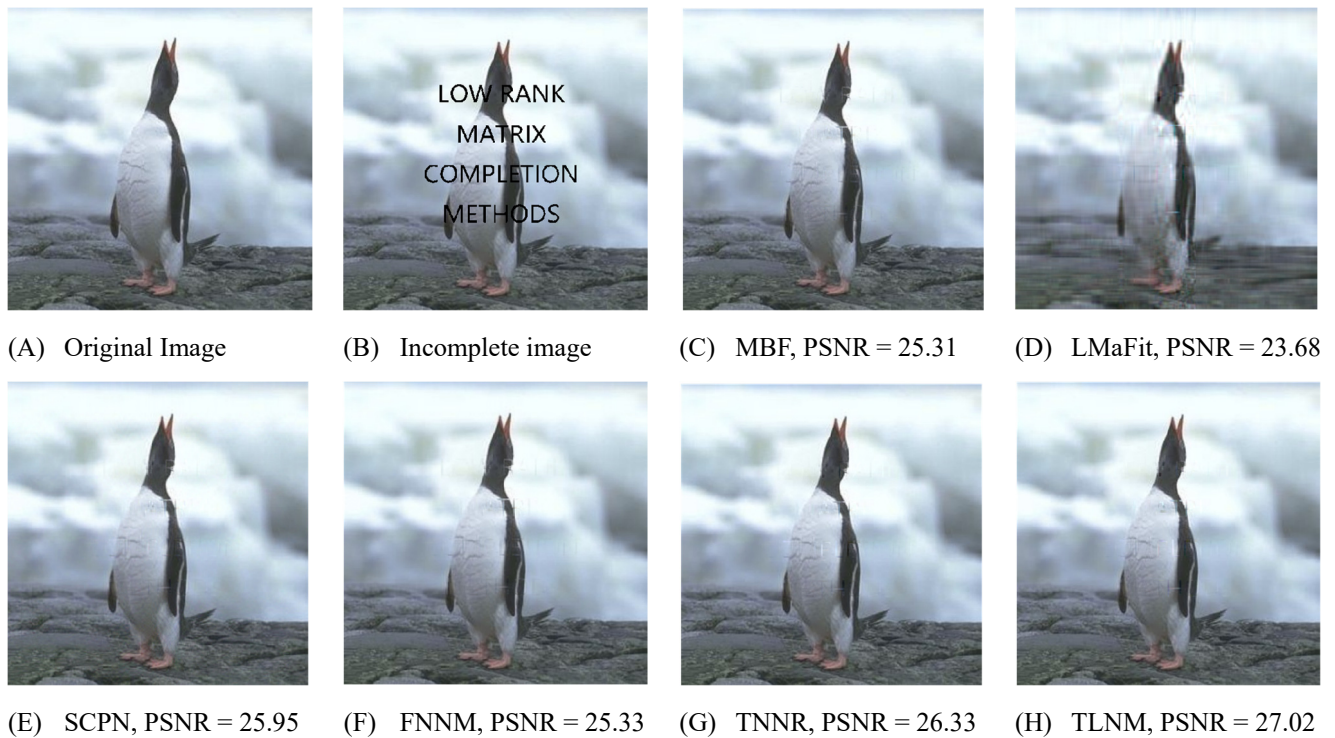


Figure 6. The recovered images of TLNM and other state-of-art methods. (A) is an original image. (B) is an incomplete image. (C), (D), ..., and (H) are the outputs of MBF, LMaFit, SCPN, FNNM, TNNR and TLNM, respectively.

Figure 6 indicates that the result given by LMaFit is not clear, because the recovered image in (D) has many noisy pixels. The result given by MBF shown in (C) is much better than that given by LMaFit. However, there are very noticeable text traces appeared in (C). The result of SCPN shown in (E) is much clearer than those of FNNM, MBF, and LMaFit. Hence, this result reaffirms a fact that the weighted nuclear norm minimization method should have a better convergence accuracy than a method based on nuclear norm minimization in most cases. The TNNR method can recover the images masked by textures successfully, because its PSNR value is much better than those of the SCPN, FNNM, LMaFit, and MBF methods. Moreover, the result given by TNNR is much better than the results given by SCPN and FNNM. However, there are some inconspicuous traces of texture remained in the recovered image given by TNNR. The TLNM method is much more accurate than the compared methods, because its PSNR, i.e., 27.02, is much better than those of the latter. The recovery result given by TLNM shown in (H) is clearer than the results of other methods, because there is almost no noises distributed in image (H).

Finally, the detailed convergence processes of TLNM and other five compared methods, i.e., the PSNR curves, are plotted in Figure 7.

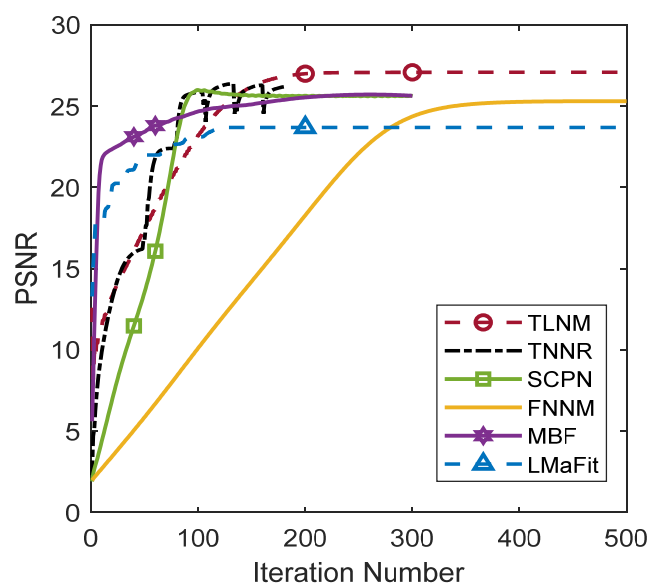


Figure 7. The curves of PSNR versus iteration number for TLNM and other five compared methods on the incomplete image shown in Figure 6(B).

From Figure 7, it is easy to see that the FNNM takes about 400 iterations to converge, which is the slowest method among the six tested methods. The SCPN and TNNR take about 100 and 180 iterations to converge, respectively. The iteration number of MBF, TLNM, and LMaFit are almost equal to 200. Although the gap in the number of iterations between the TLNM and TNNR methods is not large, their CPU times vary greatly. The main reason is that the computation cost of TLNM is smaller than that of TNNR. In view of the CPU times shown in Fig. 5 and the PSNR curves in Figure 7, it is not difficult to see that the CPU time of each iteration for TLNM is about ten percent of that of TNNR, which means that the main computation costs of TNNR and TLNM are mainly consumed by QR decomposition and SVD decomposition, respectively. Hence, the methods based on QR, i.e., TLNM, LmaFit, and MBF, are faster than the TNNR, SCPN, and FNNM methods using SVD.

5. Conclusions

In this paper, a truncated $L_{2,1}$ norm minimization method, which is abbreviated as TLNM, is proposed for improving the speed and convergence accuracy of TNNR. On one side, for improving the convergence accuracy of TNNR, a novel truncated $L_{2,1}$ norm is designed as a better relaxation of rank function than the truncated nuclear norm in the proposed TLNM. Second, for improving the speed of TNNR, the optimization problem of TLNM is solved using QR decomposition to update the key variables. Experimental results using color images show that TLNM is more accurate than the fast methods based on matrix factorization and other state-of-art methods, such as LmaFit, MBF, SCPN, FNNM, and TNNR. Since the speed of QR is about 7 times as fast as that of SVD, the proposed TLNM method is faster than the compared state-of-art methods using SVD, i.e., SCPN, TNNR, and FNNM.

Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62102002, in part by the Natural Science Foundation of Anhui Province of China under Grant 2008085QF291, by the Research Start-up Fund of West Anhui University, No. WGKQ2021053, in part by the transverse project of designing and processing of gas gun driven by high pressure air mixed with Gas (0045021079).

Conflict of interest

The authors declare that there are no conflicts of interest.

References

1. J. Miao, K. I. Kou, Color image recovery using low-rank quaternion matrix completion algorithm, *IEEE Trans. Image Process.*, **31** (2022), 190–201. <https://doi.org/10.1109/TIP.2021.3128321>
2. Z. Jia, Q. Jin, M. K. Ng, X. Zhao, Non-local robust quaternion matrix completion for large-scale color image and video inpainting, *IEEE Trans. Image Process.*, **31** (2022), 3868–3883. <https://doi.org/10.1109/TIP.2022.3176133>
3. X. Li, H. Zhang, R. Zhang, Matrix completion via non-convex relaxation and adaptive correlation learning, *IEEE Trans. Pattern Anal. Mach. Intell.*, **45** (2023), 1981–1991. <https://doi.org/10.1109/TPAMI.2022.3157083>
4. H. Cai, L. Huang, P. Li, D. Needell, Matrix completion with cross-concentrated sampling: Bridging uniform sampling and CUR sampling, *IEEE Trans. Pattern Anal. Mach. Intell.*, **45** (2023), 10100–10113. <https://doi.org/10.1109/TPAMI.2023.3261185>
5. S. Bhattacharya, S. Chatterjee, Matrix completion with data-dependent missingness probabilities, *IEEE Trans. Inf. Theory*, **68** (2022), 6762–6773. <https://doi.org/10.1109/TIT.2022.3170244>
6. Y. Yang, C. Ma, Optimal tuning-free convex relaxation for noisy matrix completion, *IEEE Trans. Inf. Theory*, **69** (2023), 6571–6585. <https://doi.org/10.1109/TIT.2023.3284341>
7. K. F. Masood, J. Tong, J. Xi, J. Yuan, Y. Yu, Inductive matrix completion and root-MUSIC-based channel estimation for intelligent reflecting surface (IRS)-aided hybrid MIMO systems, *IEEE Trans. Wireless Commun.*, **22** (2023), 7917–7931. <https://doi.org/10.1109/TWC.2023.3257138>
8. O. Elnahas, Y. Ma, Y. Jiang, Z. Quan, Clock synchronization in wireless networks using matrix completion-based maximum likelihood estimation, *IEEE Trans. Wireless Commun.*, **19** (2020), 8220–8231. <https://doi.org/10.1109/TWC.2020.3020191>
9. Q. Liu, X. Li, H. Cao, Two-dimensional localization: Low-rank matrix completion with random sampling in massive MIMO system, *IEEE Syst. J.*, **15** (2020), 3628–3631. <https://doi.org/10.1109/JSYST.2020.3012775>

10. Z. Liu, X. Li, H. C. So, ℓ_0 -norm minimization based robust matrix completion approach for MIMO radar target localization, *IEEE Trans. Aerosp. Electronic Syst.*, **59** (2023), 6759–6770. <https://doi.org/10.1109/TAES.2023.3280462>
11. M. Fazel, *Matrix Rank Minimization with Applications*, Ph.D thesis, Stanford University, 2002.
12. E. J. Candès, B. Recht, Exact matrix completion via convex optimization, *Found. Comput. Math.*, **9** (2009), 717–772. <https://doi.org/10.1007/s10208-009-9045-5>
13. J. Cai, E. Candès, Z. Shen, A singular value thresholding method for matrix completion, *SIAM J. Optim.*, **20** (2010), 1956–1982. <https://doi.org/10.1137/080738970>
14. S. Ma, D. Goldfarb, L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, *Math. Program.*, **128** (2011), 321–353. <https://doi.org/10.1007/s10107-009-0306-5>
15. K. C. Toh, S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems, *Pacific J. Optim.*, **6** (2010), 615–640.
16. R. Meka, P. Jain, I. Dhillon, Guaranteed rank minimization via singular value projection, in *Proceedings of Advances in Neural Information Processing Systems*, (2009), 937–945.
17. 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** (2013), 2117–2130. <https://doi.org/10.1109/TPAMI.2012.271>
18. Y. Liu, L. Jiao, F. Shang, F. Yin, F. Liu, An efficient matrix bi-factorization alternative optimization method for low-rank matrix recovery and completion, *Neural Networks*, **48** (2013), 8–18. <https://doi.org/10.1016/j.neunet.2013.06.013>
19. Y. Liu, L. Jiao, F. Shang, A fast tri-factorization method for low-rank matrix recovery and completion, *Pattern Recognit.*, **46** (2013), 163–173. <https://doi.org/10.1016/j.patcog.2012.07.003>
20. Z. Wen, W. Yin, Y. Zhang, Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation, *Math. Program. Comput.*, **4** (2012), 333–361. <https://doi.org/10.1007/s12532-012-0044-1>
21. S. Gu, L. Zhang, W. Zuo, X. Feng, Weighted nuclear norm minimization with application to image denoising, in *Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition*, (2014).
22. C. Wen, W. Qian, Q. Zhang, F. Cao, Algorithms of matrix recovery based on truncated Schatten p -norm, *Int. J. Mach. Learn. Cybern.*, **12** (2021), 1557–1570. <https://doi.org/10.1007/s13042-020-01256-7>
23. G. Li, G. Guo, S. Peng, C. Wang, S. Yu, J. Niu, et al., Matrix completion via Schatten capped p norm, *IEEE Trans. Knowl. Data Eng.*, **34** (2022), 394–404. <https://doi.org/10.1109/TKDE.2020.2978465>
24. Q. Liu, Z. Lai, Z. Zhou, F. Kuang, Z. Jin, A truncated nuclear norm regularization method based on weighted residual error for matrix completion, *IEEE Trans. Image Process.*, **25** (2016), 316–330. <https://doi.org/10.1109/TIP.2015.2503238>
25. W. Zeng, H. C. So, Outlier-robust matrix completion via ℓ_p -minimization, *IEEE Trans. Signal Process.*, **66** (2018), 1125–1140. <https://doi.org/10.1109/TSP.2017.2784361>
26. 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. <https://doi.org/10.1109/LSP.2020.2988596>
27. M. Muma, W. Zeng, A. M. Zoubir, Robust M-estimation based matrix completion, in *ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (2019), 5476–5480. <https://doi.org/10.1109/ICASSP.2019.8682657>

28. Y. He, F. Wang, Y. Li, J. Qin, B. Chen, Robust matrix completion via maximum correntropy criterion and half-quadratic optimization, *IEEE Trans. Signal Process.*, **68** (2019), 181–195. <https://doi.org/10.1109/TSP.2019.2952057>
29. Z. Shi, X. P. Li, W. Li, T. Yan, J. Wang, Y. Fu, Robust low-rank matrix recovery as mixed integer programming via ℓ_0 -norm optimization, *IEEE Signal Process. Lett.*, **30** (2023), 1012–1016. <https://doi.org/10.1109/LSP.2023.3301244>
30. Q. Zhao, D. Meng, Z. Xu, W. Zuo, Y. Yan, L1-norm low rank matrix factorization by variational Bayesian method, *IEEE Trans. Neural Network Learn. Syst.*, **26** (2015), 825–839. <https://doi.org/10.1109/TNNLS.2014.2387376>
31. X. P. Li, Z. Shi, Q. Liu, H. C. So, Fast robust matrix completion via entry-wise ℓ_2 -norm minimization, *IEEE Trans. Cybern.*, **53** (2023), 7199–7212. <https://doi.org/10.1109/TCYB.2022.3224070>
32. Q. Liu, F. Davoine, J. Yang, Y. Cui, Z. Jin, F. Han, A fast and accurate matrix completion method based on QR decomposition and L2,1-norm minimization, *IEEE Trans. Neural Networks Learn. Syst.*, **30** (2019), 803–817. <https://doi.org/10.1109/TNNLS.2018.2851957>
33. Q. Liu, C. Peng, P. Yang, X. Zhou, Z. Liu, A fast matrix completion method based on matrix bifactorization and QR decomposition, *Wireless Commun. Mobile Comput.*, **2023** (2023), 2117876. <https://doi.org/10.1155/2023/2117876>
34. M. Yang, Y. Li, J. Wang, Feature and nuclear norm minimization for matrix completion, *IEEE Trans. Knowl. Data Eng.*, **34** (2022), 2190–2199. <https://doi.org/10.1109/TKDE.2020.3005978>
35. Q. Fan, Y. Liu, T. Yang, H. Peng, Fast and accurate spectrum estimation via virtual co-array interpolation based on truncated nuclear norm regularization, *IEEE Signal Process. Lett.*, **29** (2021), 169–173. <https://doi.org/10.1109/LSP.2021.3130018>
36. G. Morison, SURE based truncated tensor nuclear norm regularization for low rank tensor completion, in *Proceedings of the 28th European Signal Processing Conference (EUSIPCO) in 2020*, (2021), 2001–2005. <https://doi.org/10.23919/Eusipco47968.2020.9287726>
37. J. Dong, Z. Xue, J. Guan, Z. F. Han, W. Wang, Low rank matrix completion using truncated nuclear norm and sparse regularizer, *Signal Process. Image Commun.*, **68** (2018), 76–87. <https://doi.org/10.1016/j.image.2018.06.007>
38. M. Zhang, M. Zhang, F. Zhao, F. Zhang, Y. Liu, A. Evans, Truncated weighted nuclear norm regularization and sparsity for image denoising, in *Proceedings of 2023 IEEE International Conference on Image Processing (ICIP)*, (2023), 1825–1829. <https://doi.org/10.1109/ICIP49359.2023.10221971>
39. Y. Song, J. Li, X. Chen, D. Zhang, Q. Tang, K. Yang, An efficient tensor completion method via truncated nuclear norm, *J. Visual Commun. Image Representation*, **70** (2020), 102791. <https://doi.org/10.1016/j.jvcir.2020.102791>
40. Y. Qiu, G. Zhou, J. Zeng, Q. Zhao, S. Xie, Imbalanced low-rank tensor completion via latent matrix factorization, *Neural Networks*, **155** (2022), 369. <https://doi.org/10.1016/j.neunet.2022.08.023>
41. Z. Hu, F. Nie, R. Wang, X. Li, Low rank regularization: A review, *Neural Networks*, **136** (2021), 218. <https://doi.org/10.1016/j.neunet.2020.09.021>
42. Q. Shen, S. Yi, Y. Liang, Y. Chen, W. Liu, Bilateral fast low-rank representation with equivalent transformation for subspace clustering, *IEEE Trans. Multimedia*, **25** (2023), 6371–6383. <https://doi.org/10.1109/TMM.2022.3207922>

43. Y. Yang, J. Zhang, S. Song, C. Zhang, D. Liu, Low-rank and sparse matrix decomposition with orthogonal subspace projection-based background suppression for hyperspectral anomaly detection, *IEEE Geosci. Remote Sens. Lett.*, **17** (2020), 1378–1382. <https://doi.org/10.1109/LGRS.2019.2948675>
44. X. Wang, X. P. Li, H. C. So, Truncated quadratic norm minimization for bilinear factorization-based matrix completion, *Signal Process.*, **214** (2024), 109219. <https://doi.org/10.1016/j.sigpro.2023.109219>
45. J. Wen, H. He, Z. He, F. Zhu, A pseudo-inverse-based hard thresholding algorithm for sparse signal recovery, *IEEE Trans. Intell. Transp. Syst.*, **24** (2023), 7621–7630. <https://doi.org/10.1109/TITS.2022.3172868>



AIMS Press

©2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)