Computational Optimization and Applications
https://doi.org/10.1007/s10589-022-00439-y

T-product factorization based method for matrix and tensor completion problems

Quan Yu[1] · **Xinzhen Zhang**[2]

Received: 17 May 2022 / Accepted: 14 November 2022

Abstract
Low rank matrix and tensor completion problems are to recover the incomplete two and higher order data of low rank structures. The essential problem in the matrix and tensor completion problems is how to improve the efficiency. For a matrix completion problem, we establish a relationship between matrix rank and tensor tubal rank, and reformulate matrix completion problem as a third order tensor completion problem. For the reformulated tensor completion problem, we adopt a two-stage strategy based on tensor factorization algorithm. In this way, a matrix completion problem of big size can be solved via some matrix computations of smaller sizes. For a third order tensor completion problem, to fully exploit the low rank structures, we introduce the double tubal rank which combines the tubal rank of two tensors, original tensor and the reshaped tensor of the mode-3 unfolding matrix of original tensor. Based on this, we propose a reweighted tensor factorization algorithm for third order tensor completion. Extensive numerical experiments demonstrate that the proposed methods outperform state-of-the-art methods in terms of both accuracy and running time.

Keywords Matrix completion · Tensor completion · Tensor factorization · Tubal rank

✉ Xinzhen Zhang
xzzhang@tju.edu.cn

Quan Yu
quanyu@hnu.edu.cn

[1] School of Mathematics, Hunan University, Changsha 410082, People's Republic of China

[2] School of Mathematics, Tianjin University, Tianjin 300354, People's Republic of China

Published online: 30 November 2022

1 Introduction

Matrix and tensor completion have received much attention in recent years, which have many applications, such as in hyperspectral data recovery [6], internet traffic data completion [19], image/video inpainting [8, 20, 35, 46–48], image classification [4, 23] and high dynamic range (HDR) imaging [17, 28, 37]. In general, such matrix and tensor data have low rank structures. Hence the problems are modeled as the rank minimization problems. Unfortunately, the rank minimization problem is NP-hard in general due to the combinational nature of the function rank$(\cdot)$ even for matrix rank.

Nuclear norm is known to be the tightest convex relaxation of matrix rank function [31]. Hence the matrix completion problem is relaxed as a nuclear norm or related norm minimization with various efficient numerical methods [5, 7, 18, 21, 24, 26, 36]. But these methods require computing matrix singular value decomposition (SVD), which become increasingly expensive with the increasing sizes of the underlying matrices. To cut down the computational cost, low rank matrix factorization methods have been proposed in [9, 13, 32, 49, 54]. However, matrix factorization methods also need expensive computation for large scale matrix data.

As a higher order generalization of matrix completion, tensor completion has attracted much more attention recently [1, 2, 12, 48, 51]. Compared to matrix rank, there are various definitions for tensor rank, including CANDECOMP/PARAFAC (CP) rank [11], Tucker rank [38], tensor train (TT) rank [29], triple rank [30] and tubal rank [14]. Since it is generally NP-hard to compute the CP rank [10], it is hard to apply CP rank to the tensor completion problem. Although the TT rank can be computed by TT singular value decomposition, it always has a fixed pattern, which might not be the optimum for specific data tensor [53]. The Tucker rank is defined on the rank of unfolding matrices, which are of big sizes. On the other hand, unfolding a tensor as a matrix would destroy the original multi-way structure of the data, leading to vital information loss and degrading performance efficiency [25, 27, 48]. Recently, tubal rank becomes more and more popular since the low tubal rank tensor completion can be solved via updating matrices of smaller sizes at each iteration [57]. However, tubal rank is defined on the third mode, which ignores the low rank structures on the other two modes. To exploit the low rank structures, Zhang et al. [55, 56] proposed 3-tubal rank and tensor fibered rank, respectively, which considered the three modes at the same time. Though this type of rank reveals more low rank structures of the tensor, the low rank structures they considered overlapped (see Lemma 8), so redundant running time is generated.

Based on these analyses, in this paper, we first propose a novel model for low rank matrix completion problem. For a large scale matrix, we reshape it as a third order tensor. Then we establish a relationship between matrix rank and tubal rank of the reshaped tensor. Based on this relationship, we reformulate a matrix completion problem as a third order tensor completion problem. Then we propose a two-stage tensor factorization based algorithm to the reformulated tensor

completion problem. By this way, a matrix completion problem of big size can be dealt with by computing matrix factorization of smaller sizes, which drastically cuts down the consumed time.

For the tensor completion problem, we consider the tubal rank and the mode-3 unfolding matrix rank together for fully exploiting the low rank structures of the tensor. For the mode-3 unfolding matrix, we adopt the tubal rank of the reshaped tensor to measure. Thus, we introduce a new tensor rank, named double tubal rank. See the definition of tensor double tubal rank in (19) for details. Based on these, a reweighted tensor factorization algorithm is proposed for the tensor completion based on double tubal rank.

In summary, our main contributions include:

(1) For a matrix completion problem, we reformulate it as a third order tensor completion problem. Then we propose a tensor factorization based algorithm to solve it. In this way, a big matrix completion problem can be solved by computing some smaller matrices, which greatly improves the efficiency of matrix completion problem.
(2) For a third order tensor completion problem, we introduce the tensor double tubal rank. Compared with tubal rank, 3-tubal rank [55] and tensor fibered rank [56], double tubal rank can fully exploit the low rank structures without redundancy. Based on the introduced double tubal rank, we propose a reweighted tensor factorization algorithm.
(3) In the proposed algorithms, we adopt the two-stage strategy, in which a good initial point is generated in the first stage and the convergence is accelerated in the second stage.
(4) The proposed algorithms converge to KKT points. Extensive numerical experiments demonstrate the outperformance of our proposed algorithms over the other compared algorithms.

The outline of this paper is given as follows. We recall the basic notations on tensor in Sect. 2. Sections 3 and 4 deal with the low rank matrix completion problem and the low rank tensor completion problem, respectively. Extensive simulation results are reported to demonstrate the validity of our proposed algorithms in Sects. 5 and 6 concludes this paper.

2 Notations and preliminaries

This section recalls some basic knowledge on tensors. We first give the basic notations and then present the tubal rank, 3-tubal rank (tensor fibered rank), and Tucker rank. We state them here in detail for the readers' convenience.

2.1 Notations

For a positive integer n, $[n] := \{1, 2, \ldots, n\}$. Scalars, vectors and matrices are denoted as lowercase letters $(a, b, c, \ldots)$, boldface lowercase letters $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \ldots)$ and uppercase letters $(A, B, C, \ldots)$, respectively. Third order tensors are denoted as calligraphic letters $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \ldots)$. For a third order tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we denote its (i, j, k) -th element as $\mathcal{A}_{ijk}$ or $\mathcal{A}(i, j, k)$ without confusion. Furthermore, we use the notation $\mathcal{A}(:, :, k)$ to denote its k-th frontal slice, denoted by $A^{(k)}$ for all $k \in [n_3]$. The inner product of two tensors $\mathcal{A}$, $\mathcal{B} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is the sum of products of their entries, i.e.

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} \mathcal{A}_{ijk} \mathcal{B}_{ijk}.$$

The Frobenius norm is $\|\mathcal{A}\|_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$. For a matrix A, A^H and $A^\dagger$ represent the conjugate transpose and the pseudo-inverse of A, respectively.

2.2 T-product, tubal rank and 3-tubal rank (tensor fibered rank)

Discrete Fourier Transformation (DFT) plays a key role in tensor-tensor product (t-product). For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, let $\bar{\mathcal{A}} \in \mathbb{C}^{n_1 \times n_2 \times n_3}$ be the result of DFT of $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ along the 3rd dimension. Specifically, let $F_{n_3} = [\boldsymbol{f_1}, \ldots, \boldsymbol{f_{n_3}}] \in \mathbb{C}^{n_3 \times n_3}$ be a DFT matrix, where

$$\boldsymbol{f_i} = \left[1; \omega^{(i-1)}; \ldots; \omega^{k(i-1)}; \ldots; \omega^{(n_3-1)(i-1)}\right] \in \mathbb{C}^{n_3}$$

with $\omega = e^{-\frac{2\pi \mathfrak{b}}{n_3}}$ and $\mathfrak{b} = \sqrt{-1}$. Then $\bar{\mathcal{A}}(i, j, :) = F_{n_3} \mathcal{A}(i, j, :)$, which can be computed by Matlab command "$\bar{\mathcal{A}} = \mathit{fft}(\mathcal{A}, [\,], 3)$". Furthermore, $\mathcal{A}$ can be computed by $\bar{\mathcal{A}}$ with the inverse DFT $\mathcal{A} = \mathit{ifft}(\bar{\mathcal{A}}, [\,], 3)$.

Lemma 1 [33] *Given any real vector $\boldsymbol{v} \in \mathbb{R}^{n_3}$, the associated vector $\bar{\boldsymbol{v}} = F_{n_3} \boldsymbol{v} \in \mathbb{C}^{n_3}$ satisfies*

$$\bar{v}_1 \in \mathbb{R} \text{ and } \operatorname{conj}\left(\bar{v}_i\right) = \bar{v}_{n_3 - i + 2}, \ i = 2, \ldots, \left\lfloor \frac{n_3 + 1}{2} \right\rfloor.$$

By Lemma 1, the frontal slices of $\bar{\mathcal{A}}$ have the following properties:

$$\begin{cases} \bar{A}^{(1)} \in \mathbb{R}^{n_1 \times n_2}, \\ \operatorname{conj}\left(\bar{A}^{(i)}\right) = \bar{A}^{(n_3 - i + 2)}, \ i = 2, \ldots, \left\lfloor \frac{n_3+1}{2} \right\rfloor. \end{cases} \tag{1}$$

For $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we define matrix $\bar{A} \in \mathbb{C}^{n_1 n_3 \times n_2 n_3}$ as

$$\bar{A} = bdiag(\bar{\mathcal{A}}) = \begin{bmatrix} \bar{A}^{(1)} & & & \\ & \bar{A}^{(2)} & & \\ & & \ddots & \\ & & & \bar{A}^{(n_3)} \end{bmatrix}. \tag{2}$$

Here, $bdiag(\cdot)$ is an operator which maps the tensor $\bar{\mathcal{A}}$ to the block diagonal matrix $\bar{A}$. The block circulant matrix $bcirc(\mathcal{A}) \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$ of $\mathcal{A}$ is defined as

$$bcirc(\mathcal{A}) = \begin{bmatrix} A^{(1)} & A^{(n_3)} & \cdots & A^{(2)} \\ A^{(2)} & A^{(1)} & \cdots & A^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ A^{(n_3)} & A^{(n_3-1)} & \cdots & A^{(1)} \end{bmatrix}.$$

Based on these notations, the T-product is presented as follows.

Definition 1 (*T-product*) [15] For $\mathcal{A} \in \mathbb{R}^{n_1 \times r \times n_3}$ and $\mathcal{B} \in \mathbb{R}^{r \times n_2 \times n_3}$, define

$$\mathcal{A} * \mathcal{B} := fold(bcirc(\mathcal{A}) \cdot unfold(\mathcal{B})) \in \mathbb{R}^{n_1 \times n_2 \times n_3}.$$

Here

$$unfold(\mathcal{B}) = \left[B^{(1)}; B^{(2)}; \ldots; B^{(n_3)}\right],$$

and its inverse operator "fold" is defined by

$$fold(unfold(\mathcal{B})) = \mathcal{B}.$$

Tensor multi-rank and tubal rank are now introduced as follows.

Definition 2 (*Tensor multi-rank and tubal rank*) [14] For a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, let $r_k = \text{rank}\left(\bar{A}^{(k)}\right)$ for all $k \in [n_3]$ with $A^{(k)}$ being the k-th frontal slice of $\bar{\mathcal{A}}$. Then multi-rank of $\mathcal{A}$ is defined as $\text{rank}_m(\mathcal{A}) = (r_1, \ldots, r_{n_3})$. The tensor tubal rank is defined as $\text{rank}_t(\mathcal{A}) = \max\left\{r_k | k \in [n_3]\right\}$.

Then, we introduce 3-tubal rank (tensor fibered rank).

Definition 3 (*3-tubal rank/tensor fibered rank*) [55, 56] For a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, let $\mathcal{A}_{(13)} \in \mathbb{R}^{n_1 \times n_3 \times n_2}$ and $\mathcal{A}_{(23)} \in \mathbb{R}^{n_2 \times n_3 \times n_1}$ with $\mathcal{A}_{ijk} = (\mathcal{A}_{(13)})_{ikj} = (\mathcal{A}_{(23)})_{jki}$. Then its tensor fibered rank is defined as

$$\text{rank}_f(\mathcal{A}) = \left(\text{rank}_t(\mathcal{A}), \text{rank}_t\left(\mathcal{A}_{(13)}\right), \text{rank}_t\left(\mathcal{A}_{(23)}\right)\right).$$

Finally, we presented lemmas that will be utilized to simplify models and do theoretical analysis.

Lemma 2 [15] *Suppose that $\mathcal{A}$, $\mathcal{B}$ are two third order tensors such that $\mathcal{F} := \mathcal{A} * \mathcal{B}$ is well defined as in Definition 1. Let $\bar{A}, \bar{B}, \bar{F}$ be the block diagonal matrices defined as in (2). Then*

(1) $\|\mathcal{A}\|_F^2 = \frac{1}{n_3}\|\bar{A}\|_F^2$;
(2) $\mathcal{F} = \mathcal{A} * \mathcal{B}$ *and* $\bar{F} = \bar{A}\bar{B}$ *are equivalent.*

Lemma 3 [57] *Suppose that* $\mathcal{A} \in \mathbb{R}^{n_1 \times r \times n_3}$ *and* $\mathcal{B} \in \mathbb{R}^{r \times n_2 \times n_3}$, *and* $\mathcal{F} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ *are three tensors. Then*

(1) *if* $\text{rank}_t(\mathcal{F}) = \hat{r}$, *then* $\mathcal{F}$ *can be written into a tensor product form* $\mathcal{F} = \mathcal{G} * \mathcal{H}$, *where* $\mathcal{G} \in \mathbb{R}^{n_1 \times \hat{r} \times n_3}$ *and* $\mathcal{H} \in \mathbb{R}^{\hat{r} \times n_2 \times n_3}$ *are two tensors of smaller sizes and they meet* $\text{rank}_t(\mathcal{G}) = \text{rank}_t(\mathcal{H}) = \hat{r}$;
(2) $\text{rank}_t(\mathcal{A} * \mathcal{B}) \leq \min\left\{\text{rank}_t(\mathcal{A}), \text{rank}_t(\mathcal{B})\right\}$.

2.3 Tucker rank

In this subsection, we are ready to present some notations on Tucker rank decomposition. More details can be found in Kolda and Bader's review on tensor decompositions [16].

The mode-s unfolding $A_{(s)}$ of tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is a matrix in $\mathbb{R}^{n_s \times N_s}$ with its (i, j)-th element being $\mathcal{A}_{i_1 \ldots i_{s-1} i i_{s+1} \ldots i_3}$, where $j = 1 + \sum_{k \neq s}(i_k - 1)\bar{n}_k$, $\bar{n}_k = \prod_{l<k, l \neq s} n_l$ and $N_s = \prod_{k \neq s} n_k$. The unfolding matrix can be obtained by "tens2mat$(\mathcal{A}, s)$" in Matlab. The opposite operation "fold$_s$" is defined as $\text{fold}_s(A_{(s)}) := \mathcal{A}$.

Based on the definition of mode-s unfolding matrix, the Tucker rank of tensor is defined as follows.

Definition 4 For a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, let $A_{(i)} \in \mathbb{R}^{n_i \times N_i}$ be the mode-i unfolding matrix. The Tucker rank of $\mathcal{A}$ is

$$\text{rank}_{tc}(\mathcal{A}) = \left(\text{rank}(A_{(1)}), \text{rank}(A_{(2)}), \text{rank}(A_{(3)})\right).$$

Next, we recall the definition of k-mode product.

Definition 5 For a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times \ldots \times n_k \times \ldots \times n_m}$ and a matrix $B \in \mathbb{R}^{J_k \times n_k}$, the mode-$k$ product of $\mathcal{A}$ with B is a tensor of $n_1 \times \ldots \times n_{k-1} \times J_k \times n_{k+1} \times \ldots \times n_m$ with its entries

$$(\mathcal{A} \times_k B)_{i_1 \ldots i_m} = \sum_{j_k=1}^{n_k} \mathcal{A}_{i_1 \ldots i_{k-1} j_k i_{k+1} \ldots i_m} B_{i_k j_k}.$$

Easy to find that, for suitable matrices B^1 and B^2, it holds

$$\mathcal{T} \times_i B^1 \times_i B^2 = \mathcal{T} \times_i \left(B^2 B^1\right).$$

Based on these notations, we are ready to present an equivalent definition of Tucker decomposition of tensor as follows.

Definition 6 Suppose that

$$\mathcal{A} = \mathcal{G} \times_1 U^1 \times_2 U^2 \times_3 U^3, \tag{3}$$

where $\mathcal{G} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$, orthogonal matrix $U^i \in \mathbb{R}^{n_i \times r_i}$ and $r_i = \operatorname{rank}\left(A_{(i)}\right)$ for all $i \in [3]$. Such $\mathcal{G}$ is called a core tensor and (3) is called a Tucker rank decomposition of $\mathcal{A}$.

3 A tensor-product factorization based method for matrix completion

Given a partially observed matrix $M \in \mathbb{R}^{n_1 \times h}$, low rank matrix completion problem can be formulated as a constrained rank minimization problem, that is,

$$\min_{X \in \mathbb{R}^{n_1 \times h}} \operatorname{rank}(X), \quad \text{s.t.} \quad P_{\tilde{\Omega}}(X - M) = 0, \tag{4}$$

where $\tilde{\Omega}$ is an index subset of observed entries of matrix, $P_{\tilde{\Omega}}(\cdot)$ is a projection operator that keeps the entries of matrix in $\tilde{\Omega}$ and makes other entries zero. When n_1 and h are very large, the required cost to recover matrix X will be very expensive. To lower the cost, we reshape the matrix as a third order tensor as follows. For a given integer n_2, we add a zero matrix $0 \in \mathbb{R}^{n_1 \times l}$ in X with the smallest l such that $X := [X, 0] \in \mathbb{R}^{n_1 \times (h+l)}$ and $n_3 := (h + l)/n_2$ is an integer. Therefore, we reshape the matrix $X \in \mathbb{R}^{n_1 \times h}$ as a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ such that

$$X^{(k)} = X\big(:, (k-1)n_2 + 1 : kn_2\big),\ k \in [n_3]. \tag{5}$$

See Fig. 1 for clearness.

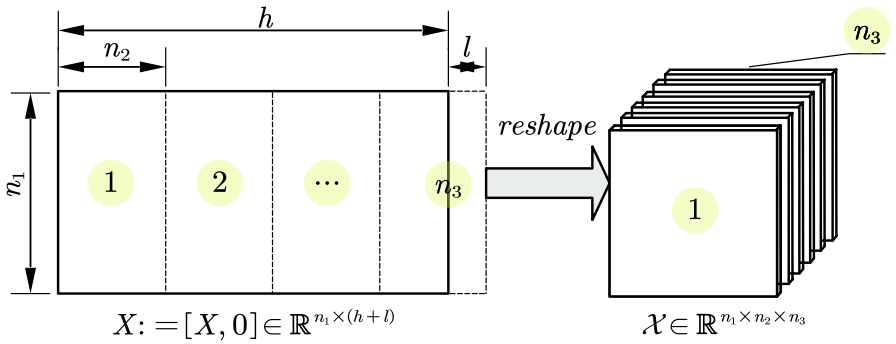


Fig. 1 Reshaping the matrix X into the tensor $\mathcal{X}$

Now we are ready to establish a relationship between $\text{rank}(X)$ and $\text{rank}_t(\mathcal{X})$. For this aim, we need the following results.

Lemma 4 *Suppose that* $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ *and* $\bar{\mathcal{A}} = fft(\mathcal{A}, [\,], 3)$, *then* $\text{rank}\big(\bar{A}_{(1)}\big) = \text{rank}\big(A_{(1)}\big)$.

Proof By $\bar{\mathcal{A}} = fft(\mathcal{A}, [\,], 3)$, we have $\bar{\mathcal{A}} = \mathcal{A} \times_3 F_{n_3}$. Let $\mathcal{A} = \mathcal{G} \times_1 U^1 \times_2 U^2 \times_3 U^3$ be a Tucker rank decomposition. Then

$$\bar{\mathcal{A}} = \mathcal{A} \times_3 F_{n_3} = \mathcal{G} \times_1 U^1 \times_2 U^2 \times_3 \big(F_{n_3} U^3\big),$$

which leads to $\text{rank}\big(\bar{A}_{(1)}\big) \le \text{rank}\big(U^1\big) = \text{rank}\big(A_{(1)}\big)$. Similarly, with $\mathcal{A} = \bar{\mathcal{A}} \times_3 F_{n_3}^{-1}$, there holds

$$\text{rank}\big(A_{(1)}\big) \le \text{rank}\big(\bar{A}_{(1)}\big).$$

In conclusion, the lemma is established now. □

Lemma 5 *Suppose that matrix* $X \in \mathbb{R}^{n_1 \times h}$ *and tensor* $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ *obtained by reshaping matrix X with* (5). *Then*

$$\begin{aligned} &\text{rank}_t(\mathcal{X}) \le \text{rank}(X) \le n_3 \text{rank}_t(\mathcal{X}), \\ &\text{rank}(X) \le \big\|\text{rank}_m(\mathcal{X})\big\|_1 \le n_3 \text{rank}(X). \end{aligned} \tag{6}$$

Proof Let $\bar{\mathcal{X}} = fft(\mathcal{X}, [\,], 3)$, then

$$\text{rank}(X) = \text{rank}\big(X_{(1)}\big) = \text{rank}\big(\bar{X}_{(1)}\big) = \text{rank}\big(\big[\bar{X}^{(1)}, \bar{X}^{(2)}, \ldots, \bar{X}^{(n_3)}\big]\big), \tag{7}$$

where the first equality follows from the way of the reshaped tensor $\mathcal{X}$, the second equality is due to Lemma 4 and the third equality comes from $\bar{X}_{(1)} = \big[\bar{X}^{(1)}, \bar{X}^{(2)}, \ldots, \bar{X}^{(n_3)}\big]$. Observe that

$$\text{rank}\big(\big[\bar{X}^{(1)}, \bar{X}^{(2)}, \ldots, \bar{X}^{(n_3)}\big]\big) \le \sum_{k=1}^{n_3} \text{rank}\big(\bar{X}^{(k)}\big) \le n_3 \text{rank}_t(\mathcal{X}) \tag{8}$$

and

$$\text{rank}\big(\big[\bar{X}^{(1)}, \bar{X}^{(2)}, \ldots, \bar{X}^{(n_3)}\big]\big) \ge \max\big\{\text{rank}\big(\bar{X}^{(k)}\big) | k \in [n_3]\big\} = \text{rank}_t(\mathcal{X}). \tag{9}$$

By (7), (8) and (9), it follows

$$\text{rank}_t(\mathcal{X}) \le \text{rank}(X) \le n_3 \text{rank}_t(\mathcal{X}).$$

On the other hand, (7) and (9) mean that

$$n_3 \text{rank}(X) \ge n_3 \text{rank}_t(\mathcal{X}) \ge \sum_{k=1}^{n_3} \text{rank}\big(\bar{X}^{(k)}\big). \tag{10}$$

Together with (7) and (8), it holds

$$n_3 \text{rank}(X) \geq \sum_{k=1}^{n_3} \text{rank}\left(\bar{X}^{(k)}\right) = \left\|\text{rank}_m(\mathcal{X})\right\|_1 \geq \text{rank}(X).$$

□

Based on these analyses, we consider the following tensor completion problem for solving the matrix completion problem (4):

$$\min_{\mathcal{X}\in\mathbb{R}^{n_1\times n_2\times n_3}} \text{rank}_t(\mathcal{X}), \quad \text{s.t.} \quad P_{\Omega}(\mathcal{X}-\mathcal{M}) = 0, \tag{11}$$

where $\Omega = \left\{(i,j,k) : (i,j+n_2(k-1)) \in \tilde{\Omega},\ i \in [n_1],\ j \in [n_2],\ k \in [n_3]\right\}$ is an index subset of index set $\{(i,j,k) : i \in [n_1], j \in [n_2], k \in [n_3]\}$; $\mathcal{M} \in \mathbb{R}^{n_1\times n_2\times n_3}$ is a tensor reshaped by matrix M, by the same way of reshaped tensor $\mathcal{X}$.

According to Lemma 3, a tensor can be factorized as a product of two tensors of smaller sizes. Applying the low rank tensor approximation model (11) introduced in [57], we consider the following model to solve (11)

$$\min_{\mathcal{X},\mathcal{P},\mathcal{Q}} \frac{1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2, \quad \text{s.t.} \quad P_{\Omega}(\mathcal{X}-\mathcal{M}) = 0, \tag{12}$$

where $\mathcal{P} \in \mathbb{R}^{n_1\times r\times n_3}$, $\mathcal{Q} \in \mathbb{R}^{r\times n_2\times n_3}$ with r being the pre-estimated tubal rank of $\mathcal{X}$. Clearly, problem (12) can be regarded as a low rank approximation version of (11).

In the following, we present how to solve (12). We adopt the alternating minimization scheme to optimize (12). Update $\mathcal{X}$, for fixed tensors $\mathcal{P}$ and $\mathcal{Q}$, by

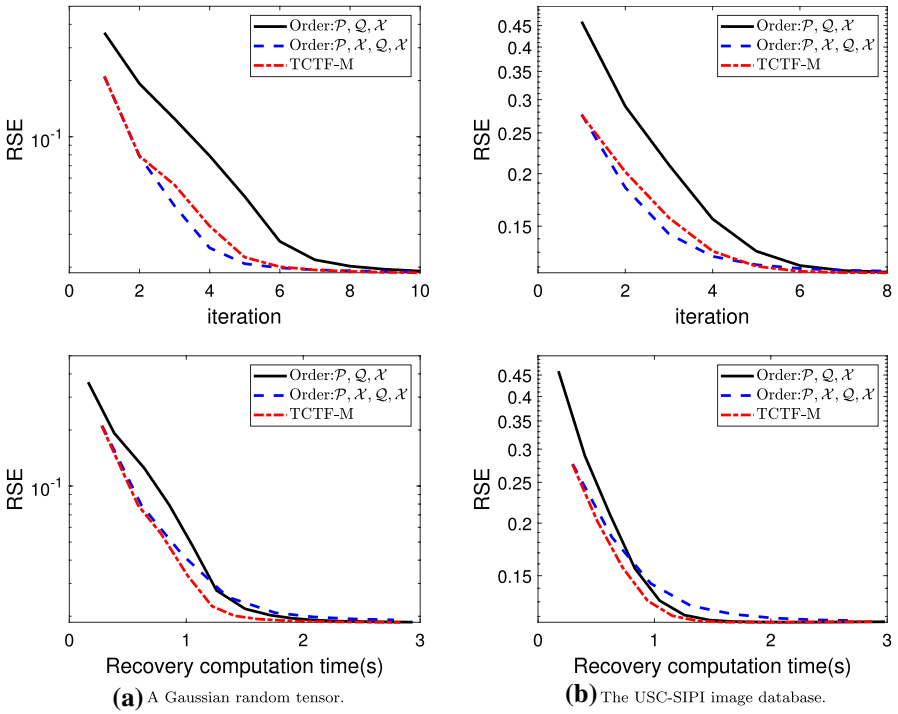
$$\mathcal{X} = \underset{P_{\Omega}(\mathcal{X}-\mathcal{M})=0}{\text{argmin}} \frac{1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2 = P_{\Omega^c}(\mathcal{P} * \mathcal{Q}) + P_{\Omega}(\mathcal{M}), \tag{13}$$

where Ω^c denotes the complement of the set Ω with respect to the set $\left\{(i,j,k) : i \in [n_1], j \in [n_2], k \in [n_3]\right\}$.

Now we update $\mathcal{P}$ and $\mathcal{Q}$ with a similar way of Algorithm TCTF proposed in Section 3 of [57]. For the ease of the reader, we present the details here. We rewrite (12) as a corresponding matrix version. Assume that $\text{rank}_m(\mathcal{X}) = (r_1, r_2, \ldots, r_{n_3})$ and $\text{rank}_t(\mathcal{X}) = \hat{r}$, where $r_k = \text{rank}\left(\bar{X}^{(k)}\right)$, $k \in [n_3]$ and $\hat{r} = \max\{r_1, \ldots, r_{n_3}\}$. For each k, $\bar{X}^{(k)}$ can be factorized as a product of two matrices $\hat{P}^{(k)}$ and $\hat{Q}^{(k)}$ with $\hat{P}^{(k)} \in \mathbb{C}^{n_1\times r_k}$ and $\hat{Q}^{(k)} \in \mathbb{C}^{r_k\times n_2}$ are the k-th block diagonal matrices of $\hat{P} \in \mathbb{C}^{n_1 n_3\times(\sum_{k=1}^{n_3} r_k)}$ and $\hat{Q} \in \mathbb{C}^{(\sum_{k=1}^{n_3} r_k)\times n_2 n_3}$. Let $\bar{P}^{(k)} = [\hat{P}^{(k)}, 0] \in \mathbb{C}^{n_1\times\hat{r}}$, $\bar{Q}^{(k)} = [\hat{Q}^{(k)}; 0] \in \mathbb{C}^{\hat{r}\times n_2}$ and $\bar{P}$, $\bar{Q}$ be the block diagonal matrices with the k-th block diagonal matrices $\bar{P}^{(k)}$ and $\bar{Q}^{(k)}$, respectively. Then $\hat{P}\hat{Q} = \bar{P}\bar{Q}$. Together with Lemma 2, it follows

$$\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2 = \frac{1}{n_3}\left\|\bar{P}\bar{Q} - \bar{X}\right\|_F^2 = \frac{1}{n_3}\left\|\hat{P}\hat{Q} - \bar{X}\right\|_F^2 = \frac{1}{n_3}\sum_{k=1}^{n_3}\left\|\hat{P}^{(k)}\hat{Q}^{(k)} - \bar{X}^{(k)}\right\|_F^2.$$

Therefore, (12) can be rewritten as



(a) A Gaussian random tensor. **(b)** The USC-SIPI image database.

Fig. 2 Results with three different orders

$$\min_{\mathcal{X},\mathcal{P},\mathcal{Q}} \frac{1}{2n_3}\sum_{k=1}^{n_3}\left\|\hat{P}^{(k)}\hat{Q}^{(k)}-\bar{X}^{(k)}\right\|_F^2, \quad \text{s.t.} \quad P_{\Omega}(\mathcal{X}-\mathcal{M})=0, \tag{14}$$

where $\hat{P}^{(k)}, \hat{Q}^{(k)}$ are defined as above. Combining with (1), we can update $\mathcal{P}$ and $\mathcal{Q}$ as follows:

$$\hat{P}^{(k)}=\begin{cases}\bar{X}^{(k)}\left(\hat{Q}^{(k)}\right)^H\left(\hat{Q}^{(k)}\left(\hat{Q}^{(k)}\right)^H\right)^{\dagger}, k=1,\ldots,\left\lceil\frac{n_3+1}{2}\right\rceil,\\ \text{conj}\left(\hat{P}^{(n_3-k+2)}\right), k=\left\lceil\frac{n_3+1}{2}\right\rceil+1,\ldots,n_3,\end{cases} \tag{15}$$

$$\hat{Q}^{(k)}=\begin{cases}\left(\left(\hat{P}^{(k)}\right)^H\hat{P}^{(k)}\right)^{\dagger}\left(\hat{P}^{(k)}\right)^H\bar{X}^{(k)}, k=1,\ldots,\left\lceil\frac{n_3+1}{2}\right\rceil,\\ \text{conj}\left(\hat{Q}^{(n_3-k+2)}\right), k=\left\lceil\frac{n_3+1}{2}\right\rceil+1,\ldots,n_3.\end{cases} \tag{16}$$

One can perform (15), (16) and (13) to update $\mathcal{P}$, $\mathcal{Q}$ and $\mathcal{X}$ in different manners. Directly applying the APG method proposed in [45] leads to the order of $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{X}$. However, since $\mathcal{X}$ interacts with $\mathcal{P}$ and $\mathcal{Q}$, updating it more frequently is expected to

speed up the convergence of the algorithm. Hence, a more efficient way would be to update the variables in the order of $\mathcal{P}, \mathcal{X}, \mathcal{Q}, \mathcal{X}$. The convergence behavior with two different updating orders on a synthetic tensor and the USC-SIPI image database[1] was shown in Fig. 2. As you can see from the figure, the final effect of the updating order $\mathcal{P}, \mathcal{Q}, \mathcal{X}$ is similar to that of $\mathcal{P}, \mathcal{X}, \mathcal{Q}, \mathcal{X}$. However, the former convergence speeds are much worse than the latter. Although the updated sequence $\{\mathcal{P}, \mathcal{X}, \mathcal{Q}, \mathcal{X}\}$ converges faster, it takes more iteration time for each step, and the reason for the faster convergence is due to the fact that the first few steps can produce a good value. For this reason, we adopt the two-stage strategy: updating order $\mathcal{P}, \mathcal{X}, \mathcal{Q}, \mathcal{X}$ in the first few steps, and $\mathcal{P}, \mathcal{Q}, \mathcal{X}$ in the subsequent steps. We denote this algorithm by TCTF-M. Similarly, we can see the convergence behavior of TCTF-M with the best performance.

For convenience, we outline TCTF-M as follows.

Algorithm 3.1 Matrix Completion Algorithm (TCTF-M)

Input: The matrix (tensor) data $M \in \mathbb{R}^{n_1 \times h}$ ($\mathcal{M} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$), the observed set $\tilde{\Omega}(\Omega)$ and a positive integer t_0.
Initialization: $\mathcal{X}^0$, $\mathcal{P}^0$, $\mathcal{Q}^0$ and the multi-rank $\boldsymbol{r}_{\mathcal{X}}^0 \in \mathbb{R}^{n_3}$.
While not converge do
1. Fix $\hat{Q}^t$ and $\mathcal{X}^t$ to update $\hat{P}^{t+1}$ by (15).
2. If $t \leq t_0$ then
   Fix $\hat{P}^{t+1}$ and $\hat{Q}^t$ to compute $\mathcal{X}^t$ by (13).
3. Fix $\hat{P}^{t+1}$ and $\mathcal{X}^t$ to update $\hat{Q}^{t+1}$ by (16).
4. Adopt the rank decreasing scheme to adjust $\boldsymbol{r}_{\mathcal{X}}^t$, adjust the sizes of $\hat{P}^{t+1}$, $\hat{Q}^{t+1}$.
5. Fix $\hat{P}^{t+1}$ and $\hat{Q}^{t+1}$ to compute $\mathcal{X}^{t+1}$ by (13).
6. Check the stop criterion: $\|\mathcal{X}^{t+1} - \mathcal{X}^t\|_F / \|\mathcal{X}^t\|_F < \varepsilon$.
7. $t \leftarrow t+1$.

end while
Output: $\mathcal{X}^{t+1}$.

Remark 1 In general, we do not know the true multi-tubal rank of optimal tensor $\mathcal{X}$ in advance. Thus, it is necessary to estimate the multi-rank of tensor $\mathcal{X}$. Similar to [40, 43, 57], we adopt the same rank estimation and rank decreasing strategy.

The strategy is described as follows. Suppose that the multi-rank of $\mathcal{X}$ is $\boldsymbol{r}^t = \left(r_1^t, r_2^t, \cdots, r_{n_3}^t\right)$ at the t-th iteration. We compute the eigenvalues λ_i of $\left(\hat{P}^{(k)}\right)^H \hat{P}^{(k)}$ with $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_{n_t}$, where $n_t = \sum_{k=1}^{n_3} r_k^t$. Compute $\hat{\lambda}_j = \lambda_j / \lambda_{j+1}$, $j \in [n_t - 1]$, $l^t = \arg\max_{1 \leqslant j \leqslant n_t - 1} \hat{\lambda}_j$ and $\tau^t = (l^t - 1)\hat{\lambda}_{l^t} / \sum_{j \neq l^t} \hat{\lambda}_j$. If $\tau^t \geqslant 10$ (a large drop in the magnitude of the eigenvalues), we will reduce $\boldsymbol{r}^t$. We find λ_{s^t} such that $\sum_{j=1}^{s^t} \lambda_j / \sum_{j=1}^{n_t} \lambda_j \geqslant 95\%$. Assume there are m_k^t eigenvalues of $\left(\hat{P}^{(k)}\right)^H \hat{P}^{(k)}$ which belong to $\left\{\lambda_{s^t+1}, \cdots, \lambda_{n_t}\right\}$. Then we set $r_k^t = r_k^t - m_k^t$. Suppose $G^{(k)} \Sigma^{(k)} \left(H^{(k)}\right)^H$ is the skinny SVD of $\hat{P}^{(k)} \hat{Q}^{(k)}$. We can update $\hat{P}^{(k)} = G_{r_k^t}^{(k)} \Sigma_{r_k^t}^{(k)}$, $\hat{Q}^{(k)} = \left(H_{r_k^t}^{(k)}\right)^H$, where $G_{r_k^t}^{(k)}$ consists of the first r_k^t columns of $G^{(k)}$ and $H_{r_k^t}^{(k)}$ consists of the first r_k^t rows of $H^{(k)}$. $\Sigma_{r_k^t}^{(k)}$ is a diagonal matrix whose diagonal entries are the largest r_k^t eigenvalues of $\Sigma^{(k)}$. In this way, we reduce $\boldsymbol{r}^t$ and pre-estimate the rank of the tensor data.

[1] http://sipi.usc.edu/database/.

Compared to TCTF, only half of matrices $\hat{P}^{(k)}$ and $\hat{Q}^{(k)}$ are calculated in (15) and (16), which cut the computational cost of $\mathcal{P}^{t+1}$ and $\mathcal{Q}^{t+1}$ when n_3 is large. When $t \geq t_0$, in each iteration, the complexity of TCTF-M is $\mathcal{O}\Big(r(n_1+n_2)n_3 \log n_3 + rn_1n_2\Big\lceil \frac{n_3+1}{2} \Big\rceil\Big)$, where $r = \text{rank}_t(\mathcal{X})$.

The convergence result of Algorithm 3.1 can be seen from [57], stated as follows.

Theorem 1 *Assume that* $g(\mathcal{P}, \mathcal{Q}, \mathcal{X}) = \frac{1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2$ *and the sequence* $\{\mathcal{P}^t, \mathcal{Q}^t, \mathcal{X}^t\}$ *generated by Algorithm* 3.1 *is bounded, then it satisfies the following properties*:

(1) $g^t := g(\mathcal{P}^t, \mathcal{Q}^t, \mathcal{X}^t)$ *is monotonically decreasing. Actually, it satisfies the following inequality*:

$$g^t - g^{t+1} \geq \frac{1}{2n_3}\left\|\hat{P}^{t+1}\hat{Q}^{t+1} - \hat{P}^t\hat{Q}^t\right\|_F^2 \geq 0,$$

where $\hat{P}, \hat{Q}$ are from $\mathcal{P}$ and $\mathcal{Q}$, respectively.

(2) *any accumulation point* $(\mathcal{P}_\star, \mathcal{Q}_\star, \mathcal{X}_\star)$ *of the sequence* $\{\mathcal{P}^t, \mathcal{Q}^t, \mathcal{X}^t\}$ *is a KKT point of problem* (12).

4 An improved tensor-product factorization based method for tensor completion

In this section, we first establish a relationship between tubal rank and Tucker rank of the third order tensor. According to such relationship, we modify the tubal rank to double tubal rank and then establish a novel low rank tensor completion model with the introduced double tubal rank.

4.1 Tensor completion model based on double tubal rank

From Lemma 5, the following result is direct.

Lemma 6 *For a tensor* $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, *it holds*

$$\text{rank}_t(\mathcal{X}) \leq \text{rank}\big(X_{(i)}\big) \leq n_3 \text{rank}_t(\mathcal{X}), \quad i \in [2]. \tag{17}$$

Compared to Tucker rank, tubal rank does not involve the low rank structure information of the mode-3 unfolding matrix from Lemma 6. Hence, we define an improved tensor rank as follows:

$$\text{rank}_{ttr}(\mathcal{X}) = \left(\text{rank}_t(\mathcal{X}), \text{rank}(X_{(3)})\right). \tag{18}$$

Based on Lemma 5, we change (18) into double tubal rank:

$$\text{rank}_{dt}(\mathcal{X}) = \left(\text{rank}_t(\mathcal{X}), \text{rank}_t(\tilde{\mathcal{X}})\right), \tag{19}$$

where $\tilde{\mathcal{X}} \in \mathbb{R}^{n_3 \times p \times q}$ ($pq = n_1 n_2$) is a tensor by reshaping the unfolding matrix $X_{(3)}$ satisfying (5) and hence $\tilde{X}_{(1)} = X_{(3)}$.

Next, we discuss the relationship between Tucker rank and double tubal rank.

Lemma 7 *Suppose that* $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ *and* $\text{rank}_{dt}(\mathcal{X})$ *is defined as in* (19). *Then*

$$\begin{aligned} \text{rank}_t(\mathcal{X}) &\le \text{rank}\left(X_{(i)}\right) \le n_3 \,\text{rank}_t(\mathcal{X}), \quad i \in [2], \\ \text{rank}_t(\tilde{\mathcal{X}}) &\le \text{rank}\left(X_{(3)}\right) \le n_3 \,\text{rank}_t(\tilde{\mathcal{X}}). \end{aligned}$$

Proof The result is immediate from Lemma 5 and Lemma 6. □

According to this lemma, the introduced double tubal rank can learn the global correlations within multi-dimensional data as well as the Tucker rank. In the next lemma, we prove a connection between double tubal rank and 3-tubal rank (tensor fibered rank).

Lemma 8 *For a tensor* $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, *we have*

$$\begin{aligned} \text{rank}_t(\tilde{\mathcal{X}})/n_2 &\le \text{rank}_t(\mathcal{X}_{(13)}) \le q\,\text{rank}_t(\tilde{\mathcal{X}}), \\ \text{rank}_t(\tilde{\mathcal{X}})/n_1 &\le \text{rank}_t(\mathcal{X}_{(23)}) \le q\,\text{rank}_t(\tilde{\mathcal{X}}). \end{aligned}$$

In particular, when $\tilde{\mathcal{X}} \in \mathbb{R}^{n_3 \times n_1 \times n_2}$, $\text{rank}_t(\tilde{\mathcal{X}}) = \text{rank}_t(\mathcal{X}_{(13)})$.

Proof By the definition of $\mathcal{X}_{(13)}$ and Lemma 6, we have

$$\text{rank}_t(\mathcal{X}_{(13)}) \le \text{rank}(X_{(3)}) \le n_2\,\text{rank}_t(\mathcal{X}_{(13)}), \quad \text{rank}_t(\tilde{\mathcal{X}}) \le \text{rank}(\tilde{X}_{(1)}) \le q\,\text{rank}_t(\tilde{\mathcal{X}}).$$

Together with $\tilde{X}_{(1)} = X_{(3)}$, one has

$$\text{rank}_t(\tilde{\mathcal{X}})/n_2 \le \text{rank}_t(\mathcal{X}_{(13)}) \le q\,\text{rank}_t(\tilde{\mathcal{X}}).$$

Similar to the analysis above, we obtain

$$\text{rank}_t(\tilde{\mathcal{X}})/n_1 \le \text{rank}_t(\mathcal{X}_{(23)}) \le q\,\text{rank}_t(\tilde{\mathcal{X}}).$$

□

Double tubal rank is a vector and its corresponding low rank tensor completion model will be a vector optimization problem. To keep things simple, we adopt the weighted rank $\gamma_1\,\text{rank}_t(\mathcal{X}) + \gamma_2\,\text{rank}_t(\tilde{\mathcal{X}})$ with a positive parameter γ_1, γ_2 as a measure of tensor rank. Hence the low rank tensor completion problem can be modeled as

$$\min_{\mathcal{X}} \ \gamma_1 \operatorname{rank}_t(\mathcal{X}) + \gamma_2 \operatorname{rank}_t(\tilde{\mathcal{X}}), \quad \text{s.t.} \quad P_{\Omega}(\mathcal{X} - \mathcal{M}) = 0. \tag{20}$$

Clearly, (20) reduces to the classical low tubal rank tensor completion model when $\gamma_1 = 1$ and $\gamma_2 = 0$.

Similar to (12), we consider the following tensor factorization model

$$\min_{\mathcal{X},\mathcal{P},\mathcal{Q},\mathcal{U},\mathcal{V}} \frac{\gamma_1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2 + \frac{\gamma_2}{2}\|\mathcal{U} * \mathcal{V} - \tilde{\mathcal{X}}\|_F^2, \quad \text{s.t.} \quad P_{\Omega}(\mathcal{X} - \mathcal{M}) = 0. \tag{21}$$

Motivated by the reweighted strategies [22, 41] and the supergradient concepts [3], problem (21) can be written as

$$\min_{\mathcal{X},\mathcal{P},\mathcal{Q},\mathcal{U},\mathcal{V}} \frac{1}{2}\rho\big(\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2\big) + \frac{1}{2}\rho\Big(\|\mathcal{U} * \mathcal{V} - \tilde{\mathcal{X}}\|_F^2\Big), \quad \text{s.t.} \quad P_{\Omega}(\mathcal{X} - \mathcal{M}) = 0. \tag{22}$$

Throughout this paper, Assumption 1 is assumed for $\rho(\cdot)$.

Assumption 1 The function $\rho(\cdot) : \mathbb{R}^+ \to \mathbb{R}^+$ is a proper, concave, lower semicontinuous function on $[0, +\infty)$, and there exist $a, b > 0$ such that $\partial\rho(t) \subset [a, b]$ for any $t \in [0, +\infty)$.

Remark 2 Since $\rho(\cdot)$ is concave on $[0, +\infty)$, by the definition of the supergradient, for any s and t, we have

$$\rho(t) \le \rho(s) + w_s(t - s), \ \forall\, w_s \in \partial\rho(s).$$

Now, we are ready to update $\mathcal{X}$, $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{U}$, $\mathcal{V}$. First of all, by Assumption 1, we can update $\mathcal{X}$ by

$$\begin{aligned}
\mathcal{X} &= \operatorname*{argmin}_{P_{\Omega}(\mathcal{X}-\mathcal{M})=0} \frac{\gamma_1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2 + \frac{\gamma_2}{2}\|\mathcal{U} * \mathcal{V} - \tilde{\mathcal{X}}\|_F^2 \\
&= \operatorname*{argmin}_{P_{\Omega}(\mathcal{X}-\mathcal{M})=0} \frac{\gamma_1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2 + \frac{\gamma_2}{2}\Big\|fold_3\big[(\mathcal{U} * \mathcal{V})_{(1)}\big] - \mathcal{X}\Big\|_F^2 \\
&= \frac{1}{\gamma_1 + \gamma_2} P_{\Omega^c}\big(\gamma_1 \mathcal{P} * \mathcal{Q} + \gamma_2 fold_3\big[(\mathcal{U} * \mathcal{V})_{(1)}\big]\big) + P_{\Omega}(\mathcal{M}).
\end{aligned} \tag{23}$$

After updating $\mathcal{X}$, we need to compute the weighting γ_1, γ_2 by

$$\gamma_1 \in \partial\rho\big(\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2\big), \quad \gamma_2 \in \partial\rho\Big(\|\mathcal{U} * \mathcal{V} - \tilde{\mathcal{X}}\|_F^2\Big). \tag{24}$$

Furthermore, since $\rho(\cdot)$ is a monotonically increasing function, $\mathcal{P}$ and $\mathcal{Q}$ can be updated by solving the following problem

$$\operatorname*{argmin}_{\mathcal{P},\,\mathcal{Q}} \frac{1}{2}\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2. \tag{25}$$

Clearly, $\mathcal{P}$ and $\mathcal{Q}$ can be updated by (15) and (16) respectively.

Similarly, we can update $\hat{U}$ and $\hat{V}$ as follows:

$$\hat{U}^{(k)} = \begin{cases} \bar{\hat{X}}^{(k)}\left(\hat{V}^{(k)}\right)^H\left(\hat{V}^{(k)}\left(\hat{V}^{(k)}\right)^H\right)^{\dagger}, k = 1, \ldots, \left\lceil \frac{q+1}{2} \right\rceil, \\ conj\left(\hat{U}^{(q-k+2)}\right), k = \left\lceil \frac{q+1}{2} \right\rceil + 1, \ldots, q, \end{cases} \tag{26}$$

$$\hat{V}^{(k)} = \begin{cases} \left(\left(\hat{U}^{(k)}\right)^H\hat{U}^{(k)}\right)^{\dagger}\left(\hat{U}^{(k)}\right)^H\bar{\hat{X}}^{(k)}, k = 1, \ldots, \left\lceil \frac{q+1}{2} \right\rceil, \\ conj\left(\hat{V}^{(q-k+2)}\right), k = \left\lceil \frac{q+1}{2} \right\rceil + 1, \ldots, q. \end{cases} \tag{27}$$

Based on above discussions, a tensor factorization based method can be outlined as Algorithm 4.1, denoted by DTRTC.

Algorithm 4.1 Double Tubal Rank Tensor Completion (DTRTC)

Input: The tensor data $\mathcal{M} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the observed set Ω, positive integer t_0 and a concave function $\rho(x)$.
Initialization: $\mathcal{X}^0$, $\mathcal{P}^0$, $\mathcal{Q}^0$, $\mathcal{U}^0$, $\mathcal{V}^0$. The initialized rank $\boldsymbol{r}_{\mathcal{X}}^0 \in \mathbb{R}^{n_3}$ and $\boldsymbol{r}_{\tilde{\mathcal{X}}}^0 \in \mathbb{R}^q$. Parameters γ_1^0, $\gamma_2^0 > 0$.
While not converge do
1. Fix $\hat{Q}^t$ and $\mathcal{X}^t$ to update $\hat{P}^{t+1}$ by (15).
2. If $t \le t_0$ then
Fix $\hat{P}^{t+1}$ and $\hat{Q}^t$ to compute $\mathcal{X}^t$ by (13).
3. Fix $\hat{P}^{t+1}$ and $\mathcal{X}^t$ to update $\hat{Q}^{t+1}$ by (16).
4. If $t \le t_0$ then
Fix $\hat{P}^{t+1}$ and $\hat{Q}^{t+1}$ to compute $\mathcal{X}^t$ by (13).
5. Fix $\hat{V}^t$ and $\mathcal{X}^t$ to update $\hat{U}^{t+1}$ by (26).
6. If $t \le t_0$ then
Fix $\hat{U}^{t+1}$ and $\hat{V}^t$ to compute $\mathcal{X}^t$ by (13).
7. Fix $\hat{U}^{t+1}$ and $\mathcal{X}^t$ to update $\hat{V}^{t+1}$ by (27).
8. Adopt the rank decreasing scheme to adjust $\boldsymbol{r}_{\mathcal{X}}^t$ and $\boldsymbol{r}_{\tilde{\mathcal{X}}}^t$, adjust the sizes of $\hat{P}^{t+1}$, $\hat{Q}^{t+1}$, $\hat{U}^{t+1}$ and $\hat{V}^{t+1}$.
9. Fix $\hat{P}^{t+1}$, $\hat{Q}^{t+1}$, $\hat{U}^{t+1}$, $\hat{V}^{t+1}$ to compute $\mathcal{X}^{t+1}$ by (23).
10. Compute γ_1^{t+1}, γ_2^{t+1} by (24).
11. Check the stop criterion: $\|\mathcal{X}^{t+1} - \mathcal{X}^t\|_F / \|\mathcal{X}^t\|_F < \varepsilon$.
12. $t \leftarrow t + 1$.
end while
Output: $\mathcal{X}^{t+1}$.

Remark 3 Similar to TCTF-M, it does not know the true multi-tubal rank of optimal tensor $\mathcal{X}$ and $\tilde{\mathcal{X}}$ in advance. Hence, we adopt the same rank estimation and rank decreasing strategy as in TCTF-M.

Complexity analysis: At each iteration, the cost of updating $\mathcal{P}$ and $\mathcal{Q}$ by (15) and (16) is $\mathcal{O}(\hat{r}_{\mathcal{X}}(n_1+n_2)n_3 \cdot \log n_3 + \hat{r}_{\mathcal{X}} n_1 n_2 \left\lceil \frac{n_3+1}{2} \right\rceil)$, respectively. The cost of updating $\mathcal{U}$ and $\mathcal{V}$ by (26) and (27) is $\mathcal{O}(\hat{r}_{\tilde{\mathcal{X}}}(n_3+p)q \log q + \hat{r}_{\tilde{\mathcal{X}}} n_3 p \left\lceil \frac{q+1}{2} \right\rceil)$, where $\hat{r}_{\mathcal{X}}$ and $\hat{r}_{\tilde{\mathcal{X}}}$ is the estimated tubal rank of $\mathcal{X}$ and $\tilde{\mathcal{X}}$, respectively. For updating $\mathcal{X}$ by (23), the computational cost for conducting the (inverse) DFT and matrix product is $\mathcal{O}(\hat{r}_{\mathcal{X}}(n_1+n_2)n_3 \log n_3 + \hat{r}_{\mathcal{X}} n_1 n_2 \left\lceil \frac{n_3+1}{2} \right\rceil + \hat{r}_{\tilde{\mathcal{X}}}(n_3+p)q \log q + \hat{r}_{\tilde{\mathcal{X}}} n_3 p \left\lceil \frac{q+1}{2} \right\rceil)$. In step 8, we use QR decomposition to estimate the target rank whose cost is $\mathcal{O}(\hat{r}_{\mathcal{X}}(n_1+n_2)n_3 \log n_3 + \hat{r}_{\mathcal{X}} n_1 n_2 \left\lceil \frac{n_3+1}{2} \right\rceil)$ and $\mathcal{O}(\hat{r}_{\tilde{\mathcal{X}}}(n_3+p)q \log q + \hat{r}_{\tilde{\mathcal{X}}} n_3 p \left\lceil \frac{q+1}{2} \right\rceil)$. In summary, the total cost at each iteration is $\mathcal{O}(\hat{r}_{\mathcal{X}}(n_1+n_2)n_3 \log n_3 + \hat{r}_{\tilde{\mathcal{X}}}(n_3+p)q \log q + \hat{r}_{\mathcal{X}} n_1 n_2 \left\lceil \frac{n_3+1}{2} \right\rceil + \hat{r}_{\tilde{\mathcal{X}}} n_3 p \left\lceil \frac{q+1}{2} \right\rceil)$.

4.2 Convergence analysis

In this subsection, we present the convergence of DTRTC. The following notation will be used in our analysis. In problem (21), Ω is an index set which locates the observed data. We use Ω^c to denote the complement of the set Ω with respect to the set $\{(i,j,k) : i \in [n_1], j \in [n_2], k \in [n_3]\}$. To simply the notation, we denote $z := (\mathcal{P}, \mathcal{Q}, \mathcal{U}, \mathcal{V}, \mathcal{X})$, $f(z) := \frac{1}{2}\rho\left(\|\mathcal{P} * \mathcal{Q} - \mathcal{X}\|_F^2\right) + \frac{1}{2}\rho\left(\left\|\mathcal{U} * \mathcal{V} - \tilde{\mathcal{X}}\right\|_F^2\right)$ and $f^t := f(z^t)$ in this subsection.

Theorem 2 *Assume that the sequence* $\left\{\mathcal{P}^t, \mathcal{Q}^t, \mathcal{U}^t, \mathcal{V}^t, \mathcal{X}^t\right\}$ *generated by Algorithm* 4.1 *is bounded, then it satisfies the following properties*:

(1) f^t *is monotonically decreasing. Actually, it satisfies the following inequality*:

$$f^t - f^{t+1} \geq \frac{\gamma_1^t}{2n_3}\left\|\hat{P}^{t+1}\hat{Q}^{t+1} - \hat{P}^t\hat{Q}^t\right\|_F^2 + \frac{\gamma_2^t}{2q}\left\|\hat{U}^{t+1}\hat{V}^{t+1} - \hat{U}^t\hat{V}^t\right\|_F^2 + a\left\|\mathcal{X}^{t+1} - \mathcal{X}^t\right\|_F^2 \geq 0,$$

where $\hat{P}, \hat{Q}, \hat{U}, \hat{V}$ are from $\mathcal{P}, \mathcal{Q}, \mathcal{U}$ and $\mathcal{V}$, respectively.

(2) *any accumulation point* $\left(\mathcal{P}_\star, \mathcal{Q}_\star, \mathcal{U}_\star, \mathcal{V}_\star, \mathcal{X}_\star\right)$ *of the sequence* $\left\{\mathcal{P}^t, \mathcal{Q}^t, \mathcal{U}^t, \mathcal{V}^t, \mathcal{X}^t\right\}$ *is a KKT point of problem* (21).

Proof According to f^t, we have that

$$\begin{aligned} f^t - f^{t+1} &= \frac{1}{2}\left(\rho\left(\|\mathcal{P}^t * \mathcal{Q}^t - \mathcal{X}^t\|_F^2\right) - \rho\left(\left\|\mathcal{P}^{t+1} * \mathcal{Q}^{t+1} - \mathcal{X}^{t+1}\right\|_F^2\right)\right) \\ &\quad + \frac{1}{2}\left(\rho\left(\left\|\mathcal{U}^t * \mathcal{V}^t - \tilde{\mathcal{X}}^t\right\|_F^2\right) - \rho\left(\left\|\mathcal{U}^{t+1} * \mathcal{V}^{t+1} - \tilde{\mathcal{X}}^{t+1}\right\|_F^2\right)\right) \\ &\geq \frac{\gamma_1^t}{2}\left(\|\mathcal{P}^t * \mathcal{Q}^t - \mathcal{X}^t\|_F^2 - \left\|\mathcal{P}^{t+1} * \mathcal{Q}^{t+1} - \mathcal{X}^{t+1}\right\|_F^2\right) \\ &\quad + \frac{\gamma_2^t}{2}\left(\left\|\mathcal{U}^t * \mathcal{V}^t - \tilde{\mathcal{X}}^t\right\|_F^2 - \left\|\mathcal{U}^{t+1} * \mathcal{V}^{t+1} - \tilde{\mathcal{X}}^{t+1}\right\|_F^2\right), \end{aligned} \tag{28}$$

where

$$\gamma_1^t \in \partial\rho\left(\|\mathcal{P}^t * \mathcal{Q}^t - \mathcal{X}^t\|_F^2\right), \quad \gamma_2^t \in \partial\rho\left(\left\|\mathcal{U}^t * \mathcal{V}^t - \tilde{\mathcal{X}}^t\right\|_F^2\right).$$

Since $\mathcal{X}^{t+1}$ is an optimal solution of $\mathcal{X}$-subproblem, we have

$$\begin{aligned}&\frac{\gamma_1^t}{2}\left\|\mathcal{P}^{t+1}*\mathcal{Q}^{t+1}-\mathcal{X}^{t+1}\right\|_F^2+\frac{\gamma_2^t}{2}\left\|\mathcal{U}^{t+1}*\mathcal{V}^{t+1}-\tilde{\mathcal{X}}^{t+1}\right\|_F^2+a\left\|\mathcal{X}^{t+1}-\mathcal{X}^t\right\|_F^2\\&\quad\le\frac{\gamma_1^t}{2}\left\|\mathcal{P}^{t+1}*\mathcal{Q}^{t+1}-\mathcal{X}^t\right\|_F^2+\frac{\gamma_2^t}{2}\left\|\mathcal{U}^{t+1}*\mathcal{V}^{t+1}-\tilde{\mathcal{X}}^t\right\|_F^2.\end{aligned}\tag{29}$$

According to the computation of $\mathcal{P}^{t+1}$, $\mathcal{Q}^{t+1}$ and Lemma 3 in [57], we have

$$\begin{aligned}&\|\mathcal{P}^t*\mathcal{Q}^t-\mathcal{X}^t\|_F^2-\left\|\mathcal{P}^{t+1}*\mathcal{Q}^{t+1}-\mathcal{X}^{t+1}\right\|_F^2\\&\quad=\left\|\mathcal{P}^{t+1}*\mathcal{Q}^{t+1}-\mathcal{X}^t\right\|_F^2-\left\|\mathcal{P}^{t+1}*\mathcal{Q}^{t+1}-\mathcal{X}^{t+1}\right\|_F^2+\frac{1}{n_3}\left\|\hat{P}^{t+1}\hat{Q}^{t+1}-\hat{P}^t\hat{Q}^t\right\|_F^2.\end{aligned}\tag{30}$$

Similar result can be obtained that

$$\begin{aligned}&\left\|\mathcal{U}^t*\mathcal{V}^t-\tilde{\mathcal{X}}^t\right\|_F^2-\left\|\mathcal{U}^{t+1}*\mathcal{V}^{t+1}-\tilde{\mathcal{X}}^{t+1}\right\|_F^2\\&\quad=\left\|\mathcal{U}^{t+1}*\mathcal{V}^{t+1}-\tilde{\mathcal{X}}^t\right\|_F^2-\left\|\mathcal{U}^{t+1}*\mathcal{V}^{t+1}-\tilde{\mathcal{X}}^{t+1}\right\|_F^2+\frac{1}{q}\left\|\hat{U}^{t+1}\hat{V}^{t+1}-\hat{U}^t\hat{V}^t\right\|_F^2.\end{aligned}\tag{31}$$

Combining (28)-(31), it holds

$$\begin{aligned}f^t-f^{t+1}\ge&\frac{\gamma_1^t}{2n_3}\left\|\hat{P}^{t+1}\hat{Q}^{t+1}-\hat{P}^t\hat{Q}^t\right\|_F^2+\frac{\gamma_2^t}{2q}\left\|\hat{U}^{t+1}\hat{V}^{t+1}-\hat{U}^t\hat{V}^t\right\|_F^2\\&+a\left\|\mathcal{X}^{t+1}-\mathcal{X}^t\right\|_F^2\ge0.\end{aligned}\tag{32}$$

Summing all the inequality (32) for all t, we obtain

$$\begin{aligned}f^1-f^{t+1}\ge&\frac{1}{2n_3}\sum_{t=1}^n\gamma_1^t\left\|\hat{P}^{t+1}\hat{Q}^{t+1}-\hat{P}^t\hat{Q}^t\right\|_F^2+\frac{1}{2q}\sum_{t=1}^n\gamma_2^t\left\|\hat{U}^{t+1}\hat{V}^{t+1}-\hat{U}^t\hat{V}^t\right\|_F^2\\&+a\sum_{t=1}^n\left\|\mathcal{X}^{t+1}-\mathcal{X}^t\right\|_F^2.\end{aligned}\tag{33}$$

Thus, there hold

$$\begin{aligned}&\lim_{t\to+\infty}\gamma_1^t\left\|\hat{P}^{t+1}\hat{Q}^{t+1}-\hat{P}^t\hat{Q}^t\right\|_F^2=0,\quad\lim_{t\to+\infty}\gamma_2^t\left\|\hat{U}^{t+1}\hat{V}^{t+1}-\hat{U}^t\hat{V}^t\right\|_F^2=0,\\&\lim_{t\to+\infty}\left\|\mathcal{X}^{t+1}-\mathcal{X}^t\right\|_F^2=0.\end{aligned}\tag{34}$$

Similar to the analysis of Equation (38)-(46) in [57] and $\gamma_i^t\in[a,b]$ $(i=1,2)$, ones have

$$\lim_{t\to+\infty}\gamma_1^t\left(\bar{X}^t-\hat{P}^t\hat{Q}^t\right)\left(\hat{Q}^t\right)^H=0,\quad\lim_{t\to+\infty}\gamma_1^t\left(\hat{P}^t\right)^H\left(\bar{X}^t-\hat{P}^t\hat{Q}^t\right)=0.$$

Since the sequence $\{\mathcal{P}^t, \mathcal{Q}^t, \mathcal{U}^t, \mathcal{V}^t, \mathcal{X}^t\}$ generated by Algorithm 4.1 is bounded, there is a subsequence $\{\mathcal{P}^{t_j}, \mathcal{Q}^{t_j}, \mathcal{U}^{t_j}, \mathcal{V}^{t_j}, \mathcal{X}^{t_j}\}$ that converges to a point $(\mathcal{P}_\star, \mathcal{Q}_\star, \mathcal{U}_\star, \mathcal{V}_\star, \mathcal{X}_\star)$. Therefore, the following two equations hold:

$$\gamma_1^\star(\bar{X}_\star - \hat{P}_\star\hat{Q}_\star)(\hat{Q}_\star)^H = 0, \quad \gamma_1^\star(\hat{P}_\star)^H(\bar{X}_\star - \hat{P}_\star\hat{Q}_\star) = 0. \tag{35}$$

Similarly, we have

$$\gamma_2^\star\left(\bar{\bar{X}}_\star - \hat{U}_\star\hat{V}_\star\right)(\hat{V}_\star)^H = 0, \quad \gamma_2^\star(\hat{U}_\star)^H\left(\bar{\bar{X}}_\star - \hat{U}_\star\hat{V}_\star\right) = 0. \tag{36}$$

On the other hand, we update $\mathcal{X}^{t+1} = \frac{1}{\gamma_1^t+\gamma_2^t}P_{\Omega^c}\left(\gamma_1^t\mathcal{P}^{t+1} * \mathcal{Q}^{t+1} + \gamma_2^t fold_3\left[(\mathcal{U}^{t+1} * \mathcal{V}^{t+1})_{(1)}\right]\right)$ $+P_\Omega(\mathcal{M})$ at each iteration. Thus, $\mathcal{X}_\star$ always satisfies the following two equations

$$P_{\Omega^c}\left(\mathcal{X}_\star - \frac{1}{\gamma_1^\star+\gamma_2^\star}\left(\gamma_1^\star\mathcal{P}_\star * \mathcal{Q}_\star + \gamma_2^\star fold_3\left[(\mathcal{U}_\star * \mathcal{V}_\star)_{(1)}\right]\right)\right) = 0, \quad P_\Omega(\mathcal{X}_\star - \mathcal{M}) = 0. \tag{37}$$

Furthermore, there exists $\Lambda_\star$ such that

$$P_\Omega\left(\mathcal{X}_\star - \frac{1}{\gamma_1^\star+\gamma_2^\star}\left(\gamma_1^\star\mathcal{P}_\star * \mathcal{Q}_\star + \gamma_2^\star fold_3\left[(\mathcal{U}_\star * \mathcal{V}_\star)_{(1)}\right]\right)\right) + \Lambda_\star = 0. \tag{38}$$

By (35)-(38), $(\mathcal{P}_\star, \mathcal{Q}_\star, \mathcal{U}_\star, \mathcal{V}_\star, \mathcal{X}_\star)$ is a KKT point of problem (21). □

5 Numerical experiments

In this section, we conduct some experiments on real-world dataset to compare the performance of TCTF-M and DTRTC to show their validity. We employ the peak signal-to-noise rate (PSNR) [39], the structural similarity (SSIM) [39], the feature similarity (FSIM) [50] and the recovery computation time to measure the quality of the recovered results. We compare TCTF-M for the matrix completion problem with four existing methods, including SRMF [34], MC-NMF [45], FPCA [24] and SPG [47]. We compare DTRTC for the tensor completion problem with WSTNN [55], TCTF [57], TNN [52], NCPC [44] and NTD [42]. All methods are implemented on the platform of Windows 10 and Matlab (R2020b) with an Intel(R) Core(TM) i7-7700 CPU at 3.60GHz and 24 GB RAM.

5.1 Grayscale image inpainting

In this subsection, we use the USC-SIPI image database[2] to evaluate our proposed method TCTF-M for grayscale image inpainting. In our test, six images are randomly selected from this database, including texture images "Plastic" and "Bark",

[2] http://sipi.usc.edu/database/.

Table 1 Grayscale image inpainting performance comparison

Methods	Picture											
	Plastic				Bark				Pentagon			
	PSNR	SSIM	FSIM	Time	PSNR	SSIM	FSIM	Time	PSNR	SSIM	FSIM	Time
TCTF-M	**31.893**	**0.902**	**0.996**	**0.664**	**30.125**	**0.906**	**0.997**	**0.731**	**30.160**	**0.857**	**0.993**	**0.855**
SRMF	27.144	0.708	0.973	15.175	25.643	0.726	0.975	15.421	26.704	0.627	0.971	15.179
MC-NMF	26.532	0.674	0.964	2.206	24.375	0.661	0.959	2.162	26.518	0.619	0.967	1.604
FPCA	21.038	0.405	0.836	32.836	19.104	0.392	0.842	31.783	22.492	0.408	0.829	34.746
SPG	29.728	0.841	0.984	10.435	29.251	0.880	0.990	10.658	28.536	0.779	0.973	8.964
Methods	Picture											
	Male				Airport				Wash			
	PSNR	SSIM	FSIM	Time	PSNR	SSIM	FSIM	Time	PSNR	SSIM	FSIM	Time
TCTF-M	**31.354**	**0.892**	**0.995**	**0.771**	**29.106**	**0.851**	**0.990**	**0.966**	**24.376**	**0.832**	**0.997**	**2.377**
SRMF	27.984	0.694	0.965	17.155	26.238	0.648	0.961	15.019	19.383	0.364	0.965	147.029
MC-NMF	14.706	0.485	0.878	2.033	25.229	0.626	0.952	1.924	19.013	0.311	0.947	14.406
FPCA	21.856	0.417	0.820	31.613	21.639	0.424	0.832	32.619	17.181	0.198	0.820	345.854
SPG	30.797	0.853	0.984	12.027	29.083	0.823	0.981	11.731	24.048	0.783	0.990	175.104

Bold values indicate the best results

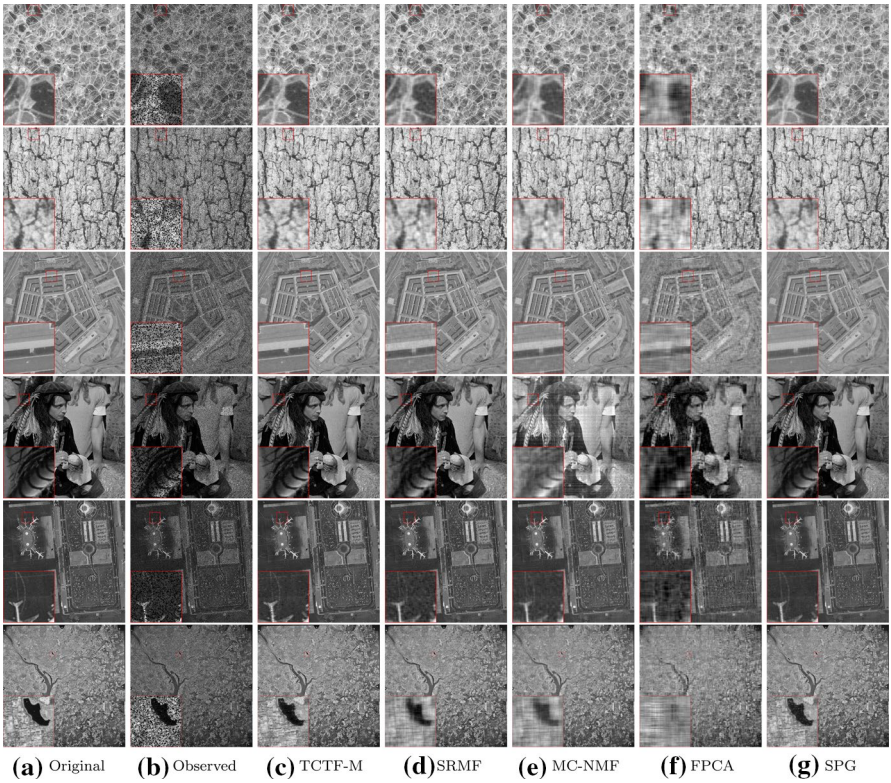


Fig. 3 Examples of grayscale image inpainting

high altitude aerial images "Pentagon" and "Wash", other images "Male" and "Airport". Among them, only the pixels of "Wash" is 2250×2250, and the others are 1024×1024. The data of images are normalized in the range $[0, 1]$.

For each taken image, we randomly sample by the sampling ratio $p = 70\%$. The initial matrix rank is set to 100 in SRMF and MC-NMF. In TCTF-M, "Wash" data sets form a tensor of size $2250 \times 18 \times 125$ and the others set form a tensor of size $1024 \times 8 \times 128$. The initial tubal rank is set to $(round(1.5r), round(r/2), \ldots, round(r/2))$ with $r = round(h/n_3/3)$ in TCTF-M. Furthermore, t_0 is set to be 2.

In Table 1, we present the results of all five methods for different images, and the best results are highlighted in bold. It is easy to see that TCTF-M outperforms the other four methods. TCTF-M is the fastest method, about 3 times faster than the second fastest method MC-NMF. MC-NMF is only slightly longer than TCTF-M in running time, but it has no exact recovery performance guarantee. Both SRMF and FPCA are far inferior to TCTF-M in terms of running time and inpainting results. Although SPG has similar PSNR, SSIM, and FSIM values as TCTF-M, its running time is almost 12 times that of TCTF-M. Especially for the more challenging image

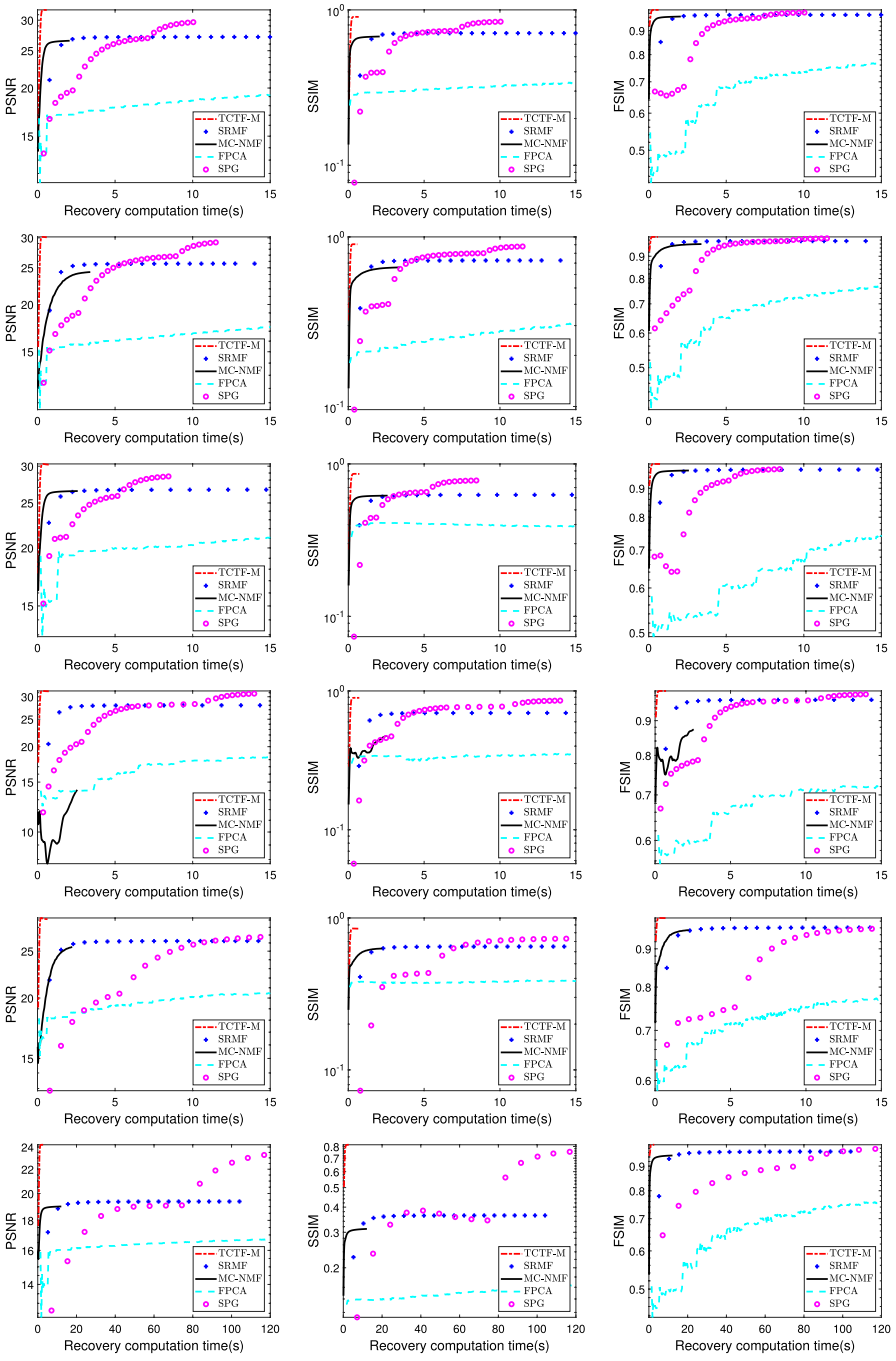


Fig. 4 Grayscale image inpainting results. From top to bottom, the results are for "Plastic", "Bark", "Pentagon", "Male", "Airport" and "Wash", respectively

"Wash" inpainting, TCTF-M is about 73.6 times faster than SPG. Since SPG has to compute SVD at each iteration, it runs slower. In summary, TCTF-M not only achieves the best inpainting results but also runs very fast.

To further demonstrate the performance, images recovered by different algorithms are shown in Fig. 3. Enlarged views of the recovered images evidently show the recovery differences. It can be seen that MC-NMF fails to recover the "Male" image. Furthermore, the recovered images of SRMF and MC-NMF still have some visible reconstruction errors, such as roads in "Pentagon" image, river edge in "Wash" image and lines in "Airport image". TCTF-M and SPG recover these details with better performance.

To further demonstrate the advantage of the proposed algorithms in terms of computational cost, we make a comparison of computation complexity for fives methods in Fig. 4, which shows the PSNR, SSIM, and FSIM values over running time. In order to see the performance, the first 15 (120) seconds are selected for comparison. We can assert that the PSNR, SSIM, and FSIM values of methods based on TCTF-M optimization rapidly increase to the highest values with less running time than other methods.

5.2 High altitude aerial image inpainting

This subsection applies DTRTC to high altitude aerial image inpainting. We also use the USC-SIPI image database to evaluate our proposed method for high altitude aerial image inpainting. In our test, four high altitude aerial images are randomly selected from this database. The first three images both are $1024 \times 1024 \times 3$ pixels and that of the last one is $2250 \times 2250 \times 3$ pixels. The data of images are normalized in the range $[0, 1]$.

For each chosen image, we randomly sample by the sampling ratio $p = 40\%, 45\%, 50\%$. We set the initial double tubal rank $\boldsymbol{r}_{\mathcal{X}}^0 = (200, 30, 30)$, $\boldsymbol{r}_{\tilde{\mathcal{X}}}^0 = (3, \ldots, 3)$ in DTRTC, the initial tubal rank $(200, 30, 30)$ in TCTF, the initial CP rank 100 in NCPC and the initial Tucker rank $(100, 100, 3)$ in NTD. In DTRTC, $\rho(x) = \sqrt{x + \varepsilon}$ and $t_0 = 1$. "Wash" data sets form a tensor of size $3 \times 50625 \times 100$ and the others set form a tensor of size $3 \times 8192 \times 128$. In experiments, the maximum iterative number is set to be 100 and precision ε is set to be 1e−4.

We present the image inpainting results of the four tested images in Table 2, and the best results are highlighted in bold. For visual comparisons, we show the results of the recovered high altitude aerial images by different methods for $p = 40\%$ in Fig. 5. The proposed DTRTC algorithm can be seen to achieve the best performance. The four methods based on tubal rank DTRTC, WSTNN, TCTF, and TNN perform better on PSNR, SSIM, and FSIM values than the method based on CP rank, NCPC, and the method based on Tucker rank, NTD except for the "Wash" image. Furthermore, TCTF and TNN do not use all low rank structures of tensors [55], DTRTC and WSTNN are more comprehensive to preserve all low rank structures of tensor data. However, it can be seen from Lemma 8 that WSTNN over-utilizes the low rank information of the tensor, resulting in too long running

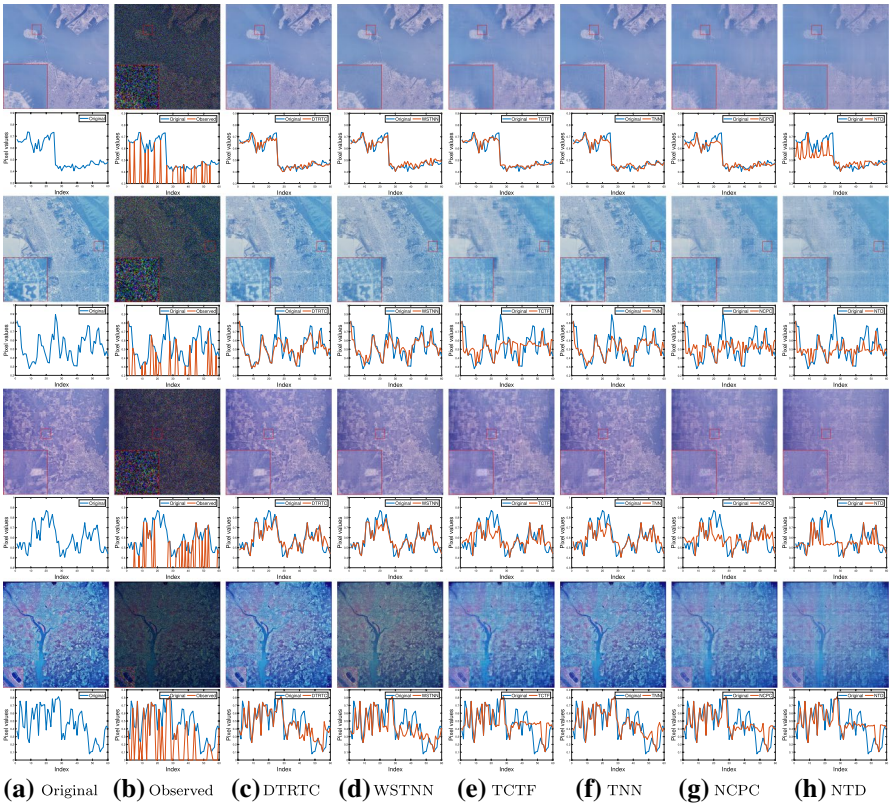


Fig. 5 Examples of high altitude aerial image inpainting with $p = 40\%$. The images are "San Francisco", "Richmond", "Shreveport" and "Wash", respectively. For better visualization, we present the zoom-in region and the corresponding partial residuals of the region

time and little improvement in PSNR, SSIM and FSIM values. For the large scale "Wash" image, the recovery of WSTNN, TCTF, NCPC and NTD is unsatisfactory, but DTRTC and TNN can successfully recover the image. However, since TNN and WSTNN require T-SVD decomposition at each step, as the tensor size increases, its calculation time increases significantly. As a result, DTRTC both produces excellent inpainting results and runs extremely fast.

5.3 Discussion of the size n_2 and n_3

In this subsection, we discuss the influence of different sizes n_2 and n_3 when we reshape a matrix as a third order tensor. Table 3 reports PSNR, SSIM, FSIM values and running time of recovered grayscale image "Wash" by TCTF-M with different n_3. From Table 3, we can see that TCTF-M is not very sensitive to changes in the size n_3 when $n_3 \in [25, 450]$. At the same time, we give PSNR, SSIM, FSIM values and running time of recovered high altitude aerial image "San Francisco" under

Table 2 High altitude aerial image inpainting performance comparison

Picture	Methods	$p = 40\%$				$p = 45\%$				$p = 50\%$			
		PSNR	SSIM	FSIM	Time	PSNR	SSIM	FSIM	Time	PSNR	SSIM	FSIM	Time
San Francisco	DTRTC	**29.997**	**0.832**	0.978	**5.732**	30.550	**0.844**	0.983	**5.416**	**31.897**	**0.884**	0.988	**4.759**
	WSTNN	29.938	0.806	**0.982**	244.549	**30.921**	0.836	**0.988**	184.873	31.807	0.858	**0.991**	181.316
	TCTF	27.159	0.752	0.915	9.992	27.706	0.776	0.932	10.597	28.907	0.802	0.969	10.416
	TNN	28.839	0.774	0.972	167.204	29.572	0.805	0.979	150.400	30.301	0.830	0.984	149.611
	NCPC	26.177	0.693	0.897	36.969	26.401	0.711	0.909	27.820	27.240	0.751	0.928	27.853
	NTD	25.586	0.703	0.878	11.062	26.276	0.730	0.902	10.480	26.776	0.754	0.918	10.578
Richmond	DTRTC	**28.923**	0.804	**0.987**	**10.283**	**30.140**	0.849	**0.991**	**7.106**	**31.186**	0.879	**0.993**	**6.303**
	WSTNN	28.631	**0.815**	0.980	192.782	29.981	**0.858**	0.988	306.987	30.985	**0.882**	0.992	353.232
	TCTF	24.777	0.661	0.874	10.271	25.750	0.701	0.921	17.995	27.008	0.743	0.963	17.807
	TNN	27.582	0.750	0.974	148.175	28.401	0.786	0.981	262.497	29.243	0.819	0.987	288.664
	NCPC	24.116	0.614	0.861	27.919	24.980	0.661	0.897	36.301	25.225	0.679	0.911	36.154
	NTD	23.853	0.627	0.843	10.748	24.405	0.661	0.869	14.232	25.042	0.695	0.896	14.201
Shreveport	DTRTC	**29.730**	0.824	**0.987**	**7.378**	30.445	0.850	**0.990**	**7.967**	**31.641**	**0.879**	**0.994**	**6.849**
	WSTNN	29.662	**0.828**	0.984	378.739	**30.625**	**0.856**	0.989	382.169	31.522	**0.879**	0.993	371.212
	TCTF	26.413	0.684	0.926	17.868	27.014	0.717	0.943	17.269	28.361	0.766	0.978	18.584
	TNN	28.251	0.752	0.976	258.789	28.984	0.786	0.982	289.570	29.731	0.817	0.987	287.477
	NCPC	25.195	0.627	0.883	25.993	25.540	0.652	0.899	35.481	25.903	0.681	0.912	36.616
	NTD	24.629	0.628	0.838	10.025	25.219	0.662	0.873	14.144	25.784	0.694	0.899	14.506
Wash	DTRTC	**22.122**	**0.694**	**0.991**	39.163	**22.703**	**0.735**	**0.993**	**31.289**	**23.339**	**0.770**	**0.995**	**26.273**
	WSTNN	13.698	0.372	0.910	1473.129	15.012	0.427	0.939	1472.944	16.488	0.485	0.960	1464.967
	TCTF	19.560	0.540	0.883	52.778	19.878	0.574	0.893	53.191	20.581	0.623	0.929	52.741
	TNN	21.727	0.644	0.980	1299.844	22.426	0.690	0.986	1306.686	23.144	0.732	0.990	1303.736
	NCPC	19.310	0.528	0.871	118.265	19.736	0.567	0.895	117.742	20.298	0.608	0.919	121.550
	NTD	18.910	0.518	0.811	**33.840**	19.499	0.562	0.863	34.513	19.655	0.589	0.868	34.959

Bold values indicate the best results

Table 3 The recovered grayscale image "Wash" by TCTF-M with different n_3

n_3	5	25	45	75	125	225	450	750	1125
PSNR	23.498	24.269	24.347	24.358	24.376	**24.426**	24.414	23.712	21.421
SSIM	0.776	0.820	0.827	0.830	0.832	**0.836**	**0.836**	0.822	0.736
FSIM	0.996	**0.997**	**0.997**	**0.997**	**0.997**	**0.997**	**0.997**	0.996	0.992
Time	4.864	3.108	3.075	3.070	2.377	2.511	**2.286**	3.663	11.087

Bold values indicate the best results

Table 4 The recovered high altitude aerial image "San Francisco" by DTRTC with different n_3

n_3	4	32	64	128	256	512	1024	2048	4096	8192
PSNR	29.092	30.440	30.550	30.587	30.730	30.902	31.024	31.050	31.065	**31.070**
SSIM	0.783	0.840	0.844	0.846	0.851	0.858	0.862	0.863	**0.864**	**0.864**
FSIM	0.979	0.983	0.983	0.984	0.984	**0.985**	**0.985**	**0.985**	**0.985**	**0.985**
Time	7.368	5.931	**5.416**	5.450	5.759	6.227	6.744	7.676	7.933	9.680

Bold values indicate the best results

sampling ratio $p = 45\%$ by DTRTC with different n_3 in Table 4. It can be seen from Table 4 that DTRTC is not very sensitive to changes of size n_3 when $n_3 \in [64, 8192]$, but they directly affect the computational complexity of the proposed algorithm. Therefore, we suggest that n_3 be about 100 in the experiment to obtain better results.

5.4 Effectiveness of adaptive selection of parameters γ_1 and γ_2

In this subsection, we further illustrate the effectiveness of adaptive selection of parameters γ_1 and γ_2 in the proposed framework. Specifically, we compare the performance of DTRTC with fixed $\gamma_1 (\gamma_2 = 1 - \gamma_1)$ from 0.1 to 0.9.

Table 5 reports the PSNR, SSIM, FSIM values and running time of recovered high altitude aerial image "San Francisco" under sampling ratio $p = 45\%$ by DTRTC with different selection of γ_1. It is easy to observe that better recovered results can be obtained by DTRTC by adaptive selection of γ_1.

Table 5 The recovered results of "San Francisco" by DTRTC with different γ_1

γ_1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	DTRTC
PSNR	29.187	29.556	29.669	29.666	29.659	29.695	29.610	29.368	28.920	**30.550**
SSIM	0.791	0.817	0.822	0.820	0.817	0.817	0.812	0.801	0.781	**0.844**
FSIM	0.979	0.980	0.980	0.980	0.980	0.980	0.980	0.979	0.977	**0.983**
Time	14.660	8.470	6.191	5.680	5.219	4.954	**4.940**	4.977	5.527	5.416

Bold values indicate the best results

6 Conclusion

In this paper, we established a relationship between matrix rank and tensor tubal rank. Based on the relationship, we modeled the matrix completion problem as a third order tensor completion problem and proposed a two-stage tensor factorization based algorithm, which made a drastic reduction on the dimension of data and hence cut down on the running time. For low rank tensor completion problem, we introduced double tubal rank. Compared to tubal rank, tensor fibered rank, double tubal rank can not only fully exploit the low rank structures of the tensor but also avoid the low rank structure redundancy. Based on this rank, a reweighted tensor factorization algorithm was proposed. The reported experiments demonstrated that our proposed algorithms were much more efficient than most of state-of-the-art matrix/tensor completion algorithms.

Acknowledgements Xinzhen Zhang was supported by the National Natural Science Foundation of China (Grant No. 11871369).

Data Availability The datasets generated by the study during and/or analyzed during the current research are available in the Dataverse repository: https://github.com/quanyumath/DTRTC.

References

1. Bai, M., Zhang, X., Ni, G., Cui, C.: An adaptive correction approach for tensor completion. SIAM J. Imaging Sci. **9**(3), 1298–1323 (2016)
2. Bengua, J.A., Phien, H.N., Tuan, H.D., Do, M.N.: Efficient tensor completion for color image and video recovery: low-rank tensor train. IEEE Trans. Image Process. **26**(5), 2466–2479 (2017)
3. Border, K.: The supergradient of a concave function (2001). https://healy.econ.ohio-state.edu/kcb/Notes/Supergrad.pdf
4. Cabral, R., De la Torre, F., Costeira, J.P., Bernardino, A.: Matrix completion for weakly-supervised multi-label image classification. IEEE Trans. Pattern Anal. Mach. Intell. **37**(1), 121–135 (2015)
5. Cai, J.F., Candès, E.J., Shen, Z.: A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**(4), 1956–1982 (2010)
6. Gandy, S., Recht, B., Yamada, I.: Tensor completion and low-n-rank tensor recovery via convex optimization. Inverse Prob. **27**(2), 025010 (2011). https://doi.org/10.1088/0266-5611/27/2/025010
7. Gao, B., Absil, P.A.: A Riemannian rank-adaptive method for low-rank matrix completion. Comput. Optim. Appl. **81**(1), 67–90 (2021). https://doi.org/10.1007/s10589-021-00328-w
8. Guo, H., Yu, Q., Zhang, X., Cheng, L.: Low rank matrix minimization with a truncated difference of nuclear norm and frobenius norm regularization. J. Ind. Manag. Optim. (2022). https://doi.org/10.3934/jimo.2022045
9. He, Y., Wang, F., Li, Y., Qin, J., Chen, B.: Robust matrix completion via maximum correntropy criterion and half-quadratic optimization. IEEE Trans. Signal Process. **68**, 181–195 (2020)
10. Hillar, C.J., Lim, L.H.: Most tensor problems are NP-hard. J. ACM **60**(6), 1–39 (2013). https://doi.org/10.1145/2512329
11. Hitchcock, F.L.: The expression of a tensor or a polyadic as a sum of products. J. Math. Phys. **6**(1–4), 164–189 (1927)
12. Jiang, T.X., Ng, M.K., Zhao, X.L., Huang, T.Z.: Framelet representation of tensor nuclear norm for third-order tensor completion. IEEE Trans. Image Process. **29**, 7233–7244 (2020)
13. Keshavan, R.H., Montanari, A., Oh, S.: Matrix completion from a few entries. IEEE Trans. Inf. Theory **56**(6), 2980–2998 (2010)
14. Kilmer, M.E., Braman, K., Hao, N., Hoover, R.C.: Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging. SIAM J. Matrix Anal. Appl. **34**(1), 148–172 (2013). https://doi.org/10.1137/110837711
15. Kilmer, M.E., Martin, C.D.: Factorization strategies for third-order tensors. Linear Algebra Appl. **435**(3), 641–658 (2011). https://doi.org/10.1016/j.laa.2010.09.020

16. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **51**(3), 455–500 (2009)
17. Lee, C., Li, Y., Monga, V.: Ghost-free high dynamic range imaging via rank minimization. IEEE Signal Process. Lett. **21**(9), 1045–1049 (2014)
18. Li, Y.F., Shang, K., Huang, Z.H.: A singular value p-shrinkage thresholding algorithm for low rank matrix recovery. Comput. Optim. Appl. **73**(2), 453–476 (2019). https://doi.org/10.1007/s10589-019-00084-y
19. Ling, C., Yu, G., Qi, L., Xu, Y.: T-product factorization method for internet traffic data completion with spatio-temporal regularization. Comput. Optim. Appl. **80**(3), 883–913 (2021). https://doi.org/10.1007/s10589-021-00315-1
20. Liu, J., Musialski, P., Wonka, P., Ye, J.: Tensor completion for estimating missing values in visual data. IEEE Trans. Pattern Anal. Mach. Intell. **35**(1), 208–220 (2013)
21. Liu, Y.J., Sun, D., Toh, K.C.: An implementable proximal point algorithmic framework for nuclear norm minimization. Math. Program. **133**(1), 399–436 (2012)
22. Lu, C., Tang, J., Yan, S., Lin, Z.: Generalized nonconvex nonsmooth low-rank minimization. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. IEEE (2014). https://doi.org/10.1109/cvpr.2014.526
23. Luo, Y., Liu, T., Tao, D., Xu, C.: Multiview matrix completion for multilabel image classification. IEEE Trans. Image Process. **24**(8), 2355–2368 (2015)
24. Ma, S., Goldfarb, D., Chen, L.: Fixed point and Bregman iterative methods for matrix rank minimization. Math. Program. **128**(1), 321–353 (2011)
25. Martin, C.D., Shafer, R., LaRue, B.: An order-p tensor factorization with applications in imaging. SIAM J. Sci. Comput. **35**(1), A474–A490 (2013)
26. Mazumder, R., Hastie, T., Tibshirani, R.: Spectral regularization algorithms for learning large incomplete matrices. J. Mach. Learn. Res. **11**(1), 2287–2322 (2010)
27. Mu, C., Huang, B., Wright, J., Goldfarb, D.: Square deal: lower bounds and improved relaxations for tensor recovery. In: Proceedings of the 31st International Conference on International Conference on Machine Learning, vol. 32, pp. II-73–II-81 (2014)
28. Oh, T.H., Lee, J.Y., Tai, Y.W., Kweon, I.S.: Robust high dynamic range imaging by rank minimization. IEEE Trans. Pattern Anal. Mach. Intell. **37**(6), 1219–1232 (2015)
29. Oseledets, I.V.: Tensor-train decomposition. SIAM J. Sci. Comput. **33**(5), 2295–2317 (2011)
30. Qi, L., Chen, Y., Bakshi, M., Zhang, X.: Triple decomposition and tensor recovery of third order tensors. SIAM J. Matrix Anal. Appl. **42**(1), 299–329 (2021). https://doi.org/10.1137/20m1323266
31. Recht, B., Fazel, M., Parrilo, P.A.: Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM Rev. **52**(3), 471–501 (2010)
32. Recht, B., Ré, C.: Parallel stochastic gradient algorithms for large-scale matrix completion. Math. Program. Comput. **5**(2), 201–226 (2013)
33. Rojo, O., Rojo, H.: Some results on symmetric circulant matrices and on symmetric centrosymmetric matrices. Linear Algebra Appl. **392**(15), 211–233 (2004). https://doi.org/10.1016/j.laa.2004.06.013
34. Roughan, M., Zhang, Y., Willinger, W., Qiu, L.: Spatio-temporal compressive sensing and internet traffic matrices (extended version). IEEE/ACM Trans. Netw. **20**(3), 662–676 (2012)
35. Sobral, A., Zahzah, Eh.: Matrix and tensor completion algorithms for background model initialization: a comparative evaluation. Pattern Recognit. Lett. **96**(1), 22–33 (2017)
36. Toh, K.C., Yun, S.: An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. Pac. J. Optim. **6**(3), 615–640 (2010)
37. Tsagkatakis, G., Tsakalides, P.: Efficient high dynamic range imaging via matrix completion. In: 2012 IEEE International Workshop on Machine Learning for Signal Processing, pp. 1–6 (2012)
38. Tucker, L.R.: Some mathematical notes on three-mode factor analysis. Psychometrika **31**(3), 279–311 (1966)
39. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004). https://doi.org/10.1109/tip.2003.819861
40. Wen, Z., Yin, W., Zhang, Y.: Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. Math. Program. Comput. **4**(4), 333–361 (2012)
41. Xie, Y., Gu, S., Liu, Y., Zuo, W., Zhang, W., Zhang, L.: Weighted Schatten p-norm minimization for image denoising and background subtraction. IEEE Trans. Image Process. **25**(10), 4842–4857 (2016). https://doi.org/10.1109/tip.2016.2599290

42. Xu, Y.: Alternating proximal gradient method for sparse nonnegative tucker decomposition. Math. Program. Comput. **7**(1), 39–70 (2015). https://doi.org/10.1007/s12532-014-0074-y
43. Xu, Y., Hao, R., Yin, W., Su, Z.: Parallel matrix factorization for low-rank tensor completion. Inverse Probl. Imaging **9**(2), 601–624 (2015). https://doi.org/10.3934/ipi.2015.9.601
44. Xu, Y., Yin, W.: A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM J. Imaging Sci. **6**(3), 1758–1789 (2013). https://doi.org/10.1137/120887795
45. Xu, Y., Yin, W., Wen, Z., Zhang, Y.: An alternating direction algorithm for matrix completion with nonnegative factors. Front. Math. China **7**(2), 365–384 (2012). https://doi.org/10.1007/s11464-012-0194-5
46. Yang, L., Huang, Z.H., Hu, S., Han, J.: An iterative algorithm for third-order tensor multi-rank minimization. Comput. Optim. Appl. **63**(1), 169–202 (2015). https://doi.org/10.1007/s10589-015-9769-x
47. Yu, Q., Zhang, X.: A smoothing proximal gradient algorithm for matrix rank minimization problem. Comput. Optim. Appl. **81**(2), 519–538 (2022). https://doi.org/10.1007/s10589-021-00337-9
48. Yu, Q., Zhang, X., Chen, Y., Qi, L.: Low tucker rank tensor completion using a symmetric block coordinate descent method. Numer. Linear Algebra Appl. (2022). https://doi.org/10.1002/nla.2464
49. Zeng, W.J., So, H.C.: Outlier-robust matrix completion via ℓ_p-minimization. IEEE Trans. Signal Process. **66**(5), 1125–1140 (2018)
50. Zhang, L., Zhang, L., Mou, X., Zhang, D.: FSIM: a feature similarity index for image quality assessment. IEEE Trans. Image Process. **20**(8), 2378–2386 (2011). https://doi.org/10.1109/tip.2011.2109730
51. Zhang, X.: A nonconvex relaxation approach to low-rank tensor completion. IEEE Trans. Neural Netw. Learn. Syst. **30**(6), 1659–1671 (2019)
52. Zhang, Z., Ely, G., Aeron, S., Hao, N., Kilmer, M.: Novel methods for multilinear data completion and de-noising based on tensor-SVD. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3842–3849 (2014)
53. Zhao, Q., Zhou, G., Xie, S., Zhang, L., Cichocki, A.: Tensor ring decomposition (2016). arXiv:1606.05535
54. Zheng, Y., Liu, G., Sugimoto, S., Yan, S., Okutomi, M.: Practical low-rank matrix approximation under robust L1-norm. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1410–1417 (2012)
55. Zheng, Y.B., Huang, T.Z., Zhao, X.L., Jiang, T.X., Ji, T.Y., Ma, T.H.: Tensor n-tubal rank and its convex relaxation for low-rank tensor recovery. Inf. Sci. **532**, 170–189 (2020). https://doi.org/10.1016/j.ins.2020.05.005
56. Zheng, Y.B., Huang, T.Z., Zhao, X.L., Jiang, T.X., Ma, T.H., Ji, T.Y.: Mixed noise removal in hyperspectral image via low-fibered-rank regularization. IEEE Trans. Geosci. Remote Sens. **58**(1), 734–749 (2020). https://doi.org/10.1109/tgrs.2019.2940534
57. Zhou, P., Lu, C., Lin, Z., Zhang, C.: Tensor factorization for low-rank tensor completion. IEEE Trans. Image Process. **27**(3), 1152–1163 (2018)