Received: 16 November 2021 | Revised: 21 May 2022 | Accepted: 11 July 2022

DOI: 10.1002/nla.2464

RESEARCH ARTICLE

WILEY

# Low Tucker rank tensor completion using a symmetric block coordinate descent method

**Quan Yu**[1] | **Xinzhen Zhang**[2] | **Yannan Chen**[3] | **Liqun Qi**[4]

[1]School of Mathematics, Hunan University, Hunan, China

[2]School of Mathematics, Tianjin University, Tianjin, China

[3]School of Mathematical Sciences, South China Normal University, Guangzhou, China

[4]Huawei Theory Research Lab, Hong Kong, China

**Correspondence**
Xinzhen Zhang, School of Mathematics, Tianjin University, Tianjin 300072, China.
Email: xzzhang@tju.edu.cn

**Funding information**
Guangdong Basic and Applied Basic Research Foundation, Grant/Award Number: 2020A1515010489; Guangdong Province Higher Education Foundation, Grant/Award Number: 2021ZDZX1071; National Natural Science Foundation of China, Grant/Award Numbers: 11871369, 12171168

**Abstract**

Low Tucker rank tensor completion has wide applications in science and engineering. Many existing approaches dealt with the Tucker rank by unfolding matrix rank. However, unfolding a tensor to a matrix would destroy the data's original multi-way structure, resulting in vital information loss and degraded performance. In this article, we establish a relationship between the Tucker ranks and the ranks of the factor matrices in Tucker decomposition. Then, we reformulate the low Tucker rank tensor completion problem as a multilinear low rank matrix completion problem. For the reformulated problem, a symmetric block coordinate descent method is customized. For each matrix rank minimization subproblem, the classical truncated nuclear norm minimization is adopted. Furthermore, temporal characteristics in image and video data are introduced to such a model, which benefits the performance of the method. Numerical simulations illustrate the efficiency of our proposed models and methods.

**KEYWORDS**

block coordinate descent, sparsity, tensor completion, truncated nuclear norm, Tucker rank

## 1 | INTRODUCTION

A tensor is a multidimensional array. Recently, more and more applications of tensors are found in signal processing,[1,2] face recognition,[3] computer vision,[4-7] high-order web link analysis,[8] data mining,[9] and collaborative filtering.[10] In real-world applications, tensor may be incomplete from the missing information or limitation of data transmission bandwidth, which leads to the low rank tensor completion problem. The low rank tensor completion problem is to reconstruct a tensor from the observed incomplete tensor, which has a wide range of realistic applications, such as seismic data reconstruction,[11] color image video recovery,[12-19] and medical image processing.[20,21] Mathematically, a unified low rank tensor completion model can be written as

$$\min \ \operatorname{rank}(\mathcal{X}) \quad \text{s.t.} \quad P_\Omega(\mathcal{X}) = P_\Omega\left(\overline{\mathcal{X}}\right), \tag{1}$$

where $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_m}$ is the underlying tensor, $\Omega$ is an index set of observed entries and $P_\Omega(\cdot)$ is a projection operator.

[Correction added on 08 September 2022, after first online publication: equations in sections 2 and 4.1 have been corrected in this version.]

Low rank matrix completion[22,23] can be regarded as a special case of low rank tensor completion. Different from the matrix case, there are various tensor decompositions, including CANDECOMP/PARAFAC (CP) decomposition,[24,25] Tucker decomposition,[26] tensor train decomposition,[27] and tensor singular value decomposition (SVD).[28,29] Corresponding to these decompositions, tensor ranks are called the CP-rank, Tucker rank, tensor train rank, and tubal rank, respectively. Different tensor ranks lead to various low rank tensor completion models. It is NP-hard to compute CP-rank.[30] Thus, it is challenging to consider the low CP-rank tensor completion model. Compared with CP-rank, the tensor train rank is much easier to compute, but it always has a fixed pattern, that is, smaller for the border cores and larger for the middle cores, which might not be the optimum for specific data tensor.[31] More recently, tubal rank, corresponding to a tensor–tensor product and the tensor SVD of a third-order tensor was introduced in Reference 28. Based on tubal rank, the low tubal rank tensor completion problem was discussed in References 28, 29, and 32. Although models and related methods based on tubal rank are effective to a certain extent, tubal rank is applicable only for third-order tensors. However, many tensor data, in reality, are of higher-order. For example, color video data is expressed by a fourth-order tensor.

Tucker rank is a commonly used definition of higher-order tensor ranks. For an $m$th order tensor, Tucker rank is a vector of all unfolding matrix ranks. Based on Tucker rank, Liu et al.[5] defined the nuclear norm of a tensor by the average of the nuclear norm of each unfolding matrix. Xu et al.[33] used matrix factorization to preserve the low rank structure of unfolded matrices. An alternating proximal gradient method in Reference 34 was proposed based on sparse nonnegative Tucker decomposition.

In this article, we establish a relationship between the factor matrix rank in Tucker decomposition and the unfolding matrix rank. Then we reformulate the low Tucker rank tensor completion problem as a multilinear low rank and sparse matrix completion problem. This reformulation not only avoids the destroying of the multi-dimensional structure, but also explores the low rank information. In recent years, truncated nuclear norm (TNN), discarding large singular values, is a better estimation of matrix rank. It was shown in Reference 35 that the truncated nuclear norm regularization can achieve better performance than nuclear norm regularization for matrix completion problems. Then TNN is adopted to measure the matrix rank. Furthermore, $l_1$ is adopted to estimate the sparsity of core tensor and factor matrices and a relaxation problem is arrived. For the relaxation problem, we propose a symmetric block coordinate descent (SBCD) algorithm.

This article is organized as follows. We first recall notation and state some basic properties of tensors in Section 2. Based on the definition of Tucker decomposition, a multilinear low rank and sparse matrix minimization model is reformulated for the low Tucker rank tensor completion problem. For the reformulated model, the truncated nuclear norm and $l_1$ norm of factor matrices, together with $l_1$ norm of core tensor are adopted to relax the reformulated problem in Section 3. To solve the relaxation problem, a SBCD algorithm is proposed. We further improve the model to fit the tensor data with temporal characteristics better and modify the proposed algorithm in Section 4. Numerical examples are reported in Section 5 to show the performance of our proposed models and methods. Section 6 briefly concludes our study and introduces the future work.

## 2 | PRELIMINARY KNOWLEDGE ON TENSOR

In this section, we recall some notations on tensor and its Tucker decomposition. More details can be found in Kolda and Bader's review.[36]

Tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ is said to be an $m$-order $(n_1, \ldots, n_m)$-dimensional real tensor, whose elements are denoted as $\mathcal{T}_{i_1 i_2 \cdots i_m}$, where $i_j \in [n_j]$ and $j \in [m]$. Here $[n] := \{1, 2, \ldots, n\}$ for a positive integer $n$. A fiber of tensor $\mathcal{T}$ is a vector defined by fixing all indices but one and a slice of tensor $\mathcal{T}$ is a matrix defined by fixing all indices but two. The mode-$s$ unfolding $T_{(s)}$ of tensor $\mathcal{T}$ is a matrix in $\mathbb{R}^{n_s \times N_s}$ with its $(i,j)$th element being $\mathcal{T}_{i_1 \cdots i_{s-1} i i_{s+1} \cdots i_m}$, where $j = 1 + \sum_{k=1, k\neq s}^{m} (i_k - 1)\overline{n}_k$, $\overline{n}_k = \prod_{l \neq s, l<k} n_l$ and $N_s = \prod_{k=1, k\neq s}^{m} n_k$. The unfolding matrix can be obtained by "tens2mat($\mathcal{T}, s$)" in Matlab. The opposite operation "fold" is defined as $\text{fold}_s(T_{(s)}) := \mathcal{T}$.

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

**Definition 1.** For a tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$, let $T_{(i)} \in n_i \times N_i$ with $N_i = n_1 \times \cdots \times n_{i-1} \times n_{i+1} \times \cdots \times n_m$ be the mode-$i$ unfolding matrix for $i \in [m]$. Then Tucker rank of $\mathcal{T}$ is defined as

$$\text{rank}_{tc}(\mathcal{T}) = \left(\text{rank}(T_{(1)}), \ldots, \text{rank}(T_{(m)})\right).$$

Now we recall the definition of $k$-mode product.

**Definition 2.** For a given tensor $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ and a matrix $M \in \mathbb{R}^{J_k \times n_k}$, the mode-$k$ product of $\mathcal{T}$ with $M$ is a tensor of $(n_1 \times \cdots \times n_{k-1} \times J_k \times n_{k+1} \times \cdots \times n_m)$ with its entries

$$(\mathcal{T} \times_k M)_{i_1 i_2 \cdots i_m} = \sum_{j_k=1}^{n_k} T_{i_1 i_2 \cdots i_{k-1} j_k i_{k+1} \cdots i_m} M_{i_k j_k}.$$

For matrices $M$, $Q$, $M^{(1)}$, and $M^{(2)}$ of appropriate sizes, there hold

$$\mathcal{T} \times_i M \times_j Q = (\mathcal{T} \times_i M) \times_j Q = (\mathcal{T} \times_j Q) \times_i M, \quad i \neq j,$$
$$\mathcal{T} \times_i M^{(1)} \times_i M^{(2)} = \mathcal{T} \times_i (M^{(2)} M^{(1)}).$$

Let $\mathcal{T} = \mathcal{G} \times_1 V^{(1)} \times_2 V^{(2)} \times \cdots \times_m V^{(m)}$. Then for any $i \in [m]$,

$$T_{(i)} = V^{(i)} G_{(i)} \left( V^{(m)} \otimes \cdots \otimes V^{(i+1)} \otimes V^{(i-1)} \otimes \cdots \otimes V^{(1)} \right)^T, \tag{2}$$

where $A \otimes B$ denotes the Kronecker product of $A$ and $B$, $G_{(i)}$ is the mode-$i$ unfolding matrix of tensor $\mathcal{G}$. Based on this notation, we are ready to present the definition of orthogonal Tucker decomposition as follows.

**Definition 3.** Suppose that

$$\mathcal{T} = \mathcal{G} \times_1 U^{(1)} \times_2 \cdots \times_m U^{(m)}, \tag{3}$$

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

Clearly, Tucker decompositions are not unique. In fact, for any orthogonal matrix $\overline{V}^{(i)} \in \mathbb{R}^{r_i \times r_i}$, the following decomposition is also an orthogonal Tucker decomposition of $\mathcal{T}$

$$\mathcal{T} = \left( \mathcal{G} \times_1 \overline{V}^{(1)} \times_2 \cdots \times_m \overline{V}^{(m)} \right) \times_1 \left( U^{(1)} \left( \overline{V}^{(1)} \right)^{-1} \right) \times_2 \cdots \times_m \left( U^{(m)} \left( \overline{V}^{(m)} \right)^{-1} \right)$$

if $\mathcal{T} = \mathcal{G} \times_1 U^{(1)} \times_2 \cdots \times_m U^{(m)}$ is an orthogonal Tucker decomposition.

Let $\mathcal{T} = \mathcal{G} \times_1 U^{(1)} \times_2 \cdots \times_m U^{(m)}$ be an orthogonal Tucker decomposition with $\mathcal{G} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_m}$ and orthogonal matrices $U^{(i)} \in \mathbb{R}^{n_i \times r_i}$ for all $i \in [m]$. Denote $\mathcal{H} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ and $V^{(i)} \in \mathbb{R}^{n_i \times n_i}$ with their entries

$$\begin{aligned} \mathcal{H}_{i_1 i_2 \cdots i_m} &= \begin{cases} \mathcal{G}_{i_1 i_2 \cdots i_m} & i_j \leq r_j,\ j \in [m], \\ 0 & \text{otherwise}, \end{cases} \\ V^{(i)}(:, j) &= \begin{cases} U^{(i)}(:, j) & j \leq r_i, \\ 0 & \text{otherwise}, \end{cases} \qquad i \in [m]. \end{aligned} \tag{4}$$

By direct computation, we have

$$\mathcal{T} = \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)}.$$

Before we end this section, some notations used later are presented here. For the same-sized tensors $\mathcal{A}$, $\mathcal{B} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$, their inner product is the sum of products of the entries, that is, $\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1, i_2, \ldots, i_m = 1}^{n_1, n_2, \ldots, n_m} \mathcal{A}_{i_1 i_2 \cdots i_m} \mathcal{B}_{i_1 i_2 \cdots i_m}$. Then the Frobenius norm of tensor $\mathcal{A}$ is $\|\mathcal{A}\|_F = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$ and $\|\mathcal{A}\|_{l_1} := \sum_{i_1, \ldots, i_m} |\mathcal{A}_{i_1 \cdots i_m}|$. For a matrix $X \in \mathbb{R}^{n \times n}$ and $s \in [n]$, the truncated nuclear norm of $X$ is $\|X\|_{*,s} = \sum_{i=n-s+1}^{n} \sigma_i(X)$, that is, the sum of $s$ smallest singular values, where $\sigma_1(X) \geq \sigma_2(X) \geq \cdots \geq \sigma_n(X)$ are the singular values of $X$.

## 3 | REFORMULATION OF LOW TUCKER RANK TENSOR COMPLETION

Tucker rank is a vector of matrix ranks, which makes tensor rank minimization problem be a vector optimization. To keep things simple, we use the sum of Tucker rank as a tensor rank, and hence model (1) can be written as

$$\min_{\mathcal{X}} \sum_{i=1}^{m} \text{rank}\left(X_{(i)}\right) \quad \text{s.t.} \quad P_\Omega(\mathcal{X}) = P_\Omega\left(\overline{\mathcal{X}}\right). \tag{5}$$

Combining the definition of Tucker decomposition and notations (4), we consider the following multilinear matrix rank minimization problem

$$\begin{aligned} \min_{\mathcal{H}, V^{(1)}, \ldots, V^{(m)}} \quad & \textstyle\sum_{i=1}^{m} \text{rank}\left(V^{(i)}\right), \\ \text{s.t.} \quad & P_\Omega\left(\mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)}\right) = P_\Omega\left(\overline{\mathcal{X}}\right), \\ & \mathcal{H} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}, \ V^{(i)} \in \mathbb{R}^{n_i \times n_i}, \quad i \in [m]. \end{aligned} \tag{6}$$

Now we are ready to establish the equivalence between problems (5) and (6). To this end, the following Lemma is needed.

**Lemma 1.** *Suppose that* $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$ *and* $\mathcal{T} = \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)}$. *Then* $\text{rank}\left(T_{(i)}\right) \le \text{rank}\left(V^{(i)}\right)$.

*Proof.* The result is clear from Equation (2) and the rank property of matrix product. ∎

**Theorem 1.** *Problems (5) and (6) are equivalent. That is, they have the same optimal values. Furthermore, any optimization solution of problem (5) returns an optimization solution of (6), and vice versa.*

*Proof.* Suppose that $\left(\mathcal{H}, V^{(1)}, \ldots, V^{(m)}\right)$ is an optimal solution tuple of problem (6) and $\mathcal{X}$ is an optimal solution of (5).

Let $\mathcal{T} = \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)}$. We first show that $\sum_{i=1}^{m}\text{rank}\left(T_{(i)}\right) = \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right)$. Otherwise, $\sum_{i=1}^{m}\text{rank}\left(T_{(i)}\right) < \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right)$ from Lemma 1. Let $\mathcal{T} = \mathcal{G} \times_1 U^{(1)} \times_2 \cdots \times_m U^{(m)}$ be an orthogonal Tucker decomposition. Let $\overline{\mathcal{H}}$ and $\overline{V^{(i)}}$ be defined by $\mathcal{G}$ and $U^{(i)}$ as in (4). Then $\mathcal{T} = \overline{\mathcal{H}} \times_1 \overline{V^{(1)}} \times_2 \cdots \times_m \overline{V^{(m)}}$ with $\text{rank}\left(\overline{V^{(i)}}\right) = \text{rank}\left(T_{(i)}\right)$ for all $i \in [m]$. Clearly, $(\overline{\mathcal{H}}, \overline{V^{(1)}}, \ldots, \overline{V^{(m)}})$ is a feasible solution tuple of (6), and hence

$$\sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right) \le \sum_{i=1}^{m}\text{rank}\left(\overline{V^{(i)}}\right) = \sum_{i=1}^{m}\text{rank}\left(T_{(i)}\right) < \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right),$$

which arrives at a contradiction. So we can assert that

$$\sum_{i=1}^{m}\text{rank}\left(T_{(i)}\right) = \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right).$$

From Definition 3 and (4), there exist $\underline{\mathcal{H}} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}$, $\underline{V^{(i)}} \in \mathbb{R}^{n_i \times n_i}$ $(i \in [m])$ such that $\mathcal{X} = \underline{\mathcal{H}} \times_1 \underline{V^{(1)}} \times_2 \cdots \times_m \underline{V^{(m)}}$ and $\text{rank}\left(\underline{V^{(i)}}\right) = \text{rank}\left(X_{(i)}\right)$ for $i \in [m]$. Obviously, such $(\underline{\mathcal{H}}, \underline{V^{(1)}}, \ldots, \underline{V^{(m)}})$ is a feasible solution tuple of (6). Hence $\sum_{i=1}^{m}\text{rank}\left(\underline{V^{(i)}}\right) \ge \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right)$. On the other hand, $\sum_{i=1}^{m}\text{rank}\left(\underline{V^{(i)}}\right) = \sum_{i=1}^{m}\text{rank}\left(X_{(i)}\right) \le \sum_{i=1}^{m}\text{rank}\left(T_{(i)}\right) = \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right)$ since $\mathcal{X}$ is an optimal solution of problem (5) and $\mathcal{T}$ is a feasible solution of (5).

Hence $\sum_{i=1}^{m}\text{rank}\left(X_{(i)}\right) = \sum_{i=1}^{m}\text{rank}\left(V^{(i)}\right)$ and the equivalence between problem (5) and problem(6) is established now.

From the above procedure, $\mathcal{T}$ is an optimal solution of (5) and $(\underline{\mathcal{H}}, \underline{V^{(1)}}, \ldots, \underline{V^{(m)}})$ is an optimal solution tuple of (6). This completes the proof. ∎

Clearly, when $\mathcal{X}$ is of lower rank on mode $i$, that is, $r_i < n_i$, its factor matrix $V^{(i)} \in \mathbb{R}^{n_i \times n_i}$ has $n_i - r_i$ columns of which elements can be all zeros. Moreover, corresponding fibers of the core tensor $\mathcal{H} \in \mathbb{R}^{n_1 \times \cdots \times n_m}$ are also zeros. To capture these zeros, we impose the $\ell_1$ norm regularization on $\mathcal{H}$ and $V^{(i)}$'s in the objective function. Since the truncated nuclear norm achieves an accurate and robust approximation to matrix rank,[35] we adopt it in problem (6) to relax matrix rank function.

Hence, we consider the following relaxed problem

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} \left( \left\| V^{(i)} \right\|_{*,r_i} + \lambda \left\| V^{(i)} \right\|_{l_1} \right) + \lambda \|\mathcal{H}\|_{l_1}, \\ \text{s.t.} \quad & P_\Omega \left( \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)} \right) - P_\Omega \left( \overline{\mathcal{X}} \right) = 0, \\ & \mathcal{H} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_m}, \ V^{(i)} \in \mathbb{R}^{n_i \times n_i}, \quad i \in [m]. \end{aligned} \tag{7}$$

To solve such a problem, we further introduce surrogate tensor variable $\mathcal{M}$ and matrices $W^{(i)}$ to rewrite the problem as:

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} \left( \left\| V^{(i)} \right\|_{*,r_i} + \lambda \left\| W^{(i)} \right\|_{l_1} \right) + \lambda \|\mathcal{H}\|_{l_1}, \\ \text{s.t.} \quad & P_\Omega(\mathcal{M} - \overline{\mathcal{X}}) = 0, \ \mathcal{M} = \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)}, \ W^{(i)} = V^{(i)}. \end{aligned} \tag{8}$$

By penalizing the constraint $\mathcal{M} = \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)}$ and $W^{(i)} = V^{(i)}$, we get the following problem

$$\begin{aligned} \min \quad & \sum_{i=1}^{m} \left( \mu \left\| V^{(i)} \right\|_{*,r_i} + \mu\lambda \left\| W^{(i)} \right\|_{l_1} + \tfrac{1}{2} \left\| W^{(i)} - V^{(i)} \right\|_F^2 \right) + \mu\lambda \|\mathcal{H}\|_{l_1} + \ell \left( \mathcal{H}, V^{(1)}, \ldots, V^{(m)}, \mathcal{M} \right), \\ \text{s.t.} \quad & P_\Omega(\mathcal{M} - \overline{\mathcal{X}}) = 0. \end{aligned} \tag{9}$$

Here $\mu > 0$ is a penalty parameter and

$$\ell(\mathcal{H}, V, \mathcal{M}) \triangleq \ell \left( \mathcal{H}, V^{(1)}, \ldots, V^{(m)}, \mathcal{M} \right) = \frac{1}{2} \left\| \mathcal{H} \times_1 V^{(1)} \times_2 \cdots \times_m V^{(m)} - \mathcal{M} \right\|_F^2.$$

## 4 | AN SBCD ALGORITHM

### 4.1 | Algorithm

In this subsection, we adopt an SBCD algorithm for solving (9). Before proceeding, we recall the following results, which are adopted to deal with the subproblems.

**Lemma 2** (37,38). *For a given matrix $Y \in \mathbb{R}^{m \times n}$ with $m \le n$, let its SVD be $Y = U_Y Diag(\boldsymbol{\gamma}) V_Y^\top$. For each vector $\boldsymbol{w} \in \mathbb{R}^m$ and a scalar $\alpha > 0$, let $s_\alpha(\boldsymbol{\gamma}, \boldsymbol{w}) = \max(0, \boldsymbol{\gamma} - \alpha \boldsymbol{w})$. Then*

$$X := S_v(Y, \boldsymbol{w}) = U_Y Diag \left( s_\alpha(\boldsymbol{\gamma}, \boldsymbol{w}) \right) V_Y^\top$$

*is an optimal solution of the problem*

$$\min_{X \in \mathbb{R}^{m \times n}} f(X) := \alpha \sum_{i=1}^{m} w_i \sigma_i(X) + \frac{1}{2} \|X - Y\|_F^2.$$

**Lemma 3** (39). *For a given vector $\boldsymbol{y} \in \mathbb{R}^m$ and a scalar $\alpha > 0$, there holds*

$$\boldsymbol{x} := T_\alpha(\boldsymbol{y}) = \operatorname{sign}(\boldsymbol{y}) \cdot \max(0, |\boldsymbol{y}| - \alpha)$$

*is an optimal solution of the problem*

$$\min_{\boldsymbol{x} \in \mathbb{R}^m} f(\boldsymbol{x}) := \alpha \|\boldsymbol{x}\|_{l_1} + \frac{1}{2} \|x - y\|_s^2.$$

Based on these results, we are ready to update $\left( \mathcal{M}^k, \mathcal{H}^k, V^{(1,k)}, \ldots, V^{(m,k)} \right)$ according to an SBCD method proposed in Reference 34. For convenience of notation, denote

$$V^{(j<i,k)} := \left( V^{(1,k)}, V^{(2,k)}, \ldots, V^{(i-1,k)} \right), \quad V^{(j \ge i,k)} := \left( V^{(i,k)}, \ldots, V^{(m,k)} \right).$$

Similarly, we denote $W^{(j<i,k)}$ and $W^{(j\geq i,k)}$.

For fixed matrix vector $\left(V^{(j<i,k)}, V^{(j\geq i,k-1)}\right)$ and tensor $\mathcal{M}^{k-1}$, core tensor $\mathcal{H}^{(i,k)}$ can be updated by solving the following problem

$$\min_{\mathcal{H}^{(i,k)}} \mu\lambda \left\|\mathcal{H}^{(i,k)}\right\|_{l_1} + \ell\left(\mathcal{H}^{(i,k)}, V^{(j<i,k)}, V^{(j\geq i,k-1)}, \mathcal{M}^{k-1}\right), \tag{10}$$

which is the classical LASSO problem with the variable $\mathcal{H}^{(i,k)}$. Although (10) is convex, there is no closed-form solution. To solve problem (10), we linearize the quadratic term of its objective function with an extrapolation point $\hat{H}^{(i,k)}$ as follows

$$\ell\left(\mathcal{H}^{(i,k)}, V^{(j<i,k)}, V^{(j\geq i,k-1)}, \mathcal{M}^{k-1}\right) \approx \ell\left(\hat{\mathcal{H}}^{(i,k)}, V^{(j<i,k)}, V^{(j\geq i,k-1)}, \mathcal{M}^{k-1}\right) + \left\langle \nabla_{\mathcal{H}}\ell, \mathcal{H}^{(i,k)} - \hat{\mathcal{H}}^{(i,k)}\right\rangle + \frac{L_{\mathcal{H}}^{(i,k)}}{2}\left\|\mathcal{H}^{(i,k)} - \hat{\mathcal{H}}^{(i,k)}\right\|_F^2. \tag{11}$$

Here $L_{\mathcal{H}}^{(i,k)} \geq L_{\mathcal{H}}(f)$ is a parameter with $L_{\mathcal{H}}(f)$ being the spectral radius of matrix

$$\left(\left(V^{(1,k)}\right)^T V^{(1,k)}\right) \otimes \cdots \otimes \left(\left(V^{(i-1,k)}\right)^T V^{(i-1,k)}\right) \otimes \left(\left(V^{(i,k-1)}\right)^T V^{(i,k-1)}\right) \otimes \cdots \otimes \left(\left(V^{(m,k-1)}\right)^T V^{(m,k-1)}\right)$$

and

$$\begin{aligned}\nabla_{\mathcal{H}}\ell &= \nabla_{\mathcal{H}}\ell\left(\hat{\mathcal{H}}^{(i,k)}, V^{(j<i,k)}, V^{(j\geq i,k-1)}, \mathcal{M}^{k-1}\right)\\ &= \left(\hat{\mathcal{H}}^{(i,k)} \times_1 V^{(1,k)} \times_2 \cdots \times_{i-1} V^{(i-1,k)} \times_i V^{(i,k-1)} \times_{i+1} \cdots \times_m V^{(m,k-1)} - \mathcal{M}^{k-1}\right)\\ &\quad \times_1 \left(V^{(1,k)}\right)^T \times_2 \cdots \times_{i-1} \left(V^{(i-1,k)}\right)^T \times_i \left(V^{(i,k-1)}\right)^T \times_{i+1} \cdots \times_m \left(V^{(m,k-1)}\right)^T.\end{aligned}$$

Then (10) can be relaxed as

$$\min_{\mathcal{H}^{(i,k)}} \mu\lambda \left\|\mathcal{H}^{(i,k)}\right\|_{l_1} + \left\langle \nabla_{\mathcal{H}}\ell, \mathcal{H}^{(i,k)} - \hat{\mathcal{H}}^{(i,k)}\right\rangle + \frac{L_{\mathcal{H}}^{(i,k)}}{2}\left\|\mathcal{H}^{(i,k)} - \hat{\mathcal{H}}^{(i,k)}\right\|_F^2. \tag{12}$$

By Lemma 3, we update core tensor $\mathcal{H}^{(i,k)}$ by

$$\begin{aligned}\mathcal{H}^{(i,k)} &= \underset{\mathcal{H}^{(i,k)}}{argmin}\ \mu\lambda\left\|\mathcal{H}^{(i,k)}\right\|_{l_1} + \left\langle \nabla_{\mathcal{H}}\ell, \mathcal{H}^{(i,k)} - \hat{\mathcal{H}}^{(i,k)}\right\rangle + \frac{L_{\mathcal{H}}^{(i,k)}}{2}\left\|\mathcal{H}^{(i,k)} - \hat{\mathcal{H}}^{(i,k)}\right\|_F^2\\ &= \underset{\mathcal{H}^{(i,k)}}{argmin}\ \mu\lambda\left\|\mathcal{H}^{(i,k)}\right\|_{l_1} + \frac{L_{\mathcal{H}}^{(i,k)}}{2}\left\|\mathcal{H}^{(i,k)} - \frac{L_{\mathcal{H}}^{(i,k)}\hat{\mathcal{H}}^{(i,k)} - \nabla_{\mathcal{H}}\ell}{L_{\mathcal{H}}^{(i,k)}}\right\|_F^2\\ &= T_{\frac{\mu\lambda}{L_{\mathcal{H}}^{(i,k)}}}\left(\frac{L_{\mathcal{H}}^{(i,k)}\hat{\mathcal{H}}^{(i,k)} - \nabla_{\mathcal{H}}\ell}{L_{\mathcal{H}}^{(i,k)}}\right).\end{aligned} \tag{13}$$

Here, we take

$$\begin{aligned}\hat{\mathcal{H}}^{(i,k)} &= \mathcal{H}^{(i,k)} + \omega_{\mathcal{H}}^{(i,k)}\left(\mathcal{H}^{(i-1,k)} - \mathcal{H}^{(i-2,k)}\right),\\ L_{\mathcal{H}}^{(i,k)} &= \left\|V^{(1,k)}\right\|_2^2 \times \cdots \times \left\|V^{(i-1,k)}\right\|_2^2 \times \left\|V^{(i,k-1)}\right\|_2^2 \times \cdots \times \left\|V^{(m,k-1)}\right\|_2^2 + 1,\end{aligned}$$

where

$$\omega_{\mathcal{H}}^{(i,k)} = \min\left(\hat{\omega}_{\mathcal{H}}^{(i,k)}, 0.9999\sqrt{\frac{L_{\mathcal{H}}^{(i-1,k)}}{L_{\mathcal{H}}^{(i,k)}}}\right), \quad \hat{\omega}_{\mathcal{H}}^{(i,k)} = \frac{t_{\mathcal{H}}^{(i-1,k)} - 1}{t_{\mathcal{H}}^{(i,k)}},$$

$$t_{\mathcal{H}}^{(0,1)} = 1,\ t_{\mathcal{H}}^{(0,k)} = t_{\mathcal{H}}^{(m,k-1)},\ \text{for } k \geq 2, \quad t_{\mathcal{H}}^{(i,k)} = \frac{1}{2}\left(1 + \sqrt{1 + 4\left(t_{\mathcal{H}}^{(i,k)}\right)^2}\right).$$

For fixed variable $V^{(i,k-1)}$, matrix $W^{(i,k)}$ is updated by

$$W^{(i,k)} := \underset{W^{(i,k)}}{argmin}\ \mu\lambda \left\|W^{(i,k)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k)} - V^{(i,k-1)}\right\|_F^2. \tag{14}$$

From Lemma 3, it holds

$$W^{(i,k)} = T_{\mu\lambda}\left(V^{(i,k-1)}\right). \tag{15}$$

For fixed variables $\mathcal{H}^{(i,k)}$, $W^{(i,k)}$, $V^{(j<i,k)}$, $V^{(j>i,k-1)}$ and tensor $\mathcal{M}^{k-1}$, factor matrix $V^{(i,k)}$ is updated by optimizing

$$\min_{V^{(i,k)}} \mu\left\|V^{(i,k)}\right\|_{*,r_i} + \ell\left(\mathcal{H}^{(i,k)}, V^{(j<i,k)}, V^{(j\geq i,k-1)}, \mathcal{M}^{k-1}\right) + \frac{1}{2}\left\|V^{(i,k)} - W^{(i,k)}\right\|_F^2 + \frac{\xi}{2}\left\|V^{(i,k)} - V^{(i,k-1)}\right\|_F^2, \tag{16}$$

where $\xi$ is a positive constant such that $\mu\left\|V^{(i,k)}\right\|_{*,r_i} + \frac{\xi}{2}\left\|V^{(i,k)} - V^{(i,k-1)}\right\|_F^2$ is convex. Let

$$v_s^{(i,k)} = \begin{cases} 0 & s = 1, \ldots, r_i, \\ \mu & s = r_i + 1, \ldots, n_i. \end{cases}$$

Then problem (16) can be written as

$$\min_{V^{(i,k)}} \sum_{s=1}^{n_i} v_s^{(i,k)}\sigma_s\left(V^{(i,k)}\right) + \frac{1}{2}\left\|V^{(i,k)}B_i^k - M_{(i)}^{k-1}\right\|_F^2 + \frac{1}{2}\left\|V^{(i,k)} - W^{(i,k)}\right\|_F^2 + \frac{\xi}{2}\left\|V^{(i,k)} - V^{(i,k-1)}\right\|_F^2, \tag{17}$$

where

$$B_i^k = H_{(i)}^{(i,k)}\left(V^{(m,k-1)} \otimes \cdots \otimes V^{(i+1,k-1)} \otimes V^{(i-1,k)} \otimes \cdots \otimes V^{(1,k)}\right)^T.$$

To get a closed-form approximate solution of $V^{(i,k)}$, we linearize the second term of (17) as follows

$$\frac{1}{2}\left\|V^{(i,k)}B_i^k - M_{(i)}^{k-1}\right\|_F^2 \approx \frac{1}{2}\left\|\hat{V}^{(i,k)}B_i^k - M_{(i)}^{k-1}\right\|_F^2 + \left\langle \nabla_{V^{(i)}}\ell, V^{(i,k)} - \hat{V}^{(i,k)}\right\rangle + \frac{L_i^k}{2}\left\|V^{(i,k)} - \hat{V}^{(i,k)}\right\|_F^2, \tag{18}$$

where

$$\begin{aligned} \hat{V}^{(i,k)} &= V^{(i,k-1)} + \omega_i^k\left(V^{(i,k-1)} - V^{(i,k-2)}\right), \\ L_i^k = \left\|B_i^k\left(B_i^k\right)^T\right\|_2 + 1, \quad \omega_i^k &= \min\left(\hat{\omega}^k, 0.9999\sqrt{\frac{L_i^{k-1}}{L_i^k}}\right), \\ \hat{\omega}^k = \frac{t^{k-1}-1}{t^k}, \quad t^0 = 1,\ t^k &= \frac{1}{2}\left(1 + \sqrt{1 + 4\left(t^{k-1}\right)^2}\right). \end{aligned}$$

By direct computation, it holds that

$$\begin{aligned} \nabla_{V^{(i)}}\ell &= \nabla_{V^{(i)}}\ell\left(\mathcal{H}^{(i,k)}, V^{(j<i,k)}, \hat{V}^{(i,k)}, V^{(j>i,k-1)}, \mathcal{M}^{k-1}\right) \\ &= \left(\mathcal{H}^{(i,k)} \times_1 V^{(1,k)} \times_2 \cdots \times_i \hat{V}^{(i,k)} \times_{i+1} V^{(i,k-1)} \times_{i+2} \cdots \times_m V^{(m,k-1)} - \mathcal{M}^{k-1}\right)_{(i)} \cdot \\ &\quad \left(\left(\mathcal{H}^{(i,k)} \times_1 V^{(1,k)} \times_2 \cdots \times_{i-1} V^{(i-1,k)} \times_{i+1} V^{(i+1,k-1)} \times_{i+2} \cdots \times_m V^{(m,k-1)}\right)_{(i)}\right)^T \\ &= \left(\hat{V}^{(i,k)}B_i^k - M_{(i)}^{k-1}\right) \cdot \left(B_i^k\right)^T. \end{aligned}$$

Plugging (18) into (17), by Lemma 2, we update factor matrix $V^{(i,k)}$ by

$$\begin{aligned} V^{(i,k)} &= \underset{V^{(i,k)}}{argmin} \sum_{s=1}^{n_i} v_s^{(i,k)} \sigma_s\left(V^{(i,k)}\right) + \left\langle \nabla_{V^{(i)}} \ell, V^{(i,k)} - \hat{V}^{(i,k)} \right\rangle \\ &\quad + \frac{L_i^k}{2}\left\|V^{(i,k)} - \hat{V}^{(i,k)}\right\|_F^2 + \frac{1}{2}\left\|V^{(i,k)} - W^{(i,k)}\right\|_F^2 + \frac{\xi}{2}\left\|V^{(i,k)} - V^{(i,k-1)}\right\|_F^2 \\ &= \underset{V^{(i,k)}}{argmin} \sum_{s=1}^{n_i} v_s^{(i,k)} \sigma_s\left(V^{(i,k)}\right) + \frac{L_i^k + \xi + 1}{2}\left\|V^{(i,k)} - \frac{L_i^k \hat{V}^{(i,k)} - \nabla_{V^{(i)}} \ell + W^{(i,k)} + \xi V^{(i,k-1)}}{L_i^k + \xi + 1}\right\|_F^2 \\ &= S_{\frac{1}{L_i^k + \xi + 1}}\left(\frac{L_i^k \hat{V}^{(i,k)} - \nabla_{V^{(i)}} \ell + W^{(i,k)} + \xi V^{(i,k-1)}}{L_i^k + \xi + 1}, \boldsymbol{v^{(i,k)}}\right). \end{aligned} \tag{19}$$

Finally, we fix variables $\mathcal{H}^k$ and $V^{(i,k)}$, $i \in [m]$, and then tensor $\mathcal{M}^k$ is updated by

$$\mathcal{M}^k = \underset{P_\Omega(\mathcal{M} - \overline{\mathcal{X}}) = 0}{\arg\min} \frac{1}{2\mu}\left\|\mathcal{H}^k \times_1 V^{(1,k)} \times_2 \cdots \times_m V^{(m,k)} - \mathcal{M}\right\|_F^2 = P_\Omega\left(\overline{\mathcal{X}}\right) + P_{\Omega^c}\left(\mathcal{H}^k \times_1 V^{(1,k)} \times_2 \cdots \times_m V^{(m,k)}\right). \tag{20}$$

Based on above analysis, our algorithm can be outlined as follows.

**Algorithm 1.** Low Tucker rank tensor completion (LTRTC)

**Input:** The tensor data $\overline{\mathcal{X}}$, the observed set $\Omega$, rank $r_i, i \in [m]$ and parameters $\lambda,\ \mu,\ \xi$.

**Initialize:** $\left(\mathcal{H}^{-1}, V^{(1,-1)}, \ldots, V^{(m,-1)}\right) = \left(\mathcal{H}^0, V^{(1,0)}, \ldots, V^{(m,0)}\right)$.

**While not converge do**

Let $\mathcal{H}^{(-1,1)} = \mathcal{H}^{(0,1)} = \mathcal{H}^0$ and $\mathcal{H}^{(-1,k)} = \mathcal{H}^{(m-1,k-1)}, \mathcal{H}^{(0,k)} = \mathcal{H}^{(m,k-1)}$ $(k \geq 2)$.

**For** $i = 1, \ldots, m$ **do**

**Step 1.** Compute $L_{\mathcal{H}}^{(i,k)}$ and set $\omega_{\mathcal{H}}^{(i,k)}$.

**Step 2.** Let $\hat{\mathcal{H}}^{(i,k)} = \mathcal{H}^{(i-1,k)} + \omega_{\mathcal{H}}^{(i,k)}\left(\mathcal{H}^{(i-1,k)} - \mathcal{H}^{(i-2,k)}\right)$.

**Step 3.** Update $\mathcal{H}^{(i,k)}$ according to (13).

**Step 4.** Update $W^{(i,k)}$ according to (15).

**Step 5.** Compute $L_i^k$ and set $\omega_i^k$.

**Step 6.** Let $\hat{V}^{(i,k)} = V^{(i,k-1)} + \omega_i^k\left(V^{(i,k-1)} - V^{(i,k-2)}\right)$.

**Step 7.** Update $V^{(i,k)}$ according to (19).

**end**

Let $\mathcal{H}^k = \mathcal{H}^{(m,k)}$.

Update $\mathcal{M}^k$ according to (20).

$k \leftarrow k + 1$.

**end while**

**Output:** $\left(\mathcal{H}^k, V^{(1,k)}, \ldots, V^{(m,k)}, W^{(1,k)}, \ldots, W^{(m,k)}, \mathcal{M}^k\right)$.

## 4.2 | Convergence analysis

Under the LTRTC algorithm framework, we establish the local convergence guarantee of Algorithm 1.

**Theorem 2.** *Suppose that the sequence* $\left\{\mathcal{H}^k, V^{(1,k)}, \ldots, V^{(m,k)}, W^{(1,k)}, \ldots, W^{(m,k)}, \mathcal{M}^k\right\}$ *generated by Algorithm 1 is bounded. Then any accumulation point of the sequence is a stationary point of problem (9).*

*Proof.* Since $\left\{\mathcal{H}^k, V^{(1,k)}, \dots, V^{(m,k)}, W^{(1,k)}, \dots, W^{(m,k)}, \mathcal{M}^k\right\}$ is bounded, there exist positive constants $L_d$, $L_u$ such that $L_{\mathcal{H}}^{(i,k)}, L_i^k \in [L_d, L_u]$. For convenience of notation, let

$$f\left(\mathcal{H}, V^{(1)}, \dots, V^{(m)}, W^{(1)}, \dots, W^{(m)}, \mathcal{M}\right) = \ell\left(\mathcal{H}, V^{(1)}, \cdots, V^{(m)}, \mathcal{M}\right) + \sum_{i=1}^{m}\left(\mu\left\|V^{(i)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i)} - V^{(i)}\right\|_F^2\right) + \mu\lambda\|\mathcal{H}\|_{l_1}.$$

For $j = 0, 1, \dots, m$, we further denote

$$\begin{aligned}
f_k &= \sum_{i=1}^{m}\left(\mu\left\|V^{(i,k)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k)} - V^{(i,k)}\right\|_F^2\right) + \mu\lambda\left\|\mathcal{H}^k\right\|_{l_1} + \ell\left(\mathcal{H}^k, V^{(1,k)}, \dots, V^{(m,k)}, \mathcal{M}^k\right),\\
h_j^{k+1} &= \sum_{i=1}^{j}\left(\mu\left\|V^{(i,k+1)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k+1)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k+1)} - V^{(i,k+1)}\right\|_F^2\right)\\
&\quad + \sum_{i=j+1}^{m}\left(\mu\left\|V^{(i,k)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k)} - V^{(i,k)}\right\|_F^2\right)\\
&\quad + \ell\left(\mathcal{H}^{(j,k+1)}, V^{(1,k+1)}, \dots, V^{(j,k+1)}, V^{(j+1,k)}, \dots, V^{(m,k)}, \mathcal{M}^k\right) + \mu\lambda\left\|\mathcal{H}^{(j,k+1)}\right\|_{l_1},\\
g_j^{k+1} &= \sum_{i=1}^{j}\left(\mu\left\|V^{(i,k+1)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k+1)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k+1)} - V^{(i,k+1)}\right\|_F^2\right)\\
&\quad + \sum_{i=j+1}^{m}\left(\mu\left\|V^{(i,k)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k)} - V^{(i,k)}\right\|_F^2\right)\\
&\quad + \ell\left(\mathcal{H}^{(j+1,k+1)}, V^{(1,k+1)}, \dots, V^{(j,k+1)}, V^{(j+1,k)}, \dots, V^{(m,k)}, \mathcal{M}^k\right) + \mu\lambda\left\|\mathcal{H}^{(j+1,k+1)}\right\|_{l_1},\\
w_j^{k+1} &= \sum_{i=1}^{j}\left(\mu\left\|V^{(i,k+1)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k+1)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k+1)} - V^{(i,k+1)}\right\|_F^2\right)\\
&\quad + \left(\mu\|V^{(j+1,k)}\|_{*,r_i} + \mu\lambda\left\|W^{(j+1,k+1)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(j+1,k+1)} - V^{(i,k)}\right\|_F^2\right)\\
&\quad + \sum_{i=j+2}^{m}\left(\mu\left\|V^{(i,k)}\right\|_{*,r_i} + \mu\lambda\left\|W^{(i,k)}\right\|_{l_1} + \frac{1}{2}\left\|W^{(i,k)} - V^{(i,k)}\right\|_F^2\right)\\
&\quad + \ell\left(\mathcal{H}^{(j+1,k+1)}, V^{(1,k+1)}, \dots, V^{(j,k+1)}, V^{(j+1,k)}, \dots, V^{(m,k)}, \mathcal{M}^k\right) + \mu\lambda\left\|\mathcal{H}^{(j+1,k+1)}\right\|_{l_1}.
\end{aligned}$$

Then

$$f_{k-1} - f_k = (f_{k-1} - g_0^k) + (g_0^k - w_0^k) + (w_0^k - h_1^k) + \sum_{j=1}^{m-1}\left(h_j^k - g_j^k + g_j^k - w_j^k + w_j^k - h_{j+1}^k\right) + (h_m^k - f_k).$$

Based on Lemma 2.1 in Reference 16 and steps 1–3 of Algorithm 1, we have that for all $j = 0, 1, \dots, m$,

$$\begin{aligned}
h_j^k - g_j^k &\geqslant \frac{L_{\mathcal{H}}^{(j+1,k)}}{2}\left\|\mathcal{H}^{(j+1,k)} - \hat{\mathcal{H}}^{(j+1,k)}\right\|_F^2 + L_{\mathcal{H}}^{(j+1,k)}\left\langle\hat{\mathcal{H}}^{(j+1,k)} - \mathcal{H}^{(j,k)}, \mathcal{H}^{(j+1,k)} - \hat{\mathcal{H}}^{(j+1,k)}\right\rangle\\
&= \frac{L_{\mathcal{H}}^{(j+1,k)}}{2}\left\|\mathcal{H}^{(j,k)} - \mathcal{H}^{(j+1,k)}\right\|_F^2 - \frac{L_{\mathcal{H}}^{(j+1,k)}\left(\omega_{\mathcal{H}}^{(j+1,k)}\right)^2}{2}\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2\\
&\geqslant \frac{L_{\mathcal{H}}^{(j+1,k)}}{2}\left\|\mathcal{H}^{(j,k)} - \mathcal{H}^{(j+1,k)}\right\|_F^2 - \frac{L_{\mathcal{H}}^{(j,k)}\delta_\omega^2}{2}\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2
\end{aligned}$$

and

$$f_{k-1} - g_0^k \geqslant \frac{L_{\mathcal{H}}^{(1,k)}}{2}\left\|\mathcal{H}^{(m,k-1)} - \mathcal{H}^{(1,k)}\right\|_F^2 - \frac{L_{\mathcal{H}}^{(m,k-1)}\delta_\omega^2}{2}\left\|\mathcal{H}^{(-1,k)} - \mathcal{H}^{(0,k)}\right\|_F^2.$$

Here $\delta_\omega = 0.9999$. Similarly, we have that for all $j = 0, 1, \ldots, m-1$,

$$w_j^k - h_{j+1}^k \geqslant \frac{L_{j+1}^k}{2}\left\|V^{(j+1,k-1)} - V^{(j+1,k)}\right\|_F^2 - \frac{L_{j+1}^{k-1}\delta_\omega^2}{2}\left\|V^{(j+1,k-1)} - V^{(j+1,k-2)}\right\|_F^2.$$

Together with the fact that $g_j^k - w_j^k \geq 0$ for all $j = 0, 1, \ldots, m-1$ and $h_m^k \geq f_k$, we have that

$$\begin{aligned}
f_{k-1} - f_k \geqslant & \frac{L_{\mathcal{H}}^{(1,k)}}{2}\left\|\mathcal{H}^{(m,k-1)} - \mathcal{H}^{(1,k)}\right\|_F^2 - \frac{L_{\mathcal{H}}^{(m,k-1)}\delta_\omega^2}{2}\left\|\mathcal{H}^{(-1,k)} - \mathcal{H}^{(0,k)}\right\|_F^2 \\
& + \sum_{j=1}^{m-1}\left(\frac{L_{\mathcal{H}}^{(j+1,k)}}{2}\left\|\mathcal{H}^{(j,k)} - \mathcal{H}^{(j+1,k)}\right\|_F^2 - \frac{L_{\mathcal{H}}^{(j,k)}\delta_\omega^2}{2}\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2\right) \\
& + \sum_{j=0}^{m-1}\left(\frac{L_{j+1}^k}{2}\left\|V^{(j+1,k-1)} - V^{(j+1,k)}\right\|_F^2 - \frac{L_{j+1}^{k-1}\delta_\omega^2}{2}\left\|V^{(j+1,k-1)} - V^{(j+1,k-2)}\right\|_F^2\right) \\
= & \frac{L_{\mathcal{H}}^{(m,k)}}{2}\left\|\mathcal{H}^{(m-1,k)} - \mathcal{H}^{(m,k)}\right\|_F^2 - \frac{L_{\mathcal{H}}^{(m,k-1)}\delta_\omega^2}{2}\left\|\mathcal{H}^{(m-1,k-1)} - \mathcal{H}^{(m,k-1)}\right\|_F^2 + \sum_{j=1}^{m-1}\frac{L_{\mathcal{H}}^{(j,k)}\left(1-\delta_\omega^2\right)}{2}\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2 \\
& + \sum_{j=1}^{m}\left(\frac{L_j^k}{2}\left\|V^{(j,k-1)} - V^{(j,k)}\right\|_F^2 - \frac{L_j^{k-1}\delta_\omega^2}{2}\left\|V^{(j,k-1)} - V^{(j,k-2)}\right\|_F^2\right).
\end{aligned}$$

Summing up the above inequality over $k = 0, 1, \ldots, K$, it gives

$$\begin{aligned}
f_0 - f_K & \geqslant \sum_{k=1}^{K}\sum_{j=1}^{m}\left(\frac{L_{\mathcal{H}}^{(j,k)}\left(1-\delta_\omega^2\right)}{2}\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2 + \frac{L_j^k\left(1-\delta_\omega^2\right)}{2}\left\|V^{(j,k-1)} - V^{(j,k)}\right\|_F^2\right) \\
& \geqslant \frac{\left(1-\delta_\omega^2\right)L_d}{2}\sum_{k=1}^{K}\sum_{j=1}^{m}\left(\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2 + \left\|V^{(j,k-1)} - V^{(j,k)}\right\|_F^2\right).
\end{aligned}$$

Since $f_k \geq 0$ is lower bounded, $\sum_{k=1}^{\infty}(f_k - f_{k+1}) = f_0 - \lim_{k\to\infty} f_k$ is bounded and hence

$$\sum_{k=1}^{\infty}\sum_{j=1}^{m}\left(\left\|\mathcal{H}^{(j-1,k)} - \mathcal{H}^{(j,k)}\right\|_F^2 + \left\|V^{(j,k-1)} - V^{(j,k)}\right\|_F^2\right) < \infty. \tag{21}$$

So we can assert that

$$\lim_{k\to\infty}\left\|V^{(i,k)} - V^{(i,k-1)}\right\|_F = \lim_{k\to\infty}\left\|\mathcal{H}^{(i-1,k)} - \mathcal{H}^{(i,k)}\right\|_F = 0, \quad \forall i \in [m].$$

This means that $\mathcal{H}^k - \mathcal{H}^{k+1} \to 0$ when $k \to \infty$. From (15) and the non-expansive property of shrinkage operator $T_{\mu\lambda}(\cdot)$, we have that

$$\lim_{k\to\infty}\left(W^{(i,k)} - W^{(i,k-1)}\right) = 0, \quad i \in [m].$$

Let $\left\{\mathcal{H}^{k_j}, V^{(1,k_j)}, \ldots, V^{(m,k_j)}, W^{(1,k_j)}, \ldots, W^{(m,k_j)}, \mathcal{M}^{k_j}\right\}$ be a subsequence converging to $\left(\overline{\mathcal{H}}, \overline{V}^{(1)}, \ldots, \overline{V}^{(m)}, \overline{W}^{(1)}, \ldots, \overline{W}^{(m)}, \overline{\mathcal{M}}\right)$. From $\left\|\mathcal{H}^{(i-1,k)} - \mathcal{H}^{(i,k)}\right\|_F \to 0$ and $\left\|V^{(i,k-1)} - V^{(i,k)}\right\|_F \to 0$, we have that

$$\hat{\mathcal{H}}^{(i,k_j+1)},\ \mathcal{H}^{(i,k_j+1)} \to \overline{\mathcal{H}}, \quad \hat{V}^{(i,k_j+1)},\ V^{(i,k_j+1)} \to \overline{V}^{(i)}, \quad \forall i \in [m].$$

From the choosing of $\mathcal{H}^{(i,k_j+1)}$, it follows that

$$\mu\lambda\partial_{\mathcal{H}}\left\|\mathcal{H}^{(i,k_j+1)}\right\|_{l_1}+\nabla_{\mathcal{H}}l\left(\mathcal{H}^{(i,k_j+1)},V^{(1,k_j)},\ldots,V^{(m,k_j)}\right)+L_{\mathcal{H}}^{(i,k)}\left(\mathcal{H}^{(i,k_j+1)}-\hat{\mathcal{H}}^{(i,k_j+1)}\right)=0.$$

When $j\to\infty$, there holds

$$\mu\lambda\partial_{\mathcal{H}}\left\|\overline{\mathcal{H}}\right\|_{l_1}+\nabla_{\mathcal{H}}l\left(\overline{\mathcal{H}},\overline{V}^{(1)},\ldots,\overline{V}^{(m)}\right)=0. \tag{22}$$

By (14), we have

$$\mu\lambda\partial_{W^{(i)}}\left\|W^{(i,k_j+1)}\right\|_{l_1}+\left(W^{(i,k_j+1)}-V^{(i,k_j)}\right)=0.$$

When $j\to\infty$, there holds

$$\mu\lambda\partial_{W^{(i)}}\left\|\overline{W}^{(i)}\right\|_{l_1}+\left(\overline{W}^{(i)}-\overline{V}^{(i)}\right)=0. \tag{23}$$

By (19), we have

$$\begin{aligned}&\mu\partial_{V^{(i)}}\left\|V^{(i,k_j+1)}\right\|_{*,r_i}+\nabla_{V^{(i)}}l\left(\mathcal{H}^{(i,k_j+1)},V^{(1,k_j+1)},\ldots,V^{(i,k_j+1)},V^{(i+1,k_j)},\ldots,V^{(m,k_j)}\right)\\&+L_i^{k_j+1}\left(\hat{V}^{(i,k_j+1)}-V^{(i,k_j+1)}\right)+\left(V^{(i,k_j+1)}-W^{(i,k_j+1)}\right)+\xi\left(V^{(i,k_j+1)}-V^{(i,k_j)}\right)=0.\end{aligned}$$

When $j\to\infty$, there holds

$$\mu\partial_{V^{(i)}}\left\|\overline{V}^{(i)}\right\|_{*,r_i}+\nabla_{V^{(i)}}l\left(\overline{\mathcal{H}},\overline{V}^{(1)},\ldots,\overline{V}^{(m)}\right)+\left(\overline{V}^{(i)}-\overline{W}^{(i)}\right)=0. \tag{24}$$

By (20), we have

$$\mathcal{M}^{k_j+1}=P_{\Omega}\left(\overline{\mathcal{X}}\right)+P_{\Omega^c}\left(\mathcal{H}^{k_j}\times_1V^{(1,k_j)}\times_2\cdots\times_mV^{(m,k_j)}\right).$$

When $j\to\infty$, there holds

$$\overline{\mathcal{M}}=P_{\Omega}\left(\overline{\mathcal{X}}\right)+P_{\Omega^c}\left(\overline{\mathcal{H}}\times_1\overline{V}^{(1)}\times_2\cdots\times_m\overline{V}^{(m)}\right). \tag{25}$$

Equalities (22)–(25) give the first-order optimality conditions of (9). That is, $\left(\overline{\mathcal{H}},\overline{V}^{(1)},\ldots,\overline{V}^{(m)},\overline{W}^{(1)},\ldots,\overline{W}^{(m)},\overline{\mathcal{M}}\right)$ is a stationary point. ■

## 4.3 | Improvement with temporal characteristics

In the real world, some characteristics are included in the related tensor data. For example, time stability exists between two adjacent frames in the video tensor data. So we consider to add $\|\mathcal{X}\times_iQ^{(i)}\|_F^2$ to the objective function of (9). Here

$$Q^{(i)}=\begin{pmatrix}1&-1&0&\cdots&0&0\\-0.5&1&-0.5&\cdots&0&0\\0&-0.5&1&\cdots&0&0\\\vdots&\vdots&\vdots&\ddots&\vdots&\vdots\\0&0&0&\cdots&1&-0.5\\0&0&0&\cdots&-1&1\end{pmatrix}_{n_i\times n_i}.$$

Motivated by these observations, matrices $Q^{(i)}$ are introduced to characterize the $i$th characteristics for each $i \in [m]$ in the related tensor data. The temporal constraint matrix $Q^{(i)}$ captures temporal stability feature of the tensor, that is, the data is similar at adjacent time slots.

Note that

$$\begin{aligned}\left\|\mathcal{X} \times_i Q^{(i)}\right\|_F &= \left\|\mathcal{H} \times_1 V^{(1)} \times \cdots \times_i \left(Q^{(i)} V^{(i)}\right) \times_{i+1} \cdots \times_m V^{(m)}\right\|_F \\ &= \left\|\left(Q^{(i)} V^{(i)}\right)\left(\mathcal{H} \times_1 V^{(1)} \times \cdots \times_{i-1} V^{(i-1)} \times_{i+1} \cdots \times_m V^{(m)}\right)_{(i)}\right\|_F \\ &\le \left\|Q^{(i)} V^{(i)}\right\|_F \left\|\mathcal{H} \times_1 V^{(1)} \times \cdots \times_{i-1} V^{(i-1)} \times_{i+1} \cdots \times_m V^{(m)}\right\|_F \\ &\le \left\|Q^{(i)} V^{(i)}\right\|_F \|\mathcal{H}\|_F \prod_{j \ne i} \left\|V^{(j)}\right\|_F \le c \left\|Q^{(i)} V^{(i)}\right\|_F \|\mathcal{H}\|_{l_1} \prod_{j \ne i} \left\|V^{(j)}\right\|_{l_1},\end{aligned}$$

where $c$ is a positive constant depending on $\|\cdot\|_{l_1}$ and $\|\cdot\|_F$. Since $\|\mathcal{H}\|_{l_1}$ and $\|V^{(j)}\|_{l_1}$ appear in the objective function, it suffices to adopt $\|Q^{(i)}V^{(i)}\|_F$ to measure $\|\mathcal{X} \times_i Q^{(i)}\|_F$ to lower the computational cost. Hence problem (9) can be improved as follows

$$\begin{aligned} \min \quad & \sum_{i=1}^m \left(\mu \left\|V^{(i)}\right\|_{*,r_i} + \mu\lambda \left\|V^{(i)}\right\|_{l_1} + \frac{\beta_i}{2}\left\|Q^{(i)}V^{(i)}\right\|_F^2\right) + \mu\lambda\|\mathcal{H}\|_{l_1} + \ell\left(\mathcal{H}, V^{(1)}, \ldots, V^{(m)}, \mathcal{M}\right), \\ \text{s.t.} \quad & P_\Omega(\mathcal{M} - \overline{\mathcal{X}}) = 0. \end{aligned} \tag{26}$$

Let $\beta_i = 0$ if there is no additional characterization on the $i$th slice of $\mathcal{X}$. Clearly, model (9) can be regarded as a special case of model (26). Obviously, $\mathcal{H}$, $W^{(i)}$ and $\mathcal{M}$ can be updated by (13), (15), and (20), respectively. Hence it suffices to update $V^{(i,k)}$ for $i \in [m]$. To this end, we consider the following problem

$$\min_{V^{(i,k)}} \quad \mu \left\|V^{(i,k)}\right\|_{*,r_i} + \ell\left(\mathcal{H}^{(i,k)}, V^{(j<i,k)}, V^{(j\ge i,k-1)}, \mathcal{M}^{k-1}\right) + \frac{\xi}{2}\left\|V^{(i,k)} - V^{(i,k-1)}\right\|_F^2 + \frac{\beta_i}{2}\left\|Q^{(i)}V^{(i,k)}\right\|_F^2. \tag{27}$$

For simplicity, let $\ell_1(\mathcal{H}, V, \mathcal{M}) = \ell(\mathcal{H}, V, \mathcal{M}) + \frac{\beta_i}{2}\left\|Q^{(i)}V^{(i)}\right\|_F^2$ to update $V^{(i,k)}$. By substituting $\ell$ as $\ell_1$ in Algorithm 1, we have the modified algorithm, denoted by T-LTRTC, similar to LTRTC and hence we omit the details here.

*Remark* 1. In the experiment, we found that the fixed $\beta_i$ is not very effective. To improve the efficiency, we update $\beta_i$ by $\beta_i^k = \max(\beta_i^0 \rho^k, \tau_i)$, where $\rho < 1$ and $\tau_i$ is set to be a given lower bound of $\beta_i^k$. So there is $k_0$ so that $\beta_i^k = \tau_i$ when $k \ge k_0$.

## 5 | NUMERICAL EXPERIMENTS

In this section, we adopt the relative error and the peak signal-to-noise ratio (PSNR) as evaluation metrics, defined by

$$rel.\ err := \frac{\left\|\mathcal{M}_{\text{opt}} - \overline{\mathcal{X}}\right\|_F}{\left\|\overline{\mathcal{X}}\right\|_F}, \qquad \text{PSNR} := 10\log_{10}\left(\frac{n_1 \cdots n_m \left\|\overline{\mathcal{X}}\right\|_\infty^2}{\left\|\overline{\mathcal{X}} - \mathcal{M}_{\text{opt}}\right\|_F^2}\right).$$

The algorithm is terminated whenever $\left\|P_\Omega\left(\mathcal{M}_{\text{opt}} - \overline{\mathcal{X}}\right)\right\|_F / \left\|P_\Omega(\overline{\mathcal{X}})\right\|_F \le \varepsilon$ for three iterations in a row. We conduct extensive experiments to evaluate our methods, and then compare them with some existing methods, including TMac,[33] NTD,[16] and TCTF.[40] All the methods are implemented on the platform of Windows 10 and Matlab (R2014a) with an Intel(R) Core(TM) i7-7700 CPU at 3.60 GHz and 8 GB RAM.

### 5.1 | Numerical simulation

In this subsection, a tensor $\overline{\mathcal{X}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is generated randomly with $\text{rank}_{tc}(\overline{\mathcal{X}}) = (r, r, r)$ following Tucker decomposition.[26,36] For this aim, we first generate a core tensor $\mathcal{G} \in \mathbb{R}^{r \times r \times r}$ with i.i.d. standard Gaussian entries. Then, we generate

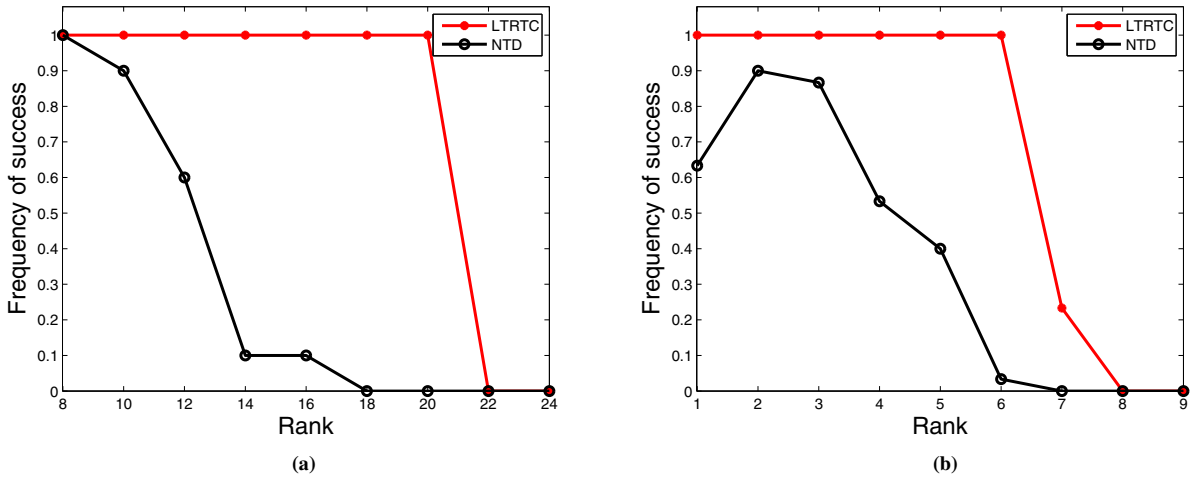


**FIGURE 1** Comparison on random data with $p = 0.5$. (a) The tensor size is $100 \times 100 \times 100$. (b) The tensor size is $80 \times 60 \times 40$

matrices $U^{(1)}$, $U^{(2)}$, $U^{(3)}$, where $U^{(i)} \in \mathbb{R}^{n_i \times r}$ with i.i.d. standard Gaussian entries. Next, we set

$$\overline{\mathcal{X}} := \mathcal{G} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}.$$

Finally, we uniformly select $pn_1 \cdots n_m$ positions of $\overline{\mathcal{X}}$ to construct $\Omega$, where $p$ is the sampling ratio. If the relative error is smaller than 1e−3, $\mathcal{M}_{\text{opt}}$ is regarded as a successful recovery of $\overline{\mathcal{X}}$. For each simulation, relative error is obtained via 30 Monte Carlo runs with different realizations of $\overline{\mathcal{X}}$ and $\Omega$. In experiments, we set the maximum iteration steps of all algorithms to 1000 steps and the termination precision is set to be 1e−6. In NTD and LTRTC, we set the initial Tucker rank to be $(r, r, r)$ in Figure 1a and $(4r, 3r, 2r)$ in Figure 1b. We set $\lambda = 0.01,\ \mu = 0.1,\ \xi = 1$ in LTRTC. For balance, the nonnegativity of NTD is missing thereafter.

From Figure 1, we can see that as the rank increases, the success ratio of the two proposed algorithms gradually decreases. However, in the case of the same rank, the success ratio of the LTRTC algorithm is higher than that of the NTD, and the advantages of LTRTC are clear when the tensor dimensions are larger.

## 5.2 | Image simulation

In this subsection, we apply LTRTC to color image inpainting. Note that color images can be expressed as third-order tensors. When the tensor data is low rank or numerical low rank, the image inpainting problem can be modeled as a low rank tensor completion problem. As shown in Figure 2, the information of an image example with low rank structure and the mode-$i$ matrix is controlled by the top 40, 40, and 3 singular values, respectively. Therefore, in NTD, TMac, LTRTC and T-LTRTC, we set the initial Tucker rank to be (40, 40, 3), and the initial tubal rank is set to be (40, 40, 40) in TCTF. We set $\lambda = 5,\ \mu = 0.2,\ \xi = 1$ in LTRTC and T-LTRTC, $\rho = 0.99$, $\beta_i = 25,\ \tau_i = 5$ for $i = 1, 2$ and $\beta_3 = 0,\ \tau_3 = 0$ in T-LTRTC. We consider the case where entries are missing randomly of sampling ratio $p = 0.3$ and the termination precision is set to be 1e−5.

### 5.2.1 | The Berkeley Segmentation Database

In our test, four pictures of "Airplane", "Church", "Woman", and "Children" are selected.1 The pixels of "Airplane" and "Churches" are $321 \times 481 \times 3$, and the pixels of "Women" and "Children" are $481 \times 321 \times 3$, respectively. We set the maximum iteration steps in all algorithms to be 300.

We can see from Figure 3 that TCTF fails to recover the image, the recovered image of NTD and TMac still have visible reconstruction errors but LTRTC and T-LTRTC can successfully recover the image. The recovered image of T-LTRTC can be seen to be slightly clear than LTRTC. To compare the recovery effects of the five methods further, we provide the rel.err and PSNR values for each method in Table 1. It is clear that T-LTRTC achieves the best results.

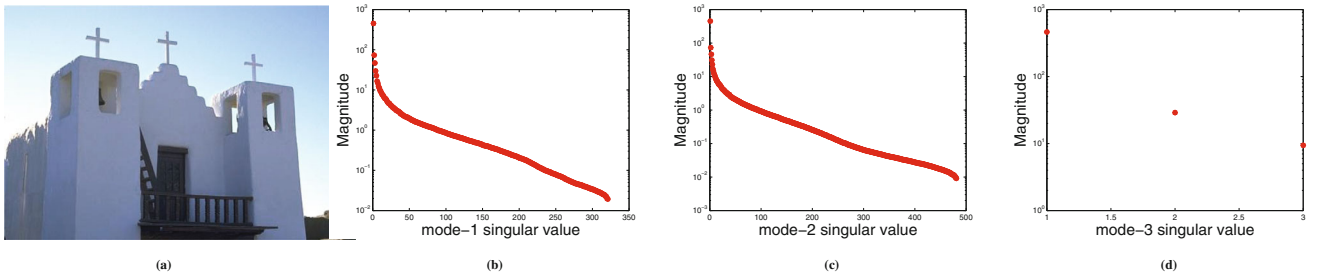


**FIGURE 2** (a) Original image. (b) The singular values of mode-1. (c) The singular values of mode-2. (d) The singular values of mode-3


**FIGURE 3** Completion results of The Berkeley Segmentation Database. Best viewed in ×2 sized color pdf file. (a) Original, (b) observed, (c) T-LTRTC, (d) LTRTC, (e) NTD, (f) TMac, and (g) TCTF

**TABLE 1** Numerical results for The Berkeley Segmentation Database

| | Airplane | | Church | | Woman | | Children | |
|---|---|---|---|---|---|---|---|---|
| **Method** | **PSNR** | **rel.err** | **PSNR** | **rel.err** | **PSNR** | **rel.err** | **PSNR** | **rel.err** |
| T-LTRTC | **30.3** | 8.75e−02 | **35.5** | 2.46e−02 | **31.3** | 7.28e−02 | **36.9** | 4.41e−02 |
| LTRTC | 28.2 | 1.12e−01 | 33.2 | 3.22e−02 | 29.4 | 9.05e−02 | 34.0 | 6.18e−02 |
| NTD | 25.3 | 1.56e−01 | 27.2 | 6.41e−02 | 25.6 | 1.41e−01 | 28.4 | 1.17e−01 |
| TMac | 25.7 | 1.48e−01 | 28.4 | 5.61e−02 | 27.7 | 1.10e−01 | 25.0 | 1.72e−01 |
| TCTF | 20.6 | 2.68e−01 | 22.4 | 1.11e−01 | 17.0 | 3.78e−01 | 19.8 | 3.16e−01 |

*Note*: The boldface number is the best.

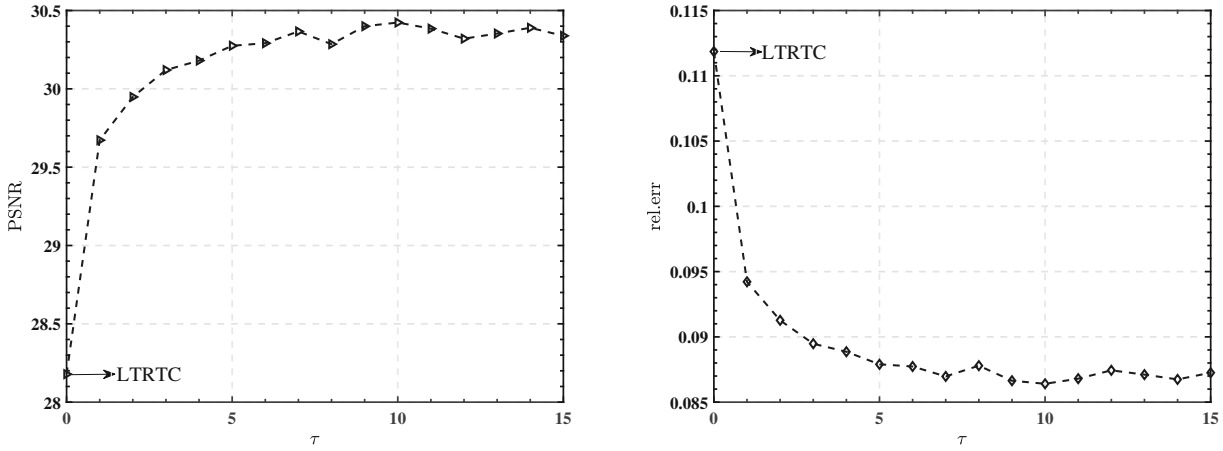


**FIGURE 4** PSNR and rel.err values with respect to different values of $\tau$ ("Airplane")


**FIGURE 5** Completion results of face images. Best viewed in ×2 sized color pdf file. (a) Original, (b) observed, (c) T-LTRTC, (d) LTRTC, (e) NTD, (f) TMac, and (g) TCTF

To illustrate the performance of T-LTRTC, Figure 4 reports the quantitative metrics against different $\tau$ ($\tau_1$, $\tau_2$). As $\tau$ increases, the effect is getting better and better, but when $\tau > 7$, the result fluctuates slightly around the optimal value. This is reasonable since larger $\tau$ captures more temporal stability information. However, pictures have limited temporal stability information, so the results fluctuate around an optimal value as $\tau$ continues to grow.

### 5.2.2 | California Institute of Technology Color Face Image Library

In our test, we consider two face pictures, each with three scenes, and their pixel size is $592 \times 896$.[2] From Figure 5, we can see that the face pictures recovered by NTD and TCTF carry a lot of noise, and the pictures recovered by TMac are

**TABLE 2** Numerical results for face images

| Image | T-LTRTC | | LTRTC | | NTD | | TMac | | TCTF | |
|---|---|---|---|---|---|---|---|---|---|---|
| | PSNR | rel.err | PSNR | rel.err | PSNR | rel.err | PSNR | rel.err | PSNR | rel.err |
| Face1 | **32.4** | 3.55e−02 | 29.9 | 4.72e−02 | 23.7 | 9.60e−02 | 25.9 | 7.47e−02 | 23.7 | 9.66e−02 |
| Face2 | **32.4** | 3.53e−02 | 29.8 | 4.77e−02 | 24.2 | 9.01e−02 | 25.6 | 7.70e−02 | 25.1 | 8.21e−02 |
| Face3 | **30.5** | 4.49e−02 | 28.2 | 5.90e−02 | 22.9 | 1.08e−01 | 24.6 | 8.89e−02 | 22.5 | 1.13e−01 |
| Face4 | **31.8** | 5.11e−02 | 30.4 | 6.01e−02 | 26.3 | 9.60e−02 | 26.0 | 9.92e−02 | 25.3 | 1.08e−01 |
| Face5 | **30.8** | 5.85e−02 | 29.5 | 6.79e−02 | 25.9 | 1.03e−01 | 25.5 | 1.08e−01 | 24.9 | 1.15e−01 |
| Face6 | **33.9** | 2.77e−02 | 32.4 | 3.33e−02 | 23.5 | 9.26e−02 | 27.3 | 5.98e−02 | 28.6 | 5.11e−02 |

*Note*: The boldface number is the best.

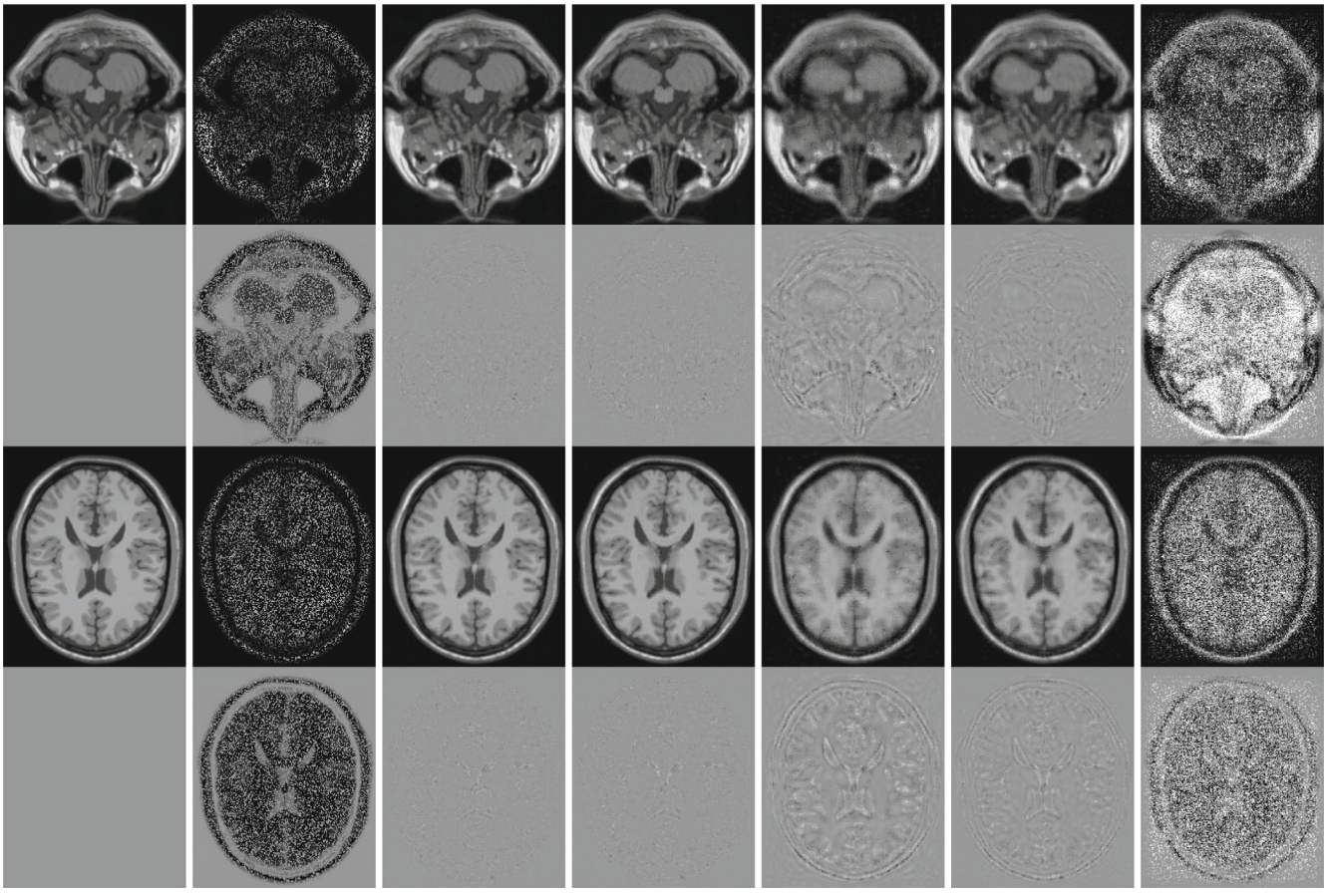


**FIGURE 6** Completion results of The MRI Volume Dataset. (a) Original, (b) observed, (c) T-LTRTC, (d) LTRTC, (e) NTD, (f) TMac, and (g) TCTF

not brightly colored. LTRTC recovers face images that are not only low noise but also colorful, and T-LTRTC is slightly clearer than LTRTC, particularly at face edges. In Table 2, it is clear that T-LTCTR outperforms other methods.

## 5.3 | MRI simulation

The resolution of the MRI volume dataset[3] is of size $217 \times 181$ with 181 slices and we pick the first 100 slices. In NTD, TMac, LTRTC and T-LTRTC, we set the initial Tucker rank to be $(40, 40, 20)$, and the initial tubal rank is set to be

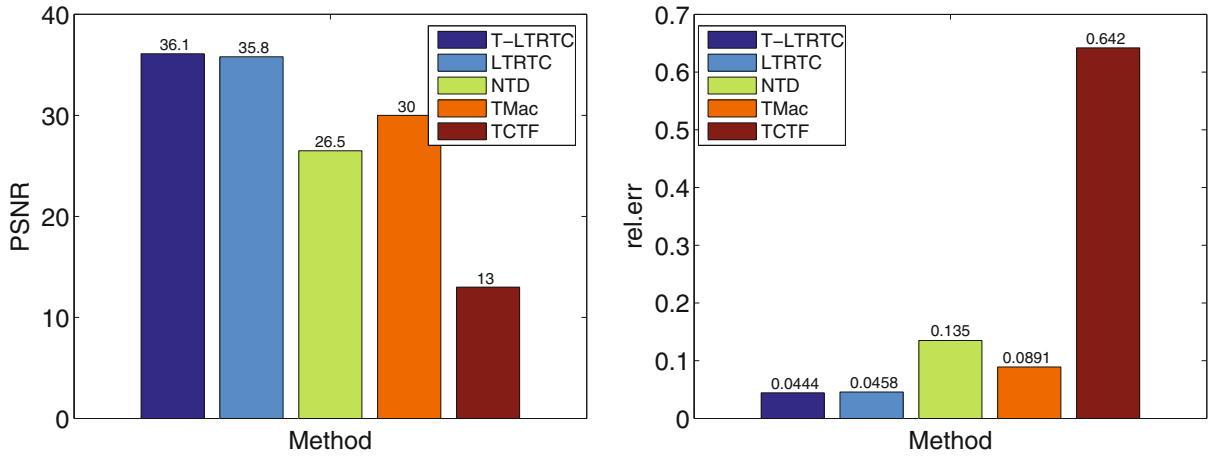


**FIGURE 7** Histogram of representation results for the MRI Volume Dataset


**FIGURE 8** Uniformly sampled video inpainting. (a) Original, (b) observed, (c) T-LTRTC, (d) LTRTC, (e) NTD, and (f) TMac

(40, … , 40) in TCTF. We set $\lambda = 1,\ \mu = 0.2,\ \xi = 1$ in LTRTC and T-LTRTC, $\rho = 0.99$, $\beta_i = 25,\ \tau_i = 9$ for $i = 1, 2$ and $\beta_3 = 0,\ \tau_i = 0$ in T-LTRTC. The maximum iteration steps of all algorithms are set to be 500. The 18th slice and the 88th slice are displayed in Figure 6. From the recovery results, T-LTRTC outperforms with more details. On the PSNR metric, T-LTRTC also achieves the best, consistent with the observation in Figure 6. At the same time, we present the resulted images of the restored images minuend the original pictures. For better visualization, we add 0.5 to the pixel. It is clear that the images corresponding to T-LTRTC and LTRTC have almost no outline of the original image, indicating the best recovery effect. From Figure 7, the effect of these five methods can be ordered as T-LTRTC, LTRTC, TMac, NTD, and TCTF.

## 5.4 | Video simulation

We evaluate our proposed methods LTRTC and T-LTRTC on the widely used YUV Video Sequences.[4] Each sequence contains at least 150 frames and we pick the first 30 frames. In the experiments, we test our proposed methods and other

**TABLE 3** Numerical results for video inpainting

| | T-LTRTC | | LTRTC | | NTD | | TMac | |
|---|---|---|---|---|---|---|---|---|
| **Video** | **PSNR** | **rel.err** | **PSNR** | **rel.err** | **PSNR** | **rel.err** | **PSNR** | **rel.err** |
| Suzie | **35.1** | 3.86e−02 | 34.3 | 4.26e−02 | 28.6 | 8.22e−02 | 29.3 | 7.53e−02 |
| News | **35.0** | 4.76e−02 | 34.6 | 5.00e−02 | 26.3 | 1.30e−01 | 28.3 | 1.03e−01 |
| Carphone | **32.9** | 4.80e−02 | 32.5 | 5.06e−02 | 27.4 | 9.03e−02 | 28.9 | 7.60e−02 |

*Note*: The boldface number is the best.


**FIGURE 9** Recovered video of lost frames. From top to bottom, the data of the 8th frame, the 8th–9th frame, the 7th–9th frame, and the 6th–10th frame are lost. (a) Observed, (b) T-LTRTC, (c) LTRTC, (d) NTD, and (e) TMac

methods on three videos. For each frame, the sizes of the three videos are all of $144 \times 176$ pixels. In NTD, TMac, LTRTC, and T-LTRTC, we set the initial Tucker rank to be $(30, 30, 3, 5)$. We set $\lambda = 1,\ \mu = 0.2,\ \xi = 1$ in LTRTC and T-LTRTC, $\rho = 0.99$, $\beta_i = 25,\ \tau_i = 1$ for $i = 1, 2$, $\beta_3 = 0,\ \tau_3 = 0$ and $\beta_4 = 100,\ \tau_4 = 10$ in T-LTRTC. The maximum iteration steps of all algorithms are set to be 1000 and the termination precision is set to be 1e−5. We consider the cases where entries are missing at random of sampling ratio $p = 0.1$.

In Figure 8, the 18th frame of the three tested videos are presented. For "Suzie" video, there are a lot of noise on face by NTD and the color of the video cannot be recovered by TMac. For "News" video, there are a lot of noise on the stage columns and ladies' clothes by NTD and TMac. For all tested videos, T-LTRTC and LTRTC recover them better than the others.

From Table 3, it is asserted that the recovery of the four algorithms on videos can be ordered as T-LTRTC, LTRTC, TMac and NTD. The video inpainting results are consistent with the image inpainting results. All these demonstrate that our proposed algorithms perform better.

**TABLE 4** Numerical results for the masked video inpainting

| | T-LTRTC | | LTRTC | | NTD | | TMac | |
|---|---|---|---|---|---|---|---|---|
| **Lost frame** | **PSNR** | **rel.err** | **PSNR** | **rel.err** | **PSNR** | **rel.err** | **PSNR** | **rel.err** |
| 8 | **59.8** | 2.27e−03 | 33.9 | 4.47e−02 | 30.1 | 6.93e−02 | 26.5 | 1.04e−01 |
| 8–9 | **55.1** | 3.88e−03 | 30.9 | 6.27e−02 | 26.6 | 1.03e−01 | 23.7 | 1.45e−01 |
| 7–9 | **50.4** | 6.64e−03 | 27.3 | 9.56e−02 | 24.3 | 1.35e−01 | 21.0 | 1.97e−01 |
| 6–10 | **35.9** | 3.55e−02 | 26.0 | 1.10e−02 | 23.3 | 1.52e−01 | 19.4 | 2.37e−01 |

*Note*: The boldface number is the best.

In real life, there may be a basic lack of data in a certain frame of the video, and worse, the basic lack of data in several consecutive frames. In order to check the perform of our proposed algorithms on this situation, we lose the data of the 8th frame, the 8-9th frame, the 7-9th frame, the 6th-10th frame, and we uniformly select 0.1% of the lost data the samples and other data points without missing frames are known.

From Figure 9 and Table 4, we can see that when only one frame of video data is missing, the T-LTRTC method can basically recover the missing data completely. However, the recovery effects of LTRTC, NTD, and TMac methods are not ideal. When several frames of video data are continuously missing, the recovery effect of T-LTRTC is also ideal. Therefore, our T-LTRTC model has a good effect on the recovery of missing data in continuous time periods.

## 6 | CONCLUSIONS AND FUTURE WORK

In this work, we established a relationship between the Tucker ranks and the ranks of the factor matrices in Tucker decomposition. Then, we reformulated the low Tucker rank tensor completion problem as a multilinear low rank matrix completion problem. An SBCD method was proposed for the reformulated problem. Furthermore, temporal characteristics in image and video data were introduced to such a model, which benefits the performance of the method. The experimental results demonstrated that our proposed models and methods led to impressive improvements over state-of-the-art methods.

In this work, a multilinear low rank matrix completion model with general sparsity on core tensor and factor matrices is introduced. However, such sparsity may be over relaxed. Hence, how to choose the sparse measure to fit the problem is our future focus.

### FUNDING INFORMATION

Xinzhen Zhang was partly supported by the National Natural Science Foundation of China (Grant No. 11871369). Yannan Chen was partly supported by the National Natural Science Foundation of China (Grant No. 12171168), Guangdong Basic and Applied Basic Research Foundation (Grant No. 2020A1515010489), and Guangdong Province Higher Education Foundation (Grant No. 2021ZDZX1071).

### CONFLICT OF INTEREST

The authors declare no potential conflict of interest.

### DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

### ENDNOTES

[1]https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
[2]http://www.datatang.com/datares/go.aspx?dataid=606195
[3]http://www.bic.mni.mcgill.ca/ServicesBrainWeb/HomePage
[4]http://trace.eas.asu.edu/yuv/

## ORCID

*Quan Yu* https://orcid.org/0000-0002-8051-7477
*Xinzhen Zhang* https://orcid.org/0000-0001-7498-6616


## REFERENCES


1. Cichocki A, Mandic D, Lathauwer LD, Zhou G, Zhao Q, Caiafa C, et al. Tensor decompositions for signal processing applications: from two-way to multiway component analysis. IEEE Signal Process Mag. 2015;32(2):145–63.
2. Lathauwer L, De Moor B. From matrix to tensor: Multilinear algebra and signal processing. Mathematics in signal processing IV. 1997:1–15.
3. Lai Z, Xu Y, Yang J, Tang J, Zhang D. Sparse tensor discriminant analysis. IEEE Trans Image Process. 2013;22(10):3904–15.
4. Chen Y, Zhang X, Qi L, Xu Y. A Barzilai–Borwein gradient algorithm for spatio-temporal internet traffic data completion via tensor triple decomposition. J Sci Comput. 2021;88(3):65.
5. Liu J, Musialski P, Wonka P, Ye J. Tensor completion for estimating missing values in visual data. IEEE Trans Pattern Anal Mach Intell. 2013;35(1):208–20.
6. Qiu D, Bai M, Ng MK, Zhang X. Robust low transformed multi-rank tensor methods for image alignment. J Sci Comput. 2021;87(1):24.
7. Yu Q. Efficient recovery of low rank tensor via triple nonconvex nonsmooth rank minimization; May 2022.
8. Kenny J, Bader B, Kolda T. Higher-order web link analysis using multilinear algebra, 2005.
9. Mørup M. Applications of tensor (multiway array) factorizations and decompositions in data mining. WIREs Data Mining Knowl Discov. 2011;1(1):24–40.
10. Karatzoglou A, Amatriain X, Baltrunas L, Oliver N. Multiverse recommendation. Proceedings of the 4th ACM Conference on Recommender Systems – RecSys'10. New York: ACM Press; 2010.
11. Kreimer N, Sacchi MD. A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation. Geophysics. 2012;77(3):V113–22.
12. Bengua JA, Phien HN, Tuan HD, Do MN. Efficient tensor completion for color image and video recovery: low-rank tensor train. IEEE Trans Image Process. 2017;26(5):2466–79.
13. Liu Y, Long Z, Zhu C. Image completion using low tensor tree rank and total variation minimization. IEEE Trans Multimed. 2019;21(2):338–50.
14. Song G, Ng MK, Zhang X. Robust tensor completion using transformed tensor singular value decomposition. Numer Linear Algebra Appl. 2020;27(3):e2299.
15. Wang PP, Li L, Cheng GH. Low-rank tensor completion with sparse regularization in a transformed domain. Numer Linear Algebra Appl. 2021;28(6):e2387.
16. 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. 2013;6(3):1758–89.
17. Yu Q, Zhang X, Huang ZH. Multi-tubal rank of third order tensor and related low rank tensor completion problem; 2020.
18. Zhao X, Bai M, Ng MK. Nonconvex optimization for robust tensor completion from grossly sparse observations. J Sci Comput. 2020;85(2):46.
19. Yu Q, Zhang X. Tensor factorization based method for low rank matrix completion and its application on tensor completion; 2022.
20. Gandy S, Recht B, Yamada I. Tensor completion and low-n-rank tensor recovery via convex optimization. Inverse Probl. 2011;27(2):025010.
21. Gao S, Fan Q. Robust schatten-p norm based approach for tensor completion. J Sci Comput. 2020;82(1):11.
22. Candès EJ, Recht B. Exact matrix completion via convex optimization. Found Comput Math. 2009;9(6):717–72.
23. Chen Y. Incoherence-optimal matrix completion. IEEE Trans Inf Theory. 2015;61(5):2909–23.
24. Hitchcock FL. The expression of a tensor or a polyadic as a sum of products. J Math Phys. 1927;6(1-4):164–89.
25. Kiers HAL. Towards a standardized notation and terminology in multiway analysis. J Chemometr. 2000;14(3):105–22.
26. Tucker LR. Some mathematical notes on three-mode factor analysis. Psychometrika. 1966;31(3):279–311.
27. Oseledets IV. Tensor-train decomposition. SIAM J Sci Comput. 2011;33(5):2295–317.
28. Kilmer ME, Martin CD. Factorization strategies for third-order tensors. Linear Algebra Appl. 2011;435(3):641–58.
29. Martin CD, Shafer R, LaRue B. An order-$p$ tensor factorization with applications in imaging. SIAM J Sci Comput. 2013;5(1):A474–90.
30. Hillar CJ, Lim LH. Most tensor problems are NP-hard. J ACM. 2013;60(6):1–39.
31. Zhao Q, Zhou G, Xie S, Zhang L, Cichocki A. Tensor ring decombposition. *arXiv*. 2016. https://arxiv.org/abs/1606.05535
32. Kilmer ME, Braman K, Hao N, Hoover RC. Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging. SIAM J Matrix Anal Appl. 2013;34(1):148–72.
33. Xu Y, Hao R, Yin W, Su Z, Parallel matrix factorization for low-rank tensor completion. Inverse Probl Imaging 2015;9(2):601–624.
34. Xu Y. Alternating proximal gradient method for sparse nonnegative Tucker decomposition. Math Program Comput. 2015;7(1):39–70.
35. Hu Y, Zhang D, Ye J, Li X, He X. Fast and accurate matrix completion via truncated nuclear norm regularization. IEEE Transa Pattern Anal Mach Intell. 2013;35(9):2117–30.
36. Kolda TG, Bader BW. Tensor decompositions and applications. SIAM Rev. 2009;51(3):455–500.
37. Chen K, Dong H, Chan KS. Reduced rank regression via adaptive nuclear norm penalization. Biometrika. 2013;100(4):901–20.
38. Lu C, Tang J, Yan S, Lin Z. Generalized nonconvex nonsmooth low-rank minimization. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Columbus, OH, USA; 2014.

39. Boyd S. Distributed optimization and statistical learning via the alternating direction method of multipliers. Found Trends Mach Learn. 2011;3(1):1–122.
40. Zhou P, Lu C, Lin Z, Zhang C. Tensor factorization for low-rank tensor completion. IEEE Trans Image Process. 2018;27(3):1152–63.

**How to cite this article:** 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;e2464. https://doi.org/10.1002/nla.2464