AIMS Mathematics, 10(3): 6492–6511.
DOI:10.3934/math.2025297
Received: 28 September 2024
Revised: 05 March 2025
Accepted: 14 March 2025
Published: 24 March 2025

http://www.aimspress.com/journal/Math

Research article

A new algorithm by embedding structured data for low-rank tensor ring completion

Ruiping Wen[1,*], **Tingyan Liu**[2] **and Yalei Pei**[2]

[1] Shanxi Key Laboratory for Intelligent Optimization Computing and Blockchain Technology, Taiyuan Normal University, Jinzhong 030619, Shanxi, China

[2] School of Mathematics and Statistics, Taiyuan Normal University, Jinzhong 030619, Shanxi, China

* **Correspondence:** Email: wenrp@163.com; Tel: 03512886656; Fax: 03512886653.

Abstract: In this paper, we put up with a new algorithm for tensor completion problems that include missing slices or row/column fibers, where embedding a structured tensor by a multi-way delay-embedding transform (MDT) makes the tensor to be completed have a special structure. The main idea is to employ a tensor completion algorithm based on the tensor ring rank, constructing latent tensor ring factors with a structure that approximates the original tensor starting from the tensor structure. It is also proved that the sequence generated by the new algorithm converges to the optimal solution. Finally, the feasibility of the proposed algorithm is verified by experiments. Compared with other completed algorithms based on tensor ring rank, the completed accuracy is improved, up to 30%.

Keywords: low-rank tensor completion; tensor ring decomposition; embedded space; structured data
Mathematics Subject Classification: 15A69, 90C06, 90C47

1. Introduction

The problem of completing a low-rank matrix or tensor from small sets of linear measurements occurs in many areas, such as hyperspectral compressive sensing [1], color images [2, 3], data analysis [4], various video processing [5, 6], and machine learning [7]. Mathematically, the optimization model of a tensor completion is described as follows [8]:

$$\begin{aligned} &\min_{\mathcal{X}} \ \operatorname{rank}(\mathcal{X}), \\ &\text{s.t.} \ \ P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}), \end{aligned} \tag{1.1}$$

where $\mathcal{X}, \mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ are both Nth-order tensors, $\mathcal{T}$ is the observed incomplete tensor, and Ω denotes the index set of the known samples, $P_\Omega(\cdot) : \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N} \to \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is the associated sampling orthogonal projection operator, which acquires only the entries indexed by the set Ω.

As we know, the rank of a tensor has many definitions based on different decompositions of a tensor. The most popular ones currently include the CANDECOMP/PARAFAC (CP) rank [9], Tucker rank (or multilinear rank) [10], tensor train (TT) rank [11], and tensor ring (TR) rank [12]. These tensor ranks have found application in various fields. The tensor completion problem is one of the most important applications of tensor decompositions, with the goal to complete an incomplete tensor from the available data. Moreover, the rank-minimization-based tensor completion models can be generated by citing different rank function; it is very challenging to specify the optimal rank beforehand. This is especially the case for Tucker, TT, and TR decompositions, for which the rank is defined as a vector; it is therefore impossible to find the optimal ranks by cross-validation due to the immense possibilities. And many theoretical results and algorithmic methods for completing a low-rank tensor have been obtained. For example, Liu et al. [8] first proposed a definition for the tensor nuclear norm, minimize the weighted sum of the nuclear norm on unfolding matrices to tackle the color image and video completion problems. Then (1.1) can be relaxed into a convex optimization problem as follows:

$$\begin{aligned} &\min \sum_{i=1}^{N} \alpha_i \|\mathcal{X}_{(i)}\|_*, \\ &\text{s.t. } P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}), \end{aligned} \tag{1.2}$$

where $\alpha_i \geq 0$ are constants satisfying $\sum_{i=1}^{N} \alpha_i = 1$, $\mathcal{X}_{(i)} \in \mathbb{R}^{I_i \times \prod_{j \neq i} I_j}$ is the mode-i unfolding of tensor $\mathcal{X}$ for $i \in \{1, \ldots, N\}$, and $\|\mathcal{X}_{(i)}\|_*$ is the nuclear norm of matrix $\mathcal{X}_{(i)}$. Later, Xu et al. [13] and Liu et al. [14] estimated the missing data with the regularized CP decomposition. Furthermore, Bengua et al. [15] applied the TT format to recover incomplete tensors. Zhang et al. [16] considered minimizing a tensor nuclear norm (TNN) on t-SVD to find a low tubal-rank tensor. Zhao et al. [17] employed a Bayesian treatment on the CP model to handle the incomplete tensor. Chen et al. [18] further explored a Bayesian Gaussian CP decomposition model on the traffic data imputation problem. Long [19] presented a Bayesian low-rank tensor completion method with the TR format for the image completion problem. The central effort, for which this paper will further study, focuses on how to retain the correlated information of data sets as much as possible under the tensor low-rank framework in a tractable way in order to achieve the high accuracy of approximations. Based on TR decomposition, Yuan et al. [20] introduced the weight optimization algorithm (TR-WOPT) and the singular value thresholding algorithm (TR-SVD); Zhao et al. [21] proposed the least squares method (TR-ALS) to approximate tensors. Recently, some works [22, 23] jointly consider the low-rank and smoothness properties, which use the hierarchical structured information. Besides, the structured low-rank model also proposed in [24]. Numerous detailed derivations on the tensor completion problem can be referred to in the references given therein.

These algorithms mentioned above were usually used to generalize tensor completion models. They directly represent data with the tensors themselves, and some structures of the tensors are assumed, and then they can be considered as one of two classes of methods for modeling tensors. By contrast, the algorithms in the second class are embedding the data into a high-dimensional feature space, and it is assumed that the data can be represented by a low-rank or a smooth manifold in the embedded space [25–28]. Typically, a time series is represented by a Hankel matrix, and its low-rank property

has been employed widely for modeling a linear time-invariant system of singles [27,28]. For example, Li et al. [29] proposed a method for modeling damped sinusoidal signals based on a low-rank Hankel approximation. Ding et al. [25] proposed the use of rank minimization of a Hankel matrix for the video inpainting problem by assuming an autoregressive moving average model.

This study focuses on a case where all of the elements in some continuous slices or row/column fibers are missing in tensor data. The completion of missing slices or row/column fibers is recognized as a difficult problem that common tensor completion algorithms usually fail to solve in this case. The research on using embedded structured data for low rank or smoothness of data has mainly been applied to matrix completion problems; only a few works [30, 31] have studied tensor completion algorithms for embedded structures. Therefore, there is still a lot of research in tensor completion problems for embedding structured data.

To obtain the structured data, the popular method of embedding data is vector delay embedding (VDT) or multi-way delay-embedding transform (MDT) [31], which is a special Hankel structure formed by embedding a set of vectors into some data. It is frequently used in the signal processing of time series [32]. It is noted that [31] have proposed two Tucker decomposition algorithms by using the same embedding method. This paper is to use the MDT technique to embed data from general tensors and expand them into high-order structured tensors, and a kernel factor with a special structure (Hankel) is constructed to approximate and achieve tensor completion for improving the approximated accuracy. We suggested a new algorithm that effectively alleviates the burden of rank selection and improves the accuracy of tensor completion.

The rest of this paper is organized as follows: the notations used throughout this paper and related preliminaries are introduced in Section 2. Section 3 establishes the proposed algorithms in detail. Convergence analysis is discussed under mild conditions in Section 4. In Section 5, we compare the proposed algorithms with the previous algorithms to show the effectiveness of the new algorithms. Finally, we conclude the paper in Section 6.

2. Notation and preliminaries

Throughout this paper, $\mathbb{N}$ stands for the set of the natural number and $\mathbb{R}$ the real number; $\mathbb{R}^{I_1}$ and $\mathbb{R}^{I_1\times I_2}$ denote the space of I_1 dimensional real column vectors and $I_1 \times I_2$ dimensional real matrices, respectively. Let $\mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ be the space of Nth-order tensors with size $I_1\times I_2\times\cdots\times I_N$. The calligraphic letters, e.g., $\mathcal{A}$, denote tensors; capital letters, e.g., A, denote matrices; lowercase letters, e.g., a, denote vectors; and Greek letters, e.g., α, denote scalars. For matrix X, we let $\sigma_1(X) \geq \sigma_2(X) \geq \cdots \geq \sigma_r(X)$ be singular values of X with rank r and $\|X\|_* = \text{trace}((X^TX)^{\frac{1}{2}}) = \sum_{i=1}^{r}\sigma_i(X)$ be the nuclear norm of X, here trace$(\cdot)$ is the matrix trace operation. For an Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\ldots\times I_N}$, its element is denoted by $\mathcal{X}_{i_1i_2\ldots i_N}$, where $1 \leq i_n \leq I_n$ and $1 \leq n \leq N$. Where appropriate, a tensor sequence can also be written as $[\mathcal{X}]$. The inner product of two tensors, $\mathcal{X}$ and $\mathcal{Y}$, of the same order and size is defined as $\langle\mathcal{X},\mathcal{Y}\rangle = \sum_{i_1,i_2,\ldots,i_N}\mathcal{X}_{i_1i_2\ldots i_N}\mathcal{Y}_{i_1i_2\ldots i_N}$. The Frobenius norm of a tensor is given by $\|\mathcal{X}\|_F = \langle\mathcal{X},\mathcal{X}\rangle^{\frac{1}{2}}$. The orthogonal projection operator onto an index set Ω, denoted by $P_\Omega(\cdot)$, is defined as

$$(P_\Omega(\mathcal{X}))_{i_1i_2\ldots i_N} = \begin{cases} \mathcal{X}_{i_1i_2\ldots i_N}, & \text{if } i_1i_2\ldots i_N \in \Omega, \\ 0, & \text{if } i_1i_2\ldots i_N \in \bar{\Omega}, \end{cases} \quad \forall\, \mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\ldots\times I_N},$$

where $\bar{\Omega}$ is the complementary set of Ω. A Hankel matrix $H = (h_{i+j-1})_{i,j=1}^{n} \in \mathbb{R}^{n\times n}$ is of the form

$$H = \begin{pmatrix} h_1 & h_2 & \cdots & h_n \\ h_2 & h_3 & \cdots & h_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ h_n & h_{n+1} & \cdots & h_{2n-1} \end{pmatrix},$$

which is determined by $2n-1$ entries, that is, the first row and the last column.

The TR decomposition represents a tensor of higher order by circular multi-linear products over a sequence of low-order latent core tensors, i.e., TR factors. For $n = 1, 2, \ldots, N$, the TR factors are denoted by $\mathcal{G}^{(n)} \in \mathbb{R}^{r_n\times I_n\times r_{n+1}}$ and each consists of two rank modes (i.e., mode-1 and mode-3) and one dimension mode (i.e., mode-2). The syntax $r = \{r_1, r_2, \ldots, r_N\}$ denotes the TR-rank, which controls the model complexity of the TR decomposition. The TR decomposition applies trace operations, and all of the TR factors are set to be 3-order; thus, the TR decomposition relaxes the rank constraint on the first and last core of the TT to $r_1 = r_{N+1}$. Moreover, the TR decomposition linearly scales to the order of the tensor, and in this way, it overcomes the curse of dimensionality. In this case, the TR can be considered as a linear combination of the TTs and hence offers a powerful and generalized representation ability. The element-wise relation of the TR decomposition and the generated tensor is given by:

$$\mathcal{X}_{i_1 i_2 \ldots i_N} = \text{trace}\left\{\prod_{n=1}^{N} G_{i_n}^{(n)}\right\}, \tag{2.1}$$

where $\mathcal{X}_{i_1 i_2 \ldots i_N}$ denotes the $(i_1, i_2, \ldots, i_N)$th element of the tensor, $G_{i_n}^{(n)} \in \mathbb{R}^{r_n\times r_{n+1}}$ is the i_n-th mode-2 slice matrix of $\mathcal{G}^{(n)}$, which can also be denoted by $\mathcal{G}^{(n)}(:, i_n, :)$ according to the MATLAB notation. For simplicity, we denote the TR decomposition by $\mathcal{X} = \Psi(\mathcal{G}^{(1)}, \mathcal{G}^{(2)}, \ldots, \mathcal{G}^{(N)})$.

There now exist three "unfold" operations $\text{unfold}_n(\mathcal{T})$ along the n-mode of an Nth-order tensor $\mathcal{T} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ are respectively defined as: $T_{\langle n\rangle} \in \mathbb{R}^{\prod_{j=1}^{n} I_j\times\prod_{j=n+1}^{N} I_j}$, the n-unfolding matrix (see [12, 30]) with its elements defined by $T_{i_1 i_2 \ldots i_N} = T_{\langle n\rangle}(\overline{i_1 \ldots i_n}, \overline{i_{n+1} \ldots i_N})$; $T_{(n)} \in \mathbb{R}^{I_n\times\prod_{j\neq n} I_j}$, the classical mode-$n$ unfolding matrix (see [8, 12, 30]) with its elements defined by $T_{i_1 i_2 \ldots i_N} = T_{(n)}(i_n, \overline{i_1 \ldots i_{n-1} i_{n+1} \ldots i_N})$; $T_{[n]} \in \mathbb{R}^{I_n\times\prod_{j\neq n} I_j}$, the mode-$n$ unfolding matrix (see [12, 30]) with its elements defined by $T_{i_1 i_2 \ldots i_N} = T_{[n]}(i_n, \overline{i_{n+1} \ldots i_N i_1 \ldots i_{n-1}})$. The difference between the latter two types of mode-n unfolding operations lies in the ordering of indices associated with the $N-1$ modes, which corresponds to a specific dimensional permutation performed on $\mathcal{T}$. Their inverse operation "fold" is respectively defined as $\text{fold}_n(T_{(n)}) = \mathcal{T}, \text{fold}_n(T_{\langle n\rangle}) = \mathcal{T}, \text{fold}_n(T_{[n]}) = \mathcal{T}$. By the way, $\|\mathcal{T}\|_F = \|T_{(n)}\|_F$ for any $1 \le n \le N$ and so on.

Definition 1. (see [12]) The subchain tensor by merging all cores except the nth core $\mathcal{G}^{(n)}$, i.e., $\mathcal{G}^{(n+1)}, \ldots, \mathcal{G}^{(N)}, \mathcal{G}^{(1)}, \ldots, \mathcal{G}^{(n-1)}$, is denoted by $\mathcal{G}^{\neq n} \in \mathbb{R}^{r_{n+1}\times\prod_{j\neq n} I_j\times r_n}$ whose slice matrices are defined by

$$G^{\neq n}(\overline{i_{n+1} \ldots i_N i_1 i_2 \ldots i_{n-1}}) = \prod_{j=n+1}^{N} G_{i_j}^{(j)} \prod_{j=1}^{n-1} G_{i_j}^{(j)}. \tag{2.2}$$

Definition 2. (see [33]) The n-mode (matrix) product of a tensor $\mathcal{T} = (\mathcal{T}_{i_1 i_2 \cdots i_N}) \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ with a matrix $X = (X_{j i_n}) \in \mathbb{R}^{J\times I_n}$ is denoted by $\mathcal{T} \times_n X$ and is of size $I_1 \times I_2 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$.

Elementwise, we have

$$(\mathcal{T} \times_n X)_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} \mathcal{T}_{i_1 i_2 \cdots i_N} X_{j i_n}. \tag{2.3}$$

Each mode-n fiber is multiplied by the matrix X. The idea can be expressed in terms of unfolded tensors:

$$\mathcal{Y} = \mathcal{T} \times_n X \Leftrightarrow Y_{(n)} = X T_{(n)},$$

that is to say, $\mathcal{Y} = \text{fold}_n(Y_{(n)}) = \text{fold}_n(X T_{(n)})$.

Definition 3. (see [31]) A standard delay embedding transform (VDT) of a vector $v = (v_1, v_2, \ldots, v_L)^T \in \mathbb{R}^L$ with τ is given

$$\mathcal{H}(v) = \begin{pmatrix} v_1 & v_2 & \cdots & v_{L-\tau+1} \\ v_2 & v_3 & \cdots & v_{L-\tau+2} \\ \vdots & \vdots & \ddots & \vdots \\ v_\tau & v_{\tau+1} & \cdots & v_L \end{pmatrix} \in \mathbb{R}^{\tau \times (L-\tau+1)},$$

it can be considered as a Hankelization operation since $\mathcal{H}(v)$ is a Hankel matrix.

If $S \in \mathbb{B}^{\tau(L-\tau+1) \times L} \subset \mathbb{R}^{\tau(L-\tau+1) \times L}$ with $\mathbb{B} = \{0, 1\}$ is a Boolean or a 0-1 matrix that satisfies

$$\text{vec}(\mathcal{H}(v)) = S\, v,$$

then the standard delay embedding transform can be obtained by

$$(\mathcal{H}(v)) = \text{fold}_{(L,\tau)}(S\, v),$$

where $\text{fold}_{(L,\tau)} : \mathbb{R}^{\tau(L-\tau+1)} \to \mathbb{R}^{\tau \times (L-\tau+1)}$ is a folding operator from a vector to a matrix. It should be noted that S is also a duplication matrix; the diagonal elements of $(S^T S)$ comprise the numbers of duplications for individual elements, which are usually $\tau \in \mathbb{N}$, but low for marginal elements.

The inverse delay embedding transform for a Hankel matrixV_H can be given by

$$\mathcal{H}_\tau^{-1}(V_H) = S^\dagger \text{vec}(V_H),$$

where $S^\dagger = (S^T S)^{-1} S^T$ stands for the Moore–Penrose pseudo-inverse of S. An inverse transform of a VDT can be decomposed into duplication and folding, so the inverse transform can also be decomposed into the individual corresponding inverse transforms: a vectorization operation and the Moore–Penrose pseudo-inverse.

Definition 4. (see [31]) A multi-way delay-embedding transform (MDT) for an Nth order tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ with $\tau = (\tau_1, \tau_2, \ldots, \tau_N) \in \mathbb{N}^N$ is defined by

$$\text{MDT}: \ \mathcal{H}_\tau(\mathcal{T}) = \text{fold}_{(I,\tau)}(\mathcal{T} \times_1 S_1 \times_2 \cdots \times_N S_N),$$

where the operator $\text{fold}_{(I,\tau)} : \mathbb{R}^{\tau_1(I_1-\tau_1+1) \times \cdots \times \tau_N(I_N-\tau_N+1)} \to \mathbb{R}^{\tau_1 \times (I_1-\tau_1+1) \times \cdots \times \tau_N \times (I_N-\tau_N+1)}$ constructs a $2N$th order tensor from the input Nth order tensor.

The inverse MDT for a Hankel tensor is given by

$$\text{invMDT}: \ \mathcal{H}_\tau^{-1}(\mathcal{T}_H) = \mathcal{T}_H \times_1 S_1^\dagger \times \cdots \times_N S_N^\dagger,$$

where $\mathcal{T}_H$ is a $2N$th order tensor, this operator reshaped an Nth order tensor from the $2N$th order tensor.

For convenience, they would be denoted by MDT($\mathcal{X}$) and invMDT($\mathcal{X}$) in the subsequent, respectively. In fact, the MDT is also a combination of multi-linear duplication and multi-way folding operations. Figure 1 shows flowcharts to illustrate MDT and invMDT for $N = 3$.

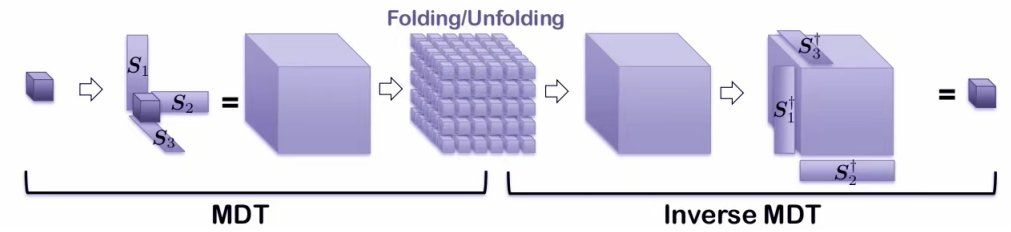

Figure 1. The technique of a multi-way delay-embedding transform (MDT) and its inverse for $N = 3$.

Let $X \in \mathbb{R}^{I_1 \times I_2}$ be a matrix with rank r. The SVD of X is given by $X = U\Sigma_r V^T$, where $U \in \mathbb{R}^{I_1 \times r}$ and $V \in \mathbb{R}^{I_2 \times r}$ have orthogonal columns formed by the left and the right singular vectors of X, respectively, and $\Sigma_r = \text{diag}(\{\sigma_i\}_{1 \le i \le r})$ is a diagonal matrix consisting of all singular values of X, see [34]. Let $\tau > 0$ be a parameter. The singular value shrinkage operator is given by $D_\tau(X) = U\text{diag}(\{\max(\sigma_i - \tau, 0)\}_{1 \le i \le r})V^T$, see [35].

3. Proposed algorithm

In this section, we introduce a new algorithm by embedding a Hankel structure for low-rank tensor ring completion.

To the best of our knowledge, there are two proposed TR-based tensor completion algorithms: the TR-ALS and TR-WOPT, mentioned in above Section 1. They apply the same optimization model, which is formulated as:

$$\min_{[\mathcal{G}]} \ \|P_\Omega(\mathcal{T} - \Psi([\mathcal{G}])\|_F^2, \tag{3.1}$$

where the optimization objective is the TR factors sequence $[\mathcal{G}]$, $P_\Omega(\mathcal{T})$ denotes all the observed entries, the set of indices of observed entries represented by Ω, and $\Psi([\mathcal{G}]) = \Psi(\mathcal{G}^{(1)}, \mathcal{G}^{(2)}, \ldots, \mathcal{G}^{(N)})$ denotes the approximated tensor generated by $[\mathcal{G}]$. Every element of $\Psi([\mathcal{G}])$ is calculated by Eq (2.1).

Completion by rank minimization-based regularization, the model (3.1) can be formulated as:

$$\min_{\mathcal{X}} \quad \text{rank}(\mathcal{X}) + \frac{\lambda}{2}\|P_\Omega(\mathcal{T} - \mathcal{X})\|_F^2, \tag{3.2}$$

where $\mathcal{X}$ is the completed low-rank tensor, and $\text{rank}(\mathcal{X})$ is a rank regularizer. The model can therefore find the low-rank structure of the data and approximate the incomplete tensor.

Because determining the tensor rank is an NP-hard problem [33, 36], work in [8, 37] extends the concept of low-rank matrix completion and defines tensor rank as a sum of the rank of the mode-n unfolding of the object tensor. A model of the tensor ring completion can be equivalently rewritten as:

$$\min_{\mathcal{G}} \quad \sum_{n=1}^{N} \text{rank}(\mathcal{G}^{(n)}) + \frac{\lambda}{2}\|P_\Omega(\mathcal{T} - \Psi([\mathcal{G}]))\|_F^2,$$

Usually, the model is solved by ADMM algorithms, and it is shown to have fast convergence and good performance when the data size is small. However, when dealing with large-scale data, the multiple SVD operations in the optimization step will be intractable due to high computational cost.

By nuclear norm regularization and imposing low rankness on each of the TR factors, a tensor completion model is formulated as follows:

$$\begin{aligned} \min_{[\mathcal{G}],\mathcal{X}} \quad & \sum_{n=1}^{N} \|\mathcal{G}^{(n)}\|_* + \tfrac{\lambda}{2}\|\mathcal{X} - \Psi([\mathcal{G}])\|_F^2, \\ \text{s.t.} \quad & P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}). \end{aligned} \tag{3.3}$$

Further, the model (3.3) was finally rewritten as the tensor ring low-rank factors (TRLRF) model in [19, 33, 36, 38]:

$$\begin{aligned} \min_{[\mathcal{G}],\mathcal{X}} \quad & \sum_{n=1}^{N}\sum_{k=1}^{3} \|G_{(k)}^{(n)}\|_* + \tfrac{\lambda}{2}\|\mathcal{X} - \Psi([\mathcal{G}])\|_F^2, \\ \text{s.t.} \quad & P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}). \end{aligned} \tag{3.4}$$

This TRLRF model has two distinctive advantages. First, the low-rank assumption is placed on tensor factors instead of on the original tensor; this greatly reduces the computational complexity of the SVD operation. Second, the low-rankness of the tensor factors can enhance the robustness to rank selection, which can alleviate the burden of searching for the optimal TR-rank and reduce the computational cost in the implementation. Immediately after, a new model is proposed that utilizes the rank relationship between a tensor and its core, which gives low-rank constraints on the two rank modes of the TR factors, i.e., the unfolding of the TR factors along mode-1 and mode-3, which can be expressed by $\sum_{n=1}^{N} \|G_{(1)}^{(n)}\|_* + \sum_{n=1}^{N} \|G_{(3)}^{(n)}\|_*$ [20]. Furthermore, it is noted that a slicing matrix of the mode-2 tensor core factor represents that important information of the original from definition 1, and so it can be used instead of mode-1 and mode-3 to improve computational efficiency and reduce computation time. So we developed it as follows:

$$\begin{aligned} \min_{[\mathcal{G}],\mathcal{X}} \quad & \sum_{n=1}^{N} 3\|G_{(2)}^{(n)}\|_* + \tfrac{\lambda}{2}\|\mathcal{X} - \Psi([\mathcal{G}]))\|_F^2, \\ \text{s.t.} \quad & P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}). \end{aligned} \tag{3.5}$$

By embedding the Hankel structure $\mathcal{H}_\mathcal{T} = \mathrm{MDT}(\mathcal{X})$, the model (3.5) is reexpressed as the following model:

$$\begin{aligned} \min_{[\mathcal{G}],\mathcal{H}_\mathcal{T}} \quad & \sum_{n=1}^{2N} 3\|\mathcal{G}_{(2)}^{(n)}\|_* + \tfrac{\lambda}{2}\|\mathcal{H}_\mathcal{T} - \Psi([\mathcal{G}])\|_F^2, \\ \text{s.t.} \quad & P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}). \end{aligned} \tag{3.6}$$

We apply the augmented Lagrange multiplier (ALM) method [39] to solve (3.6). By introducing $\mathcal{M}, \mathcal{Y}$, then the augmented Lagrangian function optimization model of (3.6) with a constraint is given as

$$L(\mathcal{G}, \mathcal{X}, \mathcal{M}, \mathcal{Y}) = \sum_{n=1}^{2N}(3\|M_{(2)}^{(n)}\|_* + \langle \mathcal{Y}_n, \mathcal{M}_n - \mathcal{G}_n\rangle + \frac{\mu}{2}\|\mathcal{M}_n - \mathcal{G}_n\|_F^2) + \frac{\mu}{2}\|\mathcal{H}_\tau - \Psi([\mathcal{G}_n])\|_F^2,$$

$$\text{s.t. } P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}), \tag{3.7}$$

where $\mu > 0$ is the penalty parameter, $\mathcal{M}_n$ is the auxiliary variable of $\mathcal{G}_n$, $\mathcal{Y}_n$ is the Lagrangian multiplier, and $\mathcal{G}_n, \mathcal{M}_n, \mathcal{Y}_n$ are independent of each other. The corresponding mathematical programming can be given in the following

$$\begin{aligned} &\min\ L(\mathcal{G}, \mathcal{X}, \mathcal{M}, \mathcal{Y}), \\ &\text{s.t. } P_\Omega(\mathcal{X}) = P_\Omega(\mathcal{T}). \end{aligned} \tag{3.8}$$

The optimization problem (3.8) is a nonconvex and nonsmooth programming. Thus, we obtain Algorithm 1.

Algorithm 1: Embedding Structured Data in Low-rank Tensor Ring Completion.

Input: $P_\Omega(\mathcal{T}), \Psi, N$;
output: $\mathcal{X}$
0. Let the subscripts of each element position in $\mathcal{G}_N$ be (i, j, k), the element values that make their subscripts equal to sum$(i + j)$ the element are the same. $\mathcal{Y}_1 = 0, \mathcal{M}_1 = 0, \lambda = 5, \mu = 10^{-1}, \mu_{\max} = 10^2$, $\rho = 1.3, \epsilon = 10^{-6}, t_{\max} = 5000$;
1. $\mathcal{X} = P_\Omega(\mathcal{T})$
2. $\mathcal{H}_\tau = \text{MDT}(\mathcal{X})$,
   for $t = 1 : t_{\max}$,
   while $n = 1 : 2N$
3. $\mathcal{H}_{\text{last}} = \mathcal{H}_\tau$
4. $\mathcal{G}_{n+1} = \text{fold}_2\left((3(\mu M^n_{(2)} + Y^n_{(2)}) + \lambda H_{\text{last}} G^{\neq n}_{<2>})(\lambda G^{\neq n,\text{T}}_{<2>} G^{\neq n}_{<2>} + 3\mu I)^{-1}\right)$
5. $\mathcal{M}_{n+1} = \text{fold}_2(\mathcal{D}_{\frac{1}{\mu}}(G^{n+1}_{(2)} - \frac{1}{\mu} Y^n_{(2)}))$
6. $\mathcal{H}_\tau = P_\Omega(\mathcal{H}_{\text{last}}) + P_{\bar{\Omega}}(\Psi([\mathcal{G}]))$
7. $\mathcal{Y}_{n+1} = \mathcal{Y}_n + \mu(\mathcal{M}_{n+1} - \mathcal{G}_{n+1})$
8. $\mu = \min(\rho\mu, \mu_{\max})$
9. **if** $\frac{\|\mathcal{H}_\tau - \mathcal{H}_{\text{last}}\|_F}{\|\mathcal{H}_\tau\|_F} < \epsilon$, **break**
   end
   end
10. $\mathcal{X} = \text{invMDT}(\mathcal{H}_\tau)$

Remark. The algorithm 1 is conceptually quite simple, comprising three steps: (1) MDT; (2) low-rank tensor approximation; and (3) inverse MDT. In (2), we obtain a low-rank approximation of $\mathcal{X}$ based on the TR decomposition model (3.5).

4. Convergence analysis

In this section, we discuss the global convergence of Algorithm 1.

For the ease of exposition, $\mathcal{H}, \mathcal{M}_k, \mathcal{G}_k, \mathcal{Y}_k$ as described in Algorithm 1, let $(\mathcal{M}^*, \mathcal{G}^*, \mathcal{Y}^*)$ be the solution of (3.8).

Lemma 1. The sequence $\{\mathcal{Y}_k\}$ generated by Algorithm 1 is bounded.

Proof. Let $\mathcal{B} = \mu_k(\mathcal{M}_{k+1} + \frac{1}{\mu_k}\mathcal{Y}_k - \mathcal{G}_k)$. We have the following from Algorithm 1,

$$\begin{aligned}\mathcal{Y}_{k+1} &= \mathcal{Y}_k + \mu_k(\mathcal{M}_{k+1} - \mathcal{G}_{k+1}) \\ &= \mathcal{Y}_k + \mu_k(\mathcal{M}_{k+1} - \mathcal{G}_k + \mathcal{G}_k - \mathcal{G}_{k+1}) \\ &= \mathcal{Y}_k + \mu_k(\mathcal{M}_{k+1} - \mathcal{G}_k) + \mu_k(\mathcal{G}_k - \mathcal{G}_{k+1}).\end{aligned}$$

By simple deduction,

$$\begin{aligned}\mu_k(\mathcal{G}_k - \mathcal{G}_{k+1}) &= \mu_k P_{\bar{\Omega}}(\mathcal{G}_k - \mathcal{G}_{k+1}) \\ &= \mu_k P_{\bar{\Omega}}(\mathcal{G}_k - (\mathcal{M}_{k+1} + {\mu_k}^{-1}\mathcal{Y}_k)) \\ &= P_{\bar{\Omega}}(\mu_k(\mathcal{M}_{k+1} + {\mu_k}^{-1}\mathcal{Y}_k - \mathcal{G}_k)) \\ &= P_{\bar{\Omega}}(\mathcal{B}).\end{aligned}$$

And

$$\begin{aligned}\|\mathcal{B}\|_F &= \|\mu_k(\mathcal{M}_{k+1} + {\mu_k}^{-1}\mathcal{Y}_k - \mathcal{G}_k)\|_F \\ &= \mu_k \sum_{i=1}^{N} \|M_{(i)}^{k+1} + {\mu_k}^{-1}Y_{(i)}^k - G_{(i)}^k)\|_F \\ &\leq N\mu_k \|M_{(2)}^{k+1} + {\mu_k}^{-1}Y_{(2)}^k - G_{(2)}^k)\|_F.\end{aligned}$$

The truncated SVD has been used in Algorithm 1, then

$$M_{(2)}^{k+1} + {\mu_k}^{-1}Y_{(2)}^k = \overline{U}_k\overline{\Sigma}_k\overline{V}_k^{\mathrm{T}} + \widetilde{U}_k\widetilde{\Sigma}_k\widetilde{V}_k^{\mathrm{T}},$$

where $\overline{U}_k\overline{\Sigma}_k\overline{V}_k^{\mathrm{T}}$ and $\widetilde{U}_k\widetilde{\Sigma}_k\widetilde{V}_k^{\mathrm{T}}$ are both the partial SVD divided by thresholding ${\mu_k}^{-1}$, that is to say, $\overline{\sigma}_k > {\mu_k}^{-1}, \widetilde{\sigma}_k \leq {\mu_k}^{-1}$.

Hence,

$$\begin{aligned}\|\mathcal{B}\|_F &= \|\mu_k(M_{(2)}^{k+1} + {\mu_k}^{-1}Y_{(2)}^k - G_{(2)}^k)\|_F \\ &\leq N\mu_k\|(M_{(2)}^{k+1} + {\mu_k}^{-1}Y_{(2)}^k - G_{(2)}^k\|_F \\ &= N\|\mu_k(\overline{U}_k\overline{\Sigma}_k\overline{V}_k^{\mathrm{T}} + \widetilde{U}_k\widetilde{\Sigma}_k\widetilde{V}_k^{\mathrm{T}} - U_k(\Sigma_k - {\mu_k}^{-1}I)\overline{V}_k^{\mathrm{T}})\|_F \\ &= N\|\mu_k({\mu_k}^{-1}\overline{U}_k I\overline{V}_k^{\mathrm{T}} + \widetilde{U}_k\widetilde{\Sigma}_k\widetilde{V}_k^{\mathrm{T}})\|_F.\end{aligned}$$

From the Lemmas 1 and 4 in [39], $\mu_k(M_{(2)}^{k+1} + {\mu_k}^{-1}Y_{(2)}^k - G_{(2)}^k) \in \nabla\|M_{(2)}^{k+1}\|_*$.

Let $M_{(2)}^{k+1} = U\Sigma V^{\mathrm{T}}$,

$$\|M_{(2)}^{k+1}\|_* = \{UV^{\mathrm{T}} + W : W \in \mathbb{R}^{n\times m}, U^{\mathrm{T}}W = 0, WV = 0, \|W\|_2 \leq 1\},$$

$$\|UV^{\mathrm{T}} + W\|_F^2 = \mathrm{trace}(VV^{\mathrm{T}} + W^{\mathrm{T}}W) \leq N,$$

$$\|\mu_k(M_{(2)}^{k+1} + {\mu_k}^{-1}Y_{(2)}^k - G_{(2)}^k)\|_F \leq \sqrt{N},$$

Then,

$$N\|\mu_k(M_{(2)}^{k+1} + \mu_k{}^{-1}Y_{(2)}^k - G_{(2)}^k)\|_F \le N\sqrt{N},$$
$$\|\mu_k(\mathcal{M}_{k+1} + \mu_k{}^{-1}\mathcal{Y}_k - \mathcal{G}_k)\|_F \le N\sqrt{N},$$
$$\|\mathcal{B}\|_F \le N\sqrt{N}.$$

Thus, Lemma 1 holds.

Theorem 1. Suppose that the series $\sum_{k=1}^{+\infty}\mu_k^{-1}$ diverges if $\mu_k \to +\infty$. Then the sequences $\{\mathcal{M}_k\}, \{\mathcal{G}_k\}$ generated by Algorithm 1 converge to the optimal solution of (3.8).

Proof. According to Algorithm 1, we obtain

$$\mathcal{Y}_{k+1} = \mathcal{Y}_k + \mu_k(\mathcal{M}_{k+1} - \mathcal{G}_{k+1}),$$
$$\mu_k^{-1}(\mathcal{Y}_{k+1} - \mathcal{Y}_k) = \mathcal{M}_{k+1} - \mathcal{G}_{k+1},$$
$$\lim_{k\to+\infty}\mu_k{}^{-1}(\mathcal{Y}_{k+1} - \mathcal{Y}_k) = \lim_{k\to+\infty}(\mathcal{M}_{k+1} - \mathcal{G}_{k+1}) = 0.$$

Let $\mathcal{M}^*, \mathcal{G}^*$ denote the optimal solution of (3.8); this means $\mathcal{M}^* = \mathcal{G}^*$. In order to prove this theorem, we now consider the formula

$$\begin{aligned}\|\mathcal{M}_{k+1} - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}^*\|_F^2 &\\ = \|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2 - \|\mathcal{M}_{k+1} - \mathcal{M}_k\|_F^2 &\\ - \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}_k\|_F^2 - 2\mu_k{}^{-1}\langle\mathcal{G}_{k+1} - \mathcal{G}^*, \widehat{\mathcal{Y}}_{k+1} - \mathcal{Y}^*\rangle.&\end{aligned}$$

Suppose that $\widehat{\mathcal{Y}}_{k+1} = \mathcal{Y}_k + \mu_k(\mathcal{M}_{k+1} - \mathcal{G}_k)$, $\mathcal{Y}^*$ is the optimal solution of the dual problem in [39].

$$\begin{aligned}\|\mathcal{M}_{k+1} - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}^*\|_F^2 &\\ = \|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2 - (\|\mathcal{M}_{k+1} - \mathcal{M}_k\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}_k\|_F^2) &\\ + 2\langle\mathcal{M}_{k+1} - \mathcal{M}^*, \mathcal{M}_{k+1} - \mathcal{M}_k\rangle + \mu_k^{-2}\langle\mathcal{Y}_{k+1} - \mathcal{Y}^*, \mathcal{Y}_{k+1} - \mathcal{Y}_k\rangle &\\ = \|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2 - (\|\mathcal{M}_{k+1} - \mathcal{M}_k\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}_k\|_F^2) &\\ + 2\langle\mathcal{M}_{k+1} - \mathcal{M}^*, \mathcal{M}_{k+1} - \mathcal{M}_k\rangle + 2\mu_k{}^{-1}\langle\mathcal{Y}_{k+1} - \mathcal{Y}^*, \mathcal{M}_{k+1} - \mathcal{G}_{k+1}\rangle &\\ = \|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2 - (\|\mathcal{M}_{k+1} - \mathcal{M}_k\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}_k\|_F^2) &\\ + 2\langle\mathcal{M}_{k+1} - \mathcal{M}^*, \mathcal{M}_{k+1} - \mathcal{M}_k\rangle + 2\mu_k{}^{-1}(\langle\mathcal{Y}_{k+1} - \mathcal{Y}^*, \mathcal{M}_{k+1} - \mathcal{M}^*\rangle &\\ + \langle\mathcal{Y}_{k+1} - \mathcal{Y}^*, \mathcal{G}^* - \mathcal{G}_{k+1}\rangle) &\\ = \|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2 - (\|\mathcal{M}_{k+1} - \mathcal{M}_k\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}_k\|_F^2) &\\ - 2\mu_k{}^{-1}\langle\mathcal{G}_{k+1} - \mathcal{G}^*, \widehat{\mathcal{Y}}_{k+1} - \mathcal{Y}^*\rangle.&\end{aligned}$$

Since $\|\mathcal{G}\|_*$ is a convex function and $\widehat{\mathcal{Y}}_{k+1} \in \nabla\|\mathcal{G}_{k+1}\|_*$, then

$$\langle\mathcal{G}_{k+1} - \mathcal{G}^*, \widehat{\mathcal{Y}}_{k+1} - \mathcal{Y}^*\rangle \ge 0.$$

It is noted that $\|\mathcal{M}_{k+1} - \mathcal{M}_k\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}_k\|_F^2 \ge 0$, and

$$\|\mathcal{M}_{k+1} - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_{k+1} - \mathcal{Y}^*\|_F^2 \le \|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2 + 2\mu_k{}^{-1}\langle\mathcal{G}_{k+1} - \mathcal{G}^*, \widehat{\mathcal{Y}}_{k+1} - \mathcal{Y}^*\rangle.$$

Then $\{\|\mathcal{M}_k - \mathcal{M}^*\|_F^2 + \mu_k^{-2}\|\mathcal{Y}_k - \mathcal{Y}^*\|_F^2\}$ is not increasing, and $\sum_{k=1}^{\infty} \mu_k{}^{-1}\langle \mathcal{M}_k - \mathcal{M}^*, \mathcal{Y}_k - \mathcal{Y}^*\rangle < +\infty$.

Consequently, $\mathcal{M}^*, \mathcal{G}^*$ is the optimal solution of (3.7) by the proof of Lemma 2 in [39].

So the theorem is proven.

Theorem 2. Assume that the tensor $\mathcal{H}_\tau$ is formed by $[\mathcal{G}_k]$. Then

$$\|\mathcal{H}_\tau - \mathcal{H}^*\|_F \leq \|\Psi([\mathcal{G}_k]) - \mathcal{H}^*\|_F,$$

here $\mathcal{H}^*$ is the solution of (3.6).

Proof. Because of

$$\begin{aligned}\|\Psi([\mathcal{G}_k]) - \mathcal{H}^*\|_F^2 =& \|\Psi([\mathcal{G}_k]) - \mathcal{H}_\tau + \mathcal{H}_\tau - \mathcal{H}^*\|_F^2 \\ =& \|\Psi([\mathcal{G}_k]) - \mathcal{H}_\tau\|_F^2 + \|\mathcal{H}_\tau - \mathcal{H}^*\|_F^2 + 2\langle \Psi([\mathcal{G}_k]) - \mathcal{H}_\tau, \mathcal{H}_\tau - \mathcal{H}^*\rangle \\ =& \|\Psi([\mathcal{G}_k]) - \mathcal{H}_\tau\|_F^2 + \|\mathcal{H}_\tau - \mathcal{H}^*\|_F^2,\end{aligned}$$

then $\|\mathcal{H}_\tau - \mathcal{H}^*\|_F \leq \|\Psi([\mathcal{G}_k]) - \mathcal{H}^*\|_F$ holds.

The theorem is proven.

5. Numerical experiments

In this section, we conducted experiments to testify to the effectiveness of our algorithm 1 by comparing some algorithms in the relative square error (RSE), peak signal-to-noise ratio (PSNR), the computing time in seconds (CPU(s)), and structural similarity (SSIM) [40]. They are defined as

$$\text{PSNR} = 10\log_{10}(255^2/\text{MSE}),$$

where the mean squared error (MSE) was defined by

$$\text{MSE} = \frac{\|\mathcal{X} - \mathcal{T}\|_F^2}{\text{num}(\mathcal{T})}, \quad \text{num}(\mathcal{T}) \text{ means the number of elements of } \mathcal{T};$$

$$\text{RSE} = \frac{\|\mathcal{X} - \mathcal{T}\|_F}{\|\mathcal{T}\|_F};$$

$$\text{SSIM}(x, y) = [(l(x, y)]^\alpha \cdot [(c(x, y)]^\beta \cdot [(s(x, y)]^\gamma,$$

where α, β, γ are all positive real numbers. As for the details of the functionals $l(x, y), c(x, y)$, and $s(x, y)$, please refer to [40].

The TR-WOPT [20] and TR-LRF [38]) algorithms were used to solve the model (3.4); algorithm 1 from [31] was used to solve the model (20) in [31]; and our algorithm 1 was exploited to solve the model (3.6). In the experiments, we selected three benchmark images, named Lena, House, and Peppers with the size of them being $256 \times 256 \times 3$, which can be considered a 3rd order tensor. p represents the sampling rate in the test.

There are three cases for the experiment in the following. In our implementations, all the codes were written by Matlab R2019b and run on a PC with an Intel Xeon E5 processor, 2.5GHz, and 20GB memory.

5.1. *Verification of the proposed algorithm using benchmark images inpainting*

We prepared two missing formats of the images. One of them is randomly missing, and the other is missing with special structures. The sampling rate of those images was set as $p = 0.5$, a light blurring. Figure 2 and Table 1 have reported the experimental results obtained after the completion of three incomplete images for four algorithms. Evidently, the result obtained by the proposed algorithm 1 was better than that by other algorithms. Table 1 shows the SSIM, PSNR, RSE, and CPU(s) for these comparisons, where the best RSE, PSNR, CPU(s) and SSIM values are emphasized in bold font.

Furthermore, it is noted that all four algorithms have completed the filling task well for two missing formats by Figure 2 and Table 1. The proposed algorithm and algorithm 1 of [31] exhibit better visual effects for the structured missing case *facade 2*. Due to their inclusion of MDT, their advantage lies not in computation time but in the completion effect.

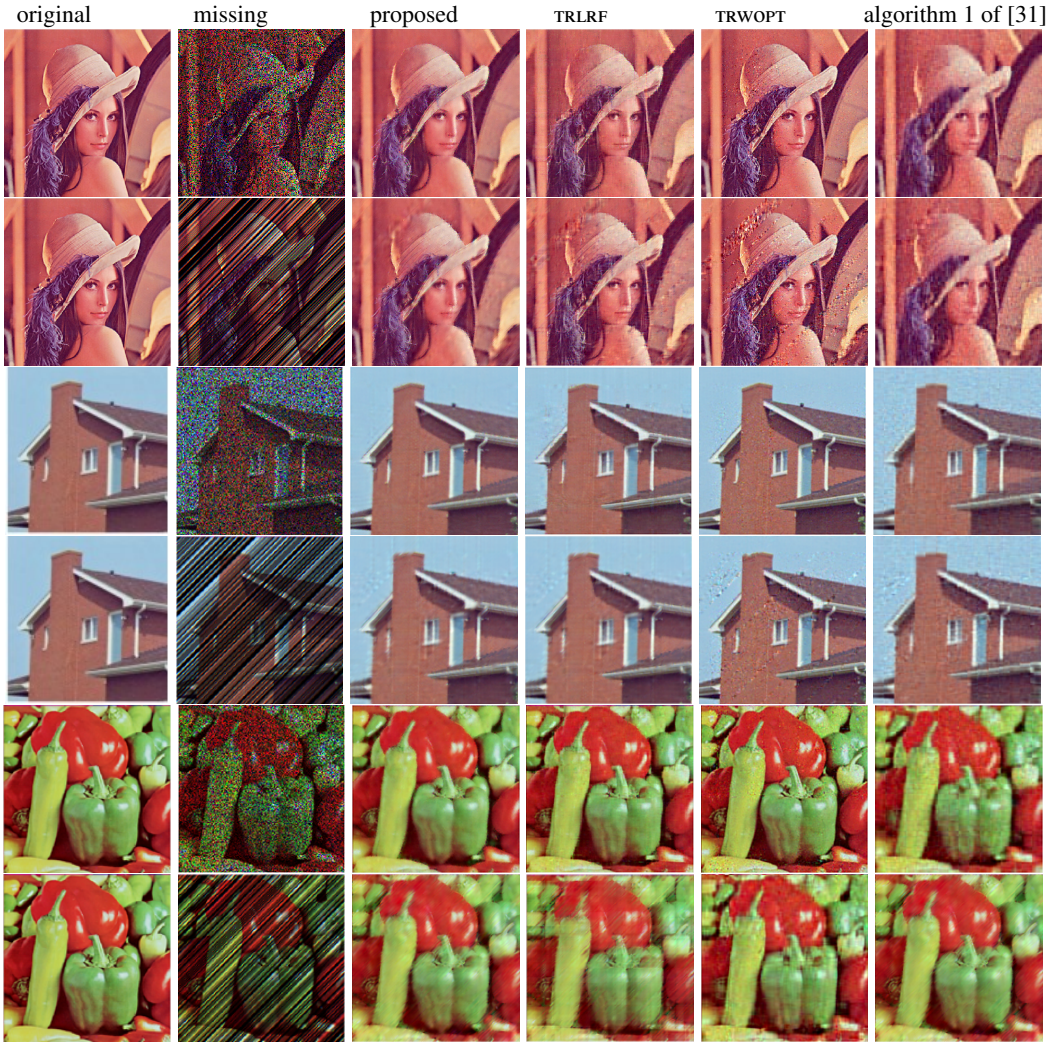

Figure 2. Color images completed with various algorithms. Two missing color images were artificially generated comprising: random missing case *facade 1*, structured missing case *facade 2*.

Table 1. Computational results for the missing images.

image	missing	item	proposed	TRLRF	TRWOPT	alg.1 of [31]
Lena	facade 1	SSIM	**0.8790**	0.7850	0.7650	0.8025
		PSNR	**27.028**	25.373	26.217	26.572
		RSE	**0.0800**	0.0970	0.0920	0.0730
		CPU(s)	8.4800	**6.3297**	6.4905	8.4411
	facade 2	SSIM	**0.8530**	0.7470	0.6130	0.7344
		PSNR	**25.665**	24.495	22.181	24.027
		RSE	**0.0920**	0.1080	0.1470	0.1338
		CPU(s)	6.1768	6.97 50	6.8912	**6.1020**
House	facade 1	SSIM	**0.9030**	0.8730	0.8170	0.8820
		PSNR	**27.587**	26.922	26.913	27.307
		RSE	**0.0710**	0.0750	0.0770	0.0723
		CPU(s)	9.0821	**6.4911**	6.5720	9.1208
	facade 2	SSIM	**0.8580**	0.8030	0.6870	0.8117
		PSNR	**26.937**	25.451	24.313	26.229
		RSE	**0.0790**	0.0890	0.1050	0.0823
		CPU(s)	**6.3728**	7.0957	7.1120	6.4432
Peppers	facade 1	SSIM	**0.8730**	0.8010	0.7130	0.8331
		PSNR	**26.678**	25.449	24.899	26.187
		RSE	0.0920	0.1070	0.1260	**0.0923**
		CPU(s)	11.0748	**9.2470**	9.4965	10.3912
	facade 2	SSIM	0.8480	0.7560	0.5920	**0.8510**
		PSNR	26.321	24.285	21.691	**26.556**
		RSE	**0.1070**	0.1360	0.1760	0.1121
		CPU(s)	10.3477	9.0231	**9.0020**	10.0015

5.2. Comparison using benchmark images inpainting for row/column fibres missing

In order to compare the performance of the new algorithm in solving images inpainting with row/column fibers missing, we prepared three row/column fibers missing formats of the images mentioned above for this experiment, which are shown in Figures 3–5. The missing rates of the image were set as p =0.3, 0.4, 0.5, and the corresponding missing fibers in the row and column are $p_r = 20, 10, 5$ and p_c =24,12,6, respectively.

From the visual effect of Figures 3–5 and the results recorded in Table 2. It is clear that the result obtained by our algorithm 1 and the algorithm 1 of [31] were much better than that of the other two algorithms without MDT. That is to say, our algorithm 1 and the algorithm 1 of [31] are more suitable for special cases of the tensor completion. Table 2 shows the SSIM, PSNR, and RSE for these comparisons, where the best RSE, PSNR, and SSIM values are emphasized in bold font.

Table 2. Completed results of three images for the different missing rate.

image	missing	item	proposed	TRLRF	TRWOPT	alg.1 of [31]
Lena	p=0.3	SSIM	**0.9499**	0.9303	0.9039	0.9019
	p_r=20	PSNR	**24.2172**	23.6465	22.3610	22.2662
	p_c=24	RSE	**0.0920**	0.1388	0.1403	0.1196
	p=0.4	SSIM	**0.9630**	0.9394	0.9383	0.9525
	p_r=10	PSNR	**26.181**	25.1286	25.0333	25.7863
	p_c=12	RSE	**0.0870**	0.1014	0.1024	0.0936
	p=0.5	SSIM	**0.9733**	0.8717	0.8130	0.9721
	p_r=5	PSNR	**28.7825**	28.1753	28.0181	28.7815
	p_c=6	RSE	**0.0606**8	0.0693	0.0740	0.0636
House	p=0.3	SSIM	**0.8808**	0.8538	0.8525	0.8170
	p_r=20	PSNR	**24.7000**	23.5651	23.5148	24.0913
	p_c=24	RSE	**0.0963**	0.1180	0.1186	0.1007
	p=0.4	SSIM	0.9179	0.8820	0.8819	**0.9670**
	p_r=10	PSNR	25.3162	25.0296	25.0206	**25.4313**
	p_c=12	RSE	0.0809	0.0923	0.0894	**0.0795**
	p=0.5	SSIM	**0.9583**	0.9417	0.9416	0.9430
	p_r=5	PSNR	**27.9485**	27.6135	27.6093	27.7811
	p_c=6	RSE	**0.0521**	0.0683	0.0683	0.0570
Peppers	p=0.3	SSIM	**0.9818**	0.8843	0.8847	0.9130
	p_r=20	PSNR	**30.3795**	22.3312	22.3587	28.8990
	p_c=24	RSE	**0.0551**	0.1393	0.1388	0.0860
	p=0.4	SSIM	0.9218	0.8824	0.9188	**0.9208**
	p_r=10	PSNR	24.5184	22.0556	23.4397	**24.691**
	p_c=12	RSE	0.1124	0.1576	0.1393	**0.1060**
	p=0.5	SSIM	**0.9588**	0.9237	0.9528	0.9536
	p_r=5	PSNR	**25.9665**	24.8717	25.1861	25.7920
	p_c=6	RSE	**0.0818**	0.1104	0.1114	0.0847

Figure 3. The completed Lena for $p = 0.5$ (Top), $p = 0.4$ (middle) and $p = 0.3$ (Bottom).

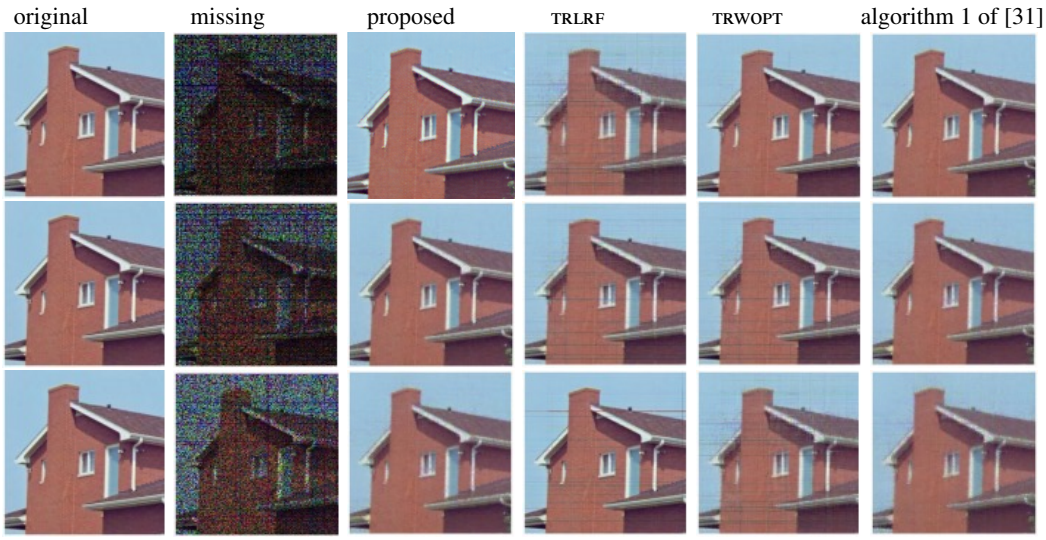

Figure 4. The completed house for $p = 0.5$ (Top), $p = 0.4$ (middle) and $p = 0.3$ (Bottom).

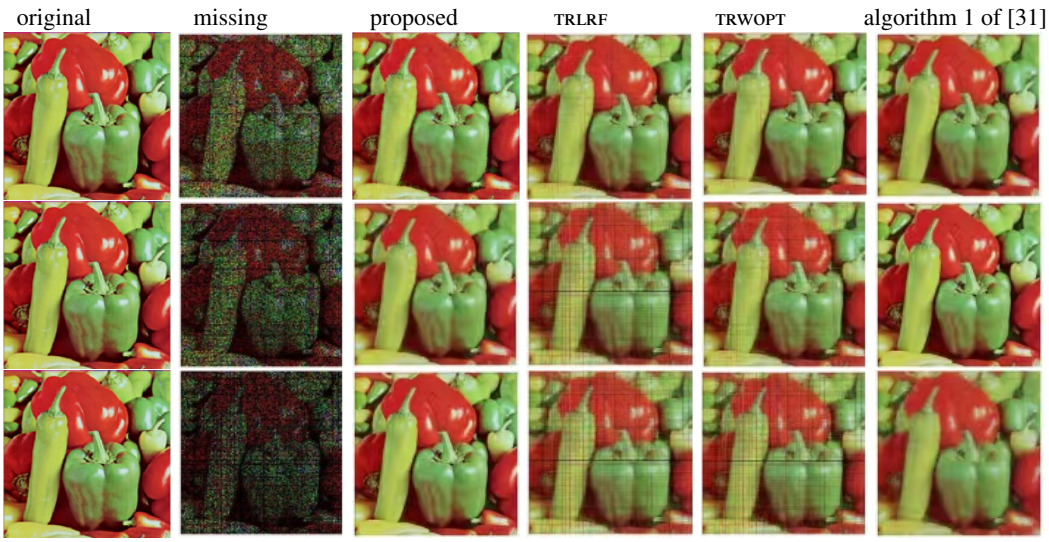

Figure 5. The completed peppers for $p = 0.5$ (Top), $p = 0.4$ (middle) and $p = 0.3$ (Bottom).

5.3. Comparison using Lena's image for missing slice

To compare the performance of the proposed algorithm with the other algorithms for solving image completion problems with a slice missing, we prepared Lena's images with badly random missing; the lower sampling rates of the image were badly set as 0.01, 0.03, and 0.05, respectively.

From the visual effect of Figure 6 and the results recorded in Table 3. It is noted that the results obtained by our algorithm 1 and algorithm 1 of [31] were much better than those by two other algorithms without MDT. That is to say, our algorithm 1 and algorithm 1 of [31] are more suitable for this special case of the tensor completion problems. Table 3 shows the SSIM, PSNR, and RSE for these comparisons, where the best RSE, PSNR, and SSIM values are emphasized in bold font.

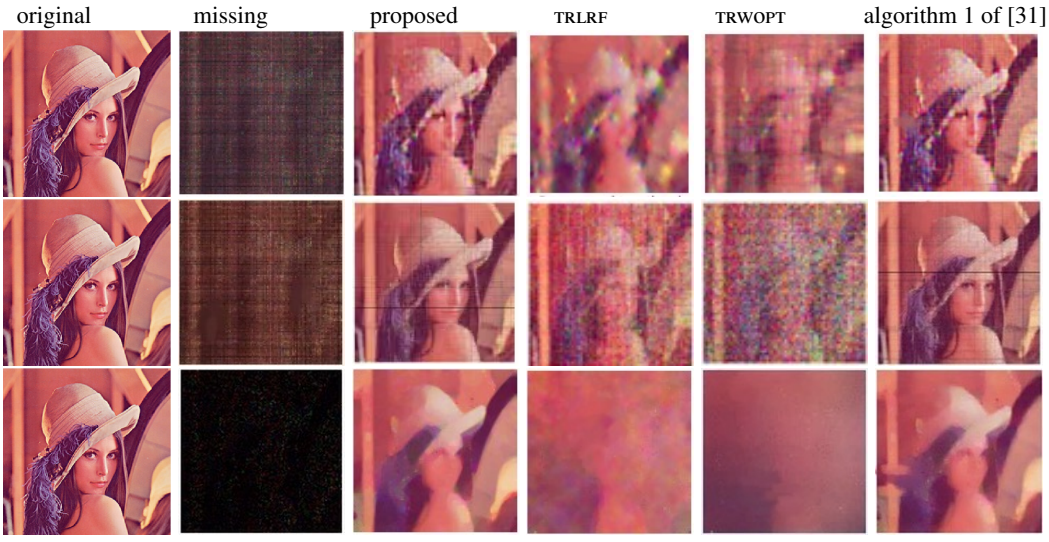

Figure 6. The completed Lena for $p = 0.05$ (Top), $p = 0.03$ (middle) and $p = 0.01$ (Bottom).

Table 3. Completed results of missing Lena for the different missing rate.

missing	item	proposed	TRLRF	TRWOPT	alg.1 of [31]
	SSIM	**0.6729**	0.5720	0.5581	0.6631
p=0.05	PSNR	**21.5228**	18.0491	19.6552	21.0217
	RSE	**0.5080**	0.7901	0.8312	0.5126
	SSIM	**0.8530**	0.6470	0.5840	0.8130
p=0.03	PSNR	**22.6065**	19.4995	18.9337	21.9181
	RSE	**0.0920**	0.8080	0.8869	0.1014
	SSIM	**0.8530**	0.4470	0.4584	0.8555
p=0.01	PSNR	**16.665**	14.4956	13.9897	16.181
	RSE	**0.2920**	0.9780	0.9921	0.3470

6. Conclusions

This paper focuses on the problem of completing a tensor with structured missing or missing slices or missing row/column fibers. In the proposed algorithm, embedding structured data in a tensor for solving a non-convex optimization model of tensor completion problems by multi-way delay-embedding transform (MDT) makes the tensor to be completed with a special structure. We have introduced an efficient and high-performance tensor completion algorithm based on the tensor ring rank, constructing latent tensor ring factors with Hankel structure the approximates the original tensor starting from the tensor structure. It has been shown to effectively handle model selection and provide much better completion results. This has been demonstrated from the extensive experiments on images inpainting.

The proposed algorithm has many potential extensions, such as using different embedding transformations in step 1 of our algorithm 1 or different approximated methods in step 2 of our algorithm 1.

Authors' contributions

Ruiping Wen: Conceptualization, Writing-original draft, Writing-review & editing; Tingyan Liu: Writing-original draft, Writing-review & editing; Yalei Pei: Writing-original draft, Writing-review & editing. All authors read and approved the final manuscript.

Use of Generative-AI tools declaration

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

Acknowledgments

The authors are very much indebted to the anonymous referees for their helpful comments and suggestions, which greatly improved the original manuscript of this paper. The authors are so thankful for the support from the NSF of China (12371381), the support from the NSF of

Shanxi (202403021222270) and Shanxi Higher Education Science and Technology Innovation Project (2023L239, 2023L243).

Conflict of interest

The authors declare that they have no conflict of interests.

References

1. A. E. Water, A. C. Sankaranarayanan, R. G. Baraniuk, SpaRCS: recovering low-rank and sparse matrices from compressive measurements, In: *NIPS'11: Proceedings of the 25th international conference on neural information processing systems*, 2011, 1089–1097.
2. M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, Image inpainting, In: *SIGGRAPH '00: Proceedings of the 27th annual conference on computer graphics and interactive techniques*, 2000, 417–424. https://doi.org/10.1145/344779.344972
3. N. Komodakis, Image completion using global optimization, In: *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, IEEE, New York, NY, USA, 17–22 June 2006, 442–452. https://doi.org/10.1109/CVPR.2006.141
4. M. Mørup, Applications of tensor (multiway array) factorizations and decompositions in data mining, *WIRS Data Min. Knowl.*, **1** (2011), 24–40. https://doi.org/10.1002/widm.1
5. P. Comon, X. Luciani, A. L. F. de Almeida, Tensor decompositions, alternating least squares and other tales, *J. Chemom.*, **23** (2009), 393–405. https://doi.org/10.1002/cem.1236
6. Q. Q. Shi, J. M. Yin, J. J. Cai, A. Cichocki, T. Yokota, L. Chen, et al., Block Hankel tensor ARIMA for mnltiple short time series forecasting, *Proceedings of the AAAI Conference on Artificial Intelligence*, **34** (2020), 5758–5766. https://doi.org/10.1609/aaai.v34i04.6032
7. M. Signoretto, R. Van de Plas, B. De Moor, J. A. K. Suykens, Tensor versus matrix completion: a comparison with application to spectral data, *IEEE Signal Process. Lett.*, **18** (2011), 403–406. https://doi.org/10.1109/LSP.2011.2151856
8. J. Liu, P. Musialski, P. Wonka, J. P. Ye, Tensor completion for estimating missing values in visual data, *IEEE Trans. Pattern Anal. Mach. Intell.*, **35** (2013), 208–220. https://doi.org/10.1109/TPAMI.2012.39
9. F. L. Hitchcock, The expression of a tensor or a polyadic as a sum of products, *Journal of Mathematics and Physics*, **6** (1927), 164–189. https://doi.org/10.1002/sapm192761164
10. L. R. Tucker, R. F. Koopman, R. L. Linn, Evaluation of factor analytic research procedures by means of simulated correlation matrices, *Psychometrika*, **34** (1969), 421–459. https://doi.org/10.1007/BF02290601
11. I. V. Oseledets, Tensor-train decomposition, *SIAM J. Sci. Comput.*, **33** (2011), 2295–2317. https://doi.org/10.1137/090752286
12. Q. B. Zhao, G. X. Zhou, S. L. Xie, L. Q. Zhang, A. Cichocki, Tensor ring decomposition, arXiv:1606.05535.

13. Y. Y. Xu, W. T. Yin, A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion, *SIAM Journal Imaging Sci.*, **6** (2013), 1758–1789. https://doi.org/10.1137/120887795

14. Y. Y. Liu, F. H. Shang, L. C. Jiao, J. Cheng, H. Cheng, Trace norm regularized candecomp/parafac decomposition with missing data, *IEEE Trans. Cybernetics*, **45** (2015), 2437–2448. https://doi.org/10.1109/TCYB.2014.2374695

15. J. A. Bengua, H. N. Phien, H. D. Tuan, M. N. Do, Efficient tensor completion for color image and video recovery: low-rank tensor train, *IEEE Trans. Image Process.*, **26** (2017), 2466–2479. https://doi.org/10.1109/TIP.2017.2672439

16. Z. M. Zhang, S. Aeron, Exact tensor completion using t-svd, *IEEE Trans. Signal Process.*, **65** (2017), 1511–1526. https://doi.org/10.1109/TSP.2016.2639466

17. Q. B. Zhao, L. Q. Zhang, A. Cichocki, Bayesian CP factorization of incomplete tensors with automatic rank determination, *IEEE Trans. Pattern Anal. Mach. Intell.*, **37** (2015), 1751–1763. https://doi.org/10.1109/tpami.2015.2392756

18. X. Y. Chen, Z. C. He, L. J. Sun, A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation, *Transport. Res. C: Emer.*, **98** (2019), 73–84. https://doi.org/10.1016/j.trc.2018.11.003

19. Z. Long, C. Zhu, J. N. Liu, Y. P. Liu, Bayesian low rank tensor ring for image recovery, *IEEE Trans. Image Process.*, **30** (2021), 3568–3580. https://doi.org/10.1109/TIP.2021.3062195

20. L. H. Yuan, J. T. Cao, X. Y. Zhao, Q. Wu, Q. B. Zhao, Higher-dimension tensor completion via low-rank tensor ring decomposition, In: *2018 Asia-Pacific signal and information processing association annual summit and conference (APSIPA ASC)*, IEEE, Honolulu, HI, USA, 12–15 November 2018, 1071–1076. https://doi.org/10.23919/APSIPA.2018.8659708

21. Q. B. Zhao, M. Sugiyama, L. H. Yuan, A. Cichocki, Learning efficient tensor representations with ring-structured networks, In: *ICASSP 2019 - 2019 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, Brighton, UK, 12–17 May 2019, 8608–8612. https://doi.org/10.1109/ICASSP.2019.8682231

22. J. Z. Xue, Y. Q. Zhao, S. G. Huang, W. Z. Liao, J. C.-W. Chan, S. G. Kong, Multilayer sparsity-based tensor decomposition for low-rank tensor completion, *IEEE Trans. Neur. Net. Learn. Syst.*, **33** (2022), 6916–6930. https://doi.org/10.1109/tnnls.2021.3083931

23. J. Z. Xue, Y. Q. Zhao, Y. Y. Bu, J. C.-W. Chan, S. G. Kong, When Laplacian scale mixture meets three-layer transform: a parametric tensor sparsity for tensor completion, *IEEE Trans. Cybernetics*, **52** (2022), 13887–13901. https://doi.org/10.1109/TCYB.2021.3140148

24. J. Z. Xue, Y.-Q. Zhao, T. L. Wu, J. C.-W. Chan, Tensor convolution-like low-rank dictionary for high-dimensional image representatio, *IEEE Trans. Circ. Syst. Vid.*, **34** (2024), 13257–13270. https://doi.org/10.1109/TCSVT.2024.3442295

25. T. Ding, M. Sznaier, O. I. Camps, A rank minimization approach to video inpainting, In: *2007 IEEE 11th international conference on computer vision*, IEEE, Riode Janeiro, Brazil, October 2007, 1–8. https://doi.org/10.1109/ICCV.2007.4408932

26. E. N. Lorenz, Deterministic nonperiodic flow, *J. Atmos. Sci.*, **20** (1963), 130–141. https://doi.org/10.1175/1520-0469(1963)020¡0130:DNF¿2.0.CO;2

27. I. Markovsky, Structured low-rank approximation and its applications, *Automatica*, **44** (2008), 891–909. https://doi.org/10.1016/j.automatica.2007.09.011
28. P. Van Overschee, B. De Moor, Subspace algorithms for the stochastic identification problem, In: *[1991] Proceedings of the 30th IEEE conference on decision and control*, IEEE, Brighton, UK, 11–13 December 1991, 1321–1326. https://doi.org/10.1109/CDC.1991.261604
29. Y. Li, K. J. R. Liu, J. Razavilar, A parameter estimation scheme for damped sinusoidal signals based on low-rank Hankel approximation, *IEEE Trans. Signal Process.*, **45** (1997), 481–486. https://doi.org/10.1109/78.554314
30. W. Q. Wang, V. Aggarwal, S. Aeron, Efficient low rank tensor ring completion, In: *2017 IEEE international conference on computer vision (ICCV)*, IEEE, Venice, Italy, 22–29 October 2017, 5698–5706. https://doi.org/10.1109/ICCV.2017.607
31. T. Yokota, B. Erem, S. Guler, S. K. Warfield, H. Hontani, Missing slice recovery for tensors using a low-rank model in embedded space, In: *2018 IEEE/CVF conference on computer vision and pattern recognition*, IEEE, Salt Lake City, UT, USA, 18–23 June 2018, 8251–8259. https://doi.org/10.1109/CVPR.2018.00861
32. Y. Li, K. J. R. Liu, J. RazavilarK, A parameter estimation scheme for damped sinusoidal signals based on low-rank Hankel approximation, *IEEE Trans. Signal Process.*, **45** (1997), 481–486. https://doi.org/10.1109/78.554314
33. T. G. Kolda, B. W. Bader, Tensor decompositions and applications, *SIAM Rev.*, **51** (2009), 455–500. https://doi.org/10.1137/07070111X
34. G. H. Golub, C. F. Van Loan, *Matrix computations*, 4 Eds., Johns Hopkins University Press, 2014.
35. J. F. Cai, E. J. Candès, Z. W. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Ooptimiz.*, **20** (2010), 1956–1982. https://doi.org/10.1137/080738970
36. C. J. Hillar, L.-H. Lim, Most tensor problems are NP-hard, *J. ACM*, **60** (2013), 45. https://doi.org/10.1145/2512329
37. M. Signoretto, Q. T. Dinh, L. De Lathauwer, J. A. K. Suykens, Learning with tensors: a framework based on convex optimization and spectral regularization, *Mach. Learn.*, **94** (2014), 303–351. https://doi.org/10.1007/s10994-013-5366-3
38. L. H. Yuan, C. Li, D. Mandic, J. T. Cao, Q. B. Zhao, Tensor ring decomposition with rank minimization on latent space: an efficient approach for tensor completion, *Proceedings of the AAAI Conference on Artificial Intelligence*, **33** (2019), 9151–9158. https://doi.org/10.1609/aaai.v33i01.33019151
39. Z. C. Lin, M. M. Chen, Y. Ma, The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices, arXiv:1009.5055.
40. Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.*, **13** (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861

AIMS Press