Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 17, No. 4, pp. 595–615, 2021

ADAPTIVE MULTI-DIMENSIONAL LOW-RANK BALANCE METHODS FOR TENSOR COMPLETION PROBLEMS

Chenjian Pan, Chen Ling* and Hongjin He

Abstract: This paper proposes two tensor product based approaches to tensor completion, which recovers missing entries of data represented by tensors. The proposed approaches are based on the tensor singular value decomposition and the related tensor tubal rank, which are able to capture hidden information from tensors thanks to the balanced consideration of multi-dimensional low-rank features. Accordingly, new optimization formulations for tensor completion are proposed as well as two new algorithms for their solutions. Some computational results for color images, multi-spectral images, videos, and magnetic resonance imaging data recovery show that our approaches perform better than some existing state-of-the-art tensor-based completion methods.

Key words: *low-rank tensor completion, tensor singular value decomposition, tensor tubal rank, weighted tensor nuclear norm, alternating direction method of multipliers*

Mathematics Subject Classification: *15A18, 15A69, 65F15, 90C33*

1 Introduction

The tensor is a multidimensional array, which can better express the complex essential structures of higher-order data arising from many applications, such as color image processing [4, 5, 6, 19], multi-spectral image (MSI) processing [1], signal processing [28], machine learning [28], and magnetic resonance imaging (MRI) data recovery [2, 30], video recovery [29], internet data completion [39], face recognition [10], to name just a few. Many of real-world problems need to be formulated as a tensor completion problem, i.e., completing the underlying tensor from an incomplete observation tensor, since it is almost impossible to collect complete information due to various reasons. The key of the tensor completion is to explore the prior information, e.g., the low-rank property of the underlying tensor. Mathematically, the low-rank tensor completion (LRTC) model can be generally expressed as follows

$$\begin{array}{ll} \min\limits_{\mathcal{X}} & \mathrm{rank}_\diamond(\mathcal{X}) \\ \text{s.t.} & \mathscr{P}_\Omega(\mathcal{X}) = \mathscr{P}_\Omega(\mathcal{H}), \end{array} \tag{1.1}$$

where $\mathrm{rank}_\diamond(\mathcal{X})$ denotes a special type of the tensor rank based on the rank assumption of the underlying tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$, $\mathcal{H} = (h_{i_1 i_2 \ldots i_N})$ is an observed incomplete tensor in $\mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$, Ω is the index set corresponding to the observed entries of $\mathcal{H}$, and $\mathscr{P}_\Omega$ is

*Corresponding author.

the linear operator extracting known elements in the set Ω and filling the elements that are not in Ω with zero values, i.e.,

$$(\mathscr{P}_\Omega(\mathcal{X}))_{i_1 i_2 \ldots i_N} = \begin{cases} h_{i_1 i_2 \ldots i_N}, & \text{if } (i_1, i_2, \ldots, i_N) \in \Omega; \\ 0, & \text{otherwise.} \end{cases} \tag{1.2}$$

It is well-known that the matrix completion problem is actually a special class of the tensor completion problem, i.e., the tensor completion problem can be viewed as an extension of the matrix completion problem. However, unlike matrix rank, the definition of the tensor rank is not unique. Many research efforts have been devoted to defining the tensor rank, and most of them are defined based on the corresponding tensor decomposition, such as the CANDECOMP/PARAFAC (CP) rank based on the CP decomposition [17, 23, 22, 31, 34], the Tucker rank based on the Tucker decomposition [8, 17, 21, 33], and the tensor tubal rank based on the tensor singular value decomposition (t-SVD) [18].

For a tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$, the CP rank is defined as the minimum number of rank-one tensors required to express a tensor [17], i.e.,

$$\operatorname{rank}_{cp}(\mathcal{X}) := \min \left\{ r \mid \mathcal{X} = \sum_{i=1}^{r} \mathbf{a}_i^1 \circ \mathbf{a}_i^2 \circ \cdots \circ \mathbf{a}_i^N \right\},$$

where '$\circ$' denotes the vector outer product. Notice that it is hard to establish a solvable relaxation form for the CP rank. The Tucker rank of $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$ is defined as a vector, i.e.,

$$\operatorname{rank}_{tc}(\mathcal{X}) := \left(\operatorname{rank}(X_{(1)}), \operatorname{rank}(X_{(2)}), \ldots, \operatorname{rank}(X_{(N)})\right),$$

where $X_{(i)}$ is the mode-i unfolding matrix of $\mathcal{X}$. To efficiently minimize (1.1) with Tucker rank function $\operatorname{rank}_{tc}(\mathcal{X})$, based on the nuclear norm concept of matrices introduced in [7], Liu et al. [21] proposed a convex relaxation model for (1.1), which can be mathematically expressed by

$$\begin{aligned} \min_{\mathcal{X}} \quad & \sum_{i=1}^{N} \alpha_i \|X_{(i)}\|_* \\ \text{s.t.} \quad & \mathscr{P}_\Omega(\mathcal{X}) = \mathscr{P}_\Omega(\mathcal{H}), \end{aligned} \tag{1.3}$$

where $\alpha_i \geq 0$ $(i = 1, 2, \cdots, N)$ satisfy $\sum_{i=1}^{N} \alpha_i = 1$ and $\|X_{(i)}\|_*$ represents the nuclear norm of the mode-i unfolding matrix $X_{(i)}$. In the literature, model (1.3) has been applied to many applications, e.g., see [1, 9, 13, 23, 24, 27, 29, 33, 32, 37] for details. However, it is not difficult to see that model (1.3) still belongs to the matrix completion, which possibly destroys the correlation among different dimensions. In fact, Yuan and Zhang [35] proved that the matricization method for the tensor completion problems can only obtain a suboptimal solution.

In this paper, we propose two new tensor completion models, which are based on the tensor T-product introduced by Kilmer et al [18]. The first model is an adaptive weighted nuclear norm minimization model, and the second one is an adaptive p-shrinkage model. More precisely, the proposed models are both based on the t-SVD and the related tensor tubal rank. As noted in [16], unfolding a tensor into a matrix along one mode usually destroys the structure information along other modes. Consequently, model (1.3) cannot maximally preserve the intrinsic structure of the tensor. Comparatively, the tubal rank models based on t-SVD are more powerful to preserve the intrinsic structure of a tensor than model (1.3), e.g., see [12, 37]. However, their models are not enough ideal to address different correlations along different modes, especially the third mode. In our two t-SVD

based models, we employ a permutation strategy on the tensor so that each mode can be considered by a series of matrix SVDs in the Fourier domain, thereby efficiently balancing the low-rank properties (correlations) along each mode. Moreover, we impose Toeplitz matrices on the tensor for the purpose of characterizing spatio-temporal structure, which make our models more flexible for real-world tensor data sets. Due to the complicated structure of the underlying models, we first introduce appropriate auxiliary variables to reformulate models as a separable optimization problem. Then, following the sequential update spirit of the well-known Alternating Direction Method of Multipliers (ADMM), we propose two implementable ADMM-type algorithms. Some computational results for color images, multi-spectral images, videos, and magnetic resonance imaging data recovery show that our approaches perform better than some existing state-of-the-art tensor completion approaches in terms of achieving higher recovery accuracy, especially for the cases with low sample ratios.

The structure of this paper is as follows. In Section 2, we summarize some notations and recall some basic definitions including (weighted) nuclear norm of the tensor, p-shrinkage mapping and proximal operators. In Section 3, we first present two T-product based tensor completion models. Further, based on the augmented Lagrangian function, we propose two implementable algorithms for the underlying tensor completion models. In Section 4, we conduct the performance of our approaches on images and videos recovery from highly under-sampled data. Finally, some concluding remarks are stated in Section 5.

2 Notation and Preliminaries

In this section, we summarize some notations and definitions on t-SVD, p-shrinkage thresholding operator and weighted nuclear norm that will be used throughout this paper.

The space of all N-th order real tensors is denoted by $\mathbb{R}^{n_1\times n_2\times\cdots\times n_N}$, where the order of a tensor is also called way or mode. Given an N-th order $\mathcal{A}\in\mathbb{R}^{n_1\times n_2\times\cdots\times n_N}$, we denote the $(i_1,i_2,\ldots,i_N)$-th component of $\mathcal{A}$ by $a_{i_1i_2\ldots i_N}$. So the N-th order tensor $\mathcal{A}$ is also denoted by $\mathcal{A}=(a_{i_1i_2\ldots i_N})$. Throughout this paper, tensors of order $N\geq 3$ are denoted by calligraphical letters, e.g., $\mathcal{A},\mathcal{B},\ldots$. Generally, we use capital letters (e.g., $A,B,\ldots$), boldfaced lowercase letters (e.g., $\mathbf{a},\mathbf{b},\ldots$), and lowercase letters (e.g., $a,b,\ldots$) to denote matrices, vectors, and scalars, respectively. For any two N-th order tensors $\mathcal{A}=(a_{i_1i_2\ldots i_N})$ and $\mathcal{B}=(b_{i_1i_2\ldots i_N})$, the inner product between $\mathcal{A}$ and $\mathcal{B}$ is given by

$$\langle\mathcal{A},\mathcal{B}\rangle:=\sum_{i_1,i_2,\ldots,i_N}a_{i_1i_2\ldots i_N}b_{i_1i_2\ldots i_N}.$$

Consequently, the Frobenius norm of a tensor $\mathcal{A}$ associated with the above inner product is given by $\|\mathcal{A}\|_F=\sqrt{\langle\mathcal{A},\mathcal{A}\rangle}$. Given an N-th order tensor $\mathcal{A}$, the mode-n matricization (or unfolding) of $\mathcal{A}$ is denoted by $A_{(n)}$, and the $(i_1,i_2,\ldots,i_N)$-th entry of tensor $\mathcal{A}$ is mapped to the (i_n,j)-th entry of matrix $A_{(n)}$ in the lexicographical order, where

$$j=1+\sum_{1\leq l\leq N,l\neq n}(i_l-1)J_l\quad\text{with}\quad J_l=\prod_{1\leq t\leq l-1,t\neq n}n_t.$$

The k-mode (matrix) product of a tensor $\mathcal{X}\in\mathbb{R}^{n_1\times n_2\times\cdots\times n_N}$ with a matrix $U\in\mathbb{R}^{J\times n_k}$ is denoted by $\mathcal{X}\times_k U$, which is of size $n_1\times\cdots\times n_{k-1}\times J\times n_{k+1}\times\cdots\times n_N$. Elementwise, we have

$$(\mathcal{X}\times_k U)_{i_1\ldots i_{k-1}ji_{k+1}\ldots i_N}=\sum_{i_k=1}^{n_k}x_{i_1\ldots i_{k-1}i_ki_{k+1}\ldots i_N}u_{ji_k}.$$

Throughout this paper, $[m] := \{1, 2, \ldots, m\}$ for given positive integer number m.

2.1 T-Product

We use X_k or $X(:,:,k)$ to denote the k-th frontal slice of a third-order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. We represent the tensor $\bar{\mathcal{X}}$ in terms of $\mathcal{X}$ after performing Fast Fourier Transform (FFT) on each tube, i.e., $\bar{\mathcal{X}} = \text{fft}(\mathcal{X}, [], 3)$ and $\mathcal{X} = \text{ifft}(\bar{\mathcal{X}}, [], 3)$ in MATLAB, we denote $\mathscr{F}(\mathcal{X})$ and $\mathscr{F}^{-1}(\bar{\mathcal{X}})$ for simplicity. The identity tensor $\mathcal{I} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ is a tensor whose first frontal slice is an identity matrix and all other frontal slices are zero matrices. An f-diagonal tensor is a tensor whose frontal slices are all diagonal matrices. We denote $\mathcal{X}^H$ as the conjugate transpose of tensor $\mathcal{X}$, whose each frontal slices are conjugate transpose and then reversing the order of the transposed frontal slices 2 through n_3.

For a given third order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the block circulant matrix $\text{bcirc}(\mathcal{X}) \in \mathbb{R}^{n_1 n_3 \times n_2 n_3}$ and the block vectorization matrix $\text{bvec}(\mathcal{X}) \in \mathbb{R}^{n_1 n_3 \times n_2}$ are defined as

$$\text{bcirc}(\mathcal{X}) = \begin{bmatrix} X_1 & X_{n_3} & \cdots & X_3 & X_2 \\ X_2 & X_1 & \cdots & X_4 & X_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ X_{n_3-1} & X_{n_3-2} & \cdots & X_1 & X_{n_3} \\ X_{n_3} & X_{n_3-1} & \cdots & X_2 & X_1 \end{bmatrix}, \text{bvec}(\mathcal{X}) = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{n_3} \end{bmatrix},$$

respectively.

Definition 2.1. (T-product [18]] Given $\mathcal{A} \in \mathbb{R}^{n_1 \times k \times n_3}$ and $\mathcal{B} \in \mathbb{R}^{k \times n_2 \times n_3}$, the t-product $\mathcal{A} * \mathcal{B}$ is a third order tensor with size $n_1 \times n_2 \times n_3$ given by

$$\mathcal{A} * \mathcal{B} = \text{bvfold}\left(\text{bcirc}(\mathcal{A}) \cdot \text{bvec}(\mathcal{B})\right),$$

where "bvfold" is the inverse operator of "bvec".

A tensor $\mathcal{Q} \in \mathbb{R}^{n \times n \times n_3}$ is orthogonal if it satisfies $\mathcal{Q} * \mathcal{Q}^H = \mathcal{Q}^H * \mathcal{Q} = \mathcal{I}$.

Theorem 2.2 (Tensor singular-value decomposition (t-SVD) [18])**.** *A third order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ can be decomposed as $\mathcal{X} = \mathcal{U} * \mathcal{S} * \mathcal{V}^H$, where $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ and $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ are othogonal tensors and $\mathcal{S} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is an f-diagonal tensor.*

It is noteworthy that the t-SVD can also be written as an "economy size" decomposition as similar to matrix SVD, i.e., $\mathcal{X} = \mathcal{U}(:, 1:r, :) * \mathcal{S}(1:r, 1:r, :) * \mathcal{V}(:, 1:r, :)^H$ with $r := \min\{n_1, n_2\}$.

Definition 2.3 ([26])**.** For any given $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, its tensor nuclear norm (TNN) is defined by

$$\|\mathcal{X}\|_* = \frac{1}{n_3} \sum_{j=1}^{n_3} \|\bar{X}_j\|_* = \frac{1}{n_3} \sum_{j=1}^{n_3} \sum_{i=1}^{r} \sigma_i(\bar{X}_j),$$

and its weighted tensor nuclear norm (WTNN) [25] is defined by

$$\|\mathcal{X}\|_{*,W} = \frac{1}{n_3} \sum_{j=1}^{n_3} \sum_{i=1}^{r} w_{ij} \sigma_i(\bar{X}_j),$$

where $\sigma_i(\bar{X}_j)$ is the i-th largest singular value of $\bar{X}_j \in \mathbb{C}^{n_1 \times n_2}$ for $j \in [n_3]$, and $W = (w_{ij}), i \in [r], j \in [n_3]$ is a weight matrix, whose (i,j)th component, as suggested in [11], is given by

$$w_{ij} = \delta / (\sigma_i(\bar{X}_j) + \epsilon), i \in [r], j \in [n_3], \tag{2.1}$$

with $\delta \in \mathbb{R}^+$ being a constant and ϵ being a small value to avoid division by 0. In algorithmic implementation, we can iteratively update W. Particularly, TNN can be considered as a special case of WTNN with $w_{ij} = 1$ for any (i, j).

2.2 Shrinkage thresholding operator

Definition 2.4 (p-shrinkage thresholding operator [3])**.** For a given vector $x \in \mathbb{R}^n$, $\lambda > 0$, and $p \leq 1$, the p-shrinkage thresholding operator is defined by

$$\mathbf{pshrink}(x, \mu, p) := \operatorname{sign}(x) \odot \max\{|x| - \lambda^{2-p}|x|^{p-1}, 0\}, \tag{2.2}$$

where 'sign$(\cdot)$' and '$|\cdot|$' are the sign function and absolute value function in component-wise, respectively, and '$\odot$' represents the component-wise product between two vectors. In particular, when setting $p = 1$, the p-shrinkage operator (2.2) immediately reduces to the well-known soft-thresholding.

It follows from [24, 25] that, for any given $\mathcal{Y} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ and a weight matrix W, the related proximal operator associated to the weighted nuclear norm function $\|\mathcal{X}\|_{*,W}$ about $\mathcal{Y}$ is defined by

$$\operatorname{prox}_{W\lambda}(\mathcal{Y}) := \underset{\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}}{\arg\min} \left\{ \|\mathcal{X}\|_{*,W} + \frac{1}{2\lambda}\|\mathcal{X} - \mathcal{Y}\|_F^2 \right\},$$

which is given by the following formula

$$\operatorname{prox}_{W\lambda}(\mathcal{Y}) = \mathcal{U} * \mathscr{F}^{-1}(\mathbf{wTshrink}(\mathcal{S}, \lambda, W)) * \mathcal{V}^H, \tag{2.3}$$

where $\mathcal{Y} = \mathcal{U} * \mathcal{S} * \mathcal{V}^H$ is the t-SVD of $\mathcal{Y}$ and $\mathbf{wTshrink}(\mathcal{S}, \lambda, W) \in \mathbb{R}^{r \times r \times n_3}$ is defined by

$$\mathbf{wTshrink}(\mathcal{S}, \lambda, W)_{iij} = \max\{\sigma_i(\bar{S}_j) - \lambda w_{ij}, 0\},\ i \in [r],\ j \in [n_3]. \tag{2.4}$$

Moreover, it follows from [3] that the p-shrinkage mapping defined in Definition 2.4 can also be interpreted as the proximal operator of a penalty function $\Phi_p(\cdot) : \mathbb{R}^n \to \mathbb{R}$, i.e.,

$$\mathbf{pshrink}(a, \lambda, p) = \underset{x \in \mathbb{R}^n}{\arg\min}\, \Phi_p(x) + \frac{1}{2\lambda}\|x - a\|^2,$$

where $\Phi_p(x) := \sum_{i=1}^n \phi_p(x_i)$ with $\phi_p(x_i)$ being even, concave, nondecreasing and continuous on $[0, \infty]$, differentiable on $(0, \infty)$, nondifferentiable at 0 with the subdifferential being $\partial\phi_p(0) = [-1, 1]$. We refer the reader to [3, Theorem 1] (also [27]) for more details. Notice that the p-shrinkage operator is available to tensor variables, i.e., $\Phi_p(\mathcal{X}) = \frac{1}{n_3}\sum_{j=1}^{n_3} \Phi_p(\bar{X}_j)$. Specifically, for the optimization problem

$$\min_{\mathcal{X}} \Phi_p(\mathcal{X}) + \frac{1}{2\lambda}\|\mathcal{X} - \mathcal{Y}\|_F^2,$$

we have the globally optimal solution

$$\mathcal{X}^* = \mathcal{U} * \mathscr{F}^{-1}(\mathbf{pTshrink}(\mathcal{S}, \lambda, p)) * \mathcal{V}^H,$$

where $\mathcal{Y} = \mathcal{U} * \mathcal{S} * \mathcal{V}^H$ is the t-SVD of $\mathcal{Y}$ and $\mathbf{pTshrink}(\mathcal{S}, \lambda, p) \in \mathbb{R}^{r \times r \times n_3}$ is defined by

$$\mathbf{pTshrink}(\mathcal{S}, \lambda, p)_{iij} = \mathbf{pshrink}(\sigma_i(\bar{S}_j), \lambda, p), i \in [r], j \in [n_3]. \tag{2.5}$$

3 Models and Algorithms

In this section, we present two t-SVD-based tensor completion models and the corresponding algorithms. The first model is an adaptive weighted nuclear norm minimization model, and the second one is an adaptive p-shrinkage model. Before presenting the models, let us introduce the following notations. For a third-order tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we denote $\mathcal{X}_{\{1\}}$ as permute$(\mathcal{X}, [2,3,1])$, $\mathcal{X}_{\{2\}}$ as permute$(\mathcal{X}, [1,3,2])$, $\mathcal{X}_{\{3\}}$ as permute$(\mathcal{X}, [1,2,3])$, respectively.

3.1 Adaptive weighted nuclear norm minimization model

In this subsection, we use the weighted nuclear norm to approximate the rank of target tensor $\mathcal{X}$, and the role of each dimension is considered in a weighted way. Accordingly, we call the model adaptive weighted nuclear norm minimization model, which can be expressed as follows

$$\begin{aligned} \min_{\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}} \quad & \sum_{i=1}^{3} \alpha_i \|\mathcal{X}_{\{i\}}\|_{*,W_i} + \sum_{i=1}^{3} \rho_i \|\mathcal{X} \times_i L_i\|_F^2 \\ \text{s.t.} \quad & \mathscr{P}_\Omega(\mathcal{X}) = \mathscr{P}_\Omega(\mathcal{H}), \end{aligned} \tag{3.1}$$

where $\alpha = (\alpha_1, \alpha_2, \alpha_3)$ and $\rho = (\rho_2, \rho_2, \rho_3)$ are nonnegative regularization parameters satisfying $\sum_{i=1}^{3} \alpha_i \approx 1$, $\mathcal{H}$ is the incomplete data tensor with missing information, W_i is defined by (2.1) with $\mathcal{X} = \mathcal{X}_{\{i\}}$. Here, L_i ($i \in [3]$) are Toeplitz matrices for spatio-temporal regularizations, i.e., $L_i = \text{Toeplitz}(0, 1, -1)$, whose central diagonals are ones, the first upper diagonal components are -1, and the others are zeros. From model (3.1), it can be clearly seen that the first part of the objective function is non-smooth, but its proximal operator has a closed-form solution due to its special structure. Moreover, the variable $\mathcal{X}$ is in a highly coupled state. In order to design effective algorithms, we equivalently transform (3.1) into a separable model of the following form

$$\begin{aligned} \min_{\mathcal{G}, \mathcal{Z}, \mathcal{X}} \quad & \sum_{i=1}^{3} \alpha_i \|(\mathcal{G}_i)_{\{i\}}\|_{*,W_i} + \sum_{i=1}^{3} \rho_i \|\mathcal{Z}_i \times_i L_i\|_F^2 + \delta_{\mathbb{K}}(\mathcal{X}) \\ \text{s.t.} \quad & (\mathcal{G}_i)_{\{i\}} = \mathcal{X}_{\{i\}}, i \in [3], \\ & Z_{i,(i)} = X_{(i)}, i \in [3], \end{aligned} \tag{3.2}$$

where $\mathcal{G}_i$ and $\mathcal{Z}_i$ are some introduced intermediate variables, $(\mathcal{G}_1)_{\{1\}} \in \mathbb{R}^{n_2 \times n_3 \times n_1}$, $(\mathcal{G}_2)_{\{2\}} \in \mathbb{R}^{n_1 \times n_3 \times n_2}$ and $(\mathcal{G}_3)_{\{3\}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, W_i is defined by (2.1) with $\mathcal{X} = (\mathcal{G}_i)_{\{i\}}$, $Z_{i,(i)} = \text{unfold}(\mathcal{Z}_i)$ for $i \in [3]$, $\mathbb{K} = \{\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \mid \mathscr{P}_\Omega(\mathcal{X} - \mathcal{H}) = 0\}$, and $\delta_{\mathbb{K}}(\cdot)$ is the indicator function of $\mathbb{K}$ defined by

$$\delta_{\mathbb{K}}(\mathcal{X}) = \begin{cases} 0, & \text{if } \mathcal{X} \in \mathbb{K}; \\ \infty, & \text{otherwise.} \end{cases}$$

Since model (3.2) is an equality-constrained optimization problem, we write the augmented Lagrangian function as follows

$$\begin{aligned}
&\mathscr{L}_{\beta,\eta}(\mathcal{G},\mathcal{Z},\mathcal{X},\mathcal{T},\mathcal{P})\\
&:=\sum_{i=1}^{3}\left(\alpha_i\left\|(\mathcal{G}_i)_{\{i\}}\right\|_{*,W_i}+\langle\mathcal{T}_i,\mathcal{X}-\mathcal{G}_i\rangle+\frac{\beta_i}{2}\|\mathcal{X}-\mathcal{G}_i\|_F^2\right)+\delta_{\mathbb{K}}(\mathcal{X})\\
&\quad+\sum_{i=1}^{3}\left(\rho_i\|\mathcal{Z}_i\times_i L_i\|_F^2+\langle\mathcal{P}_i,\mathcal{X}-\mathcal{Z}_i\rangle+\frac{\eta_i}{2}\|\mathcal{X}-\mathcal{Z}_i\|_F^2\right)\\
&=\sum_{i=1}^{3}\left(\alpha_i\left\|(\mathcal{G}_i)_{\{i\}}\right\|_{*,W_i}+\left\langle(\mathcal{T}_i)_{\{i\}},\mathcal{X}_{\{i\}}-(\mathcal{G}_i)_{\{i\}}\right\rangle+\frac{\beta_i}{2}\left\|X_{(i)}-(\mathcal{G}_i)_{\{i\}}\right\|_F^2\right)\\
&\quad+\sum_{i=1}^{3}\left(\rho_i\left\|L_iZ_{i,(i)}\right\|_F^2+\left\langle P_{i,(i)},X_{(i)}-Z_{i,(i)}\right\rangle+\frac{\eta_i}{2}\left\|X_{(i)}-Z_{i,(i)}\right\|_F^2\right)+\delta_{\mathbb{K}}(\mathcal{X}),
\end{aligned}\tag{3.3}$$

where $\mathcal{G}=(\mathcal{G}_1,\mathcal{G}_2,\mathcal{G}_3)$, $\mathcal{Z}=(\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3)$, $\mathcal{T}=(\mathcal{T}_1,\mathcal{T}_2,\mathcal{T}_3)$ with $\mathcal{T}_i$ being the Lagrangian multiplier associated to $\mathcal{G}_i=\mathcal{X}$, $\mathcal{P}=(\mathcal{P}_1,\mathcal{P}_2,\mathcal{P}_3)$ with $\mathcal{P}_i$ being the Lagrangian multiplier associated to $\mathcal{Z}_i=\mathcal{X}$, and $\{\beta^k\}=\{(\beta_1^k,\beta_2^k,\beta_3^k)\}$ and $\{\eta^k\}=\{(\eta_1^k,\eta_2^k,\eta_3^k)\}$ are two non-decreasing sequences, i.e., $\{\beta_i^k\}_{k=0}^{\infty}$ and $\{\eta_i^k\}_{k=0}^{\infty}$ are non-decreasing for $i\in[3]$. Here, the last equation is due to the fact that

$$\langle\mathcal{A},\mathcal{B}\rangle=\langle\mathcal{A}_{\{i\}},\mathcal{B}_{\{i\}}\rangle=\langle\mathcal{A}_{(i)},\mathcal{B}_{(i)}\rangle\ \text{ and }\ \|\mathcal{A}\|_F=\|\mathcal{A}_{\{i\}}\|_F=\|A_{i,(i)}\|_F,\quad i\in[3].$$

By using $\mathscr{L}_{\beta,\eta}(\mathcal{G},\mathcal{Z},\mathcal{X},\mathcal{T},\mathcal{P})$, we present an ADMM iterative scheme for (3.2). Specially, for the latest variables $\mathcal{G}^k,\mathcal{X}^k,\mathcal{Z}^k,\mathcal{T}^k,\mathcal{P}^k$, we generate the next iterate in a sequential order, i.e., $\mathcal{G}^{k+1}\to\mathcal{Z}^{k+1}\to\mathcal{X}^{k+1}\to\mathcal{T}^{k+1}\to\mathcal{P}^{k+1}$ for updating the $(k+1)$-th iteration.

- Update the variable $\mathcal{G}=(\mathcal{G}_1,\mathcal{G}_2,\mathcal{G}_3)$. For every $i\in[3]$, obtain $(\mathcal{G}_i)_{\{i\}}^{k+1}$ via the following optimization problem:

$$\begin{aligned}
(\mathcal{G}_i)_{\{i\}}^{k+1}&=\underset{(\mathcal{G}_i)_{\{i\}}}{\arg\min}\,\mathscr{L}_{\beta^k,\eta^k}\big(\mathcal{G},\mathcal{Z}^k,\mathcal{X}^k,\mathcal{T}^k,\mathcal{P}^k\big)\\
&=\underset{(\mathcal{G}_i)_{\{i\}}}{\arg\min}\left\{\frac{\alpha_i}{\beta_i^k}\left\|(\mathcal{G}_i)_{\{i\}}\right\|_{*,W_i}+\frac{1}{2}\left\|\mathcal{X}_{\{i\}}^k-(\mathcal{G}_i)_{\{i\}}+\frac{1}{\beta_i^k}(\mathcal{T}_i)_{\{i\}}^k\right\|_F^2\right\}\\
&=\mathcal{U}*\mathscr{F}^{-1}\big(\mathbf{wTShrink}\big(\mathcal{S},\frac{\alpha_i}{\beta_i^k},W_i\big)\big)*\mathcal{V}^H,
\end{aligned}\tag{3.4}$$

  where $\mathcal{X}_{\{i\}}^k+\frac{1}{\beta_i^k}(\mathcal{T}_i)_{\{i\}}^k=\mathcal{U}*\mathcal{S}*\mathcal{V}^H$ and '$\mathbf{wTshrink}(\cdot,\cdot,\cdot)$' is given by (2.4), and the weight matrix W_i is updated by (2.1) for $i\in[3]$. Furthermore, $\mathcal{G}_i$ can be obtained $\mathcal{G}_i=\text{ipermute}((\mathcal{G}_i)_{\{i\}})$ by the MATLAB script function 'ipermute'.

- Update the variable $\mathcal{Z}=(\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3)$. For every $i\in[3]$, we compute $\mathcal{Z}_i^{k+1}$ via

$$\begin{aligned}
\mathcal{Z}_i^{k+1}&=\underset{\mathcal{Z}_i}{\arg\min}\,\mathscr{L}_{\beta^k,\eta^k}\big(\mathcal{G}^{k+1},\mathcal{Z},\mathcal{X}^k,\mathcal{T}^k,\mathcal{P}^k\big)\\
&=\underset{\mathcal{Z}_i}{\arg\min}\left\{\rho_i\|L_iZ_{i,(i)}\|_F^2+\frac{\eta_i^k}{2}\|X_{(i)}^k-\mathcal{Z}_{i,(i)}+\frac{1}{\eta_i^k}P_{i,(i)}^k\|_F^2\right\}\\
&=\text{fold}\left(\big(2\rho_iL_i^\top L_i+\eta_i^kI\big)^{-1}\big(\eta_i^kX_{(i)}^k+P_{i,(i)}^k\big)\right).
\end{aligned}\tag{3.5}$$

- Update the variable $\mathcal{X}$. After obtaining $\mathcal{G}^{k+1}$ and $\mathcal{Z}^{k+1}$, we obtain $\mathcal{X}^{k+1}$ via

$$\begin{aligned}\mathcal{X}^{k+1} &= \arg\min_{\mathcal{X}} \mathscr{L}_{\beta^k,\eta^k}\big(\mathcal{G}^{k+1},\mathcal{Z}^{k+1},\mathcal{X},\mathcal{T}^k,\mathcal{P}^k\big)\\ &= \arg\min_{\mathcal{X}} \sum_{i=1}^{3}\left(\langle \mathcal{T}_i^k, \mathcal{X}-\mathcal{G}_i^{k+1}\rangle + \frac{\beta_i^k}{2}\|\mathcal{X}-\mathcal{G}_i^{k+1}\|_F^2\right) + \delta_{\mathbb{K}}(\mathcal{X})\\ &\quad + \sum_{i=1}^{3}\left(\langle \mathcal{P}_i^k, \mathcal{X}-\mathcal{Z}_i^{k+1}\rangle + \frac{\eta_i^k}{2}\|\mathcal{X}-\mathcal{Z}_i^{k+1}\|_F^2\right)\\ &= \arg\min_{\mathcal{X}} \sum_{i=1}^{3}\frac{\beta_i^k}{2}\big\|\mathcal{X}-\mathcal{G}_i^{k+1}+\frac{1}{\beta_i^k}\mathcal{T}_i^k\big\|_F^2\\ &\quad + \delta_{\mathbb{K}}(\mathcal{X}) + \sum_{i=1}^{3}\frac{\eta_i^k}{2}\big\|\mathcal{X}-\mathcal{Z}_i^{k+1}+\frac{1}{\eta_i^k}\mathcal{P}_i^k\big\|_F^2. \end{aligned} \tag{3.6}$$

  From (3.6), we obtain a closed-form solution for $\mathcal{X}^{k+1}$ as follows

$$(\mathcal{X}^{k+1})_{ijl} = \begin{cases} h_{ijl}, & \text{if } (i,j,l)\in\Omega;\\ \hat{\mathcal{X}}^k{}_{ijl}/\tau_k, & \text{otherwise.}\end{cases} \tag{3.7}$$

  where $\hat{\mathcal{X}}^k = \sum_{i=1}^{3}\beta_i^k(\mathcal{G}_i^{k+1}-\frac{1}{\beta_i^k}\mathcal{T}_i^k)+\sum_{i=1}^{3}\eta_i^k(\mathcal{Z}_i^k-\frac{1}{\eta_i^k}\mathcal{P}_i^k)$ and $\tau_k=\sum_{i=1}^{3}(\beta_i^k+\eta_i^k)$.

- Update the variables $\mathcal{T}=(\mathcal{T}_1,\mathcal{T}_2,\mathcal{T}_3)$ and $\mathcal{P}=(\mathcal{P}_1,\mathcal{P}_2,\mathcal{P}_3)$ via

$$\mathcal{T}_i^{k+1}=\mathcal{T}_i^k+\beta_i^k\big(\mathcal{X}^{k+1}-\mathcal{G}_i^{k+1}\big) \quad\text{and}\quad \mathcal{P}_i^{k+1}=\mathcal{P}_i^k+\eta_i^k\big(\mathcal{X}^{k+1}-\mathcal{Z}_i^{k+1}\big), \tag{3.8}$$

  for $i\in[3]$, respectively.

Formally, we can summarize the iterative schemes for (3.2) in Algorithm 1.

Algorithm 1 ADMM for Tensor Completion Model (3.2)

Input: Observed tensor $\mathcal{H}$, parameters: $\alpha=(\alpha_1,\alpha_2,\alpha_3)$, $\rho=(\rho_1,\rho_2,\rho_3)$, $\beta^0=(\beta_1^0,\beta_2^0,\beta_3^0)$, $\eta^0=(\eta_1^0,\eta_2^0,\eta_3^0)$, δ, ϵ, κ_1, $\kappa_2>1$.

Step 0: Initialization: choose initial guess $(\mathcal{G}^0,\mathcal{Z}^0,\mathcal{X}^0,\mathcal{T}^0,\mathcal{P}^0)$. Set $k:=0$.
Step 1: Update $\mathcal{G}^{k+1}$ by (3.4),
Step 2: Update $\mathcal{Z}^{k+1}$ by (3.5),
Step 3: Update $\mathcal{X}^{k+1}$ by (3.7),
Step 4: Update $\mathcal{T}^{k+1}$ and $\mathcal{P}^{k+1}$ by (3.8),
Step 5: Update $\beta^{k+1}=\kappa_1\beta^k$ and $\eta^{k+1}=\kappa_2\eta^k$,
Step 6: Unless a termination criterion is fulfilled, set $k:=k+1$ and go to Step 1.

Output: Completed tensor $\mathcal{X}^*$.

3.2 Adaptive t-SVD-based p-shrinkage tensor completion model

In this subsection, we further consider another scheme, which approximates low-rankness adaptively. Recently, the so-called p-shrinkage thresholding algorithm [2, 3, 20, 27] outperforms the classical iterative soft thresholding algorithm induced by nuclear norm for low rank

and sparse recovery problems. Hence, we propose the following tensor completion problem model:

$$\begin{array}{ll}\min\limits_{\mathcal{X}} & \sum\limits_{i=1}^{3}\alpha_i\Phi_p(\mathcal{X}_{\{i\}})+\sum\limits_{i=1}^{3}\rho_i\|\mathcal{X}\times_i L_i\|_F^2\\ \text{s.t.} & \mathscr{P}_\Omega(\mathcal{X})=\mathscr{P}_\Omega(\mathcal{H}).\end{array}\tag{3.9}$$

Compared with model (3.1), the only difference is that the $\|\mathcal{X}_{\{i\}}\|_{*,W}$ is replaced by $\Phi_p(\mathcal{X}_{\{i\}})$, where $p\le 1$ and $\Phi_p(\mathcal{X}_{\{i\}})$ is defined in Section 2, and Ω is the index set for the known entries. By introducing intermediate variables $\mathcal{G}=(\mathcal{G}_1,\mathcal{G}_2,\mathcal{G}_3)$ and $\mathcal{Z}=(\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3)$, we can rewrite model (3.9) as follows

$$\begin{array}{ll}\min\limits_{\mathcal{X}} & \sum\limits_{i=1}^{3}\alpha_i\Phi_p((\mathcal{G}_i)_{\{i\}})+\sum\limits_{i=1}^{3}\rho_i\|\mathcal{Z}_i\times_i L_i\|_F^2+\delta_{\mathbb{K}}(\mathcal{X})\\ \text{s.t.} & (\mathcal{G}_i)_{\{i\}}=\mathcal{X}_{\{i\}}, i\in[3],\\ & Z_{i,(i)}=X_{(i)}, i\in[3],\end{array}\tag{3.10}$$

Taking a close look at model (3.10), it is also a linearly equality-constrained optimization problem. By introducing the Lagrangian multipliers $\mathcal{T}=(\mathcal{T}_1,\mathcal{T}_2,\mathcal{T}_3)$ and $\mathcal{P}=(\mathcal{P}_1,\mathcal{P}_2,\mathcal{P}_3)$ with penalty parameters $\beta=(\beta_1,\beta_2,\beta_3)$ and $\rho=(\rho_1,\rho_2,\rho_3)$ to linear constraints in (3.10), we can get its augmented Lagrangian function as follows

$$\begin{aligned}&\mathscr{L}_{\beta,\eta}(\mathcal{G},\mathcal{Z},\mathcal{X},\mathcal{T},\mathcal{P})\\ &:=\sum_{i=1}^{3}\left(\alpha_i\Phi_p((\mathcal{G}_i)_{\{i\}})+\left\langle(\mathcal{T}_i)_{\{i\}},\mathcal{X}_{\{i\}}-(\mathcal{G}_i)_{\{i\}}\right\rangle+\frac{\beta_i}{2}\left\|\mathcal{X}_{\{i\}}-(\mathcal{G}_i)_{\{i\}}\right\|_F^2\right)\\ &\quad+\sum_{i=1}^{3}\left(\rho_i\|L_iZ_{i,(i)}\|_F^2+\left\langle P_{i,(i)},X_{(i)}-Z_{i,(i)}\right\rangle+\frac{\eta_i}{2}\left\|X_{(i)}-Z_{i,(i)}\right\|_F^2\right)+\delta_{\mathbb{K}}(\mathcal{X}).\end{aligned}\tag{3.11}$$

Based on $\mathscr{L}_{\beta,\eta}(\mathcal{G},\mathcal{Z},\mathcal{X},\mathcal{T},\mathcal{P})$ defined by (3.11), we present an ADMM iterative scheme for (3.10) as follows.

- Update variable $\mathcal{G}=(\mathcal{G}_1,\mathcal{G}_2,\mathcal{G}_3)$. For every $i\in[3]$, we compute $(\mathcal{G}_i)_{\{i\}}^{k+1}$ via

$$\begin{aligned}(\mathcal{G}_i)_{\{i\}}^{k+1}&=\arg\min_{(\mathcal{G}_i)_{\{i\}}}\mathscr{L}_{\beta^k,\eta^k}\left(\mathcal{G},\mathcal{Z}^k,\mathcal{X}^k,\mathcal{T}^k,\mathcal{P}^k\right)\\ &=\arg\min_{(\mathcal{G}_i)_{\{i\}}}\left\{\frac{\alpha_i}{\beta_i^k}\Phi_p((\mathcal{G}_i)_{\{i\}})+\frac{1}{2}\left\|\mathcal{X}_{\{i\}}^k-(\mathcal{G}_i)_{\{i\}}+\frac{1}{\beta_i^k}(\mathcal{T}_i)_{\{i\}}^k\right\|_F^2\right\}\\ &=\mathcal{U}*\mathscr{F}^{-1}\left(\mathbf{pTshrink}\left(\mathcal{S},\frac{\alpha_i}{\beta_i^k},p\right)\right)*\mathcal{V}^H,\end{aligned}\tag{3.12}$$

  where $\mathcal{X}_{\{i\}}^k+\frac{1}{\beta_i^k}(\mathcal{T}_i)_{\{i\}}^k=\mathcal{U}*\mathcal{S}*\mathcal{V}^H$ and '$\mathbf{pTshrink}(\cdot,\cdot,\cdot)$' is given by (2.5). Furthermore, $\mathcal{G}_i$ can be obtained $\mathcal{G}_i=\text{ipermute}(\mathcal{G}_i)_{\{i\}}$ by the MATLAB script function 'ipermute'.

- Update the variable $\mathcal{Z}=(\mathcal{Z}_1,\mathcal{Z}_2,\mathcal{Z}_3)$. For every $i\in[3]$, we compute $\mathcal{Z}_i^{k+1}$ via

$$\begin{aligned}\mathcal{Z}_i^{k+1}&=\arg\min_{\mathcal{Z}_i}\mathscr{L}_{\beta^k,\eta^k}\left(\mathcal{G}^{k+1},\mathcal{Z},\mathcal{X}^k,\mathcal{T}^k,\mathcal{P}^k\right)\\ &=\arg\min_{\mathcal{Z}_i}\left\{\rho_i\|L_iZ_{i,(i)}\|_F^2+\frac{\eta_i^k}{2}\left\|X_{(i)}^k-\mathcal{Z}_{i,(i)}+\frac{1}{\eta_i^k}P_{i,(i)}^k\right\|_F^2\right\}\\ &=\text{fold}\left(\left(2\rho_iL_i^\top L_i+\eta_i^kI\right)^{-1}\left(\eta_i^kX_{(i)}^k+P_{i,(i)}^k\right)\right).\end{aligned}\tag{3.13}$$

- Update variable $\mathcal{X}$. We obtain $\mathcal{X}^{k+1}$ via following optimization problem:

$$\begin{aligned}\mathcal{X}^{k+1} &= \mathscr{L}_{\beta^k,\eta^k}\big(\mathcal{G}^{k+1},\mathcal{Z}^{k+1},\mathcal{X},\mathcal{T}^k,\mathcal{P}^k\big)\\ &= \arg\min_{\mathcal{X}} \sum_{i=1}^{3}\frac{\beta_i^k}{2}\big\|\mathcal{X}-\mathcal{G}_i^{k+1}+\frac{1}{\beta_i^k}\mathcal{T}_i^k\big\|_F^2+\delta_{\mathbb{K}}(\mathcal{X})\\ &\quad+\sum_{i=1}^{3}\frac{\eta_i^k}{2}\big\|\mathcal{X}-\mathcal{Z}_i^{k+1}+\frac{1}{\eta_i^k}\mathcal{P}_i^k\big\|_F^2.\end{aligned} \tag{3.14}$$

  From (3.14), we have the closed-form solution for $\mathcal{X}^{k+1}$ as follows

$$(\mathcal{X}^{k+1})_{ijl}=\begin{cases}h_{ijl}, & \text{if } (i,j,l)\in\Omega,\\ \hat{\mathcal{X}}^k{}_{ijl}/\tau_k, & \text{otherwise,}\end{cases} \tag{3.15}$$

  where $\hat{\mathcal{X}^k}=\sum_{i=1}^{3}\beta_i^k(\mathcal{G}_i^{k+1}-\frac{1}{\beta_i^k}\mathcal{T}_i^k)+\sum_{i=1}^{3}\eta_i^k(\mathcal{Z}_i^k-\frac{1}{\eta_i^k}\mathcal{P}_i^k)$ and $\tau_k=\sum_{i=1}^{3}(\beta_i^k+\eta_i^k)$.

- Update the variables $\mathcal{T}=(\mathcal{T}_1,\mathcal{T}_2,\mathcal{T}_3)$ and $\mathcal{P}=(\mathcal{P}_1,\mathcal{P}_2,\mathcal{P}_3)$ via

$$\mathcal{T}_i^{k+1}=\mathcal{T}_i^k+\beta_i^k\big(\mathcal{X}^{k+1}-\mathcal{G}_i^{k+1}\big)\ \text{ and }\ \mathcal{P}_i^{k+1}=\mathcal{P}_i^k+\eta_i^k\big(\mathcal{X}^{k+1}-\mathcal{Z}_i^{k+1}\big), \tag{3.16}$$

  for $i\in[3]$, respectively.

Formally, we can summarize the iterative schemes for (3.10) in Algorithm 2.

Algorithm 2 ADMM for Tensor Completion Model (3.10)

Input: Observed tensor $\mathcal{H}$, parameters: $\alpha=(\alpha_1,\alpha_2,\alpha_3)$, $\rho=(\rho_1,\rho_2,\rho_3)$, $\beta^0=(\beta_1^0,\beta_2^0,\beta_3^0)$, $\eta^0=(\eta_1^0,\eta_2^0,\eta_3^0)$, $p\le 1$, κ_1, $\kappa_2>1$.

Step 0: Initialization: choose initial guess $(\mathcal{G}^0,\mathcal{Z}^0,\mathcal{X}^0,\mathcal{T}^0,\mathcal{P}^0))$. Set $k:=0$.
Step 1: Update $\mathcal{G}_i^{k+1}$ by (3.12),
Step 2: Update $\mathcal{Z}_i^{k+1}$ by (3.13),
Step 3: Update $\mathcal{X}^{k+1}$ by (3.15),
Step 4: Update $\mathcal{T}_i^{k+1}$ and $\mathcal{P}_i^{k+1}$ by (3.16),
Step 5: Update $\beta_i^{k+1}=\kappa_1\beta_i^k$ and $\eta_i^{k+1}=\kappa_2\eta_i^k$,
Step 6: Unless a termination criterion is fulfilled, set $k:=k+1$ and go to Step 1.

Output: Recovered tensor $\mathcal{X}^*$.

4 Experiments

In this section, we implement some experiments to verify the effectiveness of our proposed approaches. We test four kinds of third-order tensor datasets: (1) color images, (2) MSI data, (3) video data, and (4) MRI data. The experiments are implemented by MATLAB R2018b (64bit) on a PC with Intel(R) Core(TM) i7-7500 CPU@2.70GHz, 2.9GHz and 8GB memory. Throughout this section, for simplicity, we denote Algorithms 1 and 2 by 'aTNNst-W' and 'aTNNst-p', respectively. Moreover, to highlight the efficiency of the two proposed approaches, we select five state-of-the-art methods as follows:

- HaLRTC [21]: High accuracy low rank tensor completion, which used a traditional tensor nuclear norm defined on Tucker decomposition and ADMM framework.

- IpST [27]: Using the p-shrinkage thresholding operetor to replace the traditonal general tensor nuclear norm in HaLRTC and solved by ADMM framework.
- WSTNN [38]: Tensor N-tubal rank and its convex relaxation for low-rank tensor recovery.
- F-TNN [15]: Framelet representation of tensor nuclear norm for third-order tensor completion.
- TNN-3DTV [14]: Anisotropic total variation regularized low-rank tensor completion based on tensor nuclear norm.

We adopt the relative change of the two successive recovered tensors, i.e.,

$$\text{RelCha} = \frac{\|\mathcal{X}^{k+1} - \mathcal{X}^k\|_F}{\|\mathcal{X}_{\text{true}}\|_F} \leq 10^{-4}, \tag{4.1}$$

as the stopping criterion for all methods, where $\mathcal{X}_{\text{true}}$ is the original tensor. The following metrics are chosen to evaluate the recovery performance of the different algorithms.

- Peak Signal-to-Noise Ratio (PSNR) is defined as:

$$\text{PSNR} = 10 \log \frac{(\mathcal{X}_{\text{true}}^{\max})^2 (\#\Omega^c)}{\|\mathcal{X}^* - \mathcal{X}_{\text{true}}\|_F^2},$$

  where $\#\Omega^c$ denotes the number of elements in the complementary set of Ω, $\mathcal{X}_{\text{true}}^{\max}$ represents the max element of $\mathcal{X}_{\text{true}}$.

- Structural Similarity (SSIM) is defined as:

$$\text{SSIM} = \frac{(2\mu_X \mu_{X^*} + a_1)(2\sigma_{XX^*} + a_2)}{(\mu_X^2 + \mu_{X^*}^2 + a_1)(\sigma_X^2 + \sigma_{X^*}^2 + a_2)},$$

  where X and X^* denote the greyscale images for the original image and its recovered image, a_1 and a_2 are constants, μ_X and μ_{X^*} denote the average values, while σ_X and σ_{X^*} denote the standard deviation of X and X^*, respectively, and σ_{XX^*} denote the covariance matrix between X and X^*.

- Feature Similarity Index Measure (FSIM), which is proposed for full reference image quality assessment, and based on the fact that human visual system (HVS) understands an image mainly according to its low-level features. We refer the reader to [36] for more details.

We set $(\alpha_1, \alpha_2, \alpha_3) = (1/3, 1/3, 1/3)$ for MRI recovery, and $(\alpha_1, \alpha_2, \alpha_3) = (1/2, 1/2, 5 \cdot 10^{-4})$ for the other datasets. Moreover, we take $p = 1/2$, $(\rho_1, \rho_2, \rho_3) = (5 \cdot 10^{-3}, 5 \cdot 10^{-3}, 0)$ for aTNNst-p, and set $\delta = 10^3$, $\epsilon = 10^{-16}$ and $(\rho_1, \rho_2, \rho_3) = (1/2, 1/2, 0)$ for aTNNst-W. In addition, for both aTNNst-p and aTNNst-W, we set $\beta^0 = (\alpha_1, \alpha_2, \alpha_3)/1000$, $\eta^0 = (10^{-4}, 10^{-4}, 10^{-4})$, and by using a dynamical strategy to update β and η with $\kappa_1 = \kappa_2 = 1.1$, All parameters of other compared algorithms are taken the same as in original literature [14, 15, 21, 27, 38].

4.1 Color images recovery

In this subsection, we apply the proposed approaches to image restoration. Here, we choose six frequently used color images ('**baboon**', '**sailboat**', '**lena**', '**barbara**', '**giant**', '**butterfly**'), which are listed in Figure 1, and can be represented by the third-order tensor with size $256 \times 256 \times 3$.

We consider two scenarios to degrade the original images. The first one is some image data being dropped in a uniformly distributed way. We investigate four cases, i.e., the '**lena**', '**sailboat**', '**barbara**' and '**butterfly**' having 3%, 5%, 10%, 20% observed information, respectively, which are shown in the first column of Figure 2 from the top to the bottom. The recovery results by using different approaches: HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p and aTNNst-W are shown from second column to the right, respectively. It can be clearly seen from Figure 2 that our approaches can perform image restoration better than the other five compared methods. Particularly, when the sample ratio sr is 3%, all the five compared approaches fail, but our approaches can still recover the approximate image contour. To further highlight the efficiency of the proposed approaches, we investigate another six sample ratios, i.e., 3%, 5%, 10%, 20%, 30%, 40%, for the six images in Figure 1. Here we only report the PSNR value comparison in Figure 3, where the results clearly show that our approaches can recover better images in terms of achieving higher PSNR values than the other solvers.

Figure 1: Six color clean images for test.

We also consider the structurally missing cases, that is, observed images have entire rows or blocks missing/or corrupted by some irregular noise as shown in the first column in Figure 4. The first row, '**lena-1**' image, loses its pixels with respect to these English letters; the second row, '**lena-2**' image, which is dropped 90% pixels in a random way and then corrupted by English letters; the '**giant**' image, i.e., the third row, which has 90% percent random pixel loss and is dropped one slice for every two slices with respect to mode-1; the last row, '**baboon**' image, is dropped 90% pixels in a random way and then corrupted by ten block slices. The recovered image shown in Figure 4 further demonstrates that our approaches performs much better than the other five approaches. The PSNR, SSIM and FSIM values of all approaches in the case of four structural missing samples are reported in Figure 5. Obviously, as shown by Figure 5, in all cases, the proposed approaches can achieve higher PSNR, SSIM and FSIM values than the others.

4.2 MSI recovery

In this subsection, we apply the approaches proposed in this paper on MSI data from the CAVE database. They can be regarded as third-tensor with $256 \times 256 \times 31$, where the spatial resolution is 256×256 and the spectral resolution is 31. Here we just take sr as 5%, 10% and 20% for comparison. Table 1 lists PSNR, SSIM and FSIM values of two our

http://www.cs.columbia.edu/CAVE/databases/multispectral.

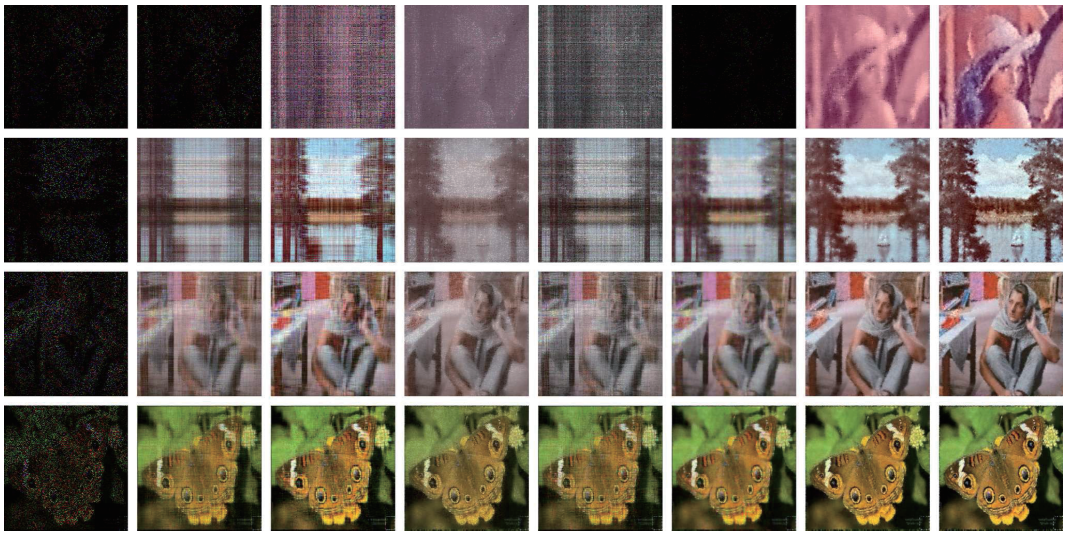

Figure 2: Results on color images ('**lena**' with sr= 3%, '**sailboat**' with sr= 5%, '**barbara**' with sr= 10% and '**butterfly**' with sr= 20%) recovered by different approaches. From the second column to right: HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p and aTNNst-W, respectively.

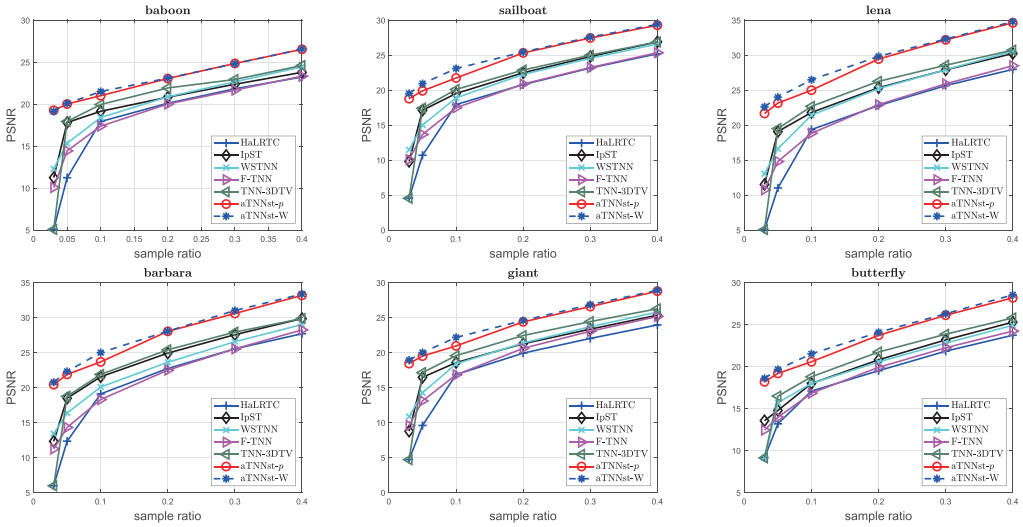

Figure 3: PSNR results of all compared approaches in the case of random missing.

Figure 4: Visualizations of the corrupted color image and performances recovered by different approaches. From the second column to right: HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p and aTNNst-W, respectively.

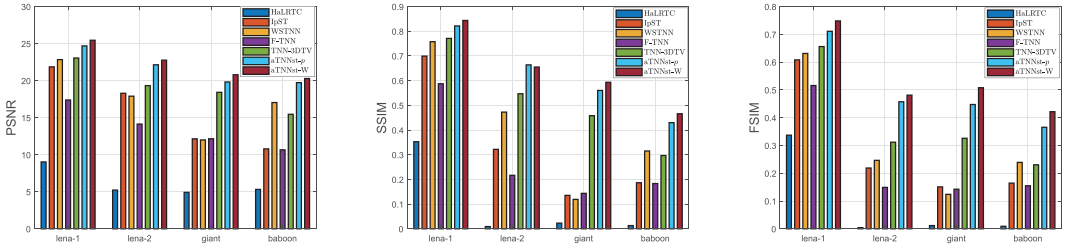

Figure 5: Numerical results of PSNR, SSIM and FSIM associated to Figure 4.

approaches and all the LRTC approaches after making four different MSI data recovery. As shown in Table 1, our approaches are significantly better than other filling approaches in terms of the numerical results of the three indexes. To further highlight the visual benefits of aTNNst-p and aTNNst-W, in Figure 6, we show three band (10th, 11th, 12th) in three tested data recovered by different methods with $\mathsf{sr} = 5\%$. In Figure 6, the observed data, recovery results by different approaches (HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p and aTNNst-W) and the clean data are shown from left to right, respectively. From Figure 6, we see that the proposed approaches are also superior to the compared ones with more clearer details and structural information, especially for the '**beads**'.

Table 1: Numerical results for MSI data recovery.

MSI Data	Method	$\mathsf{sr} = 0.05$			$\mathsf{sr} = 0.10$			$\mathsf{sr} = 0.20$		
		PSNR	SSIM	FSIM	PSNR	SSIM	FSIM	PSNR	SSIM	FSIM
	HaLRTC	14.41	0.102	0.476	14.65	0.137	0.555	15.16	0.214	0.659
	IpST	18.65	0.362	0.649	22.55	0.622	0.790	27.21	0.838	0.898
beads	WTSNN	24.19	0.763	0.859	29.28	0.920	0.943	35.57	0.979	0.983
$256 \times 256 \times 31$	F-TNN	20.92	0.590	0.504	25.22	0.651	0.711	31.01	0.812	0.849
	TNN-3DTV	20.44	0.501	0.674	22.53	0.654	0.784	25.48	0.810	0.877
	aTNNst-p	27.93	**0.866**	0.912	33.23	**0.958**	0.966	39.54	**0.990**	0.991
	aTNNst-W	**28.40**	0.864	**0.912**	**33.72**	0.957	**0.967**	**40.44**	**0.990**	**0.992**
	HaLRTC	11.66	0.052	0.517	11.90	0.080	0.539	12.40	0.131	0.565
	IpST	27.65	0.773	0.882	33.53	0.910	0.945	37.97	0.961	0.975
pompoms	WTSNN	31.65	0.924	0.955	36.93	0.986	0.979	44.02	0.992	0.994
$256 \times 256 \times 31$	F-TNN	26.47	0.532	0.611	33.11	0.751	0.804	39.27	0.886	0.915
	TNN-3DTV	26.21	0.806	0.878	30.12	0.882	0.922	34.93	0.943	0.962
	aTNNst-p	35.84	0.951	0.968	42.13	0.987	0.990	48.21	0.996	0.997
	aTNNst-W	**36.78**	**0.955**	**0.975**	**42.21**	**0.986**	**0.991**	**49.22**	**0.997**	**0.998**
	HaLRTC	13.35	0.182	0.616	13.58	0.222	0.629	14.10	0.295	0.648
	IpST	27.41	0.818	0.869	31.12	0.902	0.921	35.38	0.956	0.963
feather	WTSNN	31.32	0.908	0.933	36.16	0.962	0.970	42.87	0.989	0.983
$256 \times 256 \times 31$	F-TNN	30.43	0.493	0.551	35.12	0.743	0.807	40.28	0.789	0.829
	TNN-3DTV	25.47	0.808	0.837	28.01	0.879	0.891	31.85	0.943	0.948
	aTNNst-p	34.24	0.930	0.948	40.20	**0.980**	**0.983**	47.00	**0.994**	**0.995**
	aTNNst-W	**34.90**	**0.931**	**0.949**	**40.50**	0.978	**0.983**	**47.66**	0.990	0.992

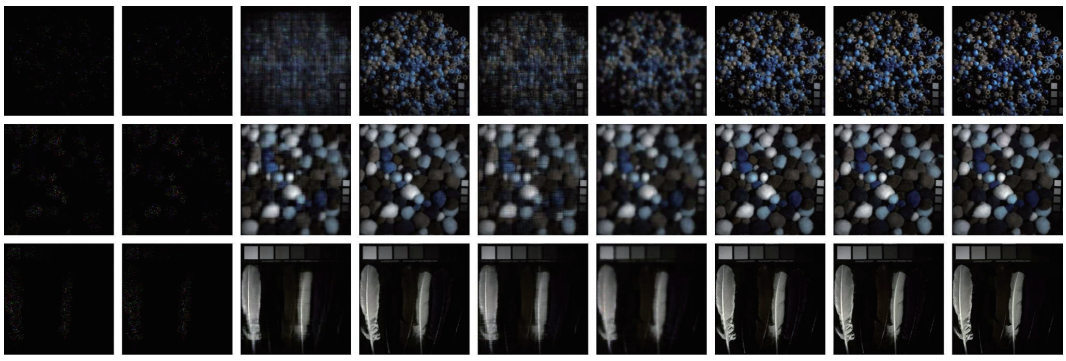

Figure 6: Visualizations of recovery results of three selected MSI datas with 5% observed elements. From left to right: Corrupted data, HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p, aTNNst-W, and Clean data, respectively.

4.3 Video recovery

Furthermore, we test four grayscale videos (named '**akiyo**', '**suzie**', '**hall**', '**seafish**', respectively) recovery, the first three of which have a size of $144 \times 176 \times 100$ and the last of which has a size of $256 \times 256 \times 30$. Table 2 lists the results of PSNR, SSIM, FSIM values for all LRTC methods with $\mathsf{sr} \in \{5\%, 10\%, 20\%\}$ in four tested datas. It can be easily seen that the proposed approaches have obvious numerical superiority in all evaluation indexes. In Figure 7, the recovery results of two frames are listed for the first three data at different sample ratios (5%, 10%, 20%, respectively).

Table 2: Numerical results for video data recovery.

Video Data	Method	$\mathsf{sr} = 0.05$			$\mathsf{sr} = 0.10$			$\mathsf{sr} = 0.20$		
		PSNR	SSIM	FSIM	PSNR	SSIM	FSIM	PSNR	SSIM	FSIM
	HaLRTC	6.61	0.012	0.458	6.84	0.019	0.445	7.35	0.032	0.424
	IpST	24.85	0.767	0.861	27.99	0.870	0.920	32.50	0.949	0.966
akiyo	WTSNN	30.55	0.938	0.906	34.00	0.969	0.958	38.23	0.987	0.991
$144 \times 176 \times 100$	F-TNN	30.44	0.920	0.814	33.60	0.879	0.903	37.23	0.983	0.955
	TNN-3DTV	24.24	0.764	0.831	26.10	0.829	0.884	29.05	0.902	0.937
	aTNNst-p	32.98	0.958	0.911	37.17	**0.984**	0.965	41.70	**0.993**	**0.995**
	aTNNst-W	**33.28**	**0.963**	**0.929**	**37.29**	0.983	**0.966**	**42.09**	**0.993**	**0.995**
	HaLRTC	7.16	0.009	0.454	7.39	0.013	0.425	7.90	0.019	0.389
	IpST	22.79	0.657	0.787	27.57	0.791	0.866	31.74	0.891	0.928
suzie	WTSNN	27.37	0.792	0.859	29.61	0.848	0.900	32.45	0.904	0.938
$144 \times 176 \times 100$	F-TNN	26.26	0.748	0.519	28.56	0.818	0.639	31.37	0.887	0.770
	TNN-3DTV	26.68	0.779	0.848	28.76	0.834	0.886	31.40	0.892	0.926
	aTNNst-p	29.49	**0.835**	**0.898**	**31.95**	**0.882**	**0.928**	**34.30**	**0.921**	**0.952**
	aTNNst-W	**29.52**	0.805	0.882	31.90	0.879	0.924	34.23	0.913	0.950
	HaLRTC	4.84	0.007	0.390	5.07	0.013	0.394	5.58	0.024	0.402
	IpST	5.18	0.019	0.430	25.56	0.852	0.882	30.48	0.936	0.945
hall	WTSNN	27.37	0.792	0.859	31.97	0.956	0.967	34.99	0.973	0.979
$144 \times 176 \times 100$	F-TNN	29.68	0.931	0.823	32.63	0.959	0.884	35.42	0.975	0.925
	TNN-3DTV	21.39	0.687	0.760	22.81	0.767	0.816	25.58	0.867	0.890
	aTNNst-p	**30.98**	**0.942**	0.891	**34.12**	**0.965**	0.970	**37.11**	**0.978**	**0.983**
	aTNNst-W	30.73	**0.942**	0.861	34.10	0.962	**0.976**	37.10	0.976	0.982
	HaLRTC	7.63	0.032	0.405	7.87	0.049	0.481	8.38	0.087	0.577
	IpST	17.00	0.314	0.658	19.63	0.543	0.763	23.25	0.774	0.868
seafish	WTSNN	22.57	0.772	0.866	26.11	0.888	0.929	29.69	0.944	0.961
$185 \times 290 \times 30$	F-TNN	22.03	0.748	0.772	25.69	0.849	0.881	29.27	0.915	0.936
	TNN-3DTV	18.17	0.373	0.610	19.41	0.507	0.702	21.51	0.686	0.814
	aTNNst-p	**24.77**	**0.861**	**0.870**	**27.84**	**0.924**	**0.949**	**31.71**	**0.963**	**0.973**
	aTNNst-W	23.18	0.806	0.818	27.51	0.918	0.947	31.34	0.960	0.971

4.4 MRI recovery

Finally, we apply the proposed approaches on MRI image recovery, whose size is $181 \times 217 \times 181$. Numerical results of different approaches for recovering the MRI image with 5%, 10% and 20% sampling ratios are shown in Table 3, which shows that the proposed approaches all have obvious numerical advantages. From Figure 8, in which three different slices are shown, and the observed image, the recovery images by different approaches and the clean image are shown from left to right, respectively, it can be clearly observed that no matter which slice is taken, the proposed approaches retain more abundant structure and texture information visually than the compared approaches.

http://trace.eas.asu.edu/yuv/.
http://brainweb.bic.mni.mcgill.ca/brainweb/selection_normal.html.

Figure 7: The recovery results of three selected videos '**akiyo**' (first two rows), '**suize**' (mid two rows), and '**hall**' (last two rows) by all approaches. From left to right: Corrupted video frames, HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p, aTNNst-W, and Clear video frames, respectively.

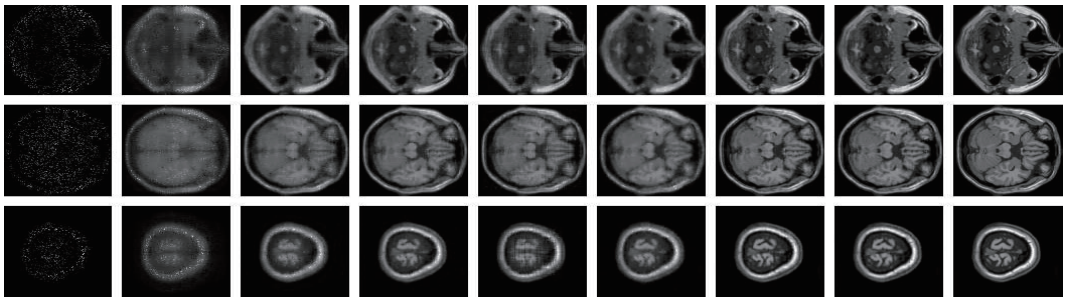

Figure 8: Visualizations of recovery results of MRI data with 10% observed information. From left to right: Corrupted data, HaLRTC, IpST, WTSNN, F-TNN, TNN-3DTV, aTNNst-p, aTNNst-W, and Clean data, respectively.

Table 3: Numerical results for MRI data recovery.

Data	Method	sr = 0.05			sr = 0.10			sr = 0.20		
		PSNR	SSIM	FSIM	PSNR	SSIM	FSIM	PSNR	SSIM	FSIM
MRI $181 \times 217 \times 181$	HaLRTC	15.40	0.241	0.608	19.03	0.390	0.699	24.30	0.653	0.826
	IpST	20.47	0.462	0.729	25.16	0.711	0.824	32.05	0.921	0.931
	WTSNN	25.60	0.714	0.827	29.02	0.835	0.887	33.46	0.931	0.941
	F-TNN	23.32	0.664	0.434	26.56	0.793	0.550	30.73	0.901	0.650
	TNN-3DTV	24.36	0.676	0.794	26.60	0.783	0.845	30.17	0.893	0.908
	aTNNst-p	29.04	0.827	0.879	33.07	**0.916**	0.929	36.89	**0.960**	**0.961**
	aTNNst-W	**30.44**	**0.846**	**0.892**	**33.58**	0.915	**0.930**	**37.18**	0.957	0.960

5 Conclusion

In this paper, we considered the tensor completion problem. By exploiting the multi-dimensional low-rankness of the involved tensors, we proposed two T-product based optimization models, which can capture the hidden structure information in the considered tensors. Due to the coupled property of these models, we proposed two easily implementable alternating updating algorithms, whose subproblems could be solved efficiently or have closed-form solutions. To illustrate the effectiveness of two proposed approaches, we applied them to color images, MSI data, videos, and MRI data recovery. A series of numerical results demonstrated that our approaches effectively exploit the correlations along all modes, while preserving the intrinsic structure of the underlying tensor. Since our approaches have to deal with some permuted tensors and the embedded matrix SVD procedure, our algorithms require more SVDs so that our methods often require more computing time than these algorithms compared in our experiments. Therefore, how to design faster and more effective approaches will be our future concerns.

Acknowledgment

The authors would like to thank the two referees' close reading and valuable comments, which helped us improve the presentation of this paper. C. Ling and H. He were supported in part by National Natural Science Foundation of China (Nos. 11971138 and 11771113) and Zhejiang Provincial Natural Science Foundation (Nos. LY19A010019, LY20A010018, and LD19A010002).

References

[1] J. A. Bengua, H.N. Phien, H. D. Tuan and M. N. Do, Efficient tensor completion for color image and video recovery: Low-rank tensor train, *IEEE Trans. Image Process.* 26 (2017) 2466–2479.

[2] R. Chartrand, Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data, in: *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 2009, pp. 262–265.

[3] R. Chartrand, Shrinkage mappings and their induced penalty functions, in: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 1026–1029.

[4] D. Coupier, A. Desolneux and B. Ycart, Image denoising by statistical area thresholding, *J. Math. Imaging Vis.* 22 (2005) 183–197.

[5] A. Criminisi, P. Perez and K. Toyama, Region filling and object removal by exemplar-based image inpainting, *IEEE Trans. Image Process.* 13 (2004) 1200–1212.

[6] A.A. Efros and T.K. Leung, Texture synthesis by non-parametric sampling, in: *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, 1999, pp. 1033–1038.

[7] M. Fazel, *Matrix Rank Minimization with Applications*, Ph.D. thesis, Stanford University, USA, 2002.

[8] M. Filipović and A. Jukić, Tucker factorization with missing data with application to low-n-rank tensor completion, *Multidim. Syst. Sign. Process.* 26 (2015) 677–692.

[9] Z.F. Han, C.-S. Leung, L.-T. Huang and H.C. So, Sparse and truncated nuclear norm based tensor completion, *Neural Process. Lett.* 45 (2017) 729–743.

[10] N. Hao, M. E. Kilmer, K. Braman and R. C. Hoover, Facial recognition using tensor-tensor decompositions, *SIAM J. Imaging Sci.* 6 (2013) 437–463.

[11] K. Hosono, S. Ono and T. Miyata, Weighted tensor nuclear norm minimization for color image denoising, in: *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3081–3085.

[12] W. Hu, D. Tao, W. Zhang, Y. Xie and Y. Yang, The twist tensor nuclear norm for video completion, *IEEE Trans. Neural. Netw Learn. Syst.* 28 (2017) 2961–2973.

[13] L. T. Huang, H. C. So, Y. Chen and W. Q. Wang, Truncated nuclear norm minimization for tensor completion, in: *IEEE Sensor Array and Multichannel Signal Processing Workshop*, 2014.

[14] F. Jiang, X.Y. Liu, H. Lu and R. Shen, Anisotropic total variation regularized low-rank tensor completion based on tensor nuclear norm for color image inpainting, in: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 1363–1367.

[15] T.X. Jiang, M.K. Ng, X.L. Zhao and T.Z. Huang, Framelet representation of tensor nuclear norm for third-order tensor completion, *IEEE Trans. Image Process.* 29 (2020) 7233–7244

[16] M.E. Kilmer, K. Braman, N. Hao and R.C. Hoover, Third-order tensors as operators on matrices: a theoretical and computational framework with applications in imaging, *SIAM J. Matrix Anal. Appl.* 34 (2013) 148–172.

[17] T. Kolda and B. Bader, Tensor decompositions and applications, *SIAM Rev.* 51 (2009) 455–500.

[18] T.G. Kolda and J.R. Mayo, Shifted power method for computing tensor eigenvalues, *SIAM J. Matrix Anal. Appl.* 32 (2011) 1095–1124.

[19] N. Komodakis, Image completion using global optimization, in: *Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 442–452.

[20] C. Liu, H. Shan and C. Chen, Tensor p-shrinkage nuclear norm for low-rank tensor completion, *Neurocomputing* 387 (2020) 255–267.

[21] J. Liu, P. Musialski, P. Wonka and J. Ye, Tensor completion for estimating missing values in visual data, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (2013) 208–220.

[22] Y. Liu, F. Shang, W. Fan, J. Cheng and H. Cheng, Generalized higher order orthogonal iteration for tensor learning and decomposition, *IEEE Trans. Neural. Netw Learn. Syst.* 27 (2016) 2551–2563.

[23] Y. Liu, F. Shang, L. Jiao, J. Cheng and H. Cheng, Trace norm regularized CANDECOMP/PARAFAC decomposition with missing data, *IEEE Trans. Cybern.* 45 (2015) 2437–2448.

[24] C. Lu, J. Tang, S.C Yan and Z.C. Lin, Nonconvex nonsmooth low rank minimization via ititerative reweighted nuclear norm, *IEEE Trans. Image Process.* 25 (2016) 829–839.

[25] Y. Mu, P. Wang, L. Lu, X. Zhang and L. Qi, Weighted tensor nuclear norm minimization for tensor completion using tensor-SVD, *Pattern Recognition Letters* 130 (2020) 4–11.

[26] O. Semerci, N. Hao, M.E. Kilmer and E.L. Miller, Tensor-based formulation and nuclear norm regularization for multienergy computed tomography, *IEEE Trans. Image Process.* 23 (2014) 1678–1693.

[27] K. Shang, Y.F. Li and Z. H. Huang, Iterative p-shrinkage thresholding algorithm for low Tucker rank tensor recovery, *Inf. Sci.* 482 (2019) 374–391.

[28] N.D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E.E. Papalexakis and C. Faloutsos, Tensor decomposition for signal processing and machine learning, *IEEE Trans. Signal Process.* 65 (2017) 3551–3582.

[29] H. Tan, B. Cheng, W. Wang, Y.J. Zhang and B. Ran, Tensor completion via a multilinear low-n-rank factorization model, *Neurocomputing* 133 (2014) 161–169.

[30] V. Nivitha Varghees, M.S. Manikandan and R. Gini, Adaptive MRI image denoising using total-variation and local noise estimation, in: *Proceedings of the 2012 International Conference on Advances in Engineering, Science and Management (ICAESM)*, 2012, pp. 506–511.

[31] Y. Wu, H. Tan, Y. Li, J. Zhang and X. Chen, A fused CP factorization method for incomplete tensors, *IEEE Trans. Neural. Netw Learn. Syst.* 30 (2019) 751–764.

[32] J. Xue, Y. Zhao, W. Liao and C.W. Chan, Nonconvex tensor rank minimization and its applications to tensor recovery, *Inf. Sci.* 503 (2019) 109–128.

[33] J. Xue, Y. Zhao, W. Liao, J.C.W. Chan and S.G. Kong, Enhanced sparsity prior model for low-rank tensor completion, *IEEE Trans. Neural. Netw Learn. Syst.* 31 (2020) 4567–4581.

[34] T. Yokota, Q. Zhao and A. Cichocki, Smooth PARAFAC decomposition for tensor completion, *IEEE Trans. Signal Process.* 64 (2016) 5423–5436.

[35] M. Yuan and C. H. Zhang, On tensor completion via nuclear norm minimization, *Found. Comput. Math.* 16 (2016) 1031–1068.

[36] L. Zhang, L. Zhang, X. Mou and D. Zhang, FSIM: a feature similarity index for image quality assessment, *IEEE Trans. Image Process.* 20 (2011) 2378–2386.

[37] Z. Zhang and S. Aeron, Exact tensor completion using t-SVD, *IEEE Trans. Signal Process.* 65 (2017)1511–1526.

[38] Y. Zheng, T. Huang, X. Zhao, T. Jiang and T. Ma., Tensor n-tubal rank and its convex relaxation for low-rank tensor recovery, *Inf. Sci.* 532 (2020) 170–189.

[39] H. Zhou, D. Zhang, K. Xie and Y. Chen, Spatio-temporal tensor completion for imputing missing internet traffic data, in: *2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC)*, 2015, pp. 1–7.

Manuscript received 30 April 2021
revised 13 June 2021
accepted for publication 22 June 2021

Chenjian Pan
Department of Mathematics, School of Science
Hangzhou Dianzi University, Hangzhou, 310018, China
E-mail address: panchenj@hdu.edu.cn

Chen Ling
Department of Mathematics, School of Science
Hangzhou Dianzi University, Hangzhou, 310018, China
E-mail address: macling@hdu.edu.cn

Hongjin He
School of Mathematics and Statistics
Ningbo University, Ningbo 315211, China
E-mail address: hehongjin@nbu.edu.cn