Information Sciences 326 (2016) 243–257

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Tensor completion using total variation and low-rank matrix factorization

Teng-Yu Ji, Ting-Zhu Huang*, Xi-Le Zhao, Tian-Hui Ma, Gang Liu

School of Mathematical Sciences/Resrarch Center for Image and Vision Computing, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, P R China

ARTICLE INFO

Article history:
Received 18 January 2015
Revised 22 May 2015
Accepted 29 July 2015
Available online 5 August 2015

Keywords:
Tensor completion
Total variation
Low-rank matrix factorization
Block coordinate descent

ABSTRACT

In this paper, we study the problem of recovering a tensor with missing data. We propose a new model combining the total variation regularization and low-rank matrix factorization. A block coordinate decent (BCD) algorithm is developed to efficiently solve the proposed optimization model. We theoretically show that under some mild conditions, the algorithm converges to the coordinatewise minimizers. Experimental results are reported to demonstrate the effectiveness of the proposed model and the efficiency of the numerical scheme.

1. Introduction

As a high-dimensional extension of matrix, tensor is an important big data format, which plays a significant role in a wide range of real-world applications [19,31–36]. Among them, one important problem is to estimate the missing data from the observed incomplete data, e.g., image inpainting [1,17], video inpainting [18], hyperspectral data recovery [22,23,41,45], magnetic resonance imaging (MRI) data recovery [38], high-order web link analysis [16], personalized web search [31], and seismic data reconstruction [19]. In this study, we specially focus on the reconstruction of low-rank tensors with randomly missing data.

Matrix completion can be regarded as the 2-mode tensor completion [4]. One powerful tool for matrix completion is to minimize the matrix rank, which can effectively estimate the missing data exploiting both the local and global information [24]. The model for low-rank matrix completion is formulated as:

$$\begin{aligned} \min_{Y} \quad & \operatorname{rank}(Y) \\ \text{s.t.} \quad & \mathcal{P}_{\Omega}(Y) = F, \end{aligned} \tag{1}$$

where $Y \in \mathbb{R}^{m\times n}$ is the underlying matrix, $F \in \mathbb{R}^{m\times n}$ is the observed matrix, and $\mathcal{P}_{\Omega}(\cdot)$ is the projection operator: see details in Section 2. However, the main difficulty of solving (1) arises from the non-convexity of the rank of matrices, which may prevent one from getting a global solution [40]. To solve the challenging problem of rank minimizing, Fazel et al. [8] and Kurucz et al. [20] proposed to use rank constraint to iteratively estimate the missing values. Another popular and effective approach is to use the trace norm, which is theoretical soundness and can be considered as the approximation for the rank of matrices [3,26,28]. And

* Corresponding author. Tel.: +86 28 61831608; fax: +86 28 61831280.
E-mail addresses: tengyu_j66@126.com (T.-Y. Ji), tingzhuhuang@126.com, tzhuang@uestc.edu.cn (T.-Z. Huang), xlzhao122003@163.com (X.-L. Zhao), nkmth0307@126.com (T.-H. Ma), wd5577@163.com (G. Liu).

http://dx.doi.org/10.1016/j.ins.2015.07.049
0020-0255/

under certain conditions [4,5], the problem (1) is converted to the following convex optimization problem:

$$\begin{aligned} \min_{Y} \quad & \|Y\|_* \\ \text{s.t.} \quad & \mathcal{P}_\Omega(Y) = F. \end{aligned} \tag{2}$$

Then the model (2) can be efficiently solved by using some optimization algorithms, such as FPCA [26], APGL [37], LMaFit [40], and the alternating direction method (ADM) [30,43].

For tensor completion, the low-rank based methods have also been widely studied [9,24,25,42,44]. The low-rank tensor completion model can be formulated as:

$$\begin{aligned} \min_{\mathcal{Y}} \quad & \text{rank}(\mathcal{Y}) \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}, \end{aligned} \tag{3}$$

where $\mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ is the underlyingtensor, and $\mathcal{F}$ is the observed data. However, there is no unique definition for the rank of tensors, such as CP-rank and n-rank [15], and both of the corresponding minimization problems are NP-hard [12]. As the tensor is a generalization of the matrix, one can generalize matrix completion problem (2) to the tensor case:

$$\begin{aligned} \min_{\mathcal{Y}} \quad & \|\mathcal{Y}\|_* \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}. \end{aligned} \tag{4}$$

A naive method is to unfold the tensor into a matrix, and thus to solve the matrix completion (2). However, the method only utilizes low-rankness to one mode of the tensor, and it cannot recover the tensor well [24,42]. Thus, it is necessary to develop methods considering low-rankness to the all mode of the tensor. Recently, Liu et al. [24] developed a theoretical framework for low-rank tensor completion and established a definition of the trace norm for tensors as a surrogate for the tensor rank:

$$\|\mathcal{Y}\|_* := \sum_{n=1}^{N} \alpha_n \|Y_{(n)}\|_*, \tag{5}$$

where $Y_{(n)}$ is the mode-n unfolding of $\mathcal{Y}$: see details in Section 2. Then low-rank tensor completion problem (4) is rewritten as:

$$\begin{aligned} \min_{\mathcal{Y}} \quad & \textstyle\sum_{n=1}^{N} \alpha_i \|Y_{(n)}\|_* \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}. \end{aligned} \tag{6}$$

Problem (6) can be solved by some optimization methods, such as FaLRTC [24] and the Douglas–Rachford splitting method [9]. Because the information of the all mode is considered, these methods [9,24] outperform the naive method. However they have to calculate singular value decomposition (SVD) for N matrices, which is expensive in term of time and memory. Considering this difficulty, Xu et al. [42] applied low-rank matrix factorization to the all-mode matricizations of the tensor as an alternative of the tensor trace norm,

$$\begin{aligned} \min_{\mathcal{Y},X,A} \quad & \sum_{n=1}^{N} \frac{\alpha_n}{2} \|Y_{(n)} - A_n X_n\|_F^2 \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}, \end{aligned} \tag{7}$$

where $A = (A_1, \ldots, A_N)$, $X = (X_1, \ldots, X_N)$, and $\alpha_n, n = 1, \ldots, N$ are positive weights satisfying $\sum_{n=1}^{N} \alpha_n = 1$. Their method (called TMac) has shown to obtain better results and take less time than FaLRTC [24].

Note that Xu et al. [42] only consider the low-rank prior. However, many real-world data exhibit the piecewise smooth prior. In particular, as one of characterizing piecewise smooth functions, the total variation (TV) norm [29] has been shown to preserve edges well in image restoration [11,21,46]. Recently, other TV based regularization methods have received great success in image processing problems, such as the image segmentation [7,39], the reconstruction for video [6], hyperspectral image [22,45] and MRI [38]. Particularly, the authors in [22,45] considered to apply TV regularization to material identification and unmixing for hyperspectral images, with the aim of exploiting the spatial contextual information presented in the hyperspectral images. Inspired by the former works, we consider to introduce the TV regularization into the tensor completion problem (7).

The contributions of this paper are mainly two folds. First, we propose a new model for low-rank tensor completion with randomly missing data. More precisely, our model is:

$$\begin{aligned} \min_{\mathcal{Y},X,A} \quad & \sum_{n=1}^{N} \frac{\alpha_n}{2} \|Y_{(n)} - A_n X_n\|_F^2 + \mu \text{TV}(X_3) \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}, \end{aligned} \tag{8}$$

where μ is the regularization parameter, $A = (A_1, \ldots, A_N)$, $X = (X_1, \ldots, X_N)$, and $\text{TV}(X_3)$ is the total variation of X_3. Similar to [2,45], A_n represents a library (each column contains a signature of the nth mode direction), and X_n is called an encoding. For example, in the unmixing problem for hyperspectral image [2,45], each column of A_3 contains a spectral signature, and each row of X_3 contains the fractional abundances of a given endmember. This interpretation is also valid for the mode-3 unfolding of video

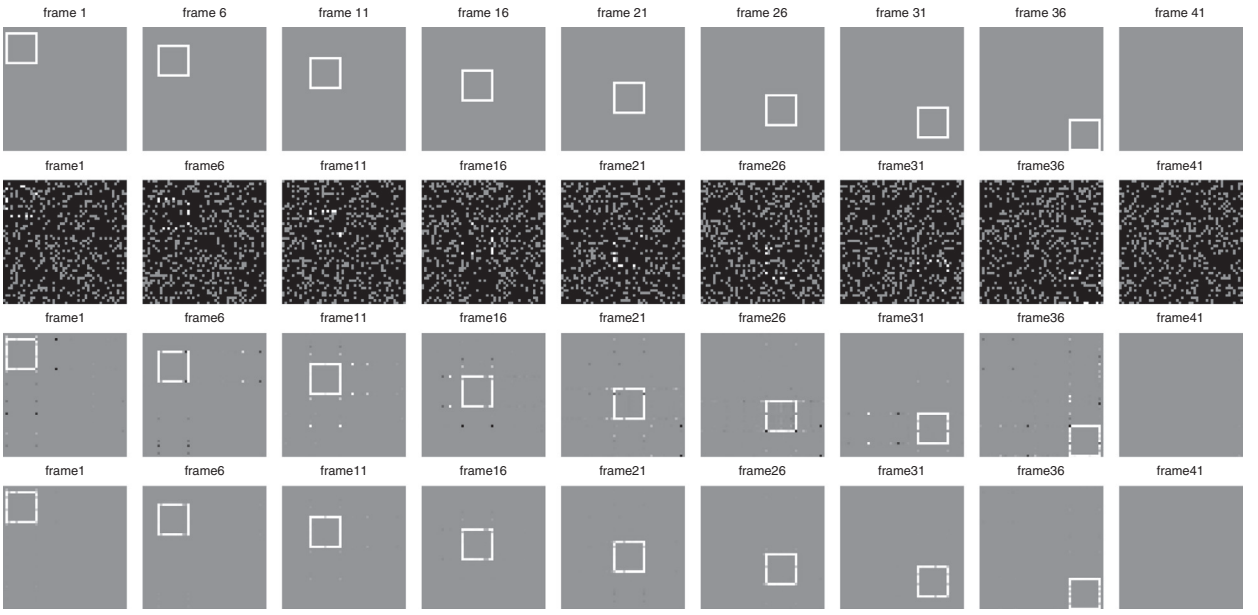

Fig. 1. The comparison results of TMac and the proposed method on synthetic data. Top row: original data. The data has 50 frames. In the first 40 frames, there is a white square moving towards lower right corner every five frames, and in the last 10 frames, there are the backgrounds (its intensity is 0.5). Second row: the marked data, which contains 80% missing entries, shown as black pixels. Bottom two rows: the recovered results by TMac and our method.

and MRI. For the other details of the model, we ask for the readers patience until Section 2. Our motivation is also illustrated by Fig. 1, where we show the results recovered by TMac and our method for synthetic data. It is clear from Fig. 1 that the recovered results by our method are visually better than that by TMac. Second, we propose an effective and convergent algorithm to efficiently solve the proposed model. The proposed algorithm is based on block coordinate descent (BCD) method [27], and used a proximal technique to ensure the strict convexity of each subproblem, so that the stability and robustness is improved. In addition, we show that the sequence generated by our proposed algorithm converges to the coordinatewise minimizers. At last, numerical examples are given to demonstrate the high effectiveness and robustness of the proposed algorithm.

The outline of this paper is as follows. In Section 2, the formulation of the proposed model is given and the BCD-based solver is elaborated. In Section 3, experimental results are reported. Finally, conclusions are given in Section 4.

2. Problem formulation and numerical scheme

2.1. Preliminary

Following [15], we use low-case letters for vectors, e.g., a, upper-case letters for matrices, e.g., A, and calligraphic letters for tensors, e.g., $\mathcal{A}$. An N-mode tensor is defined as $\mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$, and $y_{i_1,\ldots,i_N}$ is its $(i_1,\ldots,i_N)$-th component. The inner product of two tensors $\mathcal{X}$ and $\mathcal{Y}$ is defined as $\langle \mathcal{X}, \mathcal{Y} \rangle := \sum_{i_1,i_2,\ldots,i_N} x_{i_1,\ldots,i_N} y_{i_1,\ldots,i_N}$. The Frobenius norm is then defined as $\|\mathcal{X}\|_F := \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$. The mode-$n$ unfolding of a tensor $\mathcal{Y}$ is denoted as $Y_{(n)} \in \mathbb{R}^{I_n \times \prod_{i \neq n} I_i}$, where the element $(i_1, i_2, \ldots, i_N)$ maps to the matrix element (i_n, j) satisfying

$$j = 1 + \sum_{k=1,k\neq n}^{N} (i_k - 1) J_k \quad \text{with} \quad J_k = \prod_{m=1,m\neq n}^{k-1} I_m. \tag{9}$$

The inverse operator of unfolding is denoted as "fold", i.e., $\mathcal{Y} = \text{fold}_n(Y_{(n)})$. In our work, we adopt the n-rank definition for the tensor $\mathcal{Y}$, which is defined as an array: $n\text{-rank}(\mathcal{Y}) = (\text{rank}(Y_{(1)}), \ldots, \text{rank}(Y_{(N)}))$. The tensor $\mathcal{Y}$ is low-rank, if $Y_{(n)}$ is low-rank for all n. Please refer to [15] for a more extensive overview. Let Ω be an index set, then $\mathcal{P}_\Omega(\mathcal{Y})$ denotes the tensor coping the entries from $\mathcal{Y}$ in the set and letting the remaining entries be zeros, i.e.,

$$(\mathcal{P}_\Omega(\mathcal{Y}))_{i_1,\ldots,i_N} = \begin{cases} y_{i_1,\ldots,i_N}, & (i_1,\ldots,i_N) \in \Omega, \\ 0, & \text{otherwise}. \end{cases}$$

Next, we give a brief introduction of the proximal operator. Given a convex function $f(x)$, the proximal operator of $f(x)$ is defined as follows:

$$\text{prox}_f(y) = \underset{x}{\text{argmin}} f(x) + \frac{\rho}{2}\|x - y\|^2, \tag{10}$$

where ρ is a positive constant. Friendly, the problem $\min_x\{f(x)\}$ is equivalent to $\min_{x,y}\{f(x)+\frac{\rho}{2}\|x-y\|^2\}$. Thus one can obtain the minimization of $\{f(x)\}$ by iteratively solving $\text{prox}_f(x^k)$, where x^k is the last update of x. The proximal operator is very attractive in that the objective function of (10) is strongly convex with respect to x so long as $f(x)$ is convex.

2.2. Proposed model

The objective function of our model (8) is as following:

$$f(X,A,\mathcal{Y})=\sum_{n=1}^{N}\frac{\alpha_n}{2}\|Y_{(n)}-A_nX_n\|_F^2+\mu\text{TV}(X_3).$$

Suppose the rank of $Y_{(n)}$ is r_n $(n=1,2,\ldots,N)$, which is given as a prior, thus, $A_n\in\mathbb{R}^{I_n\times r_n}$ and $X_n\in\mathbb{R}^{r_n\times s_n}$, where $s_n=\prod_{i\neq n}I_i$. We define two sets $\mathbb{T}_1$ and $\mathbb{T}_2$ by Cartesian product: $\mathbb{T}_1:=\mathbb{R}^{r_1\times s_1}\times\cdots\times\mathbb{R}^{r_N\times s_N}$ and $\mathbb{T}_2:=\mathbb{R}^{I_1\times r_1}\times\cdots\times\mathbb{R}^{I_N\times r_N}$, thus $X\in\mathbb{T}_1$ and $A\in\mathbb{T}_2$.

We first explain the reason why we adopt the TV regularization of X_3. The TV regularization measures the difference between a pixel and its neighbors. The smaller the difference is, the better the TV regularization plays. Because the data is piecewise smooth with respect to the 1st- and 2nd-mode direction, the difference between the pixel of X_n and its 1st- and 2nd-mode direction neighbors is small. Thus, we can introduce the TV regularization of X_n at the 1st- and 2nd-mode direction into the tensor completion problem. However, we find that X_1 and X_2 do not contain the complete information of the 1st- and 2nd-mode for $\mathcal{Y}$, because the rank of $Y_{(1)}$ is r_1 $(r_1<I_1)$, that is, the dimension of the corresponding tensor is $r_1\times I_2\times\cdots\times I_N$. Thus, we introduce the TV regularization of X_{n_0}, $n_0\in\{3,4,\ldots,N\}$. Without loss of generality, we adopt the TV regularization of X_3. According to the rules of unfolding a tensor (9), we find that X_3 has a good structure:

$$X_3=\begin{bmatrix}X_3^{(1,1)} & X_3^{(1,2)} & \cdots & X_3^{(1,\hat{s})}\\ X_3^{(2,1)} & X_3^{(2,2)} & \cdots & X_3^{(2,\hat{s})}\\ \vdots & \vdots & \ddots & \vdots\\ X_3^{(r_3,1)} & X_3^{(r_3,2)} & \cdots & X_3^{(r_3,\hat{s})}\end{bmatrix}\in\mathbb{R}^{r_3\times s_3},\tag{11}$$

where $\hat{s}=\prod_{i=4}^{N}I_i$ and $X_3^{(i,k)}\in\mathbb{R}^{1\times I_1I_2}$ is a vector by lexicographical ordering of the entries of the matrix $\mathcal{X}(:,:,k,i_4,\ldots,i_N)\in\mathbb{R}^{I_1\times I_2}$ with $i=1+\sum_{p=4}^{N}(i_p-1)J_p$ and $J_p=\prod_{m=4}^{p-1}I_m$, which is a slice of the tensor $\mathcal{X}=\text{fold}_3(X_3)\in\mathbb{R}^{I_1\times I_2\times r_3\times I_4\times\cdots\times I_N}$.

In dealing with tensor completion, the isotropic TV[1] is defined as follows:

$$\text{TV}_i(X_3):=\sum_{k=1}^{\hat{s}}\sum_{i=1}^{r_3}\sum_{j=1}^{I_1I_2}\sqrt{|\tilde{D}_{j,1}X_3^{(i,k)}|^2+|\tilde{D}_{j,2}X_3^{(i,k)}|^2},\tag{12}$$

where $X_3^{(i,k)}$ refers to the kth block of ith row of X_3, $\tilde{D}_{j,1}X_3^{(i,k)}$ and $\tilde{D}_{j,2}X_3^{(i,k)}$ are the gradient values of $X_3^{(i,k)}$ at the 1st- and 2nd-mode directions of the jth pixel in $X_3^{(i,k)}$, and $\tilde{D}_{j,1}$ and $\tilde{D}_{j,2}$ are the corresponding discrete gradient operators at the 1st- and 2nd-mode directions, respectively. Particularly, for a 3-mode tensor ($\hat{s}=1$), X_3 can be rewritten as:

$$X_3=\left[(X_3^{(1)})^T,(X_3^{(2)})^T,\ldots,(X_3^{(r_3)})^T\right]^T,\tag{13}$$

thus the $\text{TV}(X_3)$ degenerates to the same measure as [45]:

$$\text{TV}_i(X_3):=\sum_{i=1}^{r_3}\sum_{j=1}^{I_1I_2}\sqrt{|\tilde{D}_{j,1}X_3^{(i)}|^2+|\tilde{D}_{j,2}X_3^{(i)}|^2}.\tag{14}$$

2.3. Proposed algorithm

We present the numerical scheme for solving the problem (8). It is easy to see that the objective function of (8) is not jointly convex for $(X,A,\mathcal{Y})$, but is convex with respect to X, A, $\mathcal{Y}$ independently. Let $\mathbb{T}_3:=\mathbb{T}_1\times\mathbb{T}_2\times\mathbb{R}^{I_1\times\cdots\times I_N}$, thus $\mathcal{Z}=(X,A,\mathcal{Y})\in\mathbb{T}_3$. In order to solve the non-convex problem effectively, we introduce the proximal operator and adopt the BCD method.

Utilizing the proximal operator, we perform the update as

$$h(\mathcal{Z},\mathcal{Z}^k)=f(\mathcal{Z})+\frac{\rho}{2}\|\mathcal{Z}-\mathcal{Z}^k\|_F^2,\tag{15}$$

[1] The anisotropic TV is defined as: $\text{TV}_a(X_3):=\sum_{k=1}^{\hat{s}}\sum_{i=1}^{r_3}\sum_{j=1}^{I_1I_2}|\tilde{D}_{j,1}X_3^{(i,k)}|+|\tilde{D}_{j,2}X_3^{(i,k)}|$. We mainly talk about the isotropic TV in our work, and the anisotropic case is similar.

where $\rho > 0$ is the proximal parameter, $\mathcal{Z} = (X, A, \mathcal{Y})$ and $\mathcal{Z}^k = (X^k, A^k, \mathcal{Y}^k)$. Note that the problem (15) can be rewritten as follows:

$$\begin{aligned} X^{k+1} &= \underset{X}{\operatorname{argmin}} \left\{ h_1(X, \mathcal{Z}_1^k) = f(X, A^k, \mathcal{Y}^k) + \frac{\rho}{2}\|X - X^k\|_F^2 \right\}, \\ A^{k+1} &= \underset{A}{\operatorname{argmin}} \left\{ h_2(A, \mathcal{Z}_2^k) = f(X^{k+1}, A, \mathcal{Y}^k) + \frac{\rho}{2}\|A - A^k\|_F^2 \right\}, \\ \mathcal{Y}^{k+1} &= \underset{\mathcal{Y}}{\operatorname{argmin}} \left\{ h_3(\mathcal{Y}, \mathcal{Z}_3^k) = f(X^{k+1}, A^{k+1}, \mathcal{Y}) + \frac{\rho}{2}\|\mathcal{Y} - \mathcal{Y}^k\|_F^2 \right\}, \end{aligned} \tag{16}$$

where $\mathcal{Z}_1^k = (X^k, A^k, \mathcal{Y}^k)$, $\mathcal{Z}_2^k = (X^{k+1}, A^k, \mathcal{Y}^k)$, $\mathcal{Z}_3^k = (X^{k+1}, A^{k+1}, \mathcal{Y}^k)$. Note that the X- and A- sub-problems can be solved parallelly, because they can be decomposed into N independent problems. Then the updates (16) can be written explicitly as:

$$\begin{aligned} X_n^{k+1} &= \begin{cases} ((A_n^k)^T A_n^k + \rho I_1)^\dagger ((A_n^k)^T Y_{(n)}^k + \rho X_n^k), & n = 1, 2, 4, \ldots, N, \\ \underset{X_3}{\operatorname{argmin}} \frac{1}{2}\|Y_{(3)}^k - A_3^k X_3\|_F^2 + \mu \mathrm{TV}(X_3) + \frac{\rho}{2}\|X_3 - X_3^k\|_F^2, & n = 3, \end{cases} \\ A_n^{k+1} &= (Y_{(n)}^k (X_n^{k+1})^T + \rho A_n^k)(X_n^{k+1}(X_n^{k+1})^T + \rho I_2)^\dagger, \ n = 1, 2, 3, 4, \ldots, N, \\ \mathcal{Y}^{k+1} &= \mathcal{P}_{\Omega^c}\left(\sum_{n=1}^N \alpha_n \mathrm{fold}_n \left(\frac{A_n^{k+1} X_n^{k+1} + \rho Y_{(n)}^k}{1+\rho} \right) \right) + \mathcal{F}, \end{aligned} \tag{17}$$

where $\mathcal{F}$ is the observed data. It can be seen that all the subproblems except for the X_3 subproblem can be easily solved. The complexity of computing $\mathcal{Y}$ is $O(r_1 I_1 s_1 + \cdots + r_N I_N s_N)$, the cost of computing A_n is $O(I_n r_n^2 + I_n r_n s_n + r_n^2 s_n)$ operations for $n = 1, 2, \ldots, N$, and the cost of computing X_n is $O(I_n r_n^2 + I_n r_n s_n + r_n^2 s_n)$ operations for $n = 1, 2, 4, \ldots, N$.

Following, we solve X_3 subproblem using ADM [30,43,45]. Let

$$g(X_3) = \underset{X_3}{\operatorname{argmin}} \frac{1}{2}\|Y_{(3)}^k - A_3^k X_3\|_F^2 + \frac{\rho}{2}\|X_3 - X_3^k\|_F^2 + \mu \mathrm{TV}(X_3).$$

In order to get the close form solution of $g(X_3)$, we solve its following equivalent problem:

$$\hat{g}(\hat{X}_3) = \underset{\hat{X}_3}{\operatorname{argmin}} \frac{1}{2}\|\hat{Y}_{(3)}^k - \hat{X}_3 \hat{A}_3^k\|_F^2 + \frac{\rho}{2}\|\hat{X}_3 - \hat{X}_3^k\|_F^2 + \mu \mathrm{TV}(\hat{X}_3),$$

where $\hat{X}$ denotes the transpose of X. Then apply ADM for $\hat{g}(\hat{X}_3)$ and rewrite the problem as follows:

$$\begin{aligned} \min_{\hat{X}_3, W} \quad & \mu \sum_{i=1}^{s_3} \sum_{j=1}^{r_3} \left\| W_{i,j} \right\|_2 + \tfrac{1}{2}\|\hat{Y}_{(3)}^k - \hat{X}_3 \hat{A}_3^k\|_F^2 + \tfrac{\rho}{2}\|\hat{X}_3 - \hat{X}_3^k\|_F^2 \\ \text{s.t.} \quad & W_1 = D_1 \hat{X}_3, \ W_2 = D_2 \hat{X}_3, \end{aligned} \tag{18}$$

where $W_{i,j} = [(W_1)_{i,j}, (W_2)_{i,j}] \in \mathbb{R}^{1\times 2}$ with $(W_1)_{i,j}$ and $(W_2)_{i,j}$ are the (i,j)th entries of W_1 and W_2, respectively. Moreover,

$$D_i := \begin{bmatrix} \tilde{D}_i & & & \\ & \tilde{D}_i & & \\ & & \ddots & \\ & & & \tilde{D}_i \end{bmatrix},$$

where, $i = 1, 2$, and $\tilde{D}_1$ and $\tilde{D}_2$ denote the assembled first-order difference matrices in the 1st- and 2nd-mode directions based on $\tilde{D}_{j,1}$ and $\tilde{D}_{j,2}$, respectively, in (12).

Now, we use ADM to solve the problem. For simplicity, let

$$\hat{g}_1(\hat{X}_3) = \frac{1}{2}\|\hat{Y}_{(3)}^k - \hat{X}_3 \hat{A}_3^k\|_F^2 + \frac{\rho}{2}\|\hat{X}_3 - \hat{X}_3^k\|_F^2,$$

and

$$\hat{g}_2([W_1\ W_2]) = \mu \sum_{i=1}^{s_3} \sum_{j=1}^{r_3} \left\| W_{i,j} \right\|_2.$$

The constraints are rewritten as follows:

$$B\hat{X}_3 + CW := \begin{bmatrix} D_1 \\ D_2 \end{bmatrix} \hat{X}_3 - I_{2s_3 \times 2s_3} \begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = 0_{2s_3 \times r_3},$$

where $I_{i\times i}$ is the i-by-i identity matrix. Then we can solve the problem (18) in two decoupled subproblems, which the convergence can be guaranteed [10].

The augmented Lagrangian function of (18) is

$$L(\hat{X}_3, W, \Lambda) = \hat{g}_1(\hat{X}_3) + \hat{g}_2(W) + \langle \Lambda, B\hat{X}_3 + CW\rangle + \frac{\beta}{2}\left\|B\hat{X}_3 + CW\right\|_F^2, \tag{19}$$

where $\Lambda = (\Lambda_1, \Lambda_2)^T$, and $\beta > 0$ is the penalty parameter. Then, the problem (19) is updated through alternating direction as:

$$\begin{cases} X_3\text{-subproblem:} & \hat{X}_3^{k+1,p+1} \in \operatorname{argmin} L(\hat{X}_3, W^p, \Lambda^p), \\ W\text{-subproblem:} & W^{p+1} \in \operatorname{argmin} L(\hat{X}_3^{k+1,p+1}, W, \Lambda^p), \\ \Lambda\text{-subproblem:} & \Lambda^{p+1} = \Lambda^p + \beta(B\hat{X}_3^{k+1,p+1} + CW^{p+1}), \end{cases} \tag{20}$$

where p is the iteration indicator for solving the problem (18).

For the X_3-subproblem, we solve the following problem:

$$\hat{X}_3^{k+1,p+1} \in \operatorname{argmin}\left\{\frac{1}{2}\|\hat{Y}_{(3)}^k - \hat{X}_3\hat{A}_3^k\|_F^2 + \frac{\rho}{2}\|\hat{X}_3 - \hat{X}_3^k\|_F^2 + \langle \Lambda^p, B\hat{X}_3 + CW^p\rangle + \frac{\beta}{2}\left\|B\hat{X}_3 + CW^p\right\|_F^2\right\}, \tag{21}$$

then $\hat{X}_3^{k+1,p+1}$ can be solved through the classical Sylvester matrix equation

$$\hat{X}_3(\hat{A}_3^k(\hat{A}_3^k)^T) + \beta B^T B\hat{X}_3 + \rho\hat{X}_3 = \rho\hat{X}_3^k + \hat{Y}_{(3)}^k(\hat{A}_3^k)^T - B^T\Lambda^p - \beta B^T CW^p. \tag{22}$$

The (22) can be reformulated as the following form using the Kronecker product notations:

$$\begin{aligned}&(\hat{A}_3^k(\hat{A}_3^k)^T \otimes I + \beta I \otimes B^T B + \rho_1 I \otimes I)\operatorname{vec}(\hat{X}_3) \\ &\quad = \operatorname{vec}(\rho\hat{X}_3^k + \hat{Y}_{(3)}^k(\hat{A}_3^k)^T - B^T\Lambda^p - \beta B^T CW^p),\end{aligned} \tag{23}$$

where vex$(\cdot)$ refers to a vector by lexicographical ordering of the entries in a matrix. Using singular value decomposition of $\hat{A}_3^k$ and the Fourier decomposition of B^TB with periodic boundary condition,

$$\hat{A}_3^k = U\Sigma V^*, \quad B^T B = F^*\Psi^2 F, \tag{24}$$

we can solve the problem (23) efficiently. The cost of computing the singular value decomposition of A_3 is $O(r_3^2 I_3)$, and the cost of computing Fourier decompositions of B^TB is $O(s_3^2 \log s_3)$. Then, (23) can be rewritten as:

$$\begin{aligned}&(U \otimes F^*)(\Sigma^2 \otimes I + \beta I \otimes \Psi^2 + \rho I \otimes I)(U^* \otimes F)\operatorname{vec}(\hat{X}_3) \\ &\quad = \operatorname{vec}(\rho\hat{X}_3^k + \hat{Y}_{(3)}^k(\hat{A}_3^k)^T - B^T\Lambda^p - \beta B^T CW^p),\end{aligned} \tag{25}$$

the solution $\operatorname{vec}(\hat{X}_3)$ is explicitly expressed as:

$$\begin{aligned}\operatorname{vec}(\hat{X}_3) = {} & (U \otimes F^*)(\Sigma^2 \otimes I + \beta I \otimes \Psi^2 + \rho I \otimes I)^{-1}(U^* \otimes F) \\ & \cdot \operatorname{vec}(\rho\hat{X}_3^k + \hat{Y}_{(3)}^k(\hat{A}_3^k)^T - B^T\Lambda^p - \beta B^T CW^p).\end{aligned} \tag{26}$$

At each iteration, the cost of computing $\operatorname{vec}(\rho\hat{X}_3^k + \hat{Y}_{(3)}^k(\hat{A}_3^k)^T - B^T\Lambda^p - \beta B^T CW^p)$ is $O(s_3 r_3)$ operations, and the cost of computing the product of matrix $(U\otimes F^*)$ (or $(U^*\otimes F)$) with an $r_3 s_3$-vector is $O(s_3 r_3^2 + s_3 r_3 \log s_3)$. Therefore, the complexity of obtaining $\operatorname{vec}(\hat{X}_3)$ is $O(2s_3 r_3^2 + s_3 r_3 \log s_3)$ at each iteration.

For the W-subproblem, we solve the following problem:

$$W^{p+1} = \operatorname{argmin}\mu\sum_{i=1}^{s_3}\sum_{j=1}^{r_3}\left\|W_{i,j}\right\|_2 + \frac{\beta}{2}\left\|B\hat{X}_3^{k+1,p+1} + CW + \frac{\Lambda^p}{\beta}\right\|_F^2, \tag{27}$$

which, can be determined independently by solving $s_3 r_3$ two-variable minimization problems:

$$\begin{aligned}&\min\mu\sqrt{|(W_1)_{i,j}|^2 + |(W_2)_{i,j}|^2} + \frac{\beta}{2}[(W_1)_{i,j} - (D_1\hat{X}_3^{k+1,p+1})_{i,j} - \frac{1}{\beta}(\Lambda_1^p)_{i,j}]^2 \\ &\quad + \frac{\beta}{2}[(W_2)_{i,j} - (D_2\hat{X}_3^{k+1,p+1})_{i,j} - \frac{1}{\beta}(\Lambda_2^p)_{i,j}]^2.\end{aligned} \tag{28}$$

And (28) can be solved by using the well-known 2-D shrinkage formula as follows:

$$[(W_1)_{i,j},\ (W_2)_{i,j}] = \max\{\|S_{i,j}\|_2 - \frac{\mu}{\beta},\ 0\}\frac{S_{i,j}}{\|S_{i,j}\|_2},\ 1 \le i \le s_3,\ 1 \le j \le r_3, \tag{29}$$

where, $S_{i,j} = [(D_1\hat{X}_3^{k+1,p+1})_{i,j} + \frac{1}{\beta}(\Lambda_1^p)_{i,j},\ (D_2\hat{X}_3^{k+1,p+1})_{i,j} + \frac{1}{\beta}(\Lambda_2^p)_{i,j}]$ for $1 \le i \le s_3, 1 \le j \le r_3$, and we assign $0 \cdot (0/0) = 0$. The cost of computing $[W_1, W_2]$ is $O(s_3 r_3)$.

At each iteration, the cost of computing all the variables A_n, X_n, and $Y_{(n)}$ is $O(I_3 r_3^2 + 2I_3 r_3 s_3 + 3r_3^2 s_3 + r_3 s_3 \log s_3 + \sum_{n\neq 3}(2I_n r_n^2 + 3I_n r_n s_n + 2r_n^2 s_n))$.

In the following, we study the convergence of the proposed algorithm. Recently, Razaviyayn et al. [27] proposed the block successive upper-bound minimization (BSUM) for non-smooth optimization problem. It is an alternative block coordinate decent method. Following, we restated the convergence result in [27] for convenience.

Lemma 1. *Given the problem* $\min f(x)$ *s.t.* $x \in \mathcal{X}$, *where* $\mathcal{X}$ *is the feasible set. Assume* $u(x, x^{k-1})$ *is an approximation of* $f(x)$ *at the* $k-1$*-th iteration, which satisfied the following conditions:*

$$\begin{aligned}
&u_i(y_i, y) = f(y),\ \forall y \in \mathcal{X}, \forall i,\\
&u_i(x_i, y) \geq f(y_1, \ldots, y_{i-1}, x_i, y_{i+1}, \ldots, y_n),\ \forall x_i \in \mathcal{X}_i, \forall y \in \mathcal{X}, \forall i,\\
&u_i'(x_i, y; d_i)\mid_{x_i=y_i} = f'(y; d),\ \forall d = (0, \ldots, d_i, \ldots, 0) \text{ s.t. } y_i + d_i \in \mathcal{X}_i,\ \forall i,\\
&u_i(x_i, y) \text{ is continuous in } (x_i, y),\ \forall i,
\end{aligned} \tag{30}$$

where $u_i(x_i, y)$ *is the sub-problem with respect to the ith block and* $f'(y; d)$ *is the direction derivative of f at the point y in direction d. Suppose* $u_i(x_i, y)$ *is quasi-convex in* x_i *for* $i = 1, \ldots, n$. *Furthermore, assume that each sub-problem* $\text{argmin} u_i(x_i, x^{k-1})$, *s.t.* $x \in \mathcal{X}_i$ *has a unique solution for any point* $x^{k-1} \in \mathcal{X}$. *Then, the iterates generated by the BSUM algorithm converge to the set of coordinatewise minimum of f.*

Next we will show that the convergence of the proposed algorithm for the model (8) is guaranteed, as it fits the framework of the BSUM method.

Theorem 1. *The iterates generated by (15) converge to the set of coordinatewise minimizers.*

Proof. It is easy to verify that $h(\mathcal{Z}, \mathcal{Z}^k)$ is an approximation and a global upper bound of $f(\mathcal{Z})$ at the kth iteration, which satisfies the following conditions:

$$\begin{aligned}
&h_i(\mathcal{Z}_i, \mathcal{Z}) = f(\mathcal{Z}),\ \forall \mathcal{Z}, i = 1, 2, 3,\\
&h_i(\bar{\mathcal{Z}}_i, \mathcal{Z}) \geq f(\mathcal{Z}_1, \ldots, \bar{\mathcal{Z}}_i, \ldots, \mathcal{Z}_3),\ \forall \bar{\mathcal{Z}}_i, \forall \mathcal{Z}, i = 1, 2, 3,\\
&h_1'(\bar{\mathcal{Z}}_1, \mathcal{Z}; \mathcal{D}_1)\mid_{\bar{\mathcal{Z}}_1=\mathcal{Z}_1} = f'(\mathcal{Z}; \mathcal{D}^1),\ \forall \mathcal{D}^1 = (\mathcal{D}_1, 0, 0),\\
&h_2'(\bar{\mathcal{Z}}_2, \mathcal{Z}; \mathcal{D}_2)\mid_{\bar{\mathcal{Z}}_2=\mathcal{Z}_2} = f'(\mathcal{Z}; \mathcal{D}^2),\ \forall \mathcal{D}^2 = (0, \mathcal{D}_2, 0),\\
&h_3'(\bar{\mathcal{Z}}_3, \mathcal{Z}; \mathcal{D}_3)\mid_{\bar{\mathcal{Z}}_3=\mathcal{Z}_3} = f'(\mathcal{Z}; \mathcal{D}^3),\ \forall \mathcal{D}^3 = (0, 0, \mathcal{D}_3),\\
&h_i(\bar{\mathcal{Z}}_i, \mathcal{Z}) \text{ is continuous in } (\bar{\mathcal{Z}}_i, \mathcal{Z})\ i = 1, 2, 3,
\end{aligned} \tag{31}$$

where $\mathcal{Z} = (X, A, \mathcal{Y})$, and $\mathcal{Z}_i$ equal $X, A, \mathcal{Y}$ for $i = 1, 2, 3$, respectively. In addition, the sub-problem h_i, $(i = 1, 2, 3)$ is strictly convex with respect to X, A, and $\mathcal{Y}$ respectively and thus each sub-problem has a unique solution. Therefore, all assumptions in Lemma 1 are satisfied. □

3. Numerical experiments

In this section, we test the performance of the proposed model for the tensor completion. The quality of the estimated tensor is measured by the peak signal-to-noise ratio (PSNR) and the relative squared error (RSE), which are defined by

$$\text{PSNR} = 10 \log_{10} \frac{\bar{\mathcal{Y}}_{\text{true}}^2}{\frac{1}{n^2}\|\mathcal{Y} - \mathcal{Y}_{\text{true}}\|_F^2},$$

and

$$\text{RSE} = \frac{\|\mathcal{Y} - \mathcal{Y}_{\text{true}}\|_F}{\|\mathcal{Y}_{\text{true}}\|_F},$$

where $\mathcal{Y}_{\text{true}}$, $\bar{\mathcal{Y}}_{\text{true}}$, and $\mathcal{Y}$ are the original tensor, the maximum pixel value of the original tensor, and the estimated tensor, respectively. We adopt the relative change of the two successive estimated tensors (RelCha), i.e.,

$$\frac{\|\mathcal{Y}^{k+1} - \mathcal{Y}^k\|_F}{\|\mathcal{Y}^k\|_F} < \text{tol},$$

as the stopping criterion for TMac[2] [42] and the proposed method. Here, the tolerance is set to be 10^{-5} for the synthetic data and be 10^{-4} for the real data. All the parameters in our method can be fixed, i.e., the weights $\alpha_n = 1/N$, the regularization parameter $\mu = 1$, the penalty parameter $\beta = 10$, and the proximal parameter $\rho = 0.1$ in the all examples. In all the experiments, the masked data are obtained by randomly removing some entries. All the tests are performed under Windows 7 and Matlab Version 8.2.0.701 (R2013b) running on a desktop with an Inter(R) Core(TM) i3-4160 CPU at 3.60 GHz and 4GB of memory.

[2] In [42] the authors adjust the rank along with iterations, but we do not use the the rank-adjusting for fairness in the research.

Table 1
PSNR and RSE comparison of the results recovered by FaLRTC [24], TMac [42] and the proposed method for different sampling rates.

	SR	FaLRTC			TMac			The proposed method		
		Whole	Average	Worst	Whole	Average	Worst	Whole	Average	Worst
PSNR	10%	**24.10**	25.19	22.56	16.24	20.51	9.56	22.45	28.95	17.71
	30%	29.25	31.06	26.03	38.71	54.16	34.21	**43.88**	56.29	40.82
	50%	34.55	36.42	29.97	44.19	96.50	34.19	**50.20**	66.08	40.41
RSE	10%	**122**	116	145	301	257	647	147	127	253
	30%	67.4	62.4	97.1	24.4	19.2	45.4	**13.9**	10.8	24.3
(10^{-3})	50%	36.6	33.7	61.7	12.1	3.89	38.0	**6.04**	3.01	18.6

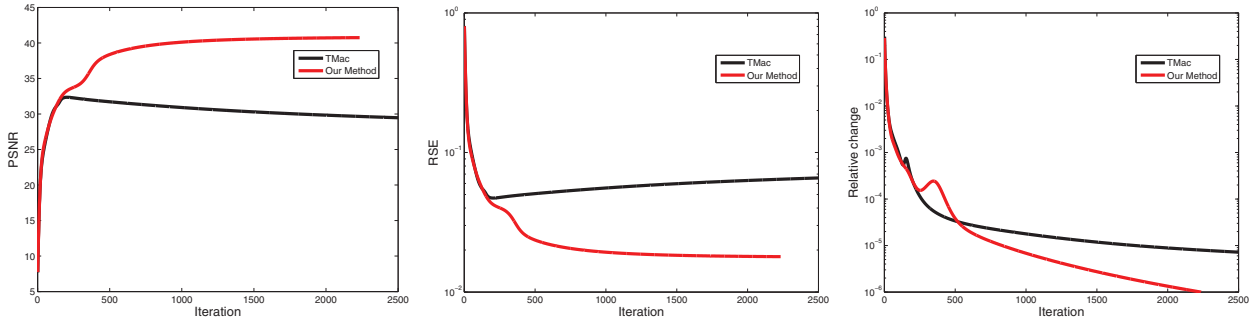

Fig. 2. The convergence behaviors of the proposed algorithm with respect to the PSNR value, the RSE value, and the RelCha value.

3.1. Synthetic data

In this subsection, we evaluate the proposed model on synthetic data. The size of the test data is $50 \times 50 \times 50$. The n-rank of the test data is (17,17,9). We study the influence of different sampling rates (SR), i.e., SR = 10%, SR = 30%, and SR = 50% for FaLRTC, TMac and the proposed method, and the results are summarized in Table 1, where the words "Whole", "Average", and "Worst" denote the PSNR values and RSE values of tensor as a whole, the average of the PSNR values and RSE values of the all frames, and the worst PSNR values and RSE values, respectively. From this table, one can observe: (1) the larger the sampling rate, the better the results recovered by both the methods; (2) for both low and high sampling rates, our results are better than the TMac results; (3) the worst frame recovered by the proposed method are better than that by TMac;(4) for SR = 10%, the result recovered by FaLRTC is better than our result. However, the average of the PSNR values and RSE values of the all recovered frames by our method is better than that by FaLRTC.

The convergence behaviors of the PSNR value, the RSE value, and the RelCha value are displayed in Fig. 2. We observe that the RelCha values of both the methods decrease almost at the same speed in the early iterations, and after some iterations, the RelCha values of our method decrease faster than TMac. In addition, the TMac exhibits the semi-convergence behavior with respect to the PSNR value and the RSE value, i.e., in the early iterations the PSNR value begins to increase and after some "optimal" iterations the PSNR value then begins to decrease, and the RSE value behaves the opposite. In contrast, the PSNR value is still increasing and the RSE value is still decreasing in the latter iterations for the proposed model, and then they keep about the same finally. In other words, our method is more stable.

3.2. Real data

3.2.1. Video

In this section, we compare the performance of FaLRTC, TMac and the proposed method on videos. We test two videos, named as "suzie" and "coastguard".[3] Both the two videos are color with the YUV format. We used Y channel in our tests. We used the first 150 frames of both "suize" and "coastguard", thus both test tensors are of size $144 \times 176 \times 150$. The numbers of the largest 0.5% singular values are used to approximate the rank of each mode. In Fig. 3, we illustrate the recovered results of two frames of "suzie" and "coastguard" by FaLRTC, TMac and the proposed method. From Fig. 3, we note that the proposed method obtain higher quality results for both 50% and 20% sampling rates. The PSNR values and the RSE values against the frame number are plotted in Fig. 4. We see that the proposed method performs quite well in terms of PSNR values and RSE values for almost frames with low and high sampling rates.

[3] The videos can be downloaded from http://trace.eas.asu.edu/yuv/.

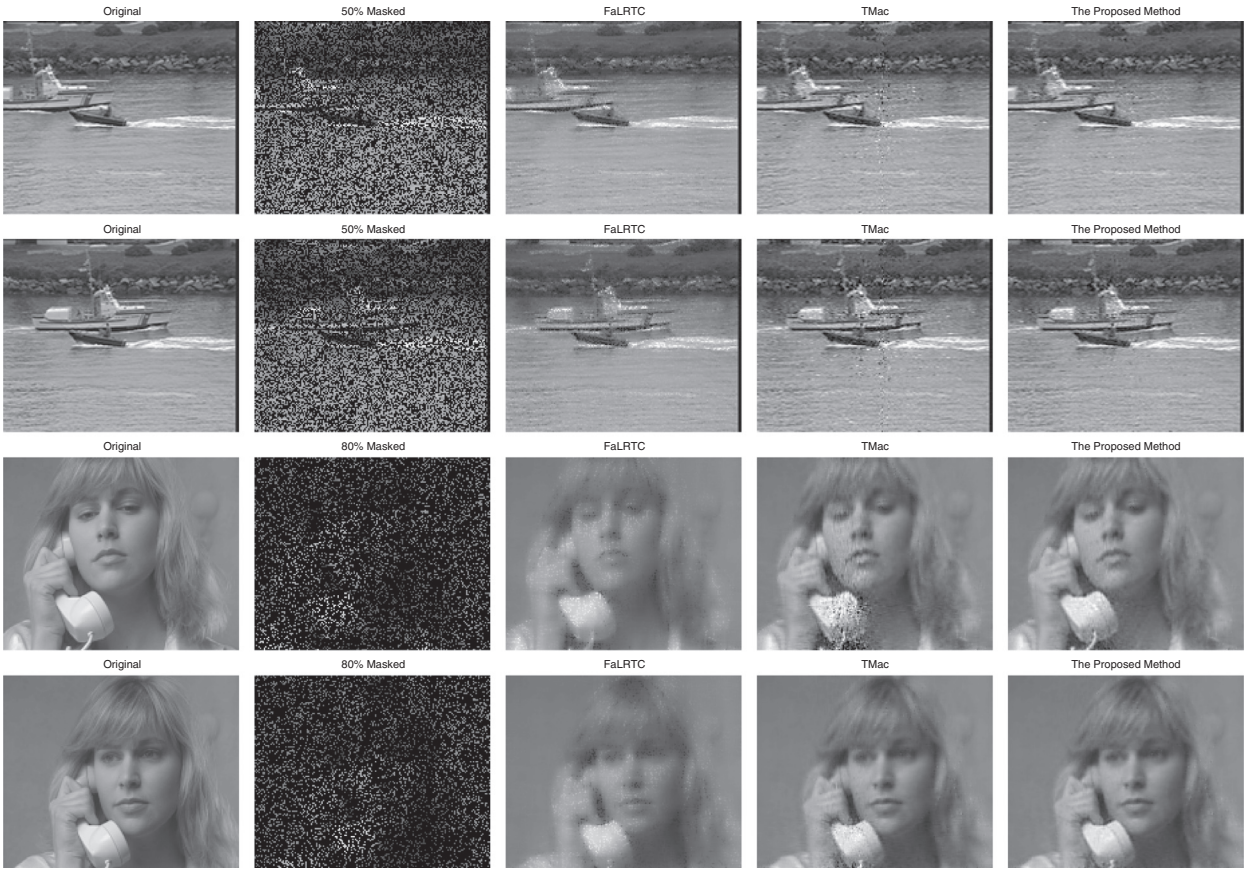

Fig. 3. From left to right: the original video, the masked video, and the estimated results by FaLRTC, TMac and the proposed method. The first two rows are for "coastguard", which has 50% pixels missing. The last two rows are for "suzie", which has 80% pixels missing.

3.2.2. MRI

In this section, we test the performance of FaLRTC, TMac and the proposed method on MRI data. We use the cubical MRI data[4], which is a 3-mode tensor of size $181 \times 217 \times 181$. The numbers of the largest 0.5% singular values are used to approximate the rank of each mode. The MRI data is tridimensional, so we display the results in three different views illustrated in Fig. 5. We randomly remove 80% elements of the test data. Fig. 6 shows the recovered results observed from different directions. Clearly, the restored results obtained by the proposed method are visually better than those obtained by FaLRTC and TMac. Fig. 7 shows the PSNR values of every frame recovered by TMac and the proposed method for three directions. We note that every frame recovered by the proposed method is better than that recovered by TMac.

3.2.3. Hyperspectral image

The Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Cuprite data[5] is considered in our real data experiment. The hyperspectral data used in experiments correspond to a 15×150 pixels subset with 188 spectral bands; see [45], [13] and [14] for more details. The numbers of the largest 0.5% singular values are used to approximate the rank of each mode. In Fig. 8, we display the recovered results by FaLRTC, TMac and the proposed method. It is clear from the figure that the estimated results by the proposed method are visually better than those by FaLRTC and TMac. The PSNR values and the RSE values of each frame are showed in Fig. 9. We see that the proposed method performs better in terms of the PSNR values and the RSE values of each frame as compared with TMac.

3.3. Robustness

In this section, we test the robustness of the proposed method with respect to the different initial guesses and different parameters. The sampling rate of this section is set to be 30%.

[4] http://brainweb.bic.mni.mcgill.ca/brainweb/selection_normal.html
[5] http://aviris.jpl.nasa.gov/html/aviris.freedata.html.

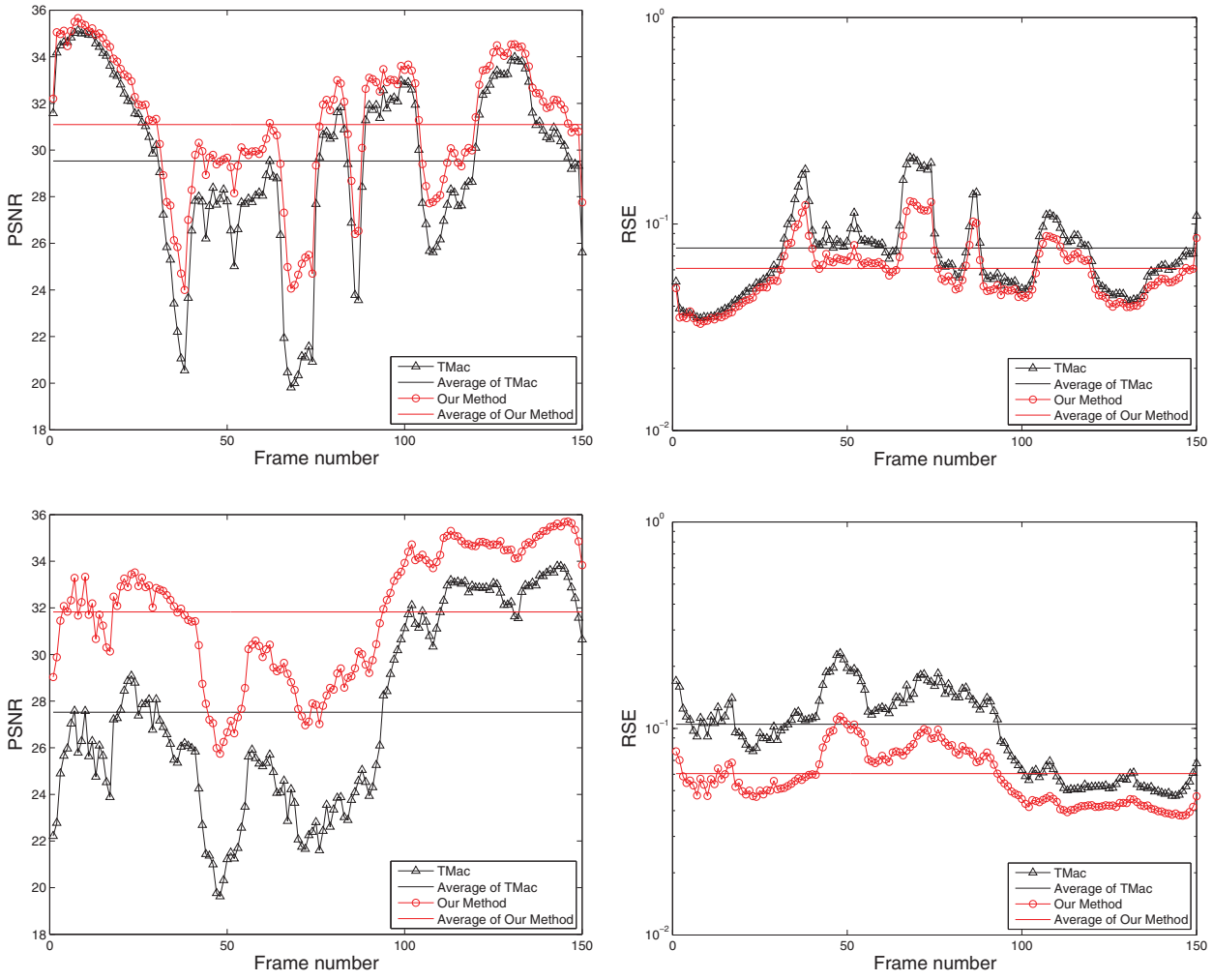

Fig. 4. The PSNR values and the RSE values recovered by TMac [42] and the proposed method for every frame. The first row is for "coastguard" and the second row is for "suzie".

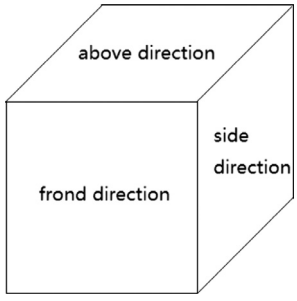

Fig. 5. The description of three directions for MRI data.

First, we study the influence of the different guesses of X^0 and A^0. We test three cases: the zero matrix, the Gaussian random matrix, and the all-ones matrix, which are denoted by "zeros", "random", and "ones", respectively. First of all, we have to clarify that it is meaningless that both the A^0 and X^0 are set to be zero matrices. In this case, $Y_{(n)}$ in (7) and (8) equals to zero matrix for all n, namely, the solution of (7) and (8) is the marked data $\mathcal{F}$. We display the PSNR values, the RSE values, and the iteration number for different guesses of X^0 and A^0 in Table 2. From the table, we can see that the proposed method can estimate the missing data from the masked data successfully for different guesses of X^0 and A^0. However, TMac fails when A^0 is set to be zero matrix. For the results recovered by the proposed method, the difference between the best RSE value and the worst RSE value is

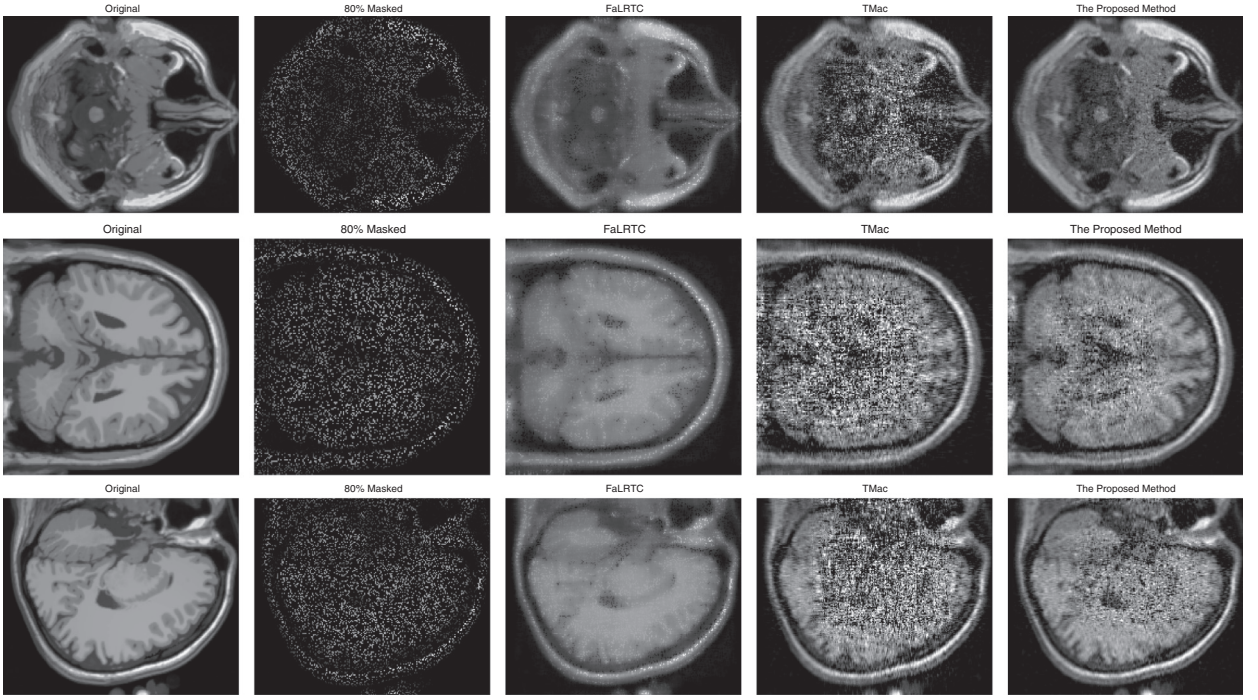

Fig. 6. The recovered results for three directions. From left to right: the original data, the masked data, and the recovered results by FaLRTC, TMac and the proposed method. From top to bottom: the results for the front direction, the side direction and the above direction, respectively.

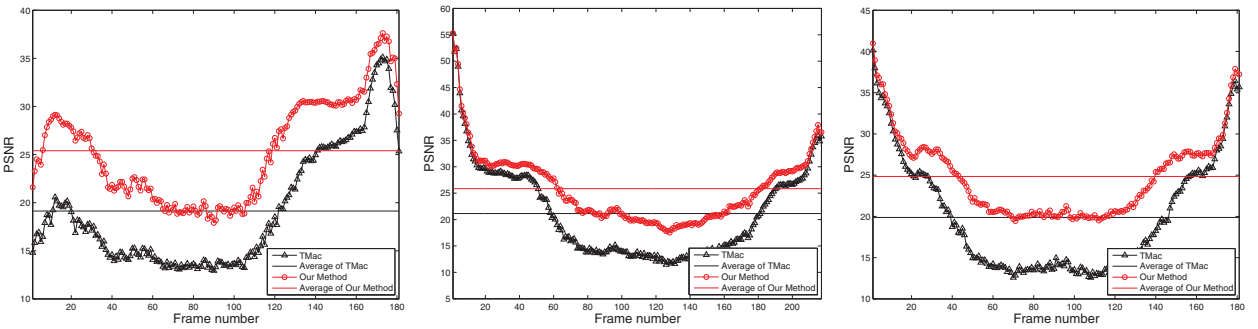

Fig. 7. The PSNR values recovered by TMac [42] and the proposed method for every frame. From left to right: the PSNR values for the front direction, the side direction, and the above direction, respectively.

Table 2
The results recovered by TMac and the proposed method with different initial guesses.

X^0	A^0	TMac			The proposed method		
		PSNR	RSE (10^{-3})	Iter	PSNR	RSE (10^{-3})	Iter
	Zeros	*	*	*	*	*	*
Zeros	Random	40.49	18.5	161	45.40	10.5	325
	Ones	26.18	96.0	32	46.04	9.75	219
	Zeros	*	*	*	41.56	16.3	1542
Random	Random	40.93	17.6	170	45.30	10.6	299
	Ones	26.18	96.0	32	40.68	18.1	1642
	Zeros	*	*	*	46.01	9.79	214
Ones	Random	42.24	15.1	170	45.56	10.3	303
	Ones	26.18	96.0	32	46.07	9.72	217

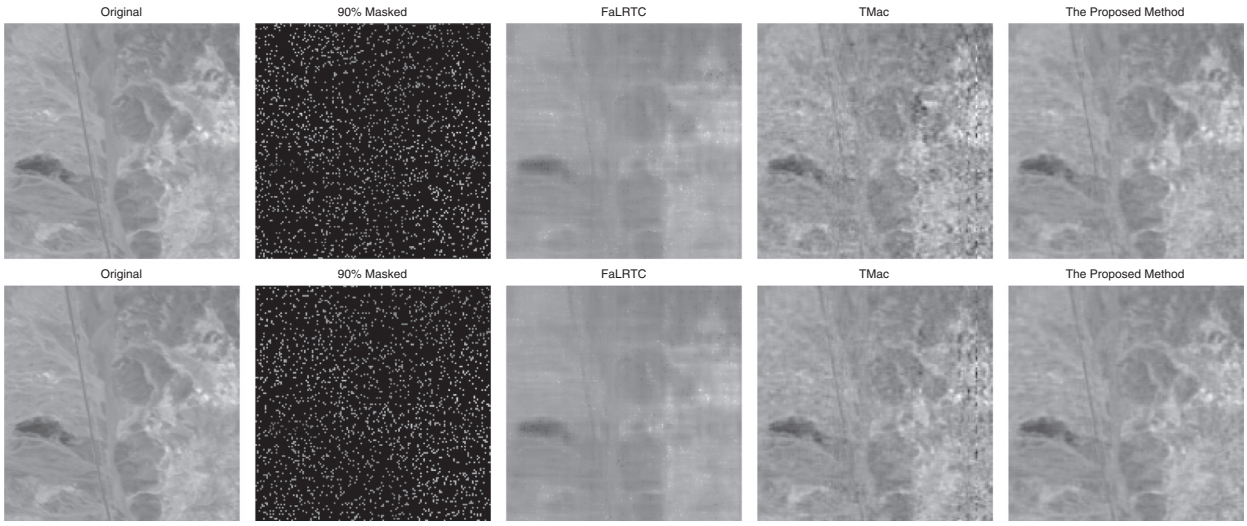

Fig. 8. The results recovered by FaLRTC, TMac and the proposed method. Each row represents a example of the recovered frames. From left to right: the original data, the masked data, the recovered results by FaLRTC, TMac and the proposed method.

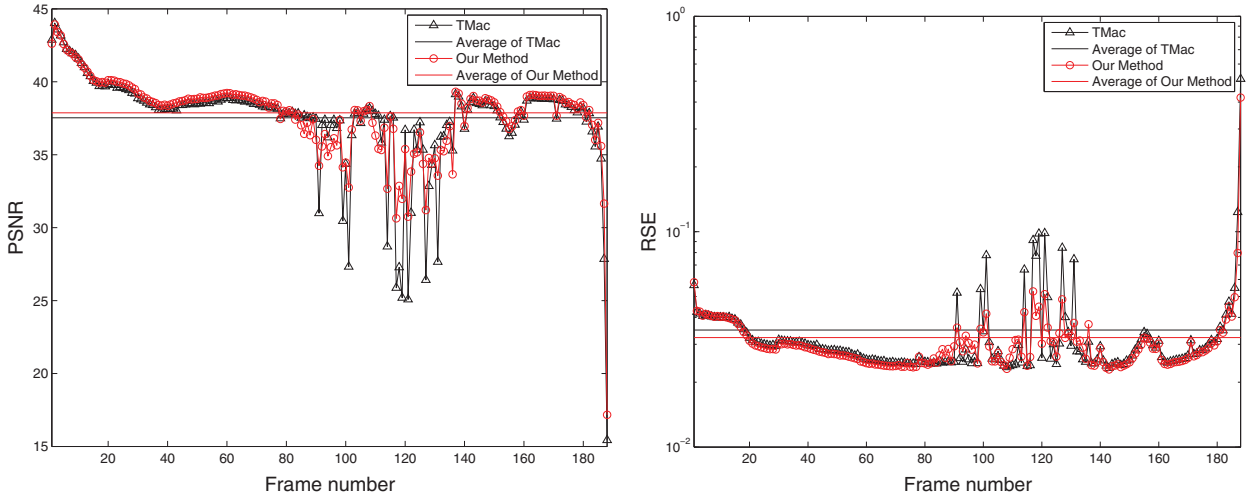

Fig. 9. The PSNR values and the RSE values recovered by TMac [42] and the proposed method for every frame.

8.40×10^{-3}, and for the results recovered by TMac, the difference is 8.09×10^{-2}. Thus, we conclude that the proposed method is more robust against the different initial guesses.

Second, we study the influence of the estimation for R. Here, $R = (17, 17, 9)$ denotes the n-rank of the synthetic data. We set the estimation to be $R + i$, where $i = -8, -6, \ldots, 32$. The PSNR value and the RSE value with respect to the estimated rank are plotted in Fig. 10. We observe that the PSNR value and the RSE value of TMac change rapidly, when the estimated rank is far away from the true one. In contrast, the change of the PSNR value and the RSE value is smaller. In other words, the proposed method is more stable.

Finally, we study the effects of the proximal parameter ρ, regularization parameter μ and penalty parameter β. The PSNR value and the RSE value with respect to iterations are displayed in Fig. 11 for different proximal parameters. The plot suggests that the proposed method can reach the same PSNR value and the same RSE value for different proximal parameters. The proximal parameter indeed affects the performance of the proposed method in terms of the iteration number. The PSNR value and the RSE value with respect to iterations are displayed in Fig. 12 for different regularization parameters. The plot suggests that we should select the regularization parameter carefully. And the regularization parameter also affects the performance of the method in terms of the iteration number. The PSNR value and RSE value with respect to iterations are reported in Fig. 13 for different penalty parameters. The convergence of the proposed method is theoretically guaranteed regardless of the penalty parameter as long as it is a positive number. In practice, we observe that the numerical performances of the proposed method are affected by the value of penalty parameter.

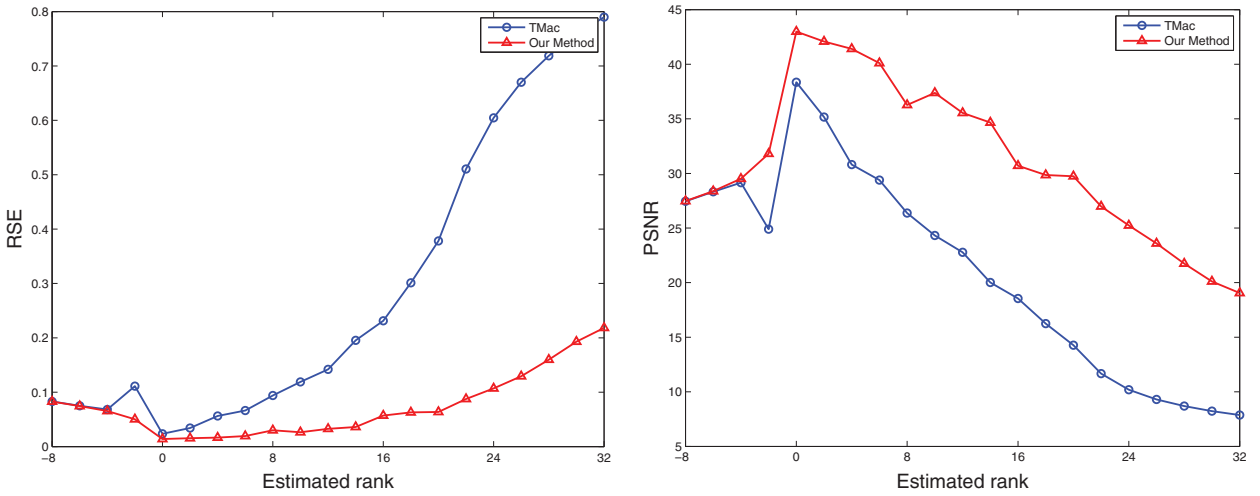

Fig. 10. The PSNR and RSE comparison against the estimated rank on synthetic data. The n-rank of the synthetic data is $R = (17, 17, 9)$ denoted as 0 in X-coordinate. And i in X-coordinate denotes R + i, $i = -8, -6, \ldots, 32$.

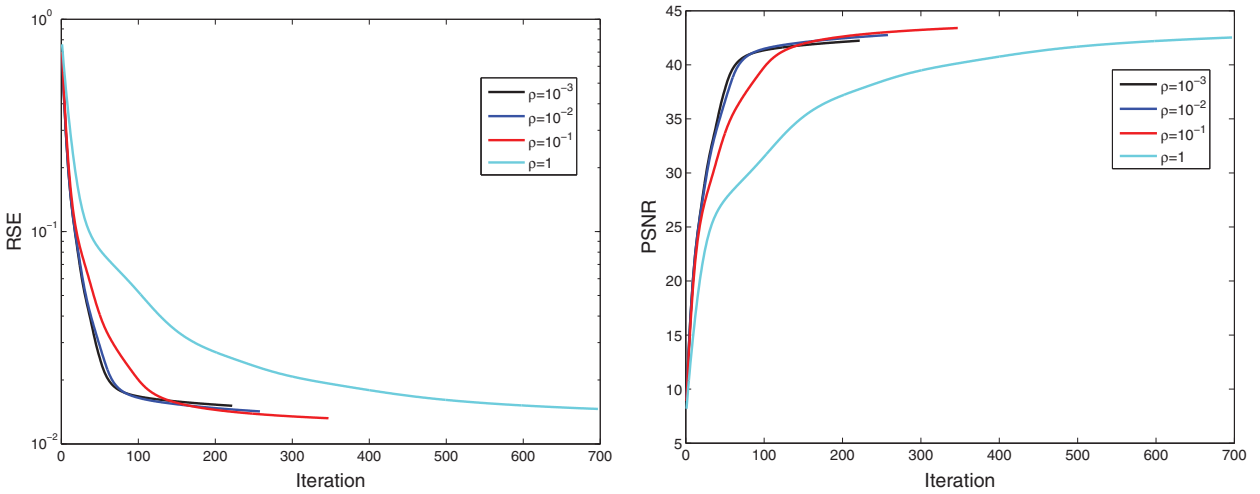

Fig. 11. The RSE values and the PSNR values with respect to iterations for different proximal parameters ρ.

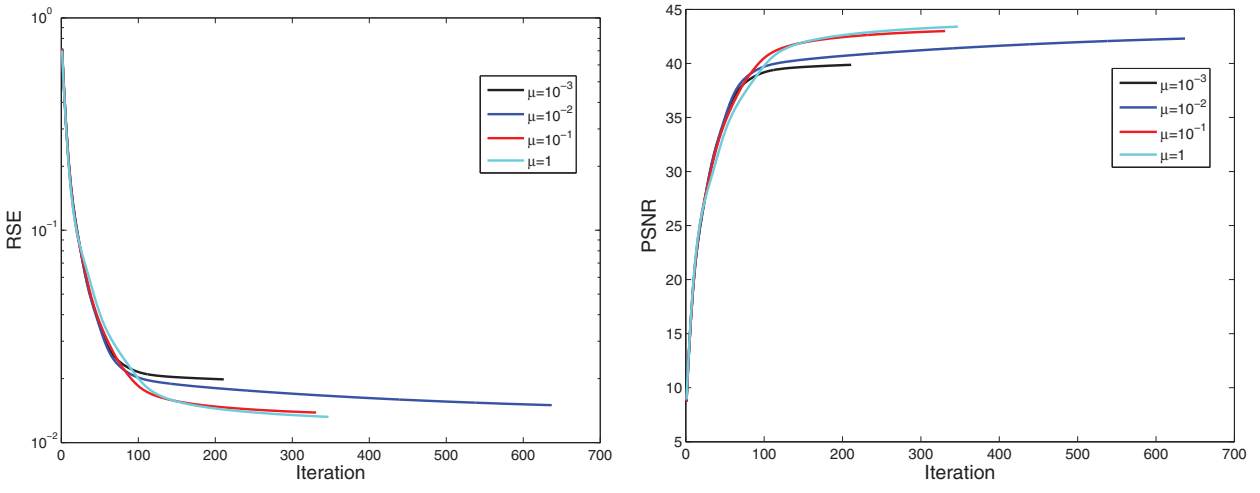

Fig. 12. The RSE values and the PSNR values with respect to iterations for different regularization parameters μ.

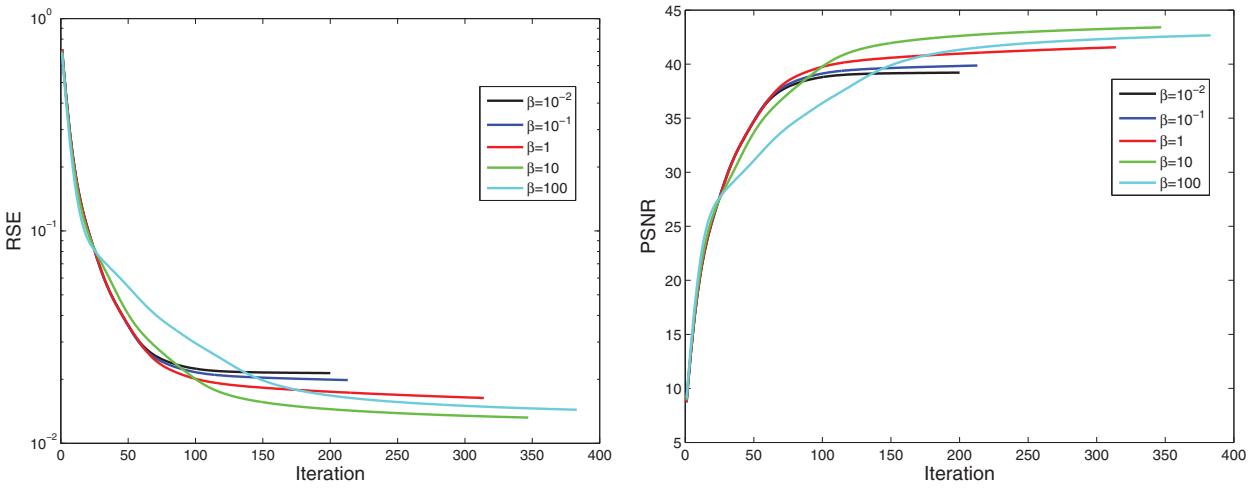

Fig. 13. The RSE values and the PSNR values with respect to iterations for different penalty parameters β.

Table 3
The restoration results for the proposed model with different iteration number for solving X_3.

Iteration number	3	6	9	12	15	18	21
PSNR	59.30	59.35	59.36	59.38	59.38	59.38	59.38
Time	4.90	5.26	5.87	6.16	6.28	6.36	6.64

On the other hand, we study the sensitivity of the number of iterations to be set for computing X_3. By using the synthetic data with 50% sampling rate, we report in Table 3 the PSNR values and the CPU-times for different numbers of iterations for computing X_3. It is obvious that when the number of iterations is higher, the PSNR value is slightly better, but the overall computational time increases. According to the results, it is sufficient to set the number of iterations to be three for computing X_3.

4. Conclusions

In this paper, we presented a model combining TV and low-rank matrix factorization to deal with tensor completion. An efficient algorithm has been developed to solve the proposed model. We demonstrate that our numerical scheme converges to the coordinatewise minimizers. Experimental results have shown that the proposed method is more effective and robust as compared with TMac.

Acknowledgments

The authors would like to express their great thankfulness to Dr. Y. Xu for sharing the codes of the TMac algorithm. The authors would like to express their sincere thanks to the editor and referees for giving thus so many valuable comments and suggestions for revising this paper. This research is supported by 973 Program (2013CB329404), NSFC (61370147, 61170311, 61402082), the Fundamental Research Funds for the Central Universities (ZYGX2013Z005, ZYGX2013J106).

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, Image inpainting, in: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 2000, pp. 417–424. SIGGRAPH '00.
[2] J.M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, J. Chanussot, Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches, IEEE J. Selected Top. Appl. Earth Observ. Rem. Sens. 5 (2) (April 2012) 354–379.
[3] J.-F. Cai, E. Candès, Z.-W. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. 20 (4) (2010) 1956–1982.
[4] E.J. Candès, B. Recht, Exact matrix completion via convex optimization, Found. Comput. Math. 9 (6) (2009) 717–772.
[5] E.J. Candès, T. Tao, The power of convex relaxation: near-optimal matrix completion, IEEE Trans. Inf. Theory 56 (5) (May 2010) 2053–2080.
[6] S.H. Chan, R. Khoshabeh, K.B. Gibson, P.E. Gill, T.Q. Nguyen, An augmented Lagrangian method for total variation video restoration, IEEE Trans. Image Process. 20 (11) (Nov 2011) 3097–3111.
[7] T. Chan, L. Vese, Active contours without edges, IEEE Trans. Image process. 10 (2) (2001) 266–277.
[8] M. Fazel, H. Hindi, S.P. Boyd, A rank minimization heuristic with application to minimum order system approximation, in: Proceedings of the American Control Conference, vol 6, pp. 4734–4739, 2001.
[9] S. Gandy, B. Recht, I. Yamada, Tensor completion and low-n-rank tensor recovery via convex optimization, Inverse Probl. 27 (2) (2011) 025010.
[10] R. Glowinski, Numerical Methods for Nonlinear Variational Problems, Springer-Verlag, New York, 2008.
[11] J. Huang, T.-Z. Huang, X.-L. Zhao, Z.-B. Xu, X.-G. Lv, Two soft-thresholding based iterative algorithms for image deblurring, Inf. Sci. 271 (0) (2014) 179–195.
[12] C.J. Hillar, L.H. Lim, Most tensor problems are np-hard, J. ACM 60 (6) (November 2013) 45:1–45:39.

[13] M.D. Iordache, J.M. Bioucas-Dias, A. Plaza, Sparse unmixing of hyperspectral data, IEEE Trans. Geosci. Rem. Sens. 49 (6) (June 2011) 2014–2039.
[14] M.D. Iordache, J.M. Bioucas-Dias, A. Plaza, Total variation spatial regularization for sparse hyperspectral unmixing, IEEE Trans. Geosci. Rem. Sens. 50 (11) (Nov 2012) 4484–4502.
[15] T.G. Kolda, B.W. Bader, Tensor decompositions and applications, SIAM Rev. 51 (3) (2009) 455–500.
[16] T.G. Kolda, B.W. Bader, J.P. Kenny, Higher-order web link analysis using multilinear algebra, in: Proceedings of the Fifth IEEE International Conference on Data Mining, November 2005, pp. 242–249.
[17] N. Komodakis, Image completion using global optimization, in: Proceedings of the Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, volume 1, June 2006, pp. 442–452.
[18] T. Korah, C. Rasmussen, Spatiotemporal inpainting for recovering texture maps of occluded building facades, IEEE Trans. Image Process. 16 (9) (September 2007) 2262–2271.
[19] N. Kreimer, M. Sacchi, A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation, GEOPHYSICS 77 (3) (2012) V113–V122.
[20] M. Kurucz, A. Benczúr, K. Csalogány, Methods for large scale SVD with missing values, in: Proceedings of the KDD Cup and Workshop, vol 12, Citeseer, 2007, pp. 31–38.
[21] J. Liu, T.-Z. Huang, I.W. Selesnick, X.-G. Lv, P.-Y. Chen, Image restoration using total variation with overlapping group sparsity, Inf. Sci. 295 (0) (2015) 232–246.
[22] F. Li, M.K. Ng, R.J. Plemmons, Coupled segmentation and denoising/deblurring models for hyperspectral material identification, Numer. Linear Algebra Appl. 19 (1) (2012) 153–173.
[23] N. Li, B.-X. Li, Tensor completion for on-board compression of hyperspectral images, in: Proceedings of the 17th IEEE International Conference on Image Processing (ICIP), September 2010, pp. 517–520.
[24] 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 (1) (January 2013) 208–220.
[25] Y. Liu, F. Shang, An efficient matrix factorization method for tensor completion, IEEE Signal Process. Lett. 20 (4) (2013) 307–310.
[26] S.-Q. Ma, D. Goldfarb, L.-F. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, Math. Program. 128 (1–2) (2011) 321–353.
[27] M. Razaviyayn, M. Hong, Z. Luo, A unified convergence analysis of block successive minimization methods for nonsmooth optimization, SIAM J. Optim. 23 (2) (2013) 1126–1153.
[28] B. Recht, M. Fazel, P. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization, SIAM Rev. 52 (3) (2010) 471–501.
[29] L.I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D: Nonlinear Phenom. 60 (3) (1992) 259–268.
[30] A.C. Sauve, A.O. Hero, W.L. Rogers, S.J. Wilderman, N.H. Clinthorne, 3D image reconstruction for a compton SPECT camera model, IEEE Trans. Nucl. Sci. 46 (6) (Dec 1999) 2075–2084.
[31] J.-T. Sun, H.-J. Zeng, H. Liu, Y.-C. Lu, Z. Chen, CubeSVD: a novel approach to personalized web search, Proceedings of the 14th International Conference on World Wide Web, ACM, New York, NY, USA, 2005, pp. 382–390. WWW '05
[32] J. Sun, T. D, C. Faloutsos, Beyond streams and graphs: dynamic tensor analysis, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 374–383.
[33] J. Sun, D. Tao, S. Papadimitriou, P.S. Yu, C. Faloutsos, Incremental tensor analysis: theory and applications, ACM Trans. Knowl. Discov. Data (TKDD) 2 (3) (2008) 11.
[34] D. Tao, X. Li, W. Hu, S. Maybank, X. Wu, Supervised tensor learning, in: Proceedings of the Fifth IEEE International Conference on Data Mining, November 2005, p. 8.
[35] D. Tao, X. Li, X. Wu, S.J. Maybank, General tensor discriminant analysis and Gabor features for gait recognition, IEEE Trans. Pattern Anal. Mach. Intell. 29 (10) (October 2007) 1700–1715.
[36] D. Tao, J. Sun, J. Shen, X. Wu, X. Li, S.J. Maybank, C. Faloutsos, Bayesian tensor analysis, in: Proceedings of the IEEE International Joint Conference on Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence)., June 2008, pp. 1402–1409.
[37] K.C. Toh, S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems, Pac. J. Optim. 6 (615–640) (2010) 15.
[38] V.N. Varghees, M.S. Manikandan, 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), March 2012, pp. 506–511.
[39] B. Wang, X.-B. Gao, D.-C. Tao, X.-L. Li, A unified tensor level set for image segmentation, IEEE Trans. Syst. Man Cybern. Part B: Cybern. 40 (3) (2010) 857–867.
[40] Z.-W. Wen, W.-T. Yin, Y. Zhang, Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm, Math. Program. Comput. 4 (4) (2012) 333–361.
[41] Z. Xing, M. Zhou, A. Castrodad, G. Sapiro, L. Carin, Dictionary learning for noisy and incomplete hyperspectral images, SIAM J. Imaging Sci. 5 (1) (2012) 33–56.
[42] Y.-Y. Xu, R.-R. Hao, W.-T. Yin, Z.-X. Su, Parallel matrix factorization for low-rank tensor completion, Inverse Probl. Imaging 9 (2) (2015) 601–624.
[43] Y.-Y. Xu, W.-T. Yin, Z.-W. Wen, Y. Zhang, An alternating direction algorithm for matrix completion with nonnegative factors, Front. Math. Ch. 7 (2) (2012) 365–384.
[44] L. Yang, Z.-H. Huang, X.-J. Shi, A fixed point iterative method for low n-rank tensor pursuit, IEEE Trans. Signal Process. 61 (11) (June 2013) 2952–2962.
[45] X.-L. Zhao, F. Wang, T.-Z. Huang, M. Ng, R. Plemmons, Deblurring and sparse unmixing for hyperspectral images, IEEE Trans. Geosci. Rem. Sens. 51 (7–1) (2013a) 4045–4058.
[46] X.-L. Zhao, W. Wang, T.-Y. Zeng, T.-Z. Huang, M.K. Ng, Total variation structured total least squares method for image restoration, SIAM J. Sci. Comput. 35 (6) (2013b) B1304–B1320.