Information Sciences 436–437 (2018) 403–417

Contents lists available at ScienceDirect

Information Sciences

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

# Matrix factorization for low-rank tensor completion using framelet prior

Tai-Xiang Jiang, Ting-Zhu Huang*, Xi-Le Zhao*, Teng-Yu Ji, Liang-Jian Deng

*School of Mathematical Sciences/Institute of Computational Science, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, PR China*

ARTICLE INFO

*Article history:*
Received 30 November 2016
Revised 12 January 2018
Accepted 15 January 2018
Available online 31 January 2018

*Keywords:*
Tensor completion
Framelet
Low-rank matrix factorization
Block successive upper-bound Minimization

ABSTRACT

In this paper, we propose a novel tensor completion model using framelet regularization and low-rank matrix factorization. An effective block successive upper-bound minimization (BSUM) algorithm is designed to solve the proposed optimization model. The convergence of our algorithm is theoretically guaranteed, and under some mild conditions, our algorithm converges to the coordinate-wise minimizers. Extensive experiments are conducted on the synthetic data and real data, and the results demonstrate the effectiveness and the efficiency of the proposed method.



## 1. Introduction

The tensor is a multidimensional array which is an important data format. Higher-order low-rank tensors play an increasingly significant role in a wide range of real-world applications [1,4,5,17–26,37,39,44,49]. Among them, many can be formulated as a missing value estimation problem (namely completion problem), e.g., color image inpainting [1,22], MRI data recovery [39], video inpainting [5,23], video rain streak removal[17], high-order web link analysis [21], personalized web search [37], seismic data reconstruction [24] and hyperspectral data recovery [25,26,44,49]. How to characterize the inner relationship between the known elements and unknown ones is the core problem [28].

The low-rank tensor completion problem can be regarded as an extension of the low-rank matrix completion (LRMC) problem. There are lots of works have been done in LRMC, e.g. [30,31], which aim at exactly recovering a low-rank matrix from an incomplete observation. These matrix completion methods minimize the matrix rank and their mathematical model can be uniformly summarized as

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

where $\mathbf{Y} \in \mathbb{R}^{I_1 \times I_2}$ is the underlying matrix, $\mathbf{F}$ is the observed data, the set $\Omega$ implies the location corresponding to the observed entries, $\mathcal{P}_{\Omega}(\cdot)$ is the projection operator extracting entries in $\Omega$ (see details of the projection operator in Section 2.3). However, as directly optimizing problem (1) is NP-hard, many methods instead minimize the nuclear norm or low-rank matrix factorization to efficiently enhance the low-rankness of the underlying results. In addition, once the nuclear norm or

* Corresponding authors.
*E-mail addresses:* taixiangjiang@gmail.com (T.-X. Jiang), tingzhuhuang@126.com (T.-Z. Huang), xlzhao122003@163.com (X.-L. Zhao), tengyu_j66@126.com (T.-Y. Ji), liangjiand1987112@126.com (L.-J. Deng).

https://doi.org/10.1016/j.ins.2018.01.035
0020-0255/

the low-rank matrix factorization is selected, this minimization problem can be efficiently solved by fixed point continuation with approximate singular value decomposition (FPCA) [29], accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems (APGL) [38], low-rank matrix fitting algorithm (LMaFit) [41], and alternating direction multipliers method (ADMM) [36,46].

Comparing with the matrix format, a tensor usually contains more essentially structural information. For instance, a video with multi-frames can represent the correlation along the time [42]; a hyperspectral image (HSI), consisting of multiple images of a real scene captured by sensors over various discrete bands, is able to facilitate a fine delivery of more faithful knowledge under real scenes [43]. Although these data can be analyzed by matrix methods after unfolding or flattening, such matricization usually fails to exploit the essential tensor structure.

It seems natural to directly extend LRMC methods to the low-rank tensor completion problem (LRTC) as

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

where $\mathcal{Y} \in \mathbb{R}^{I_1,\ldots,I_N}$ is the underlying tensor, $\mathcal{F}$ is the observed data, and $\mathcal{P}_\Omega(\cdot)$ is the projection operator. Unfortunately, when it comes to the rank of a tensor, the numerical algebra of tensors is fraught with hardness results [12]. On the one hand, there is not an exact (or unique) definition for tensor rank and the most popular rank definitions are CANDECOMP/PARAFAC rank (CP-rank) and $n$-rank [20]. On the other hand, both of the minimization problems corresponding to CP-rank and $n$-rank are NP-hard. To tackle this difficulty, Liu et al. [28] proposed the first definition of the nuclear norm of the tensor, extending the matrix case to the tensor case,

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

where $\|\mathcal{Y}\|_* := \sum_{n=1}^{N} \alpha_n \|\mathbf{Y}_{(n)}\|_*$, $\mathbf{Y}_{(n)} \in \mathbb{R}^{I_n \times \prod_{m\neq n} I_m}$ is the mode-$n$ unfolding of a tensor $\mathcal{Y}$ and $\alpha_n$, $n = 1,\ldots,N$, is positive weight satisfying $\sum_{n=1}^{N} \alpha_n = 1$. It can be efficiently solved by fast low-rank tensor completion (FaLRTC) [28] and the Douglas–Rachford splitting method [11]. Based on this definition, Ji et al. further proposed a non-convex apporach [15]. However, these methods involved the singular value decomposition (SVD) of $\mathbf{Y}_{(n)}$, which is time-consuming. To cope with this issue, Xu et al. [45] adopted low-rank matrix factorization,

$$\begin{aligned} \min_{\mathcal{Y},\mathbf{X},\mathbf{A}} \quad & \sum_{n=1}^{N} \frac{\alpha_n}{2} \|\mathbf{Y}_{(n)} - \mathbf{A}_n\mathbf{X}_n\|_F^2 \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}, \end{aligned} \tag{4}$$

where $\mathbf{A} = (\mathbf{A}_1,\ldots,\mathbf{A}_N)$ and $\mathbf{X} = (\mathbf{X}_1,\ldots,\mathbf{X}_N)$. Their method, Low-rank **T**ensor Completion by Parallel **M**atrix F**ac**torization (TMac), obtained better results with less running time than FaLRTC [45].

In general, introducing additional information, or say prior knowledge, may be helpful for the tensor completion problem which is an ill-posed inverse problem [8,48]. In addition to the afore-discussed low-rankness, the piece-wise smoothness can be taken into consideration to further enhance the potential capacity of the LRTC methods. For instance, a linear hyperspectral mixing model can be written as

$$\mathbf{Y} = \mathbf{W}\mathbf{H} \quad (\text{or} \quad \mathbf{Y}_{(3)} = \mathbf{A}_3\mathbf{X}_3), \tag{5}$$

where $\mathbf{Y}$ (or $\mathbf{Y}_{(3)}$) indicates the mode-3 unfolding of the hyperspectral image, and $\mathbf{W}$ (or $\mathbf{A}_3$) and $\mathbf{H}$ (or $\mathbf{X}_3$) are respectively the *standard spectral library* and the *abundance matrix* in hyperspectral unmixing [13,14,47]. As the matter distribution is locally continuous in nature, the *abundance maps* possess the piecewise smooth prior. Enlightened by this, Zhao et al. [49] successfully introduced the Total Variation (TV) regularization into sparse hyperspectral unmixing. To better solve the ill-conditioned problem (4), Ji et al. [16] further generalized [49] by adding a TV regularization to $\mathbf{X}_3$ as follows

$$\begin{aligned} \min_{\mathcal{Y},\mathbf{X},\mathbf{A}} \quad & \sum_{n=1}^{N} \frac{\alpha_n}{2} \|\mathbf{Y}_{(n)} - \mathbf{A}_n\mathbf{X}_n\|_F^2 + \mu \mathrm{TV}(\mathbf{X}_3) \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}. \end{aligned} \tag{6}$$

However, Ji et al. did not discuss the reason of the outperformance of their model further in [16]. To comprehend their model more intuitively, we rewrite the mixing model (5) in a tensor format

$$\mathcal{Y} = \mathcal{X}_3 \times_3 \mathbf{A}_3, \tag{7}$$

where $\times_3$ indicates the mode-3 (matrix) multiplication (see details in Section 2.1), $\mathrm{unfold}_3(\mathcal{Y}) = \mathbf{Y} = \mathbf{Y}_{(3)}$ in (5), $\mathrm{unfold}_3(\mathcal{X}_3) = \mathbf{H} = \mathbf{X}_3$ in (5) and each horizontal slice of $\mathcal{X}_3$ is the *fractional abundances map* of one kind of end-member in hyperspectral unmixing. It's easy to prove that (7) is mathematically equivalent to (5). Furthermore, in (7) every horizontal slice of $\mathcal{Y}$, which is an image (a frame of a video or a band of a hyperspectral image), is a linear combination of horizontal slices of $\mathcal{X}_3$. So that, it's easy to understand that the horizontal slices of $\mathcal{X}_3$ are the underlying factor images [50], and maintain the priors which the original images possess. Figuring out this, the potential capacity of the model (6) thus still has room to be further enhanced.

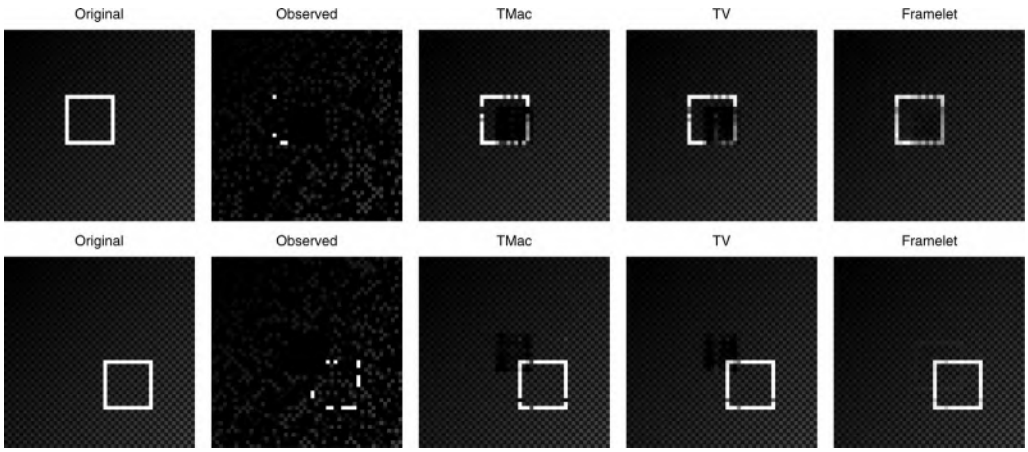


**Fig. 1.** The results recovered by TMac [45], TV based method [16], and the proposed method. Each row represents a slice of the recovered synthetic data respectively. From left to right: the original data, the observed data (80% entries and a $10 \times 10 \times 50$ cube in the center missing), the recovered result by TMac, TV based method and the proposed method, respectively.

It is well known that the TV regularizer causes the staircase artifacts [7,10,27,33,51], so that the details and geometric features of the factor images will not be well preserved in (6). However, the details and geometric features are important in real-world applications. Meanwhile, the framelet is a generation of the orthogonal basis which relaxes the requirements of the orthogonality and linear independence. Its redundancy leads to a capability of preserving geometric features and details [3]. Motivated by this, we introduce the framelet regularization into the tensor completion problem,

$$\begin{aligned} \min_{\mathcal{Y},\mathbf{X},\mathbf{A}} \quad & \sum_{n=1}^{N} \frac{\alpha_n}{2} \|\mathbf{Y}_{(n)} - \mathbf{A}_n\mathbf{X}_n\|_F^2 + \lambda \|\mathbf{W}\mathbf{X}_3^{\mathrm{T}}\|_{1,1} \\ \text{s.t.} \quad & \mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}, \end{aligned} \tag{8}$$

where $\lambda$ is a positive regularization parameter, $\mathbf{W}$ indicates the framelet transform matrix, and $\mathbf{W}\mathbf{X}_3^{\mathrm{T}}$ is the coefficients of the factor images in the framelet transform domain. The difference between our model and the model of Xu et al. (TMac) [45] is our consideration of the piece-wise smooth prior of the factor matrix. Then, the difference between our model and the model of Ji et al. (TV based method) [16] lies on not only the superior of the framelet regularizer to the total variation regularizer but also a deeper insight of the factor prior.

In Fig. 1, we display the recovered results for synthetic data by TMac [45], TV based method [16] and our framelet based method. It is clear from the Fig. 1 that the estimated results by our framelet based method are visually the best among those by TMac and TV based method.

Apart from the superiority of framelet over TV, the outstanding performance of our method also lies in the following two aspects. On the one hand, if traditional regularization techniques are directly applied to the original image squeeze and the correlation between different frames or bands would be unavoidably destroyed. We regularize the coefficients of the factor matrix in the framelet domain so that the piece-wise smoothness of the factor matrix is boosted while the correlation between different frames of bands is simultaneously well preserved. Thus, our model is more robust in contrast with the traditional variational techniques. On the other hand, in our model, the low-rank factorization fidelity term is used to capture the global information and the framelet regularization term is used to capture the local information. Therefore, our method, utilizing both the local and global information, is reasonably expected to yield better results.

The main contribution of this paper mainly consists of two folds. First, on the foundation of a better comprehension of the factor prior, we propose a new tensor completion model, utilizing both the global and local information. Second, an effective and efficient algorithm is proposed to efficiently solve our model. Our algorithm fits the framework of the BSUM [34]. Thus, the convergence of the proposed algorithms is theoretically guaranteed. Moreover, numerical experiments are conducted and the experimental results demonstrate the effectiveness and robustness of the proposed method.

The outline of this paper is given as follows. Some preliminary knowledge of the tensor, framelet and some operators is exhibited in Section 2. Section 3, gives the formulation of our model as well as the BSUM-based solver. Experimental results are reported In Section 4. Finally, we draw some conclusions in Section 5.

**Table 1**
Tensor Notations.

| Notations | Explanations |
|---|---|
| $\mathcal{X}, \mathbf{X}, \mathbf{x}, x$ | tensor, matrix, vector, scalar. |
| $\mathbf{x}(:\, i_2 i_3 \cdots i_N)$ | **fiber** of tensor $\mathcal{X}$ defined by fixing every index but one. |
| $\mathbf{X}(::\, i_3 \cdots i_N)$ | **slice** of a tensor defined by fixing all but two indices. |
| $\langle \mathcal{X}, \mathcal{Y} \rangle$ | **inner product** of two same-sized tensors $\mathcal{X}$ and $\mathcal{Y}$. |
| $\|\mathcal{X}\|_F$ | **Frobenius** norm of tensor $\mathcal{X}$. |
| $\mathbf{X}_{(n)}$, $\text{unfold}_n(\mathcal{X})$ | **mode-n unfolding** of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ denoted as $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times \prod_{i \neq n} I_i}$. |
| $(r_1, r_2, \cdots, r_N)$ | **n-rank**, where $r_n = \text{rank}(\mathbf{X}_{(n)})$, $n = 1, 2, \cdots, N$. |
| $\mathcal{Y} = (\mathcal{X} \times_n \mathbf{A})$ | **n-mode (matrix) multiplication** of a tensor $\mathcal{X} \in \mathbb{R}^{I_1, I_2, \cdots, I_n, \cdots, I_N}$ with a matrix $A \in \mathbb{R}^{J \times I_n}$. |

## 2. Preliminary

### 2.1. Tensor basics

Following [20], 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 $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, and $x_{i_1 i_2 \cdots i_N}$ is its $(i_1, i_2, \cdots, i_N)$-th component.

**Fibers** are defined by fixing every index but one. Third-order tensors have column, row, and tube fibers, denoted by $\mathbf{x}_{:\, jk}$, $\mathbf{x}_{i:\, k}$, and $\mathbf{x}_{ij:}$, respectively. When extracted from the tensor, fibers are always assumed to be oriented as column vectors.

**Slices** are two-dimensional sections of a tensor, defined by fixing all but two indices. The horizontal, lateral, and frontal slides of a third-order tensor $\mathcal{X}$, denoted by $\mathbf{X}_{i::}$, $\mathbf{X}_{:\, j:}$, and $\mathbf{X}_{::\, k}$, respectively. Alternatively, the $k$th frontal slice of a third-order tensor, $\mathbf{X}_{::\, k}$, may be denoted more compactly as $\mathbf{X}_k$.

The **inner product** of two same-sized tensors $\mathcal{X}$ and $\mathcal{Y}$ is defined as

$$\langle \mathcal{X}, \mathcal{Y} \rangle := \sum_{i_1, i_2, \cdots, i_N} x_{i_1 i_2 \cdots i_N} \cdot y_{i_1 i_2 \cdots i_N}.$$

The corresponding norm (**Frobenius norm**) is then defined as $\|\mathcal{X}\|_F := \sqrt{\langle \mathcal{X}, \mathcal{X} \rangle}$.

The **mode-n unfolding** of a tensor $\mathcal{X}$ is denoted as $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times \prod_{i \neq n} I_i}$, where the tensor element $(i_1, i_2, \cdots, i_N)$ 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., $\mathcal{X} = \text{fold}_n(\mathbf{X}_{(n)})$.

The **n-rank**, which we adopt in our work, is defined as an array

$$n\text{-rank}(\mathcal{X}) = (\text{rank}(\mathbf{X}_{(1)}), \text{rank}(\mathbf{X}_{(2)}), \cdots, \text{rank}(\mathbf{X}_{(N)})).$$

The tensor $\mathcal{X}$ is low-rank, if $\mathbf{X}_{(n)}$ is low-rank for all $n$.

The **n-mode (matrix) product** of a tensor $\mathcal{X} \in \mathbb{R}^{I_1, I_2, \cdots, I_n, \cdots, I_N}$ with a matrix $\mathbf{A} \in \mathbb{R}^{J \times I_n}$ is denoted by $\mathcal{X} \times_n \mathbf{A}$ and is of size $I_1 \times I_2 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$. Elementwise, we have

$$(\mathcal{X} \times_n \mathbf{A})_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n = 1}^{I_n} x_{i_1 i_2 \cdots i_n \cdots i_N} \cdot a_{j i_n}. \tag{10}$$

Each mode-$n$ fiber is multiplied by the matrix **A**. This idea can also be expressed in terms of unfolded tensors

$$\mathcal{Y} = (\mathcal{X} \times_n \mathbf{A}) \quad \Leftrightarrow \quad \mathbf{Y}_{(n)} = \mathbf{A} \cdot \text{unfold}_n(\mathcal{X}).$$

The notations are listed in Table 1. Please refer to [20] for a more extensive overview.

### 2.2. Framelet

A countable set $X \subset L_2(\mathbb{R})$ is called a tight frame of $L_2(\mathbb{R})$ if

$$f = \sum_{g \in X} \langle f, g \rangle g, \quad \forall f \in L_2(\mathbb{R}).$$

This is equivalent to

$$\|f\|^2_{L_2(\mathbb{R})} = \sum_{g \in X} |\langle f, g \rangle|^2, \quad \forall f \in L_2(\mathbb{R}),$$

where $\langle\cdot,\ \cdot\rangle$ is the inner product in $L^2(\mathbb{R})$, and $\|\cdot\|_{L^2(\mathbb{R})} = \langle\cdot,\cdot\rangle^{\frac{1}{2}}$. Then $X$ is called a tight frame of $L^2(\mathbb{R})$. For given $\Psi := \{\psi_1, \psi_2, \cdots, \psi_r\} \subset L^2(\mathbb{R})$, the affine (or wavelet) system is defined by the collection of the dilations and the shifts of $\Psi$ as

$$X(\Psi) := \{\psi_{l,j,k} : 1 \le l \le r;\ j,k \in \mathbb{Z}\},$$

where $\psi_{l,j,k} := 2^{j/2}\psi_l(2^j\cdot - k)$. When $X(\Psi)$ forms a tight frame of $L^2(\mathbb{R})$, it is called a tight wavelet frame, and $\psi_l, l = 1, 2, \cdots, r$ are called the (tight) framelets.

To construct wavelet tight frames, one starts from a compactly supported refinable function (a scaling function) $\psi \in L_2(\mathbb{R})$ with a refinement mask (low-pass filter) $\xi_0 \in L_2(\mathbb{Z})$ satisfying a refinement equation

$$\psi(x) = \sum_{l\in\mathbb{Z}} \xi_0(l)\psi(2x - l).$$

Then, for the given compactly supported refinable function, a tight framelet system can be constructed by finding an appropriate set of framelets $\Psi = \{\phi_1, \phi_2, \cdots, \phi_r\} \subset L^2(\mathbb{R})$. Let $\{xi_1, \xi_2, \cdots, \xi_r\} \subset L^2(\mathbb{Z})$ be a set of framelet masks (high-pass filters), then the framelets are defined as

$$\phi_j = \sum_{l\in\mathbb{Z}} \xi_j(l)\psi(2x - l), \quad j = 1, 2, \cdots, r.$$

Thus, the construction of framelet $\Psi$ amounts to design the filters $\xi_1$, $\xi_2$, $\cdots$, $\xi_r$.

The unitary extension principle (UEP) in [35] gives the condition for $X(\Psi)$ to form as a tight frame system, i.e., the filers $\xi_1$, $\xi_2$, $\cdots$, $\xi_r$ satisfy

$$\zeta_{\xi_0}(\omega)\overline{\zeta_{\xi_0}(\omega)(\omega + \gamma\pi)} + \sum_{j=1}^{r} \zeta_{\xi_j}(\omega)\overline{\zeta_{\xi_j}(\omega)(\omega + \gamma\pi)} = \sigma(\gamma), \quad \gamma = 0, 1,$$

for almost all $\omega \in \mathbb{R}$. Here, $\zeta_\xi(\omega) = \sum_l \xi(l)e^{il\omega}$ and $\sigma(\gamma)$ is a delta function.

In particular, one can construct tight framelet systems from B-splines. Here, we give one example which will be used in our numerical experiments. It is derived from piece-wise linear B-spline whose refinement mask is $h_0 = \frac{1}{4}[1, 2, 1]$. The two corresponding high-pass filters are

$$h_1 = \frac{\sqrt{2}}{4}[1, 0, -1], \quad h_2 = \frac{1}{4}[-1, 2, -1].$$

In the numerical scheme of image processing, the framelet transform (decomposition operator) can be represented by a matrix $\mathbf{W}$. The processes of generating such matrices have been detailed in many literatures such as [2]. We omit them here for readability.

For a given filter $h = \{h(j)\}_{j=-J}^{J}$, let the matrix $\mathbf{S}(h)$ be convolution operator with filter $h$ under Neumann (symmetric) boundary condition

$$\mathbf{S}(h) = \begin{bmatrix} h(0) & \cdots & h(-J) & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h(J) & \ddots & \ddots & \ddots & h(-J) \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & h(J) & \cdots & h(0) \end{bmatrix} + \begin{bmatrix} h(1) & h(2) & \cdots & h(J) & 0 \\ h(2) & \ddots & \ddots & \ddots & h(-j) \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h(J) & \ddots & \ddots & \ddots & h(-2) \\ 0 & h(-J) & \cdots & h(-2) & h(-1) \end{bmatrix}.$$

In our numerical experiments in Section 4, we use a multi-level tight frame system corresponding to the tight framelet decomposition without down sampling. To introduce it, we recall that for a given filter $h = \{h(j)\}_{j=-J}^{J}$, the filters $h^{(l)}$ at level $l$ corresponding to the decomposition without down sampling is

$$\begin{aligned} h_{(l)} =& \{h(-J), \underbrace{0, \cdots, 0}_{2^{l-1}-1}, h(-J+1), 0, \cdots, 0, h(-1), \underbrace{0, \cdots, 0}_{2^{l-1}-1}, h(0), \\ & \underbrace{0, \cdots, 0}_{2^{l-1}-1}, h(1), 0, \cdots, 0, h(j-1), \underbrace{0, \cdots, 0}_{2^{l-1}-1}, h(j)\}. \end{aligned}$$

Given the low- and high-pass filter $\{h_i\}_{i=0}^{r}$, let $\mathbf{H}_i^{(l)} \equiv \mathbf{S}(h_i^{(l)})$. Then the multi-level decomposition operator up to level $L$ (without down sampling) is given by

$$\mathbf{W} = \begin{bmatrix} \prod_{l=0}^{L-1} \mathbf{H}_0^{(L-l)} \\ H_1^{(L)} \prod_{l=1}^{L-1} \mathbf{H}_0^{(L-l)} \\ \vdots \\ \mathbf{H}_r^{(L)} \prod_{l=1}^{L-1} \mathbf{H}_0^{(L-l)} \\ \vdots \\ \mathbf{H}_1^{(1)} \\ \vdots \\ \mathbf{H}_r^{(1)} \end{bmatrix}.$$

Then the decomposition of a discrete image (vector form) $\boldsymbol{f} \in \mathbb{R}^{mn}$, can be written as $\mathbf{W}\boldsymbol{f}$. Besides, the UEP asserts that $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, where $\mathbf{W}^{\mathrm{T}}$ is the inverse framelet transform.

### 2.3. Operators

The **Proximal Operator** of a given convex function $f(x)$ is defined as

$$\mathrm{prox}_f(y) := \arg\min_x f(x) + \frac{\rho}{2}\|x - y\|^2, \tag{11}$$

where $\rho$ 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 $\mathrm{prox}_f(x^k)$, where $x^k$ is the latest update of $x$. The proximal operator is very attractive in that the objective function (11) is strongly convex with respect to $x$ so long as $f(x)$ is convex.

Let $\Omega$ be an index set, then the **Projection Operator** $\mathcal{P}_\Omega(\mathcal{Y})$ denotes the tensor copying the entries from $\mathcal{Y}$ in the set $\Omega$ and letting the remaining entries be zeros, i.e.,

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

The matrix nonnegative **Soft-thresholding Operator** $\mathcal{S}_v(\cdot)$ is defined as

$$\mathcal{S}_v(\mathbf{X}) = \bar{\mathbf{X}} \text{ with } \bar{x}_{i,j} = \begin{cases} x_{i,j} - v, & x_{i,j} > v, \\ 0, & \text{otherwise}. \end{cases}$$

## 3. Proposed model and algorithm

### 3.1. Proposed model

The objective function of our model is:

$$f(\mathbf{X}, \mathbf{A}, \mathcal{Y}) = \sum_{n=1}^{N} \frac{\alpha_n}{2}\|\mathbf{Y}_{(n)} - \mathbf{A}_n\mathbf{X}_n\|_F^2 + \lambda\|\mathbf{W}\mathbf{X}_3^{\mathrm{T}}\|_{1,1}. \tag{12}$$

There are two significantly important parts of the proposed model. One is the low-rank matrix factorization fidelity term, $\sum_{n=1}^{N}\frac{\alpha_n}{2}\|\mathbf{Y}_{(n)} - \mathbf{A}_n\mathbf{X}_n\|_F^2$. Suppose the $n$-rank of $\mathcal{Y}$ is a given prior $(r_1, r_2, r_3)$ and $r_n < I_n$ $(n = 1, 2, 3)$, thus, $\mathbf{A}_n \in \mathbb{R}^{I_n \times r_n}$ and $\mathbf{X}_n \in \mathbb{R}^{r_n \times \prod_{m\neq n} I_m}$ is the low-rank matrix factorization of $\mathbf{Y}_{(n)}$ $(n = 1, 2, 3)$. This fidelity term is to enhance the low-rankness of the underlying tensor $\mathcal{Y}$ in each mode. Furthermore, this low-rankness term is to capture the global information, since sometimes the values of the missing elements depend on the entries which are far away [28].

The other one is the regularization term, $\|\mathbf{W}\mathbf{X}_3^{\mathrm{T}}\|_{1,1}$, derived from the factor framelet prior. Specifically, the $i$-th row of $\mathbf{X}_3$ is the $i$-th vectorized factor image, which maintains the piece-wise smoothness. As discussed in Section 1, the framelet regularization is selected to both enhance this property and preserve geometric features and details.

According to the rule of unfolding, the factor images $\mathbf{X}_3$ has a good structure,

$$\mathbf{X}_3 = \left[\mathbf{x}_{31}, \mathbf{x}_{32}, \cdots, \mathbf{x}_{3i}, \cdots, \mathbf{x}_{3r_n}\right]^{\mathrm{T}} \in \mathbb{R}^{r_n \times I_1 I_2},$$

where $\mathbf{x}_{3i} \in \mathbb{R}^{I_1 I_2}$ $(i = 1, 2, \cdots, r_n)$ is the $i$-th vectorized factor image. As mentioned in Section 2.2, for a vectorized discrete image $\boldsymbol{f} \in \mathbb{R}^{mn}$, its coefficients in the framelet domain is calculated by $\mathbf{W}\boldsymbol{f}$. Thus, the framelet decomposition of the $i$th factor image is $\mathbf{W}\mathbf{x}_{3i}$, and we have $\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{x}_{3i} = \mathbf{x}_{3i}$. Moreover, benefit from the good structure of $\mathbf{X}_3$, the coefficients of all the factor images can be concisely calculated by $\mathbf{W}\mathbf{X}_3^{\mathrm{T}}$, and it is not difficult to obtain that $\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{X}_3^{\mathrm{T}} = \mathbf{X}_3^{\mathrm{T}}$, from the unitary extension principle [35]. Additionally, this regularization term is to utilize the local information.

In a brief summary, both the global information and local information, i.e., low-rankness and piece-wise smoothness, are taken into consideration in the proposed model. The low-rank matrix factorization and framelet regularization would respectively be proved efficient to boost these two priors, i.e. low-rankness and piecewise smoothness, in our numerical experiments. In the next subsection, we would present the numerical scheme for solving the proposed model.

### 3.2. Proposed algorithm

In this subsection, we present the numerical scheme for solving the minimization problem in (8). It is easy to obtain that the objective function (12) is not jointly convex for $(\mathbf{X}, \mathbf{A}, \mathcal{Y})$, but is convex with respect to **X, A**, $\mathcal{Y}$ independently. In order to solve the non-convex problem effectively, we adopt the BSUM method.

Utilizing the proximal operator, we perform the update as:

$$h(\mathcal{Z}, \mathcal{Z}^k) = f(\mathcal{Z}) + \frac{\rho}{2}\|\mathcal{Z} - \mathcal{Z}^k\|_F^2, \tag{13}$$

where $\mathcal{Z} = (\mathbf{X}, \mathbf{A}, \mathcal{Y})$ and $\mathcal{Z}^k = (\mathbf{X}^k, \mathbf{A}^k, \mathcal{Y}^k)$. Let

$$\begin{cases} h_1(\mathbf{X}, \mathcal{Z}_1^k) = f(\mathbf{X}, \mathbf{A}^k, \mathcal{Y}^k) + \frac{\rho}{2}\|\mathbf{X} - \mathbf{X}^k\|_F^2, \\ h_2(\mathbf{A}, \mathcal{Z}_2^k) = f(\mathbf{X}^{k+1}, \mathbf{A}, \mathcal{Y}^k) + \frac{\rho}{2}\|\mathbf{A} - \mathbf{A}^k\|_F^2, \\ h_3(\mathcal{Y}, \mathcal{Z}_3^k) = f(\mathbf{X}^{k+1}, \mathbf{A}^{k+1}, \mathcal{Y}) + \frac{\rho}{2}\|\mathcal{Y} - \mathcal{Y}^k\|_F^2, \end{cases}$$

and $\mathcal{Z}_1^k = (\mathbf{X}^k, \mathbf{A}^k, \mathcal{Y}^k)$, $\mathcal{Z}_2^k = (\mathbf{X}^{k+1}, \mathbf{A}^k, \mathcal{Y}^k)$, $\mathcal{Z}_3^k = (\mathbf{X}^{k+1}, \mathbf{A}^{k+1}, \mathcal{Y}^k)$. Then, problem (13) can be rewritten as follows

$$\begin{cases} \mathbf{X}^{k+1} = \underset{X}{\operatorname{argmin}}\ h_1(\mathbf{X}, \mathcal{Z}_1^k), \\ \mathbf{A}^{k+1} = \underset{A}{\operatorname{argmin}}\ h_2(\mathbf{A}, \mathcal{Z}_2^k), \\ \mathcal{Y}^{k+1} = \underset{\mathcal{P}_\Omega(\mathcal{Y}) = \mathcal{F}}{\operatorname{argmin}}\ h_3(\mathcal{Y}, \mathcal{Z}_3^k). \end{cases} \tag{14}$$

Note that **X**-, $\mathcal{Y}$- and **A**-sub-problem can be solved alternately. Therein, since $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3)$, the **X**-sub-problem in our model can be decomposed into three independent problems, i.e. $\mathbf{X}_1$- $\mathbf{X}_2$- and $\mathbf{X}_3$-sub-problem. It goes the same for **A**-sub-problem, where $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3)$. Then all the sub-problems in (14) except $\mathbf{X}_3$-sub-problem have close-formed solutions:

$$\begin{cases} \mathbf{X}_n^{k+1} = ((\mathbf{A}_n^k)^{\mathrm{T}}\mathbf{A}_n^k + \rho\mathbf{I}_1)^{\dagger}((\mathbf{A}_n^k)^{\mathrm{T}}\mathbf{Y}_{(n)}^k + \rho\mathbf{X}_n^k), & n = 1, 2, \\ \mathbf{A}_n^{k+1} = (\mathbf{Y}_{(n)}^k\mathbf{X}_n^{k+1^{\mathrm{T}}} + \rho\mathbf{A}_n^k)(\mathbf{X}_n^{k+1}\mathbf{X}_n^{k+1^{\mathrm{T}}} + \rho\mathbf{I}_2)^{\dagger}, & n = 1, 2, 3, \\ \mathcal{Y}^{k+1} = \mathcal{P}_{\Omega^c}(\sum_{n=1}^{3}\alpha_n \mathrm{fold}_n(\frac{\mathbf{A}_n^{k+1}\mathbf{X}_n^{k+1} + \rho\mathbf{Y}_{(n)}^k}{1+\rho})) + \mathcal{F}, & n = 1, 2, 3, \end{cases} \tag{15}$$

where $\mathbf{A}\dagger$ denotes the Moore–Penrose pseudo-inverse of **A**. The complexity of computing $\mathcal{Y}$ is $\mathrm{O}(r_1 I_1 s_3 + r_1 I_2 s_2 + r_1 I_3 s_3)$, where $s_n = \prod_{m=1, m\neq n}^{3} I_m$. The cost of computing $\mathbf{A}_n$ is $\mathrm{O}(I_n r_n^2 + I_n r_n s_n + r_n^2 s_n)$ operations for $n = 1, 2, 3$, while he complexity of computing $\mathbf{X}_n$ is $\mathrm{O}(I_n r_n^2 + I_n r_n s_n + r_n^2 s_n)$ operations for $n = 2, 3$.

The $\mathbf{X}_3$-sub-problem, which can be written as follows

$$\mathbf{X}_3^{k+1} = \underset{\mathbf{X}_3}{\arg\min}\ \frac{\alpha_3}{2}\|\mathbf{Y}_{(3)}^k - \mathbf{A}_3^k\mathbf{X}_3\|_F^2 + \lambda\|\mathbf{W}\mathbf{X}_3^{\mathrm{T}}\|_{1,1} + \frac{\rho}{2}\|\mathbf{X}_3 - \mathbf{X}_3^k\|_F^2,$$

can be easily solved by alternating direction method (ADM) [6,9,32,36,46,49].

Following, we solve $\mathbf{X}_3$-sub-problem. In order to separate the $l_1$ and $l_2$ components, we introduce an auxiliary variable and rewrite it as the following equivalent constrained problem

$$\begin{aligned} \underset{\mathbf{X},\mathbf{Z}}{\arg\min}\quad & \frac{\mu}{2}\|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F^2 + \|\mathbf{Z}\|_{1,1} + \frac{\rho}{2}\|\mathbf{X} - \mathbf{X}^k\|_F^2 \\ \text{s.t.}\quad & \mathbf{Z} = \mathbf{W}\mathbf{X}^{\mathrm{T}}. \end{aligned} \tag{16}$$

The augmented Lagrangian function of (16) is

$$\begin{aligned} L_\beta(\mathbf{X},\mathbf{Z},\mathbf{\Lambda}) =& \frac{\mu}{2}\|\mathbf{Y}-\mathbf{A}\mathbf{X}\|_F^2 + \|\mathbf{Z}\|_{1,1} + \frac{\rho}{2}\|\mathbf{X}-\mathbf{X}^k\|_F^2 \\ &+ \langle \mathbf{\Lambda}, \mathbf{W}\mathbf{X}^{\mathrm{T}} - \mathbf{Z}\rangle + \frac{\beta}{2}\|\mathbf{W}\mathbf{X}^{\mathrm{T}} - Z\|_F^2 \\ =& \frac{\mu}{2}\|\mathbf{Y}-\mathbf{A}\mathbf{X}\|_F^2 + \|\mathbf{Z}\|_{1,1} + \frac{\rho}{2}\|\mathbf{X}-\mathbf{X}^k\|_F^2 \\ &+ \frac{\beta}{2}\|\mathbf{W}\mathbf{X}^{\mathrm{T}} - \mathbf{Z} + \frac{\mathbf{\Lambda}}{\beta}\|_F^2 + \text{const.}, \end{aligned} \tag{17}$$

where $\mathbf{\Lambda}$ is the Lagrange Multiplier.

Then, let $\mathbf{\Theta} = \frac{\mathbf{\Lambda}}{\beta}$, the problem $\arg\min_{\mathbf{X},\mathbf{Z},\mathbf{\Theta}} L_\beta(\mathbf{X},\mathbf{Z},\mathbf{\Theta})$ in (17) can be updated through alternating direction as:

$$\begin{cases} \mathbf{X}^{p+1} = \dfrac{\mu\mathbf{A}^{\mathrm{T}}\mathbf{Y} + \rho\mathbf{X}^k + \beta[\mathbf{W}^{\mathrm{T}}(\mathbf{Z}^p - \mathbf{\Theta}^p)]^{\mathrm{T}}}{\mu\mathbf{A}^{\mathrm{T}}\mathbf{A} + (\beta+\rho)\mathbf{I}}, \\ \mathbf{Z}^{p+1} = \mathcal{S}_{\frac{1}{\beta}}\left[\mathbf{W}(\mathbf{X}^{p+1})^{\mathrm{T}} + \mathbf{\Theta}\right], \\ \mathbf{\Theta}^{p+1} = \mathbf{\Theta}^p + \mathbf{W}(\mathbf{X}^{p+1})^{\mathrm{T}} - \mathbf{Z}^{p+1}. \end{cases} \tag{18}$$

As for the $\mathbf{X}_3$-sub-problem, at each iteration, the complexities of computing $\mathbf{X}_3$, $\mathbf{Z}$ and $\mathbf{\Theta}$ are all $O(r_3 s_3^2)$.

The pseudocode of the proposed algorithm is given in Algorithm 1. Specifically, the inputs are the observed data, which

**Algorithm 1** BSUM based optimization algorithm for minimizing (8).

**Input:** The observed tensor, $\mathcal{F}$; The set of index of observed entries, $\Omega$; The given $n$-rank, $\mathbf{r} = (r_1, r_2, r_3)$;
**Output:** The completed tensor $\mathcal{Y}$;

```
Initialization: A_n^0 = rand(I_n × r_n), X_n^0 = rand(r_n × ∏_{m=1,m≠n}^{k-1} I_m), (n = 1, 2, 3), Y = P_Ω(F)
while not converged do
    Updating X_1 and X_2 via (15);
    while not converged do
        Updating X_3, Z and Θ via (18);
    end while
    Updating A_1, A_2 and A_3 via (15);
    Updating Y via (15);
end while
return Y;
```

is the partially observed entries of the underlying tensor, the set of the index of the observed entries, which indicates the location of observed entries, and the estimated rank of the underlying tensor. And the output of the proposed algorithm is the reconstructed tensor, e.g., the complete hyperspectral image, video or MRI data. Additionally, in Algorithm 1, the cost of computing all the variables at each iteration is $O(r_1 I_1 s_3 + r_1 I_2 s_2 + r_1 I_3 s_3 + I_3 r_3^2 + r_3^2 s_3 + \sum_{n=1}^{2}(I_n r_n^2 + I_n r_n s_n + r_n^2 s_n) + r_3 s_3^2)$.

### 3.3. Convergence analysis

In the following, we study the convergence of the proposed algorithm. Recently, M. Razaviyayn et al. [34] proposed the BSUM for the non-smooth optimization problem. It is an alternative inexact block coordinate descent method. Following, we restated the convergence result in [34] 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:*

$$\begin{cases} u_i(y_i, y) = f(y), \ \forall y \in \mathcal{X}, \forall i; \\ u_i(x_i, y) \geq f(y_1, \cdots, y_{i-1}, x_i, y_{i+1}, \cdots, y_n), \ \forall x_i \in \mathcal{X}_i, \ \forall y \in \mathcal{X}, \ \forall i; \\ u_i'(x_i, y; d_i)\,|_{x_i = y_i} = f'(y; d), \ \forall d = (0, \cdots, d_i, \cdots, 0), \ \text{s.t. } y_i + d_i \in \mathcal{X}_i, \ \forall i; \\ u_i(x_i, y) \text{ is continuous in } (x_i, y), \ \forall i; \end{cases} \tag{19}$$

*where* $u_i(x_i, y)$ *is the sub-problem with respect to the* $i$*-th 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* $i = 1, \cdots, n$. *Furthermore, assume that each sub-problem* $\arg\min u_i(x_i, x^{k-1})$, *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* $f$. *In addition, if* $f(\cdot)$ *is regular at* $z$, *then* $z$ *is a stationary point.*

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 (13) 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 $k$-th iteration, which satisfies the following conditions:

$$\begin{cases} h_i(\mathcal{Z}_i, \mathcal{Z}) = f(\mathcal{Z}), \ \forall \mathcal{Z}, \ i = 1, 2, 3; \\ h_i(\bar{\mathcal{Z}}_i, \mathcal{Z}) \geq f(\mathcal{Z}_1, \cdots, \bar{\mathcal{Z}}_i, \cdots, \mathcal{Z}_3), \ \forall \bar{\mathcal{Z}}_i, \ \forall \mathcal{Z}, \ i = 1, 2, 3; \\ h_i'(\bar{\mathcal{Z}}_i, \mathcal{Z}; d_i) \mid_{\bar{\mathcal{Z}}_i = \mathcal{Z}_i} = f'(\mathcal{Z}; d), \ \forall d = (0, \cdots, d_i, \cdots, 0), \ i = 1, 2, 3; \\ h_i(\bar{\mathcal{Z}}_i, \mathcal{Z}) \text{ is continuous in } (\bar{\mathcal{Z}}_i, \mathcal{Z}), \ i = 1, 2, 3; \end{cases} \tag{20}$$

where $\mathcal{Z} = (\mathbf{X}, \mathbf{A}, \mathcal{Y})$, and $\mathcal{Z}_i$ is equal to $\mathbf{X}, \mathbf{A}, \mathcal{Y}$ 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. □

## 4. Numerical experiments

In this section, the performance of the proposed method for the tensor completion is to be reported. Specifically, we will test our method (denoted as "Framelet") for the reconstruction of synthetic data, video data, hyperspectral image and MRI data. To measure the reconstruction accuracies, we employ the peak signal-to-noise ratio (PSNR), the relative squared error (RSE), and the structural similarity index (SSIM). PSNR and RSE are defined as

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

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. SSIM measures the structural similarity of two images, please see [40] for details. Better completion results correspond to larger values in PSNR and SSIM and smaller values in RSE. All algorithms are implemented on the platform of Windows 10 and Matlab (R2014a) with an Intel(R) Core(TM) i5-4590 CPU at 3.30 GHz and 8 GB RAM.

We adopt the relative change (RelCha) of the two successive reconstructed tensors

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

as the stopping criterion of the proposed method. Here, the tolerance is set to be $10^{-5}$ for the synthetic data and $10^{-4}$ for the real data. All parameters ($\alpha$, $\mu$, $\beta$, $\rho$) used in our experiments are empirically selected from a candidate set: {0, $10^{-4}$, $10^{-3}$, $10^{-2}$, $10^{-1}$, 1, $10^1$, $10^2$, $10^3$, $10^4$}. If not specified, the parameters are selected as: the weights $\alpha_i = 1/3$ $(i = 1, 2, 3)$, the regularization parameter $\mu = 10$, the penalty parameter $\beta = 1000$ and the proximal parameter $\rho = 0.1$ for all experiments.

### 4.1. Synthetic data

In this subsection, we evaluate our method on synthetic data of size $50 \times 50 \times 50$, some frames of which can be viewed in Fig. 1 in Section 1. The $n$-rank of the simulated tensor is set as (20, 20, 14). The percentage of the randomly sampled elements varies from 10% to 40%. We reconstruct the synthetic tensor using three LRTC methods: TMac [45], TV based method [16] and our framelet based method. Following [16], except the PSNR, RSE and SSIM of the whole tensor, we also exhibit the average and the worst values of the PSNR, RSE and SSIM of the all frames to measure the quality of the reconstructed tensors (denoted them as "Whole", "Average", and "Worst" in Table 2, respectively).

From Table 2, it can be found that: 1) the larger the sampling rate, the better the results recovered by all methods; 2) with varied sampling rates, our results show the best performance among all results; 3) even the *worst* frame recovered by our method are still better than the ones by TMac and TV based methods.

### 4.2. Video data

In this subsection, we compare the performance of previously mentioned three methods (TMac [45], TV based method [16] and our framelet based method) on videos. We test 9 videos, named as "carphone", "suzie", "hall", "coastguard", "news", "salesman", "highway", "foreman", and "claire".[1] All videos are color with the YUV format. YUV is a color space typically used

[1] http://trace.eas.asu.edu/yuv/.

**Table 2**
PSNR, RSE ($10^{-2}$) and SSIM comparison of the results recovered by TMac [45], TV based method [16] and the Framelet based method for different sampling rates (SR).

| Index | SR | TMac | | | TV | | | Framelet | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Whole | Average | Worst | Whole | Average | Worst | Whole | Average | Worst |
| PSNR | 10% | 16.318 | 18.974 | 11.143 | 21.370 | 23.000 | 18.152 | **23.514** | 25.572 | 19.036 |
| | 20% | 27.056 | 31.468 | 21.668 | 33.308 | 37.060 | 28.630 | **38.022** | 40.425 | 33.291 |
| | 30% | 32.169 | 36.386 | 27.274 | 28.219 | 45.587 | 32.837 | **43.232** | 48.052 | 36.237 |
| | 40% | 51.486 | 64.967 | 41.678 | 57.557 | 59.427 | 50.885 | **60.266** | 61.245 | 56.969 |
| RSE | 10% | 92.2 | 79.9 | 156 | 51.6 | 46.8 | 70.2 | **40.3** | 35.8 | 63.2 |
| | 20% | 26.8 | 21.7 | 46.6 | 13.0 | 10.8 | 20.9 | **7.58** | 6.56 | 12.2 |
| | 30% | 14.9 | 12.2 | 24.6 | 7.41 | 5.57 | 13.0 | **4.16** | 3.13 | 8.77 |
| | 40% | 1.61 | 0.751 | 4.65 | 0.800 | 7.06 | 1.61 | **0.585** | 0.551 | 0.803 |
| SSIM | 10% | 0.50103 | 0.56217 | 0.20860 | 0.76229 | 0.78671 | 0.58224 | **0.91625** | 0.93098 | 0.85531 |
| | 20% | 0.94825 | 0.95333 | 0.90005 | 0.98798 | 0.9880 | 0.96113 | **0.99332** | 0.99347 | 0.99080 |
| | 30% | 0.98302 | 0.98500 | 0.96599 | 0.99717 | 0.99747 | 0.98828 | **0.99809** | 0.99816 | 0.99467 |
| | 40% | 0.99948 | 0.99941 | 0.99416 | 0.99988 | 0.99986 | 0.99909 | **0.99993** | 0.99993 | 0.99987 |

**Table 3**
The whole PSNR, RSE ($10^{-2}$) and SSIM comparisons of TMac [45], TV based method [16] and our framelet based method for video data with different sampling rates.

| video | SR | 10% | | | 20% | | | 30% | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Method | PSNR | RSE | SSIM | PSNR | RSE | SSIM | PSNR | RSE | SSIM |
| carphone | TMac | 12.541 | 50.2 | 0.41308 | 19.331 | 23.0 | 0.73523 | 34.113 | 4.19 | 0.94191 |
| | TV | 14.074 | 42.1 | 0.45843 | 30.586 | 6.29 | 0.88142 | 34.381 | 4.07 | 0.94494 |
| | Framelet | **14.772** | **38.9** | **0.51402** | **31.951** | **5.38** | **0.91328** | **34.824** | **3.86** | **0.9511** |
| suzie | TMac | 17.344 | 30.5 | 0.49540 | 22.837 | 16.2 | 0.79804 | 33.442 | 4.78 | 0.92967 |
| | TV | 21.030 | 20.0 | 0.59265 | 32.174 | 5.53 | 0.89762 | 34.668 | 4.15 | 0.93646 |
| | Framelet | **24.134** | **14.0** | **0.66559** | **32.585** | **5.28** | **0.90274** | **35.124** | **3.94** | **0.93979** |
| hall | TMac | 20.269 | 16.5 | 0.77694 | 32.270 | 4.13 | 0.91817 | 34.720 | 3.12 | 0.95994 |
| | TV | 22.936 | 12.1 | 0.80628 | 33.087 | 3.76 | 0.91879 | 34.759 | 3.10 | 0.96070 |
| | Framelet | **23.501** | **11.3** | **0.82554** | **33.773** | **3.48** | **0.92234** | **34.892** | **3.06** | **0.96203** |
| coastguard | TMac | 7.6936 | 85.0 | 0.03725 | 11.016 | 53.5 | 0.20093 | 17.026 | 26.8 | 0.48833 |
| | TV | 8.5586 | 76.9 | 0.06307 | 12.886 | 43.1 | 0.24497 | 24.682 | 11.1 | 0.75402 |
| | Framelet | **10.723** | **60.0** | **0.11708** | **15.991** | **30.2** | **0.28367** | **27.644** | **7.98** | **0.84089** |
| news | TMac | 11.200 | 73.9 | 0.18216 | 14.862 | 48.7 | 0.45001 | 20.693 | 24.8 | 0.74674 |
| | TV | 12.439 | 64.1 | 0.21153 | 17.808 | 34.7 | 0.53682 | 25.783 | 13.8 | 0.82495 |
| | Framelet | **14.965** | **47.9** | **0.34634** | **21.456** | **22.8** | **0.65449** | **31.205** | **7.38** | **0.90833** |
| salesman | TMac | 14.726 | 59.3 | 0.28229 | 19.926 | 32.6 | 0.71697 | 29.916 | 10.3 | 0.91208 |
| | TV | 16.762 | 46.9 | 0.34183 | 28.796 | 11.7 | 0.87136 | 34.381 | 6.22 | 0.94780 |
| | Framelet | **20.681** | **29.9** | **0.58184** | **33.121** | **7.14** | **0.93854** | **35.787** | **5.25** | **0.95937** |
| highway | TMac | 31.541 | 3.78 | 0.87220 | 33.813 | 2.91 | 0.92000 | 34.410 | 2.72 | 0.92856 |
| | TV | 31.694 | 3.71 | 0.87526 | 33.800 | 2.91 | 0.92011 | 34.394 | 2.72 | 0.92870 |
| | Framelet | **31.819** | **3.66** | **0.87939** | **33.927** | **2.87** | **0.92392** | **34.430** | **2.71** | **0.92951** |
| foreman | TMac | 10.969 | 42.8 | 0.17088 | 18.924 | 17.1 | 0.71089 | 32.447 | 3.67 | 0.92349 |
| | TV | 12.826 | 34.6 | 0.24775 | 29.529 | 5.06 | 0.85824 | 32.549 | 3.57 | 0.92513 |
| | Framelet | **13.791** | **31.0** | **0.28729** | **29.770** | **4.92** | **0.86879** | **32.707** | **3.51** | **0.92802** |
| claire | TMac | 30.248 | 7.09 | 0.93392 | 38.915 | 2.61 | 0.98247 | 39.608 | 2.41 | 0.98263 |
| | TV | 33.730 | 4.75 | 0.95127 | 39.044 | 2.58 | 0.98316 | 39.601 | 2.42 | 0.98291 |
| | Framelet | **34.488** | **4.35** | **0.95754** | **39.347** | **2.49** | **0.98501** | **39.707** | **2.39** | **0.98351** |

as part of a color image pipeline. Y stands for the luma component (the brightness) and U and V are the chrominance (color) components. Being same as [16], we only conduct the methods on the Y channel and the first 150 frames of all videos. All test videos can be viewed as 3-mode tensors of size $144 \times 176 \times 150$. We select the number of the largest 1% singular values to approximate the $n$-rank and vary the sampling rate from 10% to 30%.

Table 3 shows the *whole* PSNR, RSE and SSIM of the reconstructed tensors by different methods. From Table 3 we can note that our framelet based method obtains highest quality results for all videos with different sampling rates. Additionally, different videos indicate different difficulty levels for the reconstruction task, since the changes between frames are of different intensities. By the way, our framelet based method performs significantly better to deal with some videos with high difficulty level, e.g., the “coastguard” and “news”. The PSNR RSE and SSIM values of each frame of two representatively reconstructed videos, “coastguard” and “news”, are plotted in Fig. 2. Fig. 2 illustrates that almost all frames, recovered by our framelet based method, obtain better quality in terms of the PSNR RSE and SSIM. Moreover, the *average* PSNR RSE and SSIM of each frame of all the videos with different sampling rate are the best by our method. Two representative results,

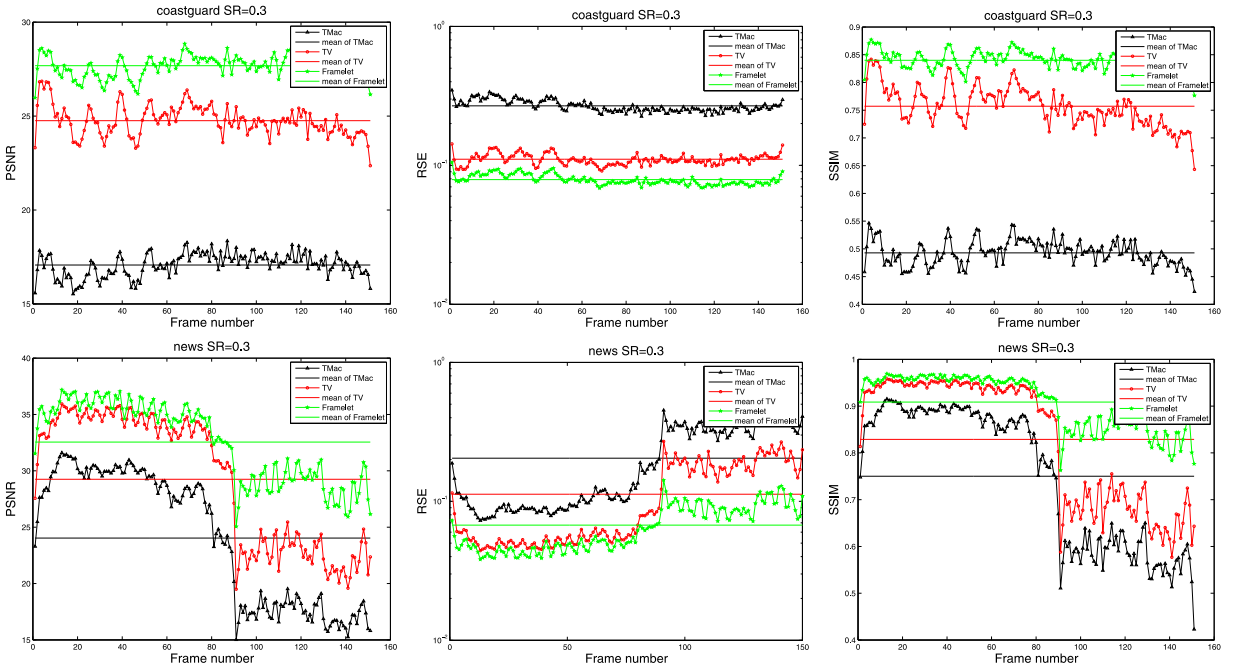


**Fig. 2.** The PSNR, RSE and SSIM of all frames of the reconstructed video results by TMac, TV based method and our framelet based method, respectively.


**Fig. 3.** The results recovered by TMac [45], TV based method [16] and our framelet based method. The first two rows represent two frames of the recovered video data *coastguard* and the other two rows represent the *news*. From left to right: the original data, the observed data with 70% missing entries, the recovered results by TMac, TV regularized and our framelet based method, respectively.

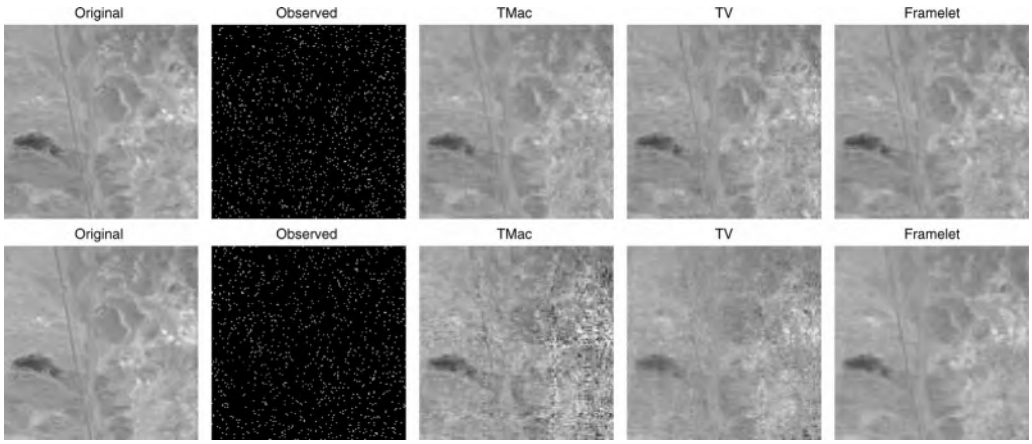


**Fig. 4.** The results recovered by TMac [45], TV based method [16] and our framelet based method. Each row represents a slice of the recovered hyperspectral image. From left to right: the original data, the observed data with 95% missing entries, the recovered results by TMac, TV based method and our framelet based method, respectively.

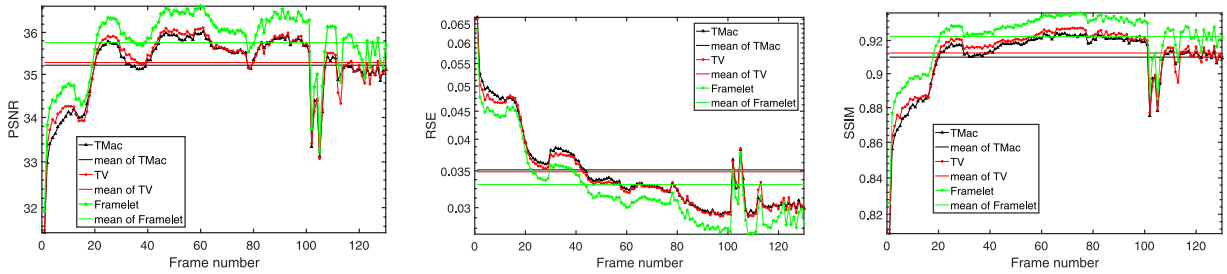


**Fig. 5.** The PSNR, RSE and SSIM of the recovered hyperspectral image by TMac [45], TV based method [16] and our framelet based method for all bands, respectively.

recovered by TMac, TV based method, and our framelet based method, is illustrated in Fig. 3. Considering geometric features and details in Fig. 3, it is obvious that our method visually outperforms TMac and TV based method.

### 4.3. Hyperspectral image

In this subsection, we use Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Cuprite data[2] to test the performance of different comparing methods. The hyperspectral data used in the experiments correspond to $150 \times 150$ pixels subset with 130 spectral bands (see [13,14,49] for more details). The number of the largest 1% singular values is chosen to approximate the rank of each mode, including two spatial modes and a spectral mode. In Fig. 4, we display the recovered results by TMac, TV based method, and our method. Fig. 4 shows that the reconstructed results by the proposed method are visually the best, compared with the results of TMac and TV based method. The PSNR, RSE, and SSIM of each frame are shown in Fig. 5. In terms of the three indices of each frame, our method surpasses other two.

### 4.4. MRI

As we mentioned in Section 1, the factor images, reshaped from row vectors of $\mathbf{X}_3$ by lexicographical ordering, maintain some priors that the original images possesses. However, for video data and hyperspectral data, we do not know the prior of $\mathbf{X}_1$ and $\mathbf{X}_2$, since the lateral and frontal slices of them could not be treated as images. Therefore, we only regularize tight framelet coefficients of $\mathbf{X}_3$ in the hyperspectral image and video reconstruction. Fortunately, slices of the MRI data form different directions can be treated as images. Hence we can simultaneously impose the piecewise smooth prior to different directions of the tensor. Similarly, we add the framelet regularizer to the factor matrix $\mathbf{X}_1$, $\mathbf{X}_2$ and $\mathbf{X}_3$ in (8). In particular, it is not difficult to solve the $\mathbf{X}_1$- and $\mathbf{X}_2$-sub-problem via (18).

Our experiments in this subsection are conducted on the cubical MRI data,[3] which is a 3-mode tensor with the size of $181 \times 217 \times 181$. We compare our three-directional framelet based method (3D-Framelet) in comparison with TMac [45], TV

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

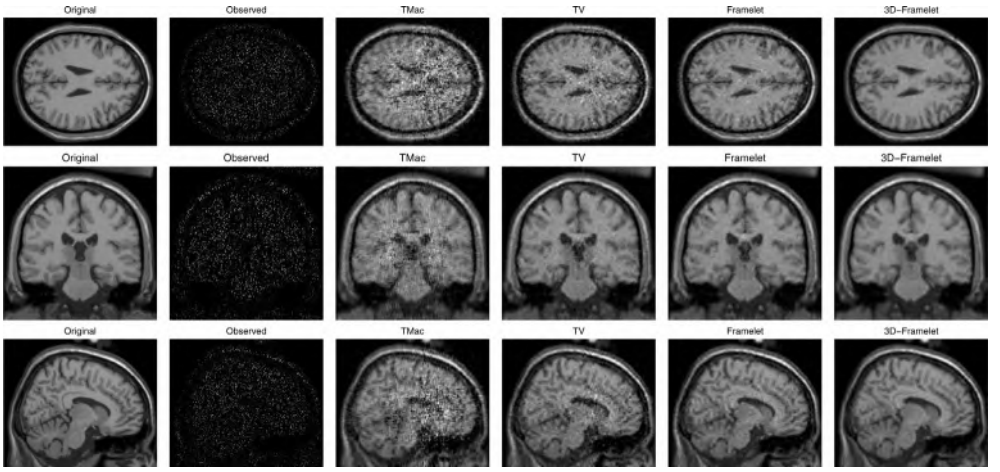


**Fig. 6.** An example of the recovered slices from three directions. From left to right: the original data, the observed data (90% entries missing), the recovered results by TMac [45], TV based method [16], our framelet based method and 3D-framelet based method, respectively.

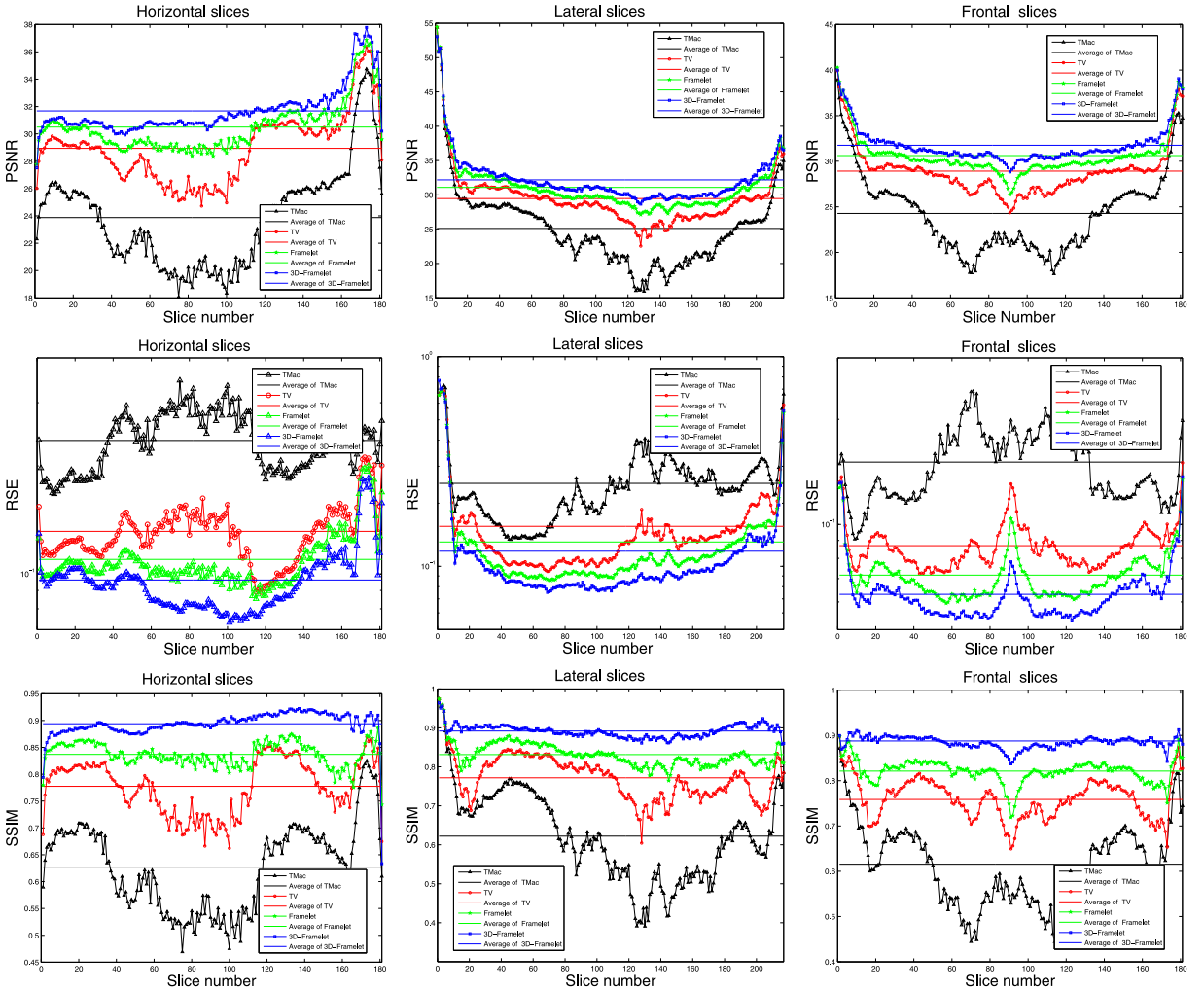


**Fig. 7.** The PSNR, RSE and SSIM of the reconstructed video results by TMac [45], TV based method [16], our framelet based method, and 3D-framelet based method for every frame. Each column represents the three numerical indices of every frame of a fixed direction, respectively.

based method [16] and our framelet based method. The $n$-rank is approximated by the numbers of the largest 1% singular values of the unfolding matrixes of the MRI data. As mentioned before, we can treat every mode equally for MRI data, e.g., as shown in Fig. 6. Therefore, in this subsection, we take the indices (PSNR, RSE, and SSIM) of slices from three directions, namely the horizontal lateral and frontal slices, into consideration. In particular, 90% entries of the original data is randomly removed in the example.

Fig. 6 shows three example slices of the recovered MRI data observed from different directions. Clearly, the recovered results obtained by the proposed methods are visually better than those obtained by TMac and TV based method. Fig. 7 presents the PSNR, RSE and SSIM results of every frame, recovered by TMac, TV based method, framelet based method and 3D-framelet based method, from three directions. It also can be found in Fig. 7 that all the frames recovered by the proposed methods are better than that recovered by TMac and TV based method. More specifically, the results reconstructed by our 3D-framelet based method are both visually and quantitatively better than the results recovered by our framelet based method, which performs significantly better than TMac and TV based method.

## 5. Conclusions

In this paper, we proposed a new tensor completion model combining framelet and low-rank matrix factorization for tensor completion with applications on the reconstruction of the video data, hyperspectral images, and MRI data. An efficient BSUM based algorithm was developed to solve the proposed model. It was demonstrated that our numerical scheme converged to the coordinatewise minimizers. Experimental results showed that: 1) our framelet based method, was more effective than TMac and TV regularized method when dealing with video and hyperspectral image; 2) when it came to the MRI data, our 3D-framelet based method visually and quantitatively obtained the best results, and our framelet based method got the second-best results.

## Acknowledgment

The authors would like to express their thanks 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 the National Science Foundation of China (61772003, 61702083), the Fundamental Research Funds for the Central Universities (ZYGX2016J132, ZYGX2016J129, ZYGX2016KYQD142).

## 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 (SIGGRAPH), 2000, pp. 417–424.
[2] J.-F. Cai, R.H. Chan, Z. Shen, A framelet-based image inpainting algorithm, Appl. Comput. Harmon. Anal. 24 (2) (2008) 131–149.
[3] J.-F. Cai, S. Osher, Z. Shen, Split Bregman methods and frame based image restoration, Multiscale Model. Simul. 8 (2) (2009) 337–369.
[4] W. Cao, Y. Wang, J. Sun, M. Deyu, C. Yang, A. Cichocki, Z. Xu, Total variation regularized tensor RPCA for background subtraction from compressive measurements, IEEE Trans. Image Process. 25 (9) (2016) 4075–4090.
[5] 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) (2011) 3097–3111.
[6] Y. Chen, T.-Z. Huang, L.-J. Deng, X.-L. Zhao, M. Wang, Group sparsity based regularization model for remote sensing image stripe noise removal, Neurocomputing (2017), doi:10.1016/j.neucom.2017.05.018.
[7] Y. Chen, T.-Z. Huang, X.-L. Zhao, L.-J. Deng, J. Huang, Stripe noise removal of remote sensing images by total variation regularization and group sparsity constraint, Remote Sens. 9 (6) (2017) 559.
[8] V. De Silva, L.H. Lim, Tensor rank and the ill-posedness of the best low-rank approximation problem, SIAM J. Matrix Anal. Appl. 30 (3) (2008) 1084–1127.
[9] L.-J. Deng, W. Guo, T.-Z. Huang, Single image super-resolution by approximated heaviside functions, Inf. Sci. 348 (2016) 107–123.
[10] D.C. Dobson, F. Santosa, Recovery of blocky images from noisy and blurred data, SIAM J. Appl. Math. 56 (4) (1996) 1181–1198.
[11] S. Gandy, B. Recht, I. Yamada, Tensor completion and low-n-rank tensor recovery via convex optimization, Inverse Prob. 27 (2) (2011) 025010.
[12] C.J. Hillar, L.-H. Lim, Most tensor problems are np-hard, J. ACM (JACM) 60 (6) (2013) 45.
[13] M.-D. Iordache, J.M. Bioucas-Dias, A. Plaza, Sparse unmixing of hyperspectral data, IEEE Trans. Geosci. Remote Sens. 49 (6) (2011) 2014–2039.
[14] M.-D. Iordache, J.M. Bioucas-Dias, A. Plaza, Total variation spatial regularization for sparse hyperspectral unmixing, IEEE Trans. Geosci. Remote Sens. 50 (11) (2012) 4484–4502.
[15] T.-Y. Ji, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, L.-J. Deng, A non-convex tensor rank approximation for tensor completion, Appl. Math. Model. 48 (2017) 410–422.
[16] T.-Y. Ji, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, G. Liu, Tensor completion using total variation and low-rank matrix factorization, Inf. Sci. 326 (2016) 243–257.
[17] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, L.-J. Deng, Y. Wang, A novel tensor-based video rain streaks removal approach via utilizing discriminatively intrinsic priors, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 4057–4066.
[18] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, T.-H. Ma, Patch-based principal component analysis for face recognition, Comput. Intell. Neurosci. (2017), doi:10.1155/2017/5317850.
[19] R. Ke, W. Li, M.K. Ng, Numerical ranges of tensors, Linear Algebra Appl. 508 (2016) 100–132.
[20] T.G. Kolda, B.W. Bader, Tensor decompositions and applications, SIAM Rev. 51 (3) (2009) 455–500.
[21] 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, 2005, pp. 242–249.
[22] N. Komodakis, Image completion using global optimization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006, pp. 442–452.
[23] T. Korah, C. Rasmussen, Spatiotemporal inpainting for recovering texture maps of occluded building facades, IEEE Trans. Image Process. 16 (9) (2007) 2262–2271.
[24] N. Kreimer, M.D. Sacchi, A tensor higher-order singular value decomposition for prestack seismic data noise reduction and interpolation, Geophysics 77 (3) (2012) V113–V122.

[25] 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.
[26] N. Li, B. Li, Tensor completion for on-board compression of hyperspectral images, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), 2010, pp. 517–520.
[27] 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 (2015) 232–246.
[28] J. Liu, P. Musialski, P. Wonka, J. Ye, Tensor completion for estimating missing values in visual data, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 208–220.
[29] S. Ma, D. Goldfarb, L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, Math. Program. 128 (1–2) (2011) 321–353.
[30] T.-H. Ma, T.-Z. Huang, X.-L. Zhao, Y. Lou, Image deblurring with an inaccurate blur kernel using a group-based low-rank image prior, Inf. Sci. 408 (2017) 213–233.
[31] T.-H. Ma, Y. Lou, T.-Z. Huang, Truncated l_1-2 models for sparse recovery and rank minimization, SIAM J. Imaging Sci. 10 (3) (2017) 1346–1380.
[32] J.-J. Mei, Y. Dong, T.-Z. Huang, W. Yin, Cauchy noise removal by nonconvex ADMM with convergence guarantees, J. Sci. Comput. (2017) 1–24, doi:10.1007/s10915-017-0460-5.
[33] M. Nikolova, Local strong homogeneity of a regularized estimator, SIAM J. Appl. Math. 61 (2) (2000) 633–658.
[34] M. Razaviyayn, M. Hong, Z.-Q. Luo, A unified convergence analysis of block successive minimization methods for nonsmooth optimization, SIAM J. Optim. 23 (2) (2013) 1126–1153.
[35] A. Ron, Z. Shen, Affine systems in $l_2(\mathbb{R}^d)$: The analysis of the analysis operator, J. Funct. Anal. 148 (2) (1997) 408–447.
[36] A.C. Sauve, A. Hero, W.L. Rogers, S. Wilderman, N. Clinthorne, 3D image reconstruction for a Compton SPECT camera model, IEEE Trans. Nucl. Sci. 46 (6) (1999) 2075–2084.
[37] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, Z. Chen, CubeSVD: a novel approach to personalized web search, in: Proceedings of the 14th International Conference on World Wide Web, 2005, pp. 382–390.
[38] K.-C. Toh, S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems, Pacific J. Optim. 6 (15) (2010) 615–640.
[39] V.N. Varghees, M.S. Manikandan, R. Gini, Adaptive MRI image denoising using total-variation and local noise estimation, in: Proceedings of the International Conference on Advances in Engineering, Science and Management (ICAESM), 2012, pp. 506–511.
[40] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[41] Z. Wen, W. Yin, Y. Zhang, Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm, Math. Programm. Comput. 4 (4) (2012) 333–361.
[42] Y. Wu, H. Tan, Y. Li, F. Li, H. He, Robust tensor decomposition based on cauchy distribution and its applications, Neurocomputing 223 (2016) 107–117.
[43] Q. Xie, Q. Zhao, D. Meng, Z. Xu, S. Gu, W. Zuo, L. Zhang, Multispectral images denoising by intrinsic tensor sparsity regularization, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1692–1700.
[44] 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.
[45] Y. Xu, R. Hao, W. Yin, Z. Su, Parallel matrix factorization for low-rank tensor completion, Inverse Prob. Imaging 9 (2) (2013) 601–624.
[46] Y. Xu, W. Yin, Z. Wen, Y. Zhang, An alternating direction algorithm for matrix completion with nonnegative factors, Front. Math. China 7 (2) (2012) 365–384.
[47] G. Zhang, Y. Xu, F. Fang, Framelet-based sparse unmixing of hyperspectral images, IEEE Trans. Image Process. 25 (4) (2016) 1516–1529.
[48] Z. Zhang, K. Batselier, H. Liu, L. Daniel, N. Wong, Tensor computation: a new framework for high-dimensional problems in eda, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 36 (4) (2017) 521–536.
[49] X.-L. Zhao, F. Wang, T.-Z. Huang, M.K. Ng, R.J. Plemmons, Deblurring and sparse unmixing for hyperspectral images, IEEE Trans. Geosci. \Remote Sens. 51 (7) (2013) 4045–4058.
[50] L. Zhuang, J.M. Bioucas-Dias, Fast hyperspectral image denoising and inpainting based on low-rank and sparserepresentations, IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens. (2017). http://www.lx.it.pt/~bioucas/files/ieee_jstars_2018_FastHyDe_submitted.pdf.
[51] 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) (2013) B1304–B1320.