Fully-Connected Tensor Network Decomposition and Its Application to Higher-Order Tensor Completion

Yu-Bang Zheng,[1] Ting-Zhu Huang,[*,1] Xi-Le Zhao,[*,1] Qibin Zhao,[2,3] Tai-Xiang Jiang[4]

[1]School of Mathematical Sciences, University of Electronic Science and Technology of China, China
[2]Tensor Learning Team, RIKEN Center for Advanced Intelligence Project (AIP), Japan
[3]School of Automation, Guangdong University of Technology, China
[4]School of Economic Information Engineering, Southwestern University of Finance and Economics, China
{zhengyubang, xlzhao122003}@163.com, tingzhuhuang@126.com, qibin.zhao@riken.jp, taixiangjiang@gmail.com

Abstract

The popular tensor train (TT) and tensor ring (TR) decompositions have achieved promising results in science and engineering. However, TT and TR decompositions only establish an operation between adjacent two factors and are highly sensitive to the permutation of tensor modes, leading to an inadequate and inflexible representation. In this paper, we propose a generalized tensor decomposition, which decomposes an Nth-order tensor into a set of Nth-order factors and establishes an operation between any two factors. Since it can be graphically interpreted as a fully-connected network, we named it fully-connected tensor network (FCTN) decomposition. The superiorities of the FCTN decomposition lie in the outstanding capability for characterizing adequately the intrinsic correlations between any two modes of tensors and the essential invariance for transposition. Furthermore, we employ the FCTN decomposition to one representative task, i.e., tensor completion, and develop an efficient solving algorithm based on proximal alternating minimization. Theoretically, we prove the convergence of the developed algorithm, i.e., the sequence obtained by it globally converges to a critical point. Experimental results substantiate that the proposed method compares favorably to the state-of-the-art methods based on other tensor decompositions.

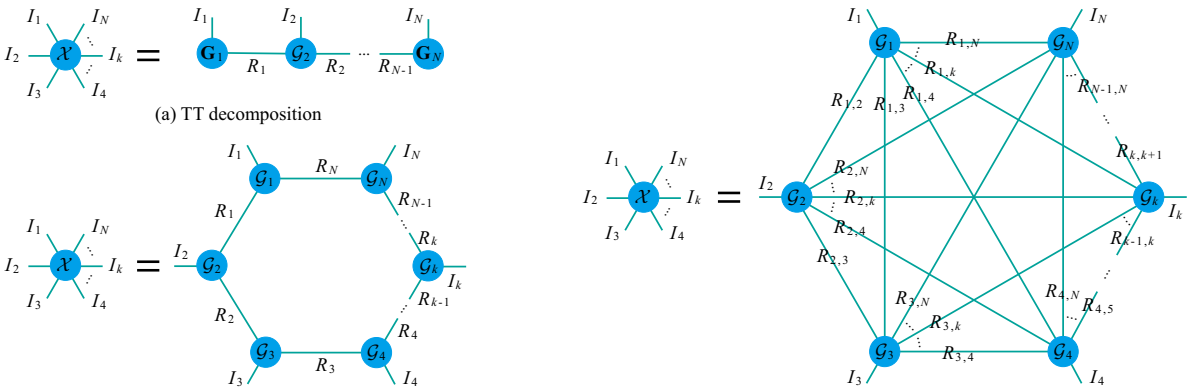
Introduction

The rapid advance in science and technology has given rise to the wide presence of higher-order data, e.g., multi-temporal, multi-spectral, and multi-scale data, which are usually expressed by higher-order tensors. Tensor decompositions focus on decomposing a higher-order tensor to a set of low-dimensional factors used to represent its latent features, which have powerful capability to capture the global correlations of tensors and have been widely applied in a variety of fields, such as signal processing, computer vision, and medical imaging (Kolda and Bader 2009; Mu et al. 2014; Anandkumar et al. 2014; Cong et al. 2015; Zhao, Zhang, and Cichocki 2015; Lu et al. 2016; Yokota and Hontani 2017; Yokota et al. 2019; Zheng et al. 2020). By designing different structures of latent factors and different multi-linear operations among them, various tensor decompositions have been proposed and attracted considerable attention. Among them, the Tucker decomposition and the CANDECOMP/PARAFAC (CP) decomposition as two most classical decompositions have achieved great success in the past decade (Kolda and Bader 2009; Gandy, Recht, and Yamada 2011; Liu et al. 2013, 2014; Zhao et al. 2016; Yokota, Zhao, and Cichocki 2016; Li, Ye, and Xu 2017; Xie et al. 2018; Yao et al. 2019; He et al. 2019; Phan et al. 2020).

More recently, an increasing number of tensor network-based tensor decompositions have emerged and shown great ability to deal with higher-order, especially beyond third-order, tensors. One of the most representative among them is the tensor train (TT) decomposition (Oseledets 2011), which decomposes an Nth-order tensor into N-2 third-order tensors located at intermediate and two matrices located at both sides (see Figure 1(a)). Besides, from the first TT factor (matrix), each factor needs to conduct a multi-linear operation with its next factor, until the last one (matrix). Subsequently, as an extension of the TT decomposition, tensor ring (TR) decomposition (Zhao et al. 2016) replaced two matrices in TT factors by third-order tensors and established an additional multi-linear operation between them (see Figure 1(b)). Since TT and TR decompositions have the outstanding capability in super-compression and computational practicability, they have been employed in many applications, such as signal restoration, compression and reconstruction, and image/video recovery (Bengua et al. 2017; Imaizumi, Maehara, and Hayashi 2017; Ding et al. 2019; Zhao et al. 2019; Yuan et al. 2018; Yuan et al. 2019; Chen et al. 2020). Many of them can be regarded as a tensor completion (TC) problem, which aims to complete a tensor from its partial observation.

However, there are two limitations to TT and TR decompositions. First, these two decompositions only establish an operation/connection between adjacent two factors, rather than any two factors, which leads to a limited characterization for correlations of tensors. Second, TT decomposition keeps the invariance only when the modes of the target tensor make a reverse permuting, while TR decomposition keeps the invariance only when the modes of the target tensor make a circular shifting or a reverse permuting. These imply that these two decompositions are highly sensitive to the permutation of tensor modes, leading to the inflexibility of decompositions and applications.

*Corresponding authors.



(a) TT decomposition

(b) TR decomposition

(c) FCTN decomposition

Figure 1: A graphical representation of TT, TR, and the proposed FCTN decompositions.

To tackle the above two limitations, we propose a fully-connected tensor network (FCTN) decomposition, which decomposes an Nth-order tensor into a set of Nth-order factors and establishes a multi-linear operation/connection between any two factors (see Figure 1(c)). The proposed FCTN decomposition has the superior capability to characterize directly the intrinsic correlations between any two modes of tensors and is proved to be essentially invariable for any permutations of tensor modes. The main contributions of this paper are summarized as three-folds:

1) We propose an FCTN decomposition, which breaks through the limitations of TT and TR decompositions in terms of correlation characterization and transpositional invariance.

2) We employ the FCTN decomposition to the TC problem and develop an efficient proximal alternating minimization (PAM)-based algorithm to solve it.

3) We theoretically demonstrate the convergence of the developed algorithm by proving the sequence obtained by it globally converges to a critical point (local minima).

Notations and Preliminaries

In this paper, we denote scalars, vectors, matrices, and tensors by x, $\mathbf{x}$, $\mathbf{X}$, and $\mathcal{X}$, respectively. For an Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, we employ $\mathcal{X}(i_1, i_2, \cdots, i_N)$ to denote its $(i_1, i_2, \cdots, i_N)$th element. The Frobenius norm of $\mathcal{X}$ is defined as $\|\mathcal{X}\|_F = \sqrt{\sum_{i_1, i_2, \cdots, i_N} |\mathcal{X}(i_1, i_2, \cdots, i_N)|^2}$. For the sake of clarity, we use $\mathcal{X}_{1:d}$ to denote $(\mathcal{X}_1, \mathcal{X}_2, \cdots, \mathcal{X}_d)$.

FCTN Decomposition

Basic Theory

Before proposing the FCTN decomposition, we first develop several basic definitions and theorems.

Definition 1 (Generalized Tensor Transposition) *Supposing that $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is an Nth-order tensor and the vector $\mathbf{n}$ is a reordering of the vector $(1, 2, \cdots, N)$. The vector $\mathbf{n}$-based generalized tensor transposition of $\mathcal{X}$ is defined as a tensor $\vec{\mathcal{X}}^{\mathbf{n}} \in \mathbb{R}^{I_{n_1} \times I_{n_2} \times \cdots \times I_{n_N}}$, which is generated by rearranging the modes of $\mathcal{X}$ in the order specified by the vector $\mathbf{n}$. We denote the corresponding operation and its inverse operation by $\vec{\mathcal{X}}^{\mathbf{n}} = \texttt{permute}(\mathcal{X}, \mathbf{n})$ and $\mathcal{X} = \texttt{ipermute}(\vec{\mathcal{X}}^{\mathbf{n}}, \mathbf{n})$, respectively.*

Definition 2 (Generalized Tensor Unfolding) *Supposing that $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is an Nth-order tensor and the vector $\mathbf{n}$ is a reordering of the vector $(1, 2, \cdots, N)$. The generalized tensor unfolding of $\mathcal{X}$ is defined as a matrix*

$$\mathbf{X}_{[n_{1:d}; n_{d+1:N}]} = \texttt{reshape}^{1}\left(\vec{\mathcal{X}}^{\mathbf{n}}, \prod_{i=1}^{d} I_{n_i}, \prod_{i=d+1}^{N} I_{n_i}\right).$$

We denote the corresponding operation and its inverse operation by $\mathbf{X}_{[n_{1:d}; n_{d+1:N}]} = \texttt{GenUnfold}(\mathcal{X}, n_{1:d}; n_{d+1:N})$ and $\mathcal{X} = \texttt{GenFold}(\mathbf{X}_{[n_{1:d}; n_{d+1:N}]}, n_{1:d}; n_{d+1:N})$, respectively.

For example, the traditional mode-k unfolding (Kolda and Bader 2009) of $\mathcal{X}$ is $\mathbf{X}_{[k; 1,2,\cdots,k-1,k+1,\cdots,N]}$, which is also simply denoted by $\mathbf{X}_{(k)}$.

Definition 3 (Tensor Contraction) *Supposing that vectors $\mathbf{n}$ and $\mathbf{m}$ are the reordering of vectors $(1, 2, \cdots, N)$ and $(1, 2, \cdots, M)$, respectively; $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and $\mathcal{Y} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_M}$ are two tensors satisfied $I_{n_i} = J_{m_i}$ with $i = 1, 2, \cdots, d$. The tensor contraction along the $n_{1:d}$th-modes of $\mathcal{X}$ and the $m_{1:d}$th-modes of $\mathcal{Y}$ yields an $(N + M - 2d)$th-order tensor*

$$\mathcal{Z} = \mathcal{X} \times_{n_{1:d}}^{m_{1:d}} \mathcal{Y} \in \mathbb{R}^{I_{n_{d+1}} \times \cdots \times I_{n_N} \times J_{m_{d+1}} \times \cdots \times J_{m_M}},$$

whose elements

$$\mathcal{Z}(i_{n_{d+1}}, \cdots, i_{n_N}, j_{m_{d+1}}, \cdots, j_{m_M}) = \sum_{i_{n_1}=1}^{I_{n_1}} \sum_{i_{n_2}=1}^{I_{n_2}} \cdots \sum_{i_{n_d}=1}^{I_{n_d}} \{\vec{\mathcal{X}}^{\mathbf{n}}(i_{n_1}, \cdots, i_{n_d}, i_{n_{d+1}}, \cdots, i_{n_N}) \vec{\mathcal{Y}}^{\mathbf{m}}(i_{n_1}, \cdots, i_{n_d}, j_{m_{d+1}}, \cdots, j_{m_M})\}.$$

Especially, here requires $n_{d+1} < n_{d+2} < \cdots < n_N$ and $m_{d+1} < m_{d+2} < \cdots < m_M$ to guarantee the uniqueness of the tensor $\mathcal{Z}$.

[1] Matlab commands.

Furthermore, Theorem 1 delivers the relationship of the tensor contraction and the matrix multiplication.

Theorem 1 *Supposing that $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and $\mathcal{Y} \in \mathbb{R}^{J_1 \times J_2 \times \cdots \times J_M}$ are two tensors, we have*

1) $\mathbf{X}^{\mathrm{T}}_{[n_{1:d};n_{d+1:N}]} = \mathbf{X}_{[n_{d+1:N};n_{1:d}]}$;

2) $\mathcal{Z} = \mathcal{X} \times_{n_{1:d}}^{m_{1:d}} \mathcal{Y} \Leftrightarrow$

$$\mathbf{Z}_{[1:N-d;N-d+1:N+M-2d]} = \mathbf{X}_{[n_{d+1:N};n_{1:d}]} \mathbf{Y}_{[m_{1:d};m_{d+1:M}]}.$$

Here vectors $\mathbf{n}$ *and* $\mathbf{m}$ *have the same setting as that in Definition 3.*

FCTN Decomposition

Definition 4 (FCTN Decomposition) *The FCTN decomposition aims to decompose an Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ into a set of Nth-order factor tensors $\mathcal{G}_k \in \mathbb{R}^{R_{1,k} \times R_{2,k} \times \cdots \times R_{k-1,k} \times I_k \times R_{k,k+1} \times \cdots \times R_{k,N}}$ with $k = 1, 2, \cdots, N$. More specifically, the element-wise form of the FCTN decomposition can be expressed as*

$$\begin{aligned}
&\mathcal{X}(i_1, i_2, \cdots, i_N) = \\
&\sum_{r_{1,2}=1}^{R_{1,2}} \sum_{r_{1,3}=1}^{R_{1,3}} \cdots \sum_{r_{1,N}=1}^{R_{1,N}} \sum_{r_{2,3}=1}^{R_{2,3}} \cdots \sum_{r_{2,N}=1}^{R_{2,N}} \cdots \sum_{r_{N-1,N}=1}^{R_{N-1,N}} \\
&\big\{\mathcal{G}_1(i_1, r_{1,2}, r_{1,3}, \cdots, r_{1,N}) \\
&\quad \mathcal{G}_2(r_{1,2}, i_2, r_{2,3}, \cdots, r_{2,N}) \cdots \\
&\quad \mathcal{G}_k(r_{1,k}, r_{2,k}, \cdots, r_{k-1,k}, i_k, r_{k,k+1}, \cdots, r_{k,N}) \cdots \\
&\quad \mathcal{G}_N(r_{1,N}, r_{2,N}, \cdots, r_{N-1,N}, i_N)\big\}.
\end{aligned} \tag{1}$$

Moreover, we denote the FCTN decomposition by $\mathcal{X} = FCTN\big(\{\mathcal{G}_k\}_{k=1}^N\big) = FCTN(\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_N)$ and call the vector collected by R_{k_1,k_2} ($1 \le k_1 < k_2 \le N$ and $k_1, k_2 \in \mathbb{N}^+$) as the FCTN-ranks.

To illustrate the FCTN decomposition vividly, Figure 1(c) gives a graphical representation of it. It is not hard to see that for second-order tensors, the FCTN decomposition is actually the matrix factorization and the FCTN-rank is actually the matrix rank. Furthermore, for higher-order tensors, any two FCTN factors $\mathcal{G}_{k_1}$ and $\mathcal{G}_{k_2}$ have an equal-sized mode R_{k_1,k_2} used to conduct the tensor contraction operation, which enables the FCTN decomposition to characterize adequately the intrinsic correlations between any two modes of the target tensor. This indicates an essential advantage of the FCTN decomposition over the TT and TR decompositions, which establish only the connection between adjacent two factors, leading to a limited characterization for correlations of tensors. Besides, the FCTN decomposition can degenerate to the TT and TR decompositions by simply setting the corresponding modes of factors to 1.

In second-order case, it is well known that the matrix factorization is essentially invariable under the transpositional condition, i.e., $\mathbf{X} = \mathbf{G}_1\mathbf{G}_2 \Leftrightarrow \mathbf{X}^{\mathrm{T}} = \mathbf{G}_2^{\mathrm{T}}\mathbf{G}_1^{\mathrm{T}}$. Naturally, it is expected to extend this property to higher-order cases.

Theorem 2 (Transpositional Invariance) *Supposing that an Nth-order tensor $\mathcal{X}$ has the following FCTN decomposition: $\mathcal{X} = FCTN(\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_N)$. Then, its vector $\mathbf{n}$-based generalized tensor transposition $\vec{\mathcal{X}}^{\mathbf{n}}$ can be expressed as $\vec{\mathcal{X}}^{\mathbf{n}} = FCTN\big(\vec{\mathcal{G}}^{\mathbf{n}}_{n_1}, \vec{\mathcal{G}}^{\mathbf{n}}_{n_2}, \cdots, \vec{\mathcal{G}}^{\mathbf{n}}_{n_N}\big)$, where $\mathbf{n} = (n_1, n_2, \cdots, n_N)$ is a reordering of the vector $(1, 2, \cdots, N)$.*

Theorem 2 illustrates another essential advantage of the FCTN decomposition as compared with the TT and TR decompositions. More specifically, the FCTN decomposition is essentially invariable, no matter how to permute the modes of the target tensor. But TR decomposition keeps the invariance only when the modes of the target tensor make a circular shifting or a reverse permuting. And TT decomposition keeps the invariance only when the modes of the target tensor make a reverse permuting.

The following theorem presents that the FCTN-ranks can bound the rank of all generalized tensor unfolding.

Theorem 3 *Supposing that an Nth-order tensor $\mathcal{X}$ can be represented by Equation (1), the following inequality holds:*

$$\mathtt{Rank}\big(\mathbf{X}_{[n_{1:d};n_{d+1:N}]}\big) \le \prod_{i=1}^{d} \prod_{j=d+1}^{N} R_{n_i,n_j},$$

where $R_{n_i,n_j} = R_{n_j,n_i}$ if $n_i > n_j$ and $(n_1, n_2, \cdots, n_N)$ is a reordering of the vector $(1, 2, \cdots, N)$.

Since the FCTN decomposition aims to characterize the intrinsic correlations between any two modes by establishing a connection between any two factors, the factors have to be designed as Nth-order tensors, which inevitably leads to the increment of the storage cost as compared to TT and TR decompositions. For an Nth-order $\mathcal{X} \in \mathbb{R}^{I \times I \times \cdots \times I}$, whose FCTN-ranks are the same value R_1, the FCTN decomposition requires $\mathcal{O}(NIR_1^{N-1})$ parameters to express it. It seems to stay on the same order of magnitude with that of the Tucker decomposition ($\mathcal{O}(NIR_2 + R_2^N)$ parameters). But when we express real-world data, the required FCTN-rank R_1 is usually far less than Tucker-rank R_2, because the FCTN decomposition uses R_1^{N-1} to bound Tucker-rank R_2 (as shown in Theorem 3). This indicates that the FCTN decomposition is superior to the Tucker decomposition regarding the storage cost[2].

Definition 5 (FCTN Composition) *We call the process of generating $\mathcal{X}$ by its FCTN factors $\mathcal{G}_k$ ($k = 1, 2, \cdots N$) as the FCTN composition, which is also denoted by $FCTN\big(\{\mathcal{G}_k\}_{k=1}^N\big)$. Furthermore, if one of the factors $\mathcal{G}_t$ ($t \in \{1, 2, \cdots, N\}$) does not participate in the composition, we denote it by $FCTN\big(\{\mathcal{G}_k\}_{k=1}^N, /\mathcal{G}_t\big)$.*

Theorem 4 *Supposing that $\mathcal{X} = FCTN\big(\{\mathcal{G}_k\}_{k=1}^N\big)$ and $\mathcal{M}_t = FCTN\big(\{\mathcal{G}_k\}_{k=1}^N, /\mathcal{G}_t\big)$, we obtain that*

$$\mathbf{X}_{(t)} = (\mathbf{G}_t)_{(t)} (\mathbf{M}_t)_{[m_{1:N-1};n_{1:N-1}]},$$

where

$$m_i = \begin{cases} 2i, & \text{if } i < t, \\ 2i-1, & \text{if } i \ge t, \end{cases} \quad \text{and} \quad n_i = \begin{cases} 2i-1, & \text{if } i < t, \\ 2i, & \text{if } i \ge t. \end{cases}$$

Theorem 4 reveals the relationship between one FCTN factor and the composition of the other factors. It is of great importance to the computation of the FCTN decomposition since computing one factor usually needs to fix the others.

[2]More experimental evidence is provided in the supplementary material, which is available at *https://yubangzheng.github.io*.

FCTN Decomposition-Based TC Method

Model and Solving Algorithm

Due to space limitations, we only apply the FCTN decomposition to one representative task, i.e., TC, which aims to recover missing elements of a higher-order tensor from its an incomplete observation. Giving an incomplete observation $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ of the target tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, the proposed FCTN decomposition-based TC (FCTN-TC) model can be formulated as

$$\min_{\mathcal{X},\mathcal{G}} \ \frac{1}{2}\|\mathcal{X} - \text{FCTN}(\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_N)\|_F^2 + \iota_{\mathbb{S}}(\mathcal{X}), \qquad (2)$$

where $\mathcal{G} = (\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_N)$ and

$$\iota_{\mathbb{S}}(\mathcal{X}) := \begin{cases} 0, & \text{if } \mathcal{X} \in \mathbb{S}, \\ \infty, & \text{otherwise}, \end{cases} \quad \text{with } \mathbb{S} := \{\mathcal{X} : \mathcal{P}_\Omega(\mathcal{X} - \mathcal{F}) = 0\}.$$

Here Ω denotes the index of the known elements and $\mathcal{P}_\Omega(\mathcal{X})$ is a projection operator which projects the elements in Ω to themselves and all others to zeros.

Since all optimization variables are coupled with each other, we employ the framework of PAM (Attouch, Bolte, and Svaiter 2013) to solve (2), whose solution can be obtained by alternately updating

$$\begin{cases} \mathcal{G}_k^{(s+1)} = \underset{\mathcal{G}_k}{\text{argmin}} \Big\{ f(\mathcal{G}_{1:k-1}^{(s+1)}, \mathcal{G}_k, \mathcal{G}_{k+1:N}^{(s)}, \mathcal{X}^{(s)}) \\ \qquad\qquad + \dfrac{\rho}{2}\|\mathcal{G}_k - \mathcal{G}_k^{(s)}\|_F^2 \Big\}, \ k = 1, 2, \cdots, N, \\ \mathcal{X}^{(s+1)} = \underset{\mathcal{X}}{\text{argmin}} \Big\{ f(\mathcal{G}^{(s+1)}, \mathcal{X}) + \dfrac{\rho}{2}\|\mathcal{X} - \mathcal{X}^{(s)}\|_F^2 \Big\}, \end{cases} \qquad (3)$$

where $f(\mathcal{G}, \mathcal{X})$ is the objective function of (2) and $\rho > 0$ is a proximal parameter.

1) Update $\mathcal{G}_k$: According to Theorem 4, the $\mathcal{G}_k$ $(k = 1, 2, \cdots, N)$-subproblems can be rewritten as

$$\begin{aligned} \mathcal{G}_k^{(s+1)} = \underset{\mathcal{G}_k}{\text{argmin}} \Big\{ & \frac{\rho}{2}\|(\mathbf{G}_k)_{(k)} - (\mathbf{G}_k^{(s)})_{(k)}\|_F^2 \\ & + \frac{1}{2}\|\mathbf{X}_{(k)}^{(s)} - (\mathbf{G}_k)_{(k)}(\mathbf{M}_k^{(s)})_{[m_{1:N-1}; n_{1:N-1}]}\|_F^2 \Big\}, \end{aligned} \qquad (4)$$

where $\mathcal{M}_k^{(s)} = \text{FCTN}\big(\mathcal{G}_{1:k-1}^{(s+1)}, \mathcal{G}_k, \mathcal{G}_{k+1:N}^{(s)}, /\mathcal{G}_k\big)$, and vectors $\mathbf{m}$ and $\mathbf{n}$ have the same setting as that in Theorem 4. The problem (4) can be directly solved as

$$\begin{aligned} & (\mathbf{G}_k^{(s+1)})_{(k)} = \\ & \big[\mathbf{X}_{(k)}^{(s)}(\mathbf{M}_k^{(s)})_{[n_{1:N-1}; m_{1:N-1}]} + \rho(\mathbf{G}_k^{(s)})_{(k)}\big] \\ & \big[(\mathbf{M}_k^{(s)})_{[m_{1:N-1}; n_{1:N-1}]}(\mathbf{M}_k^{(s)})_{[n_{1:N-1}; m_{1:N-1}]} + \rho\mathbf{I}\big]^{-1}, \end{aligned} \qquad (5)$$

and $\mathcal{G}_k^{(s+1)} = \texttt{GenFold}\big((\mathbf{G}_k^{(s+1)})_{(k)}, k; 1, \cdots, k-1, k+1, \cdots, N\big)$.

2) Update $\mathcal{X}$: The $\mathcal{X}$-subproblem has the following closed-form solution since it is a least square problem:

$$\mathcal{X}^{(s+1)} = \mathcal{P}_{\Omega^c}\left(\frac{\text{FCTN}\big(\{\mathcal{G}_k^{(s+1)}\}_{k=1}^N\big) + \rho\mathcal{X}^{(s)}}{1+\rho}\right) + \mathcal{P}_\Omega(\mathcal{F}). \qquad (6)$$

Algorithm 1 PAM-Based Solver for the FCTN-TC Model.

1: **Input:** The incomplete tensor $\mathcal{F} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, the index Ω, the maximal FCTN-rank $R^{\max}$, and $\rho = 0.1$.
2: **Initialization:** $s = 0$, $s^{\max} = 1000$, $\mathcal{X}^{(0)} = \mathcal{F}$, the initial FCTN-rank $R = \max\{\texttt{ones}^1(N(N-1)/2, 1), R^{\max} - 5\}$, and $\mathcal{G}_k^{(0)} = \texttt{rand}^1(R_{1,k}, R_{2,k}, \cdots, R_{k-1,k}, I_k, R_{k,k+1}, \cdots, R_{k,N})$, where $k = 1, 2, \cdots, N$.
3: **while** not converged and $s < s^{\max}$ **do**
4: Update $\mathcal{G}_k^{(s+1)}$ via (5).
5: Update $\mathcal{X}^{(s+1)}$ via (6).
6: Let $R = \min\{R + 1, R^{\max}\}$ and expand $\mathcal{G}_k^{(s+1)}$ if
$$\|\mathcal{X}^{(s+1)} - \mathcal{X}^{(s)}\|_F / \|\mathcal{X}^{(s)}\|_F < 10^{-2}.$$
7: Check the convergence condition:
$$\|\mathcal{X}^{(s+1)} - \mathcal{X}^{(s)}\|_F / \|\mathcal{X}^{(s)}\|_F < 10^{-5}.$$
8: Let $s = s + 1$.
9: **end while**
10: **Output:** The reconstructed tensor $\mathcal{X}$.

The whole process of the PAM-based solver for the FCTN-TC model is summarized in Algorithm 1. Especially, if the observed tensor $\mathcal{F}$ is complete (no missing elements), the way for iteratively solving the factors $\mathcal{G}_k$ in the proposed FCTN-TC method can be regarded as a strategy for obtaining its one FCTN decomposition.

Computational Complexity Analysis

For an Nth-order incomplete tensor $\mathcal{F} \in \mathbb{R}^{I \times I \times \cdots \times I}$, we analyze the computational complexity of the proposed FCTN-TC method by simply setting the FCTN-ranks R_{k_1,k_2} $(1 \le k_1 < k_2 \le N$ and $k_1, k_2 \in \mathbb{N}^+)$ as the same value R. The computational cost lies on two part: 1) updating $\mathcal{G}_k$ $(k = 1, 2, \cdots, N)$ and 2) updating $\mathcal{X}$. In (5), updating $\mathcal{G}_k$ involves the FCTN composition, the matrix multiplication, and the matrix inversion, which costs $\mathcal{O}\big(N\sum_{k=2}^N I^k R^{k(N-k)+k-1} + NI^{N-1}R^{2(N-1)} + NR^{3(N-1)}\big)$. In (6), updating $\mathcal{X}$ requires the FCTN composition costing $\mathcal{O}\big(\sum_{k=2}^N I^k R^{k(N-k)+k-1}\big)$. Therefore, the whole computational complexity at each iteration in the Algorithm 1 is $\mathcal{O}\big(N\sum_{k=2}^N I^k R^{k(N-k)+k-1} + NI^{N-1}R^{2(N-1)} + NR^{3(N-1)}\big)$.

Convergence Analysis

In this section, we provide a theoretical guarantee for the convergence of the developed PAM-based algorithm.

Theorem 5 *The sequence $\{\mathcal{G}^{(s)}, \mathcal{X}^{(s)}\}_{s\in\mathbb{N}}$ obtained by the Algorithm 1 globally converges to a critical point of (2).*

To prove the Theorem 5, we only need to justify that the following four conditions hold (Attouch, Bolte, and Svaiter 2013):

1) $\mathcal{G}^{(s)}$ and $\mathcal{X}^{(s)}$ $(s \in \mathbb{N})$ are bounded;

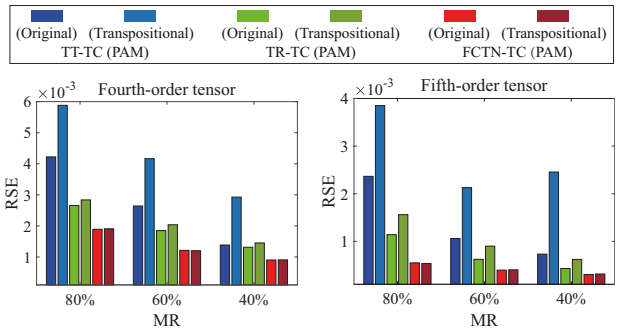


Figure 2: Reconstructed results on the synthetic dataset with different permutations.

2) $f(\mathcal{G},\mathcal{X})$ is a proper lower semi-continuous function;
3) $f(\mathcal{G},\mathcal{X})$ satisfies the K-Ł property at $\{\mathcal{G}^{(s)},\mathcal{X}^{(s)}\}_{s\in\mathbb{N}}$;
4) $\{\mathcal{G}^{(s)},\mathcal{X}^{(s)}\}_{s\in\mathbb{N}}$ satisfies Lemmas 1 and 2.

Lemma 1 (Sufficient Decrease) *Letting* $\{\mathcal{G}^{(s)},\mathcal{X}^{(s)}\}_{s\in\mathbb{N}}$ *be the sequence obtained by the Algorithm 1, then it satisfies*

$$f(\mathcal{G}^{(s+1)}_{1:k},\mathcal{G}^{(s)}_{k+1:N},\mathcal{X}^{(s)})+\frac{\rho}{2}\|\mathcal{G}^{(s+1)}_k-\mathcal{G}^{(s)}_k\|_F^2 \le f(\mathcal{G}^{(s+1)}_{1:k-1},\mathcal{G}^{(s)}_{k:N},\mathcal{X}^{(s)}),\ k=1,2,\cdots,N;$$
$$f(\mathcal{G}^{(s+1)},\mathcal{X}^{(s+1)})+\frac{\rho}{2}\|\mathcal{X}^{(s+1)}-\mathcal{X}^{(s)}\|_F^2\le f(\mathcal{G}^{(s+1)},\mathcal{X}^{(s)}).$$

Lemma 2 (Relative Error) *Letting* $\{\mathcal{G}^{(s)},\mathcal{X}^{(s)}\}_{s\in\mathbb{N}}$ *be the sequence obtained by the Algorithm 1, then there exists* $\mathcal{A}^{(s+1)}_k\in\mathbf{0}$ *and* $\mathcal{A}^{(s+1)}\in\partial_{\mathcal{X}}\iota_{\mathbb{S}}(\mathcal{X}^{(s+1)})$ *satisfied*

$$\|\mathcal{A}^{(s+1)}_k+\nabla_{\mathcal{G}_k}h(\mathcal{G}^{(s+1)}_{1:k},\mathcal{G}^{(s)}_{k+1:N},\mathcal{X}^{(s)})\|_F \le \rho\|\mathcal{G}^{(s+1)}_k-\mathcal{G}^{(s)}_k\|_F,\ k=1,2,\cdots,N;$$
$$\|\mathcal{A}^{(s+1)}+\nabla_{\mathcal{X}}h(\mathcal{G}^{(s+1)},\mathcal{X}^{(s+1)})\|_F\le\rho\|\mathcal{X}^{(s+1)}-\mathcal{X}^{(s)}\|_F,$$

where $h(\mathcal{G},\mathcal{X})=\frac{1}{2}\|\mathcal{X}-FCTN(\{\mathcal{G}_k\}_{k=1}^N)\|_F^2$.

The detailed proof for the above four conditions is presented in the supplementary materials.

Numerical Experiments

We test the performance of the proposed FCTN-TC method[3] by conducting synthetic data and real data experiments. The missing ratio (MR) is defined as the ratio of the number of missing elements and the total elements.

Synthetic Data Experiments

This section mainly aims to verify the superiorities of the proposed FCTN decomposition over the TT and TR decompositions by contrasting the performance of their corresponding TC methods, i.e., FCTN-TC, TT-TC, and TR-TC. All methods are solved by PAM to get rid of the influence of the algorithm. Since TT decomposition is a special case of TR and FCTN decompositions, we generate synthetic tensors by the TT composition of TT factors, sampled from uniform distribution $U(0,1)$, for the sake of fairness. The testing data includes a fourth-order tensor of size $20\times20\times20\times20$ and a fifth-order tensor of size $12\times12\times12\times12\times12$, whose TT-ranks are $(10,80,10)$ and $(8,48,48,8)$, respectively. Besides, for each data, we test two permutations, i.e., its original ordering and one generalized transposition ($\mathbf{n}$ are $(2,4,1,3)$ and $(2,4,1,3,5)$ for fourth-order and fifth-order tensors, respectively). The relative error between the reconstructed tensor and the ground truth, i.e., RSE $=\|\mathcal{X}-\mathcal{X}_{\text{true}}\|_F/\|\mathcal{X}_{\text{true}}\|_F$, is adopted to evaluate the performance of different methods. The methodology for selecting the missing elements is purely random sampling. The hyper-parameters needed to adjust in the proposed FCTN-TC method only involve the maximal FCTN-ranks, i.e., $R^{\max}_{k_1,k_2}$ $(1\le k_1<k_2\le N$ and $k_1,k_2\in\mathbb{N}^+)$, which are simply set as the same value.

[3]The code is available at *https://yubangzheng.github.io*.

Dataset	MR	95%	90%	80%	Mean time (s)
news	Observed	8.7149	8.9503	9.4607	—
	HaLRTC	14.490	18.507	22.460	36.738
	TMac	25.092	27.035	29.778	911.14
	t-SVD	25.070	28.130	31.402	74.807
	TMacTT	24.699	27.492	31.546	465.75
	TRLRF	22.558	27.823	31.447	891.96
	FCTN-TC	**26.392**	**29.523**	**33.048**	473.50
containe	Observed	4.5969	4.8315	5.3421	—
	HaLRTC	18.617	21.556	25.191	34.528
	TMac	26.941	26.142	32.533	1224.4
	t-SVD	28.814	34.912	39.722	71.510
	TMacTT	28.139	31.282	37.088	450.70
	TRLRF	30.631	32.512	38.324	640.41
	FCTN-TC	**30.805**	**37.326**	**42.974**	412.72
elephants	Observed	3.8499	4.0847	4.5946	—
	HaLRTC	16.651	20.334	24.813	38.541
	TMac	26.753	28.648	31.010	500.70
	t-SVD	21.810	27.252	30.975	63.994
	TMacTT	25.918	28.880	32.232	204.64
	TRLRF	27.120	28.361	32.133	592.13
	FCTN-TC	**27.780**	**30.835**	**34.391**	455.71
bunny	Observed	6.4291	6.6638	7.1736	—
	HaLRTC	14.561	19.128	23.396	32.882
	TMac	25.464	28.169	30.525	779.78
	t-SVD	21.552	26.094	30.344	66.294
	TMacTT	26.252	29.512	33.096	264.15
	TRLRF	27.749	29.034	33.224	652.03
	FCTN-TC	**28.337**	**32.230**	**36.135**	468.25
HSV	Observed	8.4157	8.6506	9.1594	—
	HaLRTC	12.008	22.140	29.000	15.456
	TMac	31.151	36.749	41.663	93.065
	t-SVD	35.328	40.379	45.801	30.533
	TMacTT	36.478	41.561	46.581	131.31
	TRLRF	37.352	41.796	46.211	4776.5
	FCTN-TC	**43.256**	**48.175**	**52.701**	501.03

Table 1: The PSNR values and the running times of all utilized methods on the CV and HSV datasets.

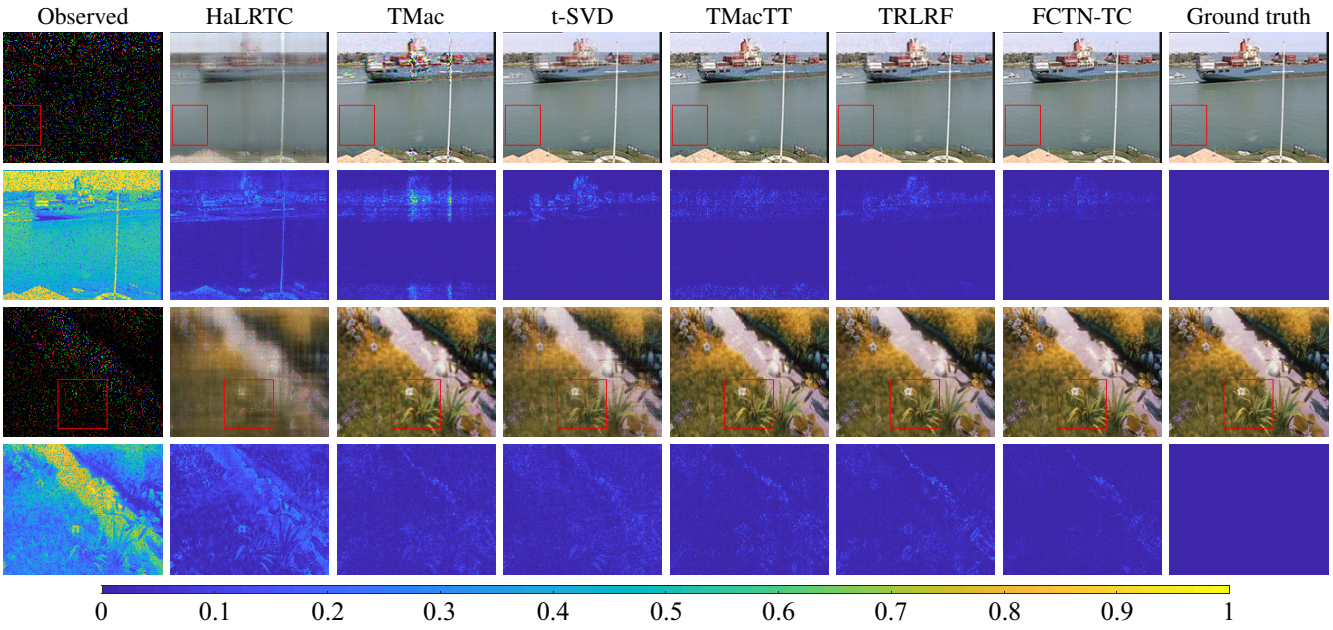


Figure 3: Reconstructed results on two testing CVs with MR=90%. From top to bottom: the odd-numbered rows are the visual results at the 1st frame of the CV *containe* and the 35th frame of the CV *bunny*, respectively; the even-numbered rows are the corresponding residual images average over three color channels.

Figure 2 shows the reconstructed results of different methods on the synthetic data under different MRs, where the RSE value under each case is averaged over the values obtained by 50 independent tests. We observe that 1) the performance of the proposed FCTN-TC method is pronouncedly robust to different permutations, while that of TT-TC and TR-TC are sensitive; and 2) the proposed FCTN-TC method always achieves the lowest RSE values among three methods under different datasets, MRs, and permutations. This is because the rearranging of tensor modes shifts the correlation among them (e.g., the correlation between the first and second modes shifts to the first and third modes), leading to the change in the performance of TT and TR decomposition-based methods. But owing that the proposed FCTN decomposition can characterize the correlations between any two modes and is essentially invariable for tensor transposition, the FCTN-TC method obtains the best and robust results. These testing results provide empirical evidence for the fore theoretical analysis regarding the superiorities of the FCTN decomposition.

Real Data Experiments

This section mainly aims to test the performance of the proposed FCTN-TC method on two types of real data by contrasting it with the state-of-the-art methods based on different tensor decompositions, including HaLRTC (Liu et al. 2013), TMac (Xu et al. 2015), t-SVD (Zhang and Aeron 2017), TMacTT (Bengua et al. 2017), and TRLRF (Yuan et al. 2019). All hyper-parameters involved in all compared methods are manually adjusted to achieve optimal performance following the authors' recommendations and codes. For instance, In TMac, TMacTT, and TRLRF, the hyper-parameters are mainly Tucker-rank, TT-rank, and TR-rank. We adjust them in a certain range. In HaLRTC and t-SVD, the hyper-parameters are mainly the threshold for the singular value thresholding operation, which is selected from the candidate set $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$.

Video Data Completion. The testing video dataset includes four color videos[4] (CVs) of size $144 \times 176 \times 3 \times 50$ (spatial height$\times$spatial width$\times$color channel$\times$frame) and a hyperspectral video[5] (HSV) of size $60 \times 60 \times 20 \times 20$ (spatial height$\times$spatial width$\times$band$\times$frame) (Mian and Hartley 2012). For each data, we test three MRs: 80%, 90%, and 95%, and employ the peak signal-to-noise ratio (PSNR) as the quantitative metric. The methodology for selecting the missing elements is purely random sampling and the pixel values of the testing data are normalized into [0, 1].

For the proposed FCTN-TC method, on CV dataset, we set $R_{1,4}^{\max}$ and $R_{2,4}^{\max}$ as the same value since they all directly characterize the correlation between the spatial modes (height and width, respectively) and the temporal mode. And we set $R_{1,3}^{\max}$, $R_{2,3}^{\max}$, and $R_{3,4}^{\max}$ as the same value, since the third mode represents the color channel. Therefore, only $R_{1,2}^{\max}$, $R_{1,3}^{\max}$, and $R_{1,4}^{\max}$ need to adjust, which are recommended to select from the candidate sets $\{10, 15, 20, 25, 30, 35, 40, 45, 50\}$, $\{2, 3\}$, and $\{4, 5, 6, 7, 8\}$, respectively. On the HSV dataset, we simply set $R_{k_1,k_2}^{\max}$ ($1 \leq k_1 < k_2 \leq 4$ and $k_1, k_2 \in \mathbb{N}^+$) as the same value recommended to select from the candidate set $\{4, 5, 6, 7, 8\}$.

[4]The data is available at *http://trace.eas.asu.edu/yuv/*.

[5]The data is available at *http://openremotesensing.net/kb/data/*.

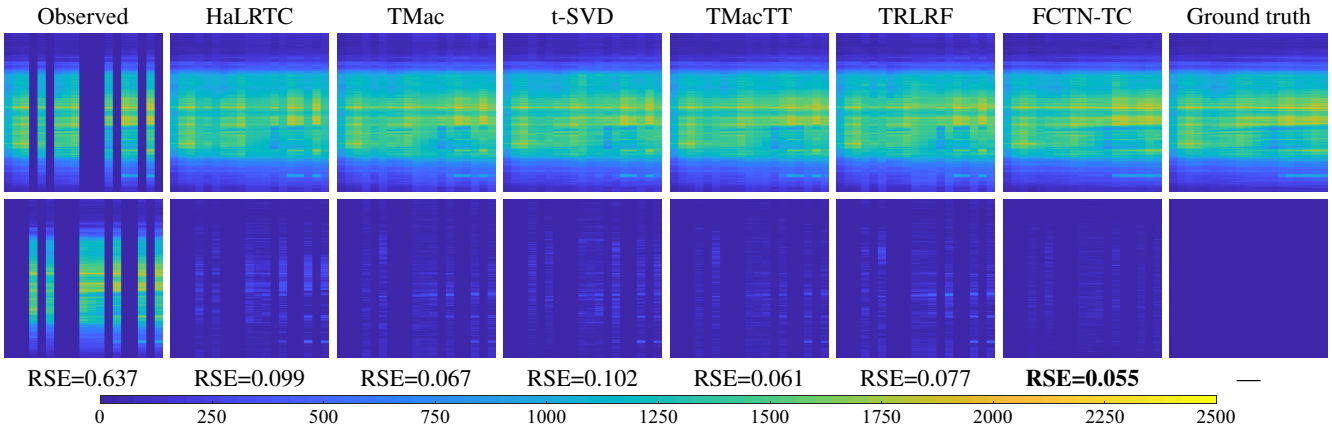


Figure 4: Reconstructed results on the traffic flow dataset with MR=40%. The first and the second rows are the results on the 2nd day and the corresponding residual results, respectively.

Table 1 reports the PSNR values and the running times of all utilized methods under different MRs on the testing CVs and HSV. In terms of effect (PSNR), we see that the proposed FCTN-TC method compares favorably to the other methods on both the testing CVs and HSV. Especially on the HSV, the FCTN-TC method is ahead by about 5dB of PSNR compared to the second-best method. In terms of efficiency (running time), we observe that on the CV dataset, the mean of the running time of the FCTN-TC method stays roughly on the same order of magnitude with that of the compared decomposition-based methods, i.e., TMac, TMacTT, and TRLRF. The main reason is that although the computational complexity of the FCTN-TC method is theoretically higher, the setting rank of it for obtaining the optimal result is usually lower than that of the compared ones.

Furthermore, Figure 3 shows the reconstructed spatial images and their corresponding residual images (the absolute difference between the reconstructed image and the ground truth) of two CVs. From Figure 3, we observe that the results obtained by the proposed FCTN-TC method are markedly superior to those by the compared ones, especially for the recovery of local details, such as ripples in the CV *containe* and grasses in the CV *bunny*.

Traffic Data Completion. This experiment was realized with traffic data provided by the NeCS team from the Grenoble Traffic Lab (GTL)[6]. The testing traffic flow dataset is collected from 19 road segments within 31 days from January 1, 2019, to January 31, 2019, and the time interval is 2 minutes. The size of it is $30 \times 24 \times 31 \times 19$ (minute$\times$hour$\times$day$\times$segment). In the traffic dataset, the damage to detectors usually leads to the data missing over some time. Therefore, we consider the slice (made up of the first and second modes) missing problem. The MR is set to be 40% and the RSE is employed as the quantitative metric. We simply set $R^{\max}_{k_1,k_2}$ ($1 \leq k_1 < k_2 \leq 4$ and $k_1, k_2 \in \mathbb{N}^+$) as the same value, i.e., 5.

In Figure 4, we present the reconstructed results on the 2nd day and the RSE value of the whole dataset. As observed, the proposed FCTN-TC method achieves the best approximation to the ground truth among different methods. This illustrates the superior of the proposed FCTN decomposition on the slice missing problem.

In summary, the above experimental results imply the superior capability of the proposed FCTN decomposition for capturing the intrinsic information of higher-order tensors, as compared to other tensor decompositions.

Conclusion

This paper proposed a novel tensor decomposition scheme, i.e., the FCTN decomposition, which factorizes an Nth-order tensor into a set of Nth-order factors with full connections. The FCTN decomposition showed its outstanding capability to adequately characterize the correlations between any two modes of tensors and was proved to be essentially transpositional invariable. Meanwhile, an FCTN-TC model was proposed with an efficient PAM-based solver, whose convergence was theoretically guaranteed. Experimental results demonstrated that the FCTN-TC method delivered an overall better performance than the compared ones.

Similar to CP and TR decompositions (Kolda and Bader 2009; Zhao et al. 2016), the proposed FCTN decomposition faces difficulty in finding the optimal FCTN-ranks. In the future, we will work to solve this challenging problem, deeply explore the essential feature of the FCTN factors, and apply the FCTN decomposition to more applications.

Acknowledgments

This work was supported by NSFC (Nos. 61772003, 61876203, 62071132, 12001446), Key Project of Applied Basic Research in Sichuan Province (No. 2020YJ0216), Project of Applied Basic Research in Sichuan Province (No. 21YYJC3042), National Key Research and Development Program of China (No. 2020YFA0714001), and JSPS KAKENHI (No. 20H04249). The authors thank the CNRS/INRIA NeCS team for providing traffic data from the GTL.

[6]Homepage: *http://gtl.inrialpes.fr/*.

References

Anandkumar, A.; Ge, R.; Hsu, D.; Kakade, S. M.; and Telgarsky, M. 2014. Tensor Decompositions for Learning Latent Variable Models. *Journal of Machine Learning Research* 15(80): 2773–2832.

Attouch, H.; Bolte, J.; and Svaiter, B. F. 2013. Convergence of Descent Methods for Semi-Algebraic and Tame Problems: Proximal Algorithms, Forward-Backward Splitting, and Regularized Gauss-Seidel Methods. *Mathematical Programming* 137(1): 91–129.

Bengua, J. A.; Phien, H. N.; Tuan, H. D.; and Do, M. N. 2017. Efficient Tensor Completion for Color Image and Video Recovery: Low-Rank Tensor Train. *IEEE Transactions on Image Processing* 26(5): 2466–2479.

Chen, Y.; Huang, T.-Z.; He, W.; Yokoya, N.; and Zhao, X.-L. 2020. Hyperspectral Image Compressive Sensing Reconstruction Using Subspace-Based Nonlocal Tensor Ring Decomposition. *IEEE Transactions on Image Processing* 29: 6813–6828.

Cong, F.; Lin, Q.-H.; Kuang, L.-D.; Gong, X.-F.; Astikainen, P.; and Ristaniemi, T. 2015. Tensor Decomposition of EEG Signals: A Brief Review. *Journal of Neuroscience Methods* 248: 59–69.

Ding, M.; Huang, T.-Z.; Ji, T.-Y.; Ji, T.-Y.; Zhao, X.-L.; and Yang, J.-H. 2019. Low-Rank Tensor Completion Using Matrix Factorization Based on Tensor Train Rank and Total Variation. *Journal of Scientific Computing* 81(2): 941–964.

Gandy, S.; Recht, B.; and Yamada, I. 2011. Tensor Completion and Low-n-Rank Tensor Recovery via Convex Optimization. *Inverse Problems* 27(2): 025010.

He, W.; Yao, Q.; Li, C.; Yokoya, N.; and Zhao, Q. 2019. Non-Local Meets Global: An Integrated Paradigm for Hyperspectral Denoising. In *IEEE Conference on Computer Vision and Pattern Recognition*, 6868–6877.

Imaizumi, M.; Maehara, T.; and Hayashi, K. 2017. On Tensor Train Rank Minimization : Statistical Efficiency and Scalable Algorithm. In *Advances in Neural Information Processing Systems*, 3930–3939.

Kolda, T. G.; and Bader, B. W. 2009. Tensor Decompositions and Applications. *SIAM Review* 51(3): 455–500.

Li, X.; Ye, Y.; and Xu, X. 2017. Low-Rank Tensor Completion with Total Variation for Visual Data Inpainting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2210–2216.

Liu, J.; Musialski, P.; Wonka, P.; and Ye, J. 2013. Tensor Completion for Estimating Missing Values in Visual Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(1): 208–220.

Liu, Y.; Shang, F.; Fan, W.; Cheng, J.; and Cheng, H. 2014. Generalized Higher-Order Orthogonal Iteration for Tensor Decomposition and Completion. In *Advances in Neural Information Processing Systems*, 1763–1771.

Lu, C.; Feng, J.; Chen, Y.; Liu, W.; Lin, Z.; and Yan, S. 2016. Tensor Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Tensors via Convex Optimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 5249–5257.

Mian, A.; and Hartley, R. 2012. Hyperspectral Video Restoration Using Optical Flow and Sparse Coding. *Optics Express* 20(10): 10658–10673.

Mu, C.; Huang, B.; Wright, J.; and Goldfarb, D. 2014. Square Deal: Lower Bounds and Improved Relaxations for Tensor Recovery. In *Proceedings of Machine Learning Research*, volume 32, 73–81.

Oseledets, I. V. 2011. Tensor-Train Decomposition. *SIAM Journal on Scientific Computing* 33(5): 2295–2317.

Phan, A.; Cichocki, A.; Oseledets, I.; Calvi, G. G.; Ahmadi-Asl, S.; and Mandic, D. P. 2020. Tensor Networks for Latent Variable Analysis: Higher Order Canonical Polyadic Decomposition. *IEEE Transactions on Neural Networks and Learning Systems* 31(6): 2174–2188.

Xie, Q.; Zhao, Q.; Meng, D.; and Xu, Z. 2018. Kronecker-Basis-Representation Based Tensor Sparsity and Its Applications to Tensor Recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40(8): 1888–1902.

Xu, Y.; Hao, R.; Yin, W.; and Su, Z. 2015. Parallel Matrix Factorization for Low-Rank Tensor Completion. *Inverse Problems and Imaging* 9(2): 601–624.

Yao, Q.; Kwok, J. T.; Wang, T.; and Liu, T.-Y. 2019. Large-Scale Low-Rank Matrix Learning with Nonconvex Regularizers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41(11): 2628–2643.

Yokota, T.; and Hontani, H. 2017. Simultaneous Visual Data Completion and Denoising Based on Tensor Rank and Total Variation Minimization and Its Primal-Dual Splitting Algorithm. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3843–3851.

Yokota, T.; Kawai, K.; Sakata, M.; Kimura, Y.; and Hontani, H. 2019. Dynamic PET Image Reconstruction Using Nonnegative Matrix Factorization Incorporated With Deep Image Prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Yokota, T.; Zhao, Q.; and Cichocki, A. 2016. Smooth PARAFAC Decomposition for Tensor Completion. *IEEE Transactions on Signal Processing* 64(20): 5423–5436.

Yuan, L.; Cao, J.; Zhao, X.; Wu, Q.; and Zhao, Q. 2018. Higher-Dimension Tensor Completion via Low-Rank Tensor Ring Decomposition. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 1071–1076.

Yuan, L.; Li, C.; Mandic, D.; Cao, J.; and Zhao, Q. 2019. Tensor Ring Decomposition with Rank Minimization on Latent Space: An Efficient Approach for Tensor Completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9151–9158.

Zhang, Z.; and Aeron, S. 2017. Exact Tensor Completion Using T-SVD. *IEEE Transactions on Signal Processing* 65(6): 1511–1526.

Zhao, Q.; Sugiyama, M.; Yuan, L.; and Cichocki, A. 2019. Learning Efficient Tensor Representations with Ring-Structured Networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 8608–8612.

Zhao, Q.; Zhang, L.; and Cichocki, A. 2015. Bayesian CP Factorization of Incomplete Tensors with Automatic Rank Determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(9): 1751–1763.

Zhao, Q.; Zhou, G.; Xie, S.; Zhang, L.; and Cichocki, A. 2016. Tensor Ring Decomposition. *arXiv preprint arXiv:1606.05535* .

Zhao, Q.; Zhou, G.; Zhang, L.; Cichocki, A.; and Amari, S. 2016. Bayesian Robust Tensor Factorization for Incomplete Multiway Data. *IEEE Transactions on Neural Networks and Learning Systems* 27(4): 736–748.

Zheng, Y.-B.; Huang, T.-Z.; Zhao, X.-L.; Jiang, T.-X.; Ma, T.-H.; and Ji, T.-Y. 2020. Mixed Noise Removal in Hyperspectral Image via Low-Fibered-Rank Regularization. *IEEE Transactions on Geoscience and Remote Sensing* 58(1): 734–749.