IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 7 June 2023, accepted 24 June 2023, date of publication 3 July 2023, date of current version 10 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3291744

RESEARCH ARTICLE

A New Model for Tensor Completion: Smooth Convolutional Tensor Factorization

HIROMU TAKAYAMA[1] AND TATSUYA YOKOTA[1,2], (Senior Member, IEEE)
[1]Nagoya Institute of Technology, Nagoya 466-8555, Japan
[2]RIKEN Center for Advanced Intelligence Project, Tokyo 103-0027, Japan

Corresponding author: Hiromu Takayama (umisaitozaki126@gmail.com)

This work was supported in part by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant 23H03419 and Grant 20H04249.

ABSTRACT Tensor completion is the problem of filling-in missing parts of multidimensional data using the values of the reference elements. Recently, Multiway Delay-embedding Transform (MDT), which considers a low-dimensional space in a delay-embedded space with high expressive capability, has attracted attention as a tensor completion method. Although MDT has a high complementary performance, its computational cost is considerably high. Therefore, we propose a new model called smooth convolutional tensor factorization (SCTF) for tensor completion based on a delay-embedded space. The proposed method is small in computational complexity because of its concise model of rank-1 decomposition in the delay-embedded space, and because it does not directly perform optimization in the delay-embedded space. In addition, a smooth constraint term is assigned to the factor tensors as a prior data structure in the optimization to improve the completion accuracy further. In our experiments, we completed clipped and random missing image data, and confirmed that the proposed method achieved high completion accuracy without high computational cost.

INDEX TERMS Tensor completion, convolution, tensor factorization, delay-embedded space.

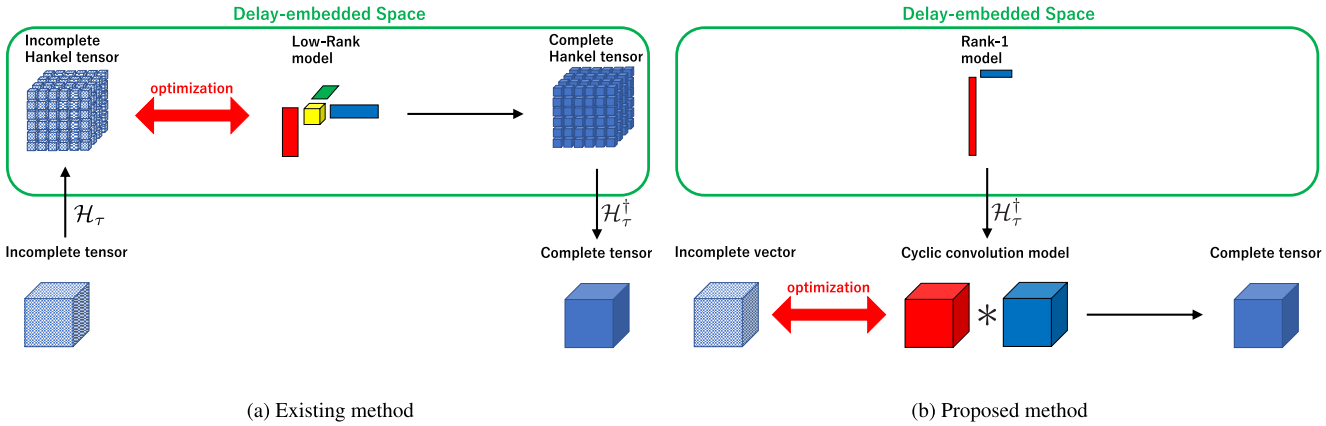
I. INTRODUCTION

Several real-world signals such as image and video data are multidimensional, and tensors are mathematical models that represent such signals [2], [8], [14], [20], [24], [27]. Such data, modeled as tensors, are often corrupted by measurement errors and missing observations. Tensor completion is the task of filling-in the missing values of the tensor data using the values of the reference elements [24], [37], [48], [52]. Because tensor completion is an ill-posed problem, we consider the prior structure in the target tensor to narrow down the solution set. The completion value should be appropriate as per the properties of the analyzed data, and the prior structure includes smoothness [15], [48], [53], nonnegativity [17], [39], [43], sparsity [23], low-rank [14], [24], etc.

Recently, low-rank tensor completion (LRTC) has attracted attention. Nuclear norm minimization is a popular low-rank model, which is a convex relaxation of rank minimization [14], [27], [31]. In some studies, this method has been applied to image and video completion [13], [24], [28]. Other important low-rank models are tensor decompositions, with CP decomposition and Tucker decomposition [4], [16], [40] as representative models for image completion [1], [43], [46], [48], [51]. Low-rank tensor models are often incorporated simultaneously with other prior structures. For example, non-negative tensor factorization (NTF) [6], [7] has been applied to various fields, such as sparse coding of images [35], traffic analysis [12], and EEG analysis [20]. Other prior structures include studies incorporating graph structures and sparse regularization to achieve a super-resolution in multispectral images [44], [49]. In the image completion subject of our study, methods that incorporate smoothness achieved highly accurate completion [15], [47], [48], [53].

Recently, tensor completion for considering low-rank structures in delay-embedded spaces has attracted attention [34], [46]. A delay-embedded space is a high-dimensional space that represents time delay. In particular, the embedding

The associate editor coordinating the review of this manuscript and approving it for publication was Yi Zhang.



(a) Existing method

(b) Proposed method

FIGURE 1. Comparison between the existing MDT-based and proposed convolution-based methods. (a) The existing method computes the optimization on the delay-embedded space. (b) Whereas, the proposed method computes the optimization in the original space, but implicitly considers the delay-emabedded space.

of tensor data is called Multiway Delay-embedding Transform (MDT), which is mathematically equivalent to multi-level Hankelization. MDT has been widely applied to the tensor completion of images and videos [25], [33], [34], [36], [37], [42], [45], [46]. MDT-Tucker [45], the original model of tensor completion using MDT, consists of the following steps:

1) Hankelization of the observed tensor by MDT.
2) Completion of the Hankelized tensor using Tucker decomposition.
3) Inverse MDT of the completed tensor.

This method considers a delay-embedded space with a high expressive capability and exhibits higher completion accuracy than existing methods [5], [24], [47], [48], [53]. However, MDT-Tucker has the disadvantages of considerable time requirement and space computational complexity. For example, for an Nth-order tensor of average size T, if the delay window size is τ, the space complexity is $\mathcal{O}(\tau^N T^N)$ and the time complexity is $\mathcal{O}(\tau^{N+1} T^N)$; thus, the complexity increases exponentially with order.

In this study, we propose a novel smooth convolutional tensor factorization (SCTF) model, which decomposes a tensor into two smooth factor tensors by convolution instead of a product. Figure 1 shows a schematic of the algorithm. This model implicitly implements tensor decomposition in the delay-embedded space, whereas optimization is performed in the original space. The model is based on the relationship between the inverse MDT of the rank-1 model and the cyclic convolution of the factor tensors. In addition, because it is a rank 1 model, the SCTF is simpler than the MDT-Tucker model, which considers the Tucker decomposition model. These properties are expected to reduce computational complexity. In addition, a smoothness constraint was imposed on the factor tensors to further narrow the solution set. Our contributions can be summarized as follows:

- We have mathematically proven that tensor decomposition based on inverse MDT has sufficient representation ability in rank-1 decomposition.
- Based on the relationship of inverse MDT of rank 1 decomposition and cyclic convolution of factor tensors and the introduction of smooth prior structure into factor tensors, we proposed a new tensor completion model named smooth convolutional tensor factorization (SCTF).
- We derived a solution method of the proposed SCTF with the Majorization-Minimization (MM) algorithm [18], [30], which is expected to provide a stable optimization in which the cost function decreases monotonically. Moreover, we exploit the equivalence of cyclic convolution in the time domain and Hadamard product in the frequency domain to reduce computation time.

The remainder of this paper is organized as follows: related works in Section II, a review of MDT in Section III, the proposed method in Section IV, experiments using the proposed method in Section V, and conclusions in Section VI.

A. MATHEMATICAL NOTATION

We follow the basic mathematical notation in [46]; that is, we use $\circledast$ for Hadamard product in this paper. In addition, we consider N matrices $\boldsymbol{U}_n \in \mathbb{R}^{I_n \times R_n}$ $(n = 1, \ldots, N)$ and an N-th order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{R_1 \times \cdots \times R_N}$. The *all-mode product* is denoted as

$$\boldsymbol{\mathcal{X}} \times \{\boldsymbol{U}\} := \boldsymbol{\mathcal{X}} \times_1 \boldsymbol{U}_1 \cdots \times_N \boldsymbol{U}_N. \tag{1}$$

II. RELATED WORKS

t-SVD [50] is a convolutional tensor decomposition method as well as the proposed method. It can achieve accurate tensor recovery based on group theory. t-SVD considers a new SVD for tensors by using some convolution, and the rank in t-SVD (tubal rank) is defined as the number of non-zero singular values. Since it is difficult to minimize the tubal rank directly, its convex relaxation is usually employed. The convex relaxation of tubal rank is given by the sum of singular values based on t-SVD, and it is called as the tensor

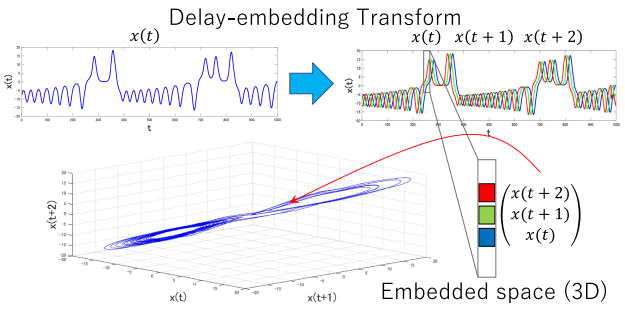


FIGURE 2. The figure shows the delay-embedded of signals generated from the Lorenz system. The embedded signal is smooth and low-dimensional in the delay-embedded space.

nuclear norm (TNN). Low-rank approximation in the t-SVD is substituted for a problem of minimizing TNN, which has the advantage of incorporating a global structure. TNN [50] is a typical model for tensor completion problems based on t-SVD, and PSTNN [19] is a further developed model. PSTNN suppresses the excessively low rank of the estimated tensor by considering partial sums of only small singular values in the tensor nuclear norm. RTF [9] and UTF [10] have also been proposed as models that avoid the high computational cost of these t-SVD models. RTF considers a factorization model of a low-rank tensor of small size and a dictionary (orthogonal) tensor. Since t-SVD is applied only to low-rank tensors of small size, the computational cost is lower than that of other t-SVD models. On the other hand, UTF uses the fact that the TNN is transformed to the minimum sum of the Frobenius norms of the two low-rank tensors so that the algorithm does not directly compute t-SVD in its calculation. UTF achieves very fast inference despite t-SVD model. However, these methods differ from the proposed method because it uses only the third-order tensor, and the convolution operation is performed only in the channel direction. Also, unlike the proposed model, these models do not have a smoothness term.

CNNM [22], [21] is a mathematical model of nuclear norm minimization of convolutional tensors applied to image completion and time series prediction. This research shows the equivalence of nuclear norm minimization of the convolution tensor and sparse approximation in Fourier space. However, the relationship between inverse MDT and cyclic convolution, and smoothness constraints is not discussed.

III. REVIEW OF MDT

This section summarizes the Multiway Delay-embedding Transform (MDT). Note that there are two types of MDT: noncyclic MDT [46] and cyclic MDT [45]. In this study, we consider a cyclic MDT. First, we discuss the Delay-embedding Transform (DT) for one-dimensional data (vectors), which is then extended to multidimensional data. Finally, we describe the drawbacks of MDT and a method, Fast-MDT-Tucker [45], for avoiding those.

A. DELAY-EMBEDDING TRANSFORM (DT)

1) OVERVIEW OF DT

A delay-embedding Transform (DT) is the transformation of data into a high-dimensional space representing a time delay. In physics, DT has been studied by reconstructing dynamic attractors from time-series data in a delay-embedded space [32]. Mathematically, the DT converts a vector into a Hankel matrix (Hankelization) [3]. When embedding an observed signal from the original space into a high-dimensional space, it is assumed that the signal is represented by a low-rank and smooth manifold in the delay-embedded space [26], [29], [41]. Figure 2 shows the results of DT of the signal generated by the Lorenz system, indicating that the transformed signal is smooth and low-dimensional in the delay-embedded space. Based on this assumption, a low-rank approximation of the Hankel matrix is used in the data analysis [3], [11].

2) MATHEMATICAL OPERATION OF DT

Following [45], the DT for an observation vector $\boldsymbol{x} = (x_1, \ldots, x_T)^{\mathrm{T}} \in \mathbb{R}^T$ with a delay window size τ is defined as

$$\boldsymbol{X} := \mathcal{H}_\tau(\boldsymbol{x}) = \begin{pmatrix} x_1 & x_2 & \cdots & x_{\tau-1} & x_\tau \\ x_2 & x_3 & \cdots & x_\tau & x_{\tau+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{T-1} & x_T & \ldots & x_{\tau-3} & x_{\tau-2} \\ x_T & x_1 & \ldots & x_{\tau-2} & x_{\tau-1} \end{pmatrix} \in \mathbb{R}^{T\times\tau}, \tag{2}$$

where the DT operation is denoted as $\mathcal{H}_\tau$. Because $\boldsymbol{X}$ is a Hankel matrix, DT is also called Hankelization. Each row of $\boldsymbol{X}$ is identical to the local window of vector $\boldsymbol{x}$. Notably, this study assumes that the signal is cyclic. Figure 3a shows a concrete example of a DT operation.

The DT can be considered a linear operation. By using the duplication matrix $\boldsymbol{S} \in \mathbb{R}^{T\tau\times T}$, we obtain

$$\boldsymbol{S}(i,j) = \begin{cases} 1 & j = (((i-1) \bmod T) + \lfloor (i-1)/T \rfloor) \bmod T + 1 \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

DT can be given by

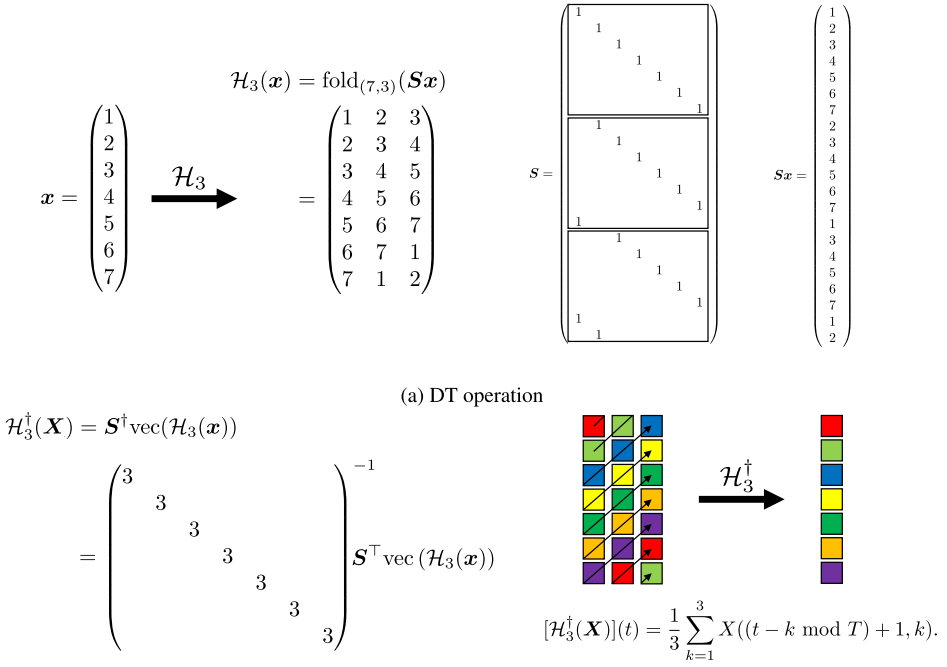
$$\mathcal{H}_\tau(\boldsymbol{x}) = \mathrm{fold}_{(T,\tau)}(\boldsymbol{S}\boldsymbol{x}), \tag{4}$$

where $\mathrm{fold}_{(V,v)}$: $\mathbb{R}^{Vv} \to \mathbb{R}^{V\times v}$ is a folding operator, that is reshaping from a vector to a matrix.

The pseudo-inverse of the DT can be expressed as the average of the anti-diagonal entries. Considering a matrix $\boldsymbol{X} \in \mathbb{R}^{T\times\tau}$, the inverse DT operation $\mathcal{H}_\tau^\dagger$ can be given by

$$\mathcal{H}_\tau^\dagger(\boldsymbol{X}) = \boldsymbol{S}^\dagger \mathrm{vec}(\boldsymbol{X}), \tag{5}$$

where $\boldsymbol{S}^\dagger := (\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S})^{-1}\boldsymbol{S}^{\mathrm{T}}$ is a Moore-Penrose pseudoinverse of $\boldsymbol{S}$, and $\mathrm{vec}(\cdot)$ represents an operation of vectorization. We note



(a) DT operation

(b) inverse DT operation

FIGURE 3. Matrix computation of DT operation and inverse DT operation. In particular, the computation of the pseudo-inverse matrix in the inverse DT corresponds to the average of anti-diagonal elements of the matrix.

that $(\boldsymbol{S}^{\mathrm{T}}\boldsymbol{S})^{-1} = \frac{1}{\tau}\boldsymbol{I}$. The tth element of the inverse DT is given by

$$[\mathcal{H}_\tau^\dagger(\boldsymbol{X})](t) = [\boldsymbol{S}^\dagger \mathrm{vec}(\boldsymbol{X})](t) = \frac{1}{\tau}\sum_{k=1}^{\tau} X((t-k \bmod T)+1, k). \quad (6)$$

Figure 3 illustrates the DT and inverse DT matrix computations.

B. MULTIWAY DELAY-EMBEDDING TRANSFORM (MDT)

1) OVERVIEW OF MDT

A Multiway Delay-embedding transform (MDT) is an embedding transform for tensor data of two or more orders. Tensor decomposition based on the MDT has various applications, particularly for tensor completion. A representative model is MDT-Tucker, which considers the Tucker decomposition model of the Hankel tensor and has been applied to image completion [46] and time-series data [36]. Another model, the HT-RPCA, was proposed in [42]. Unlike general RPCA, HT-RPCA solves the rank minimization of the tensor Hankelized by MDT, instead of the rank minimization of the matrix. This method enables anomaly detection by considering the time series. Furthermore, the TT and TR decomposition models of the Hankel tensor have been proposed and applied to image completion and time-series data, [33] [34].

2) MATHEMATICAL OPERATION OF MDT

The DT can be naturally extended to an N-th order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{T_1 \times \cdots \times T_N}$ of size $\boldsymbol{T} = (T_1, \ldots, T_N) \in \mathbb{R}^N$. Let us consider N duplication matrices $\boldsymbol{S}_n \in \{0,1\}^{T_n \tau_n \times T_n}$ $(n = 1, \ldots, N)$ with a window size $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_N) \in \mathbb{R}^N$ (see Equations (3)). The MDT is defined using an all-mode product and folding as follows:

$$\mathcal{H}_\tau(\boldsymbol{\mathcal{X}}) := \mathrm{fold}_{(\boldsymbol{T},\boldsymbol{\tau})}(\boldsymbol{\mathcal{X}} \times \{\boldsymbol{S}\}), \quad (7)$$

where $\mathrm{fold}_{(\boldsymbol{V},\boldsymbol{v})} : \mathbb{R}^{V_1 v_1 \times \cdots \times V_N v_N} \to \mathbb{R}^{V_1 \times v_1 \times \cdots \times V_N \times v_N}$ is the folding operator from an N-th order tensor to a $2N$-th order tensor. Conversely, the inverse MDT is defined as

$$\mathcal{H}_\tau^\dagger(\boldsymbol{\mathcal{X}}) := \mathrm{unfold}_{(\boldsymbol{T},\boldsymbol{\tau})}(\boldsymbol{\mathcal{X}}) \times \{\boldsymbol{S}^\dagger\}, \quad (8)$$

where $\mathrm{unfold}_{(\boldsymbol{V},\boldsymbol{v})} : \mathbb{R}^{V_1 \times v_1 \times \cdots \times V_N \times v_N} \to \mathbb{R}^{V_1 v_1 \times \cdots \times V_N v_N}$ is an unfolding operator from the $2N$-th order tensor of an N-th order tensor.

C. FAST-MDT-TUCKER

The methods introduced in Section III-B1 have the disadvantage of considerable time requirement and space computational complexity because of the Hankelization in each mode of the tensor. Fast-MDT-Tucker [45] was proposed to improve the high computational complexity of MDT-Tucker. This method focuses on the redundant structure of the Hankel matrix and improves the time complexity to $\mathcal{O}(NT^N \log NT)$ and the space complexity to $\mathcal{O}(T^N)$ using two techniques:

1) Omission of duplicate computations.
2) Equivalence of cyclic convolution in the time domain and Hadamard product in the frequency domain.

In 2), the Fast-MDT-Tucker exploits the relationship between the inverse MDT and cyclic convolution. Fast-MDT-Tucker provides a fast and accurate completion; however, only low-rank priors are available.

The proposed method is also an algorithm based on the relationship between the inverse MDT and cyclic convolution and similarly avoids the issues of MDT. Note that the low-rank model of the proposed method is not a Tucker decomposition but a rank-1 decomposition. In addition, the proposed method imposes a smoothness constraint on the factor tensors.

IV. PROPOSED METHOD

The proposed method solves the optimization problem by assuming that the observation tensor can be represented by a cyclic convolution of two smooth factors of the same size (See Figure 1). We describe the key theory behind the proposed method in Section IV-A, the smoothness constraints in Section IV-B, and the formulation and algorithm in Section IV-C.

A. KEY THEORY OF PROPOSED METHOD

1) RELATIONSHIP BETWEEN THE INVERSE DT AND CYCLIC CONVOLUTION

Any rank-R matrix $\boldsymbol{X} \in \mathbb{R}^{T\times\tau}$ has a singular value decomposition that can be expressed as

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} = \sum_{r=1}^{R} \sigma_r \boldsymbol{u}_r \boldsymbol{v}_r^{\mathrm{T}}, \tag{9}$$

where $\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_R] \in \mathbb{R}^{T\times R}$ and $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_R] \in \mathbb{R}^{\tau\times R}$ are orthonormal matrices and $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_R) \in \mathbb{R}^{R\times R}$ is a diagonal matrix. Because the inverse DT is a linear operation, $\mathcal{H}_\tau^\dagger(\boldsymbol{X})$ can be separated into rank-1 bases:

$$\mathcal{H}_\tau^\dagger(\boldsymbol{X}) = \sum_{r=1}^{R} \sigma_r \mathcal{H}_\tau^\dagger(\boldsymbol{u}_r \boldsymbol{v}_r^{\mathrm{T}}). \tag{10}$$

From Equations (6) and (10), the tth element of the inverse DT for a single basis $\boldsymbol{u}_r\boldsymbol{v}_r^{\mathrm{T}}$ is given by

$$[\mathcal{H}_\tau^\dagger(\boldsymbol{u}_r \boldsymbol{v}_r^{\mathrm{T}})](t) = \frac{1}{\tau}\sum_{k=1}^{\tau} u_r((t-k \bmod T)+1)v_r(k). \tag{11}$$

Now, let us consider the matrix $\boldsymbol{P} = (\boldsymbol{I}_\tau\ \boldsymbol{O})^{\mathrm{T}} \in \mathbb{R}^{T\times\tau}$ and set $\boldsymbol{v} \in \mathbb{R}^\tau$ to be the same vector as the dimension of $\boldsymbol{u} \in \mathbb{R}^T$, i.e., zero padding operation is given by

$$\tilde{\boldsymbol{v}} := \boldsymbol{P}\boldsymbol{v} = [v(1), v(2), \cdots v(\tau), \underbrace{0, \cdots, 0}_{T-\tau}]^{\mathrm{T}} \in \mathbb{R}^T. \tag{12}$$

Note that the sizes of $\boldsymbol{u}$ and $\tilde{\boldsymbol{v}}$ are equal and the elements of $\tilde{\boldsymbol{v}}$ are zero after the size of the delay window τ. From Equations (11) and (12), the inverse DT of the rank-1 basis $\boldsymbol{u}\boldsymbol{v}^{\mathrm{T}}$ is given by

$$\begin{aligned}[\mathcal{H}_\tau^\dagger(\boldsymbol{u}_r \boldsymbol{v}_r^{\mathrm{T}})](t) &= \frac{1}{\tau}\sum_{k=1}^{T} u_r((t-k \bmod T)+1)\tilde{v}_r(k) \\ &= \frac{1}{\tau}[\boldsymbol{u}_r * \tilde{\boldsymbol{v}}_r](t),\end{aligned} \tag{13}$$

where $*$ denotes a cyclic convolution operation. From Equation (13), the inverse DT of the rank-1 basis can be formulated in terms of a cyclic convolution. Eventually, from Equations (10) and (13), the inverse DT of $\boldsymbol{X}$ is

$$\mathcal{H}_\tau^\dagger(\boldsymbol{X}) = \frac{1}{\tau}\sum_{r=1}^{R} \sigma_r \boldsymbol{u}_r * \tilde{\boldsymbol{v}}_r. \tag{14}$$

2) SUFFICIENT REPRESENTATION ABILITY EVEN WITH A RANK-1 MATRIX

We now discuss the rank-1 representation of $\boldsymbol{X}$. From Equation (14), the rank $\boldsymbol{X}$ denotes the number of convolutional bases. The degrees of freedom of each convolutional basis determine the representational ability of the model. In this study, we consider $\boldsymbol{X}$ to be rank-1 and show that it has sufficient representation ability for vector reconstruction. Rank-1 matrix model $\boldsymbol{X} = \boldsymbol{u}\boldsymbol{v}^{\mathrm{T}} \in \mathbb{R}^{T\times\tau}$ can generate any $\boldsymbol{x} \in \mathbb{R}^T$. Let us put

$$\boldsymbol{u} = \begin{bmatrix}\boldsymbol{x}\end{bmatrix}, \quad \boldsymbol{v} = \begin{bmatrix}\tau \\ \boldsymbol{0}_{\tau-1}\end{bmatrix}, \tag{15}$$

where $\boldsymbol{0}_{\tau-1}$ is a $(\tau-1)$-dimensional vector of zeros and we have

$$\mathcal{H}_\tau^\dagger(\boldsymbol{u}\boldsymbol{v}^{\mathrm{T}}) = \mathcal{H}_\tau^\dagger\left(\begin{bmatrix}\tau\boldsymbol{x} & \boldsymbol{0}_{T,\tau-1}\end{bmatrix}\right) = \boldsymbol{x}. \tag{16}$$

This suggests that the inverse DT of a matrix, even rank-1, is over-parameterized and does not work as a model. Therefore, $\boldsymbol{u}$ and $\boldsymbol{v}$ must impose constraints to narrow the solution.

3) EXTENTION TO MDT

The properties of DT discussed in Sections IV-A1 and IV-A2 can be applied to the MDT. First, we show the relationship between the inverse MDT and N-dimensional cyclic convolution. Let us consider factor tensors $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{T_1\times\cdots\times T_n}$, $\boldsymbol{\mathcal{B}} \in \mathbb{R}^{\tau_1\times\cdots\times\tau_n}$, and we define $\boldsymbol{a} := \mathrm{vec}(\boldsymbol{\mathcal{A}}) \in \mathbb{R}^{\prod_n T_n}$, $\boldsymbol{b} := \mathrm{vec}(\boldsymbol{\mathcal{B}}) \in \mathbb{R}^{\prod_n \tau_n}$. We assume $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{T_1\times\tau_1\times\cdots\times T_N\times\tau_N}$ is given by

$$\begin{aligned}\mathrm{bunfold}_{(\boldsymbol{T},\boldsymbol{\tau})}(\boldsymbol{\mathcal{X}}) &= \mathrm{vec}(\boldsymbol{\mathcal{A}})\mathrm{vec}(\boldsymbol{\mathcal{B}})^{\mathrm{T}} \\ &= \boldsymbol{a}\boldsymbol{b}^{\mathrm{T}} \in \mathbb{R}^{\prod_n T_n\times\prod_n \tau_n},\end{aligned} \tag{17}$$

where $\mathrm{bunfold}_{(\boldsymbol{V},\boldsymbol{v})} : \mathbb{R}^{V_1\times v_1\times\cdots\times V_N\times v_N} \to \mathbb{R}^{\prod_n V_n\times\prod_n v_n}$ is the unfolding operator from an $2N$-th order tensor to the block matrix. We also define $\mathrm{bfold}_{(\boldsymbol{V},\boldsymbol{v})} : \mathbb{R}^{\prod_n V_n\times\prod_n v_n} \to \mathbb{R}^{V_1\times v_1\times\cdots\times V_N\times v_N}$ as the inverse transform of $\mathrm{bunfold}_{(\boldsymbol{V},\boldsymbol{v})}$. Using the zero padding matrix $\boldsymbol{P}_n = (\boldsymbol{I}_{\tau_n}\ \boldsymbol{O})^{\mathrm{T}} \in \{0,1\}^{T_n\times\tau_n}$ $(n = 1, \ldots, N)$, we define a tensor $\tilde{\boldsymbol{\mathcal{B}}} = \boldsymbol{\mathcal{B}} \times$

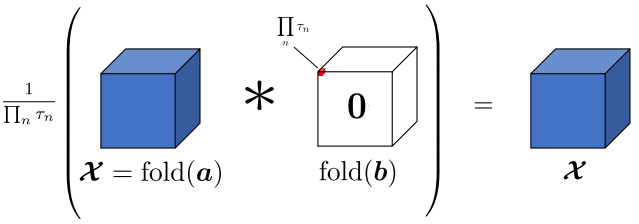


FIGURE 4. Cyclic convolution of a tensor $\mathcal{X}$ with only one element fold(v) corresponds to the Equation (20). This operation is identically derived from the tensor $\mathcal{X}$.

$\{\boldsymbol{P}\} \in \mathbb{R}^{T_1 \times \cdots \times T_n}$ of the same size as $\mathcal{A}$. The inverse MDT of $\mathcal{X} = \text{bfold}_{(\boldsymbol{\tau},\boldsymbol{T})}(\boldsymbol{a}\boldsymbol{b}^{\mathrm{T}})$ is derived by

$$\begin{aligned}
&\left[\mathcal{H}_{\tau}^{\dagger}(\text{bfold}_{(\boldsymbol{\tau},\boldsymbol{T})}(\boldsymbol{a}\boldsymbol{b}^{\mathrm{T}}))\right](t_1,\ldots,t_N) \\
&= \frac{1}{\prod_n \tau_n} \sum_{k_1=0}^{\tau_1-1} \cdots \sum_{k_N=0}^{\tau_N-1} \\
&\quad \mathcal{A}(t_1 - k_1 \bmod T_1, \ldots, t_N - k_N \bmod T_N)\mathcal{B}(k_1,\ldots,k_N) \\
&= \frac{1}{\prod_n \tau_n} \sum_{k_1=0}^{T_1-1} \cdots \sum_{k_N=0}^{T_N-1} \\
&\quad \mathcal{A}(t_1 - k_1 \bmod T_1, \ldots, t_N - k_N \bmod T_N)\tilde{\mathcal{B}}(k_1,\ldots,k_N) \\
&= \frac{1}{\prod_n \tau_n} \left[\mathcal{A} * \tilde{\mathcal{B}}\right](t_1,\ldots,t_N). \qquad (18)
\end{aligned}$$

Thus, the inverse MDT is represented by an N-dimensional cyclic convolution.

Furthermore, we show that the tensor which is folded from rank-1 matrix has sufficient representation ability, as discussed in Subsection IV-A2. Rank-1 tensor model of $\boldsymbol{X} := \text{bunfold}_{(\boldsymbol{T},\boldsymbol{\tau})}(\mathcal{X}) = \boldsymbol{a}\boldsymbol{b}^{\mathrm{T}} \in \mathbb{R}^{\prod_n T_n \times \prod_n \tau_n}$ can generate any $\mathcal{X} \in \mathbb{R}^{T_1 \times \cdots \times T_n}$. Let us

$$\boldsymbol{a} = \left[\text{vec}(\mathcal{X})\right], \quad \boldsymbol{b} = \begin{bmatrix} \prod_n \tau_n \\ \boldsymbol{0}_{\prod_n \tau_n - 1} \end{bmatrix}, \qquad (19)$$

where $\boldsymbol{0}_{\prod_n \tau_n - 1}$ is an $(\prod_n \tau_n - 1)$-dimensional vector of zeros, and then we have

$$\left[\mathcal{H}_{\tau}^{\dagger}(\text{bfold}_{(\boldsymbol{\tau},\boldsymbol{T})}(\boldsymbol{a}\boldsymbol{b}^{\mathrm{T}}))\right](t_1,\ldots,t_N) = \mathcal{X}. \qquad (20)$$

Because this operation is equivalent to the cyclic convolution of a tensor $\mathcal{X}$ with only one element, the tensor $\mathcal{X}$ is derived identically (See Figure 4).

As discussed in Section IV-A2, the inverse MDT of an unconstrained tensor, even rank-1, is over-parameterized and does not work as a model. In this study, additional constraints were imposed on $\mathcal{A}$ and $\mathcal{B}$ (see Section IV-B).

B. SMOOTHNESS CONSTRAINTS

Because the convolution of factor tensors can represent any tensor (even rank 1 models), it is necessary to impose constraints to narrow down the candidate solutions. In this study, smoothness is used as a constraint. The reasons for introducing smoothness as a constraint are as follows.

- As shown in Figure 2, the embedded data is represented by a smooth manifold on the delay embedded space.
- The data mainly targeted in our study are images, and there are many reports that smooth constraints are effective in image completion [48], [15], [53], [47].

Note that we do not introduce smoothness for the reconstructed tensor but the factor tensors. Unlike the model which smoothens the reconstructed tensor, the proposed model enables completion without excessive smoothing. We also set the scale adjustment terms for both $\mathcal{A}$ and $\mathcal{B}$ in the optimization equation to avoid smoothing by increasing only one factor of the tensors.

C. OPTIMIZATION FORMULAS AND ALGORITHM

1) OPTIMIZATION FORMULAS

In this paper, we propose a new tensor completion model. We assume that the observed tensor $\mathcal{Y} \in \mathbb{R}^{T_1 \times \cdots \times T_N}$ is incomplete and that some entries have no values. The projection tensor $\mathcal{O} \in \{0,1\}^{T_1 \times \cdots \times T_N}$ passes the observed entries and makes the missing entries equal to zero. The entries are given by

$$\mathcal{O}(t_1,\ldots,t_N) = \begin{cases} 1 & \mathcal{Y}(t_1,\ldots,t_N) \text{ is observed} \\ 0 & \text{otherwise.} \end{cases} \qquad (21)$$

The problem involves obtaining the complete tensor $\mathcal{A} * \mathcal{B}$. In this study, we impose a smoothness constraint on $\mathcal{A}$ and $\mathcal{B}$. The optimization problem is then given by

$$\begin{aligned}
&\min_{\mathcal{A},\mathcal{B}} \left\| \mathcal{O} \circledast (\mathcal{Y} - \mathcal{A} * \tilde{\mathcal{B}}) \right\|_F^2 \\
&\quad + \sum_n \lambda_{A,n} \|\mathcal{L}_n * \mathcal{A}\|_F^2 + \sum_n \lambda_{B,n} \|\mathcal{L}_n * \tilde{\mathcal{B}}\|_F^2 \\
&\quad + \eta_A \|\mathcal{A}\|_F^2 + \eta_B \|\tilde{\mathcal{B}}\|_F^2 \\
&\text{s.t. } \tilde{\mathcal{B}} = \mathcal{B} \times \{\boldsymbol{P}\}, \qquad (22)
\end{aligned}$$

where

$$\begin{aligned}
&\mathcal{L}_n := \text{fold}_{\boldsymbol{T}}(\boldsymbol{l}_N \otimes \cdots \otimes \boldsymbol{l}_1) \in \mathbb{R}^{T_1 \times \cdots \times T_N} \\
&i = 1,\ldots,N \\
&\boldsymbol{l}_i = \begin{cases} [1,-1,0,\ldots,0] & (i = n) \\ [1,0,\ldots,0] & (i \neq n) \end{cases} \in \mathbb{R}^{T_i}
\end{aligned}$$

is a differential filter, and $\mathcal{A} \in \mathbb{R}^{T_1 \times \cdots \times T_N}$, $\mathcal{B} \in \mathbb{R}^{\tau_1 \times \cdots \times \tau_N}$ are factor tensors and $\text{fold}_{\boldsymbol{V}} : \mathbb{R}^{V_1 V_2 \cdots V_N} \to \mathbb{R}^{V_1 \times V_2 \times \cdots \times V_N}$ is a folding operator from a vector to the N-th order tensor. Equation (22) evaluates the reconstruction loss in the first term. The second and third terms are smooth penalties for $\mathcal{A}$ and $\mathcal{B}$, and the fourth and fifth terms adjust the scales of $\mathcal{A}$ and $\mathcal{B}$. The equality constraint is for zero padding, based on Equation (12). Note that when $\tau = 1$, Equation (22) is equivalent to QV regularization. The relaxation of optimization problem (22) for an unconstrained optimization problem

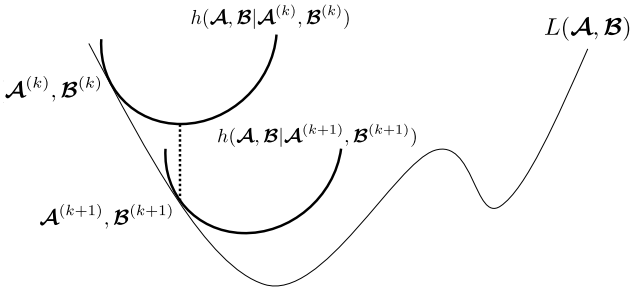


FIGURE 5. Concept of MM Algorithm.

including a penalty term yields the following equation:

$$\begin{aligned}\min_{\mathcal{A},\mathcal{B}} L(\mathcal{A},\mathcal{B}) := &\|\mathcal{O}\circledast(\mathcal{Y}-\mathcal{A}*\mathcal{B})\|_F^2 \\ &+\gamma\|\mathcal{I}_\tau\circledast\mathcal{B}\|_F^2 \\ &+\sum_n \lambda_{A,n}\|\mathcal{L}_n*\mathcal{A}\|_F^2+\sum_n\lambda_{B,n}\|\mathcal{L}_n*\mathcal{B}\|_F^2 \\ &+\eta_A\|\mathcal{A}\|_F^2+\eta_B\|\mathcal{B}\|_F^2, \end{aligned} \tag{23}$$

where $\mathcal{I}_\tau = \text{fold}_{\boldsymbol{T}}(\boldsymbol{i}_{\tau_1}\otimes\ldots\otimes\boldsymbol{i}_{\tau_N})\in\mathbb{R}^{T_1\times\cdots\times T_N}$, $\boldsymbol{i}_{\tau_n} := [\underbrace{0,\cdots,0}_{\tau_n},\underbrace{1,\cdots,1}_{T_n-\tau_n}]\in\mathbb{R}^{T_n}$. $\boldsymbol{i}_{\tau_n}$ serves as a penalty for $\mathcal{B}$ and simulates zero padding $\{\boldsymbol{P}\}$. Note that we also redefine the size of $\mathcal{B}$ as $T_1\times\cdots\times T_N$.

2) ALGORITHM FOR SOLVING OPTIMIZATION

In this study, we solved the optimization problem (23) using the majorization-minimization (MM) algorithm [18], [30]. The MM algorithm is an iterative method involving two steps.

1) Constructs a auxiliary function $h(\mathcal{A},\mathcal{B}|\mathcal{A}^{(k)},\mathcal{B}^{(k)})$ for $L(\mathcal{A},\mathcal{B})$ at $\mathcal{A}^{(k)},\mathcal{B}^{(k)}$. Note,

$$\begin{aligned}&\forall\mathcal{A},\mathcal{B}\quad L(\mathcal{A},\mathcal{B})\le h(\mathcal{A},\mathcal{B}|\mathcal{A}^{(k)},\mathcal{B}^{(k)})\\ &L(\mathcal{A}^{(k)},\mathcal{B}^{(k)}) = h(\mathcal{A}^{(k)},\mathcal{B}^{(k)}|\mathcal{A}^{(k)},\mathcal{B}^{(k)}).\end{aligned}$$

2) Update as in

$$\mathcal{A}^{(k+1)}\leftarrow\arg\min_{\mathcal{A}} h(\mathcal{A},\mathcal{B}^{(k)}|\mathcal{A}^{(k)},\mathcal{B}^{(k)}). \tag{24}$$

3) Update as in

$$\mathcal{B}^{(k+1)}\leftarrow\arg\min_{\mathcal{B}} h(\mathcal{A}^{(k+1)},\mathcal{B}|\mathcal{A}^{(k+1)},\mathcal{B}^{(k)}). \tag{25}$$

A conceptual diagram of the algorithm is shown in Figure 5. The MM algorithm was used because of convergence due to its monotonic convergence and ease of analytical computation.

The auxiliary function h is defined as follows:

$$\begin{aligned}L(\mathcal{A},\mathcal{B}) &\le h(\mathcal{A},\mathcal{B}|\mathcal{A}^{(k)},\mathcal{B}^{(k)})\\ &:= \|\mathcal{O}\circledast(\mathcal{Y}-\mathcal{A}*\mathcal{B})\|_F^2+\gamma\|\mathcal{I}_\tau\circledast\mathcal{B}\|_F^2\\ &\quad+\left\|\overline{\mathcal{O}}\circledast(\mathcal{A}^{(k)}*\mathcal{B}^{(k)}-\mathcal{A}*\mathcal{B})\right\|_F^2\\ &\quad+\gamma\|\mathcal{I}_{\overline{\tau}}\circledast(\mathcal{B}-\mathcal{B}^{(k)})\|_F^2\end{aligned}$$

Algorithm 1 MM Algorithm in the Proposed Method

Require: $\mathcal{Y}, \mathcal{O}, \mathcal{I}_\tau, \gamma, \lambda_A, \lambda_B, \eta_A, \eta_B$, maxiter

$\hat{\mathcal{A}}\leftarrow$ FFT($\mathcal{A}$)
$\hat{\mathcal{B}}\leftarrow$ FFT($\mathcal{B}$)
for $i=1$ to maxiter **do**
  Update $\mathcal{Z}$ by (27)
  $\hat{\mathcal{Z}}\leftarrow$ FFT($\mathcal{Z}$)
  Update $\mathcal{W}$ by (28)
  $\hat{\mathcal{W}}\leftarrow$ FFT($\mathcal{W}$)
  Update $\hat{\mathcal{A}}$ by (31)
  Update $\hat{\mathcal{B}}$ by (32)
  $\mathcal{A}\leftarrow$ IFFT($\hat{\mathcal{A}}$)
  $\mathcal{B}\leftarrow$ IFFT($\hat{\mathcal{B}}$)
  Calculate $L(\mathcal{A},\mathcal{B})$.
  if convergence of L **then** break
  end if
end for

$$\begin{aligned}&+\sum_n\lambda_{A,n}\|\mathcal{L}_n*\mathcal{A}\|_F^2+\sum_n\lambda_{B,n}\|\mathcal{L}_n*\mathcal{B}\|_F^2\\ &+\eta_A\|\mathcal{A}\|_F^2+\eta_B\|\mathcal{B}\|_F^2\\ &=\|\mathcal{Z}-\mathcal{A}*\mathcal{B}\|_F^2+\gamma\|\mathcal{W}-\mathcal{B}\|_F^2\\ &+\sum_n\lambda_{A,n}\|\mathcal{L}_n*\mathcal{A}\|_F^2+\sum_n\lambda_{B,n}\|\mathcal{L}_n*\mathcal{B}\|_F^2\\ &+\eta_A\|\mathcal{A}\|_F^2+\eta_B\|\mathcal{B}\|_F^2,\end{aligned} \tag{26}$$

where

$$\mathcal{Z}=\mathcal{O}\circledast\mathcal{Y}+\overline{\mathcal{O}}\circledast\left(\mathcal{A}^{(k)}*\mathcal{B}^{(k)}\right), \tag{27}$$

and

$$\mathcal{W}=\mathcal{I}_{\overline{\tau}}\circledast\mathcal{B}^{(k)}. \tag{28}$$

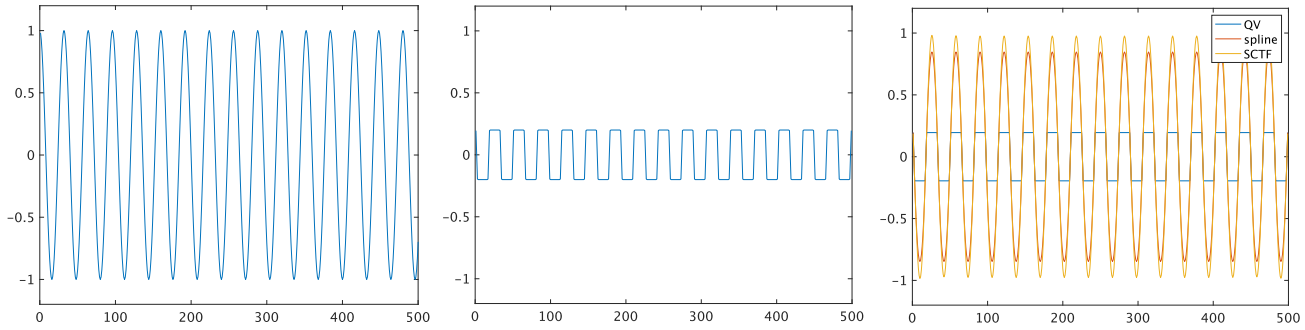
Furthermore, we exploit the property that the cyclic convolution of the time domain is a Hadamard product in the frequency domain to reduce the time complexity of the optimization problem. Because the Frobenius norm is invariant to the Fourier transform, the auxiliary function h is redefined as

$$\begin{aligned}h(\mathcal{A},\mathcal{B}|\mathcal{A}^{(k)},\mathcal{B}^{(k)}) &= \hat{h}(\hat{\mathcal{A}},\hat{\mathcal{B}}|\hat{\mathcal{A}}^{(k)},\hat{\mathcal{B}}^{(k)})\\ &:=\left\|\hat{\mathcal{Z}}-\hat{\mathcal{A}}\circledast\hat{\mathcal{B}}\right\|_F^2+\gamma\|\hat{\mathcal{W}}-\hat{\mathcal{B}}\|_F^2\\ &\quad+\sum_n\lambda_{A,n}\|\hat{\mathcal{L}}_n\circledast\hat{\mathcal{A}}\|_F^2\\ &\quad+\sum_n\lambda_{B,n}\|\hat{\mathcal{L}}_n\circledast\hat{\mathcal{B}}\|_F^2\\ &\quad+\eta_A\|\hat{\mathcal{A}}\|_F^2+\eta_B\|\hat{\mathcal{B}}\|_F^2,\end{aligned} \tag{29}$$

where $\hat{\mathcal{L}}_n, \hat{\mathcal{A}}, \hat{\mathcal{B}}, \hat{\mathcal{Z}}, \hat{\mathcal{W}}$ are the Fourier transform of $\mathcal{L}_n, \mathcal{A}, \mathcal{B}, \mathcal{Z}, \mathcal{W}$.

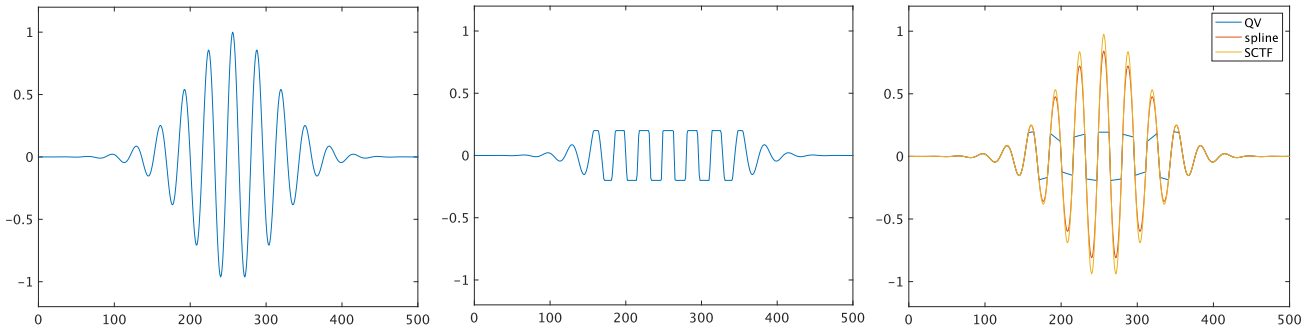
The final auxiliary function is $\hat{h}$, which is minimized using Equations (24) and (25). To minimize $\hat{h}$, we derive $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$ such that

$$\frac{\partial\hat{h}}{\partial\hat{\mathcal{A}}}=\mathbf{0},\qquad\frac{\partial\hat{h}}{\partial\hat{\mathcal{B}}}=\mathbf{0}. \tag{30}$$



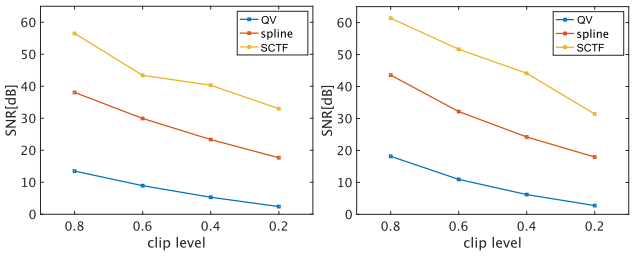
(a) Original signal (b) Clipped signal (c) De-clipped signals

FIGURE 6. Example of declipping experiment: (a) original signal of the sine function, (b) clipped signal with clipping level = 0.2, and (c) these reconstructed signals by using QV regularization, cubic spline interpolation, and SCTF.



(a) Original signal (b) Clipped signal (c) De-clipped signals

FIGURE 7. Example of declipping experiment: (a) original signal of wavelet function, (b) clipped signal with clipping level = 0.2, and (c) these reconstructed signals by using QV regularization, cubic spline interpolation, and SCTF.



(a) sine (b) wavelet

FIGURE 8. Values of SNR in declipping experiments with various clipping levels.

Note that (30) is substituted by alternating the optimization with (24) and (25) because they are not satisfied simultaneously. Thus, at every optimization step, although the cost function L decreases monotonically, the auxiliary function h is not always optimal. After solving Equation (30), we obtain

$$\hat{\mathcal{A}} = \left\{\hat{\mathcal{Z}}^* \circledast \hat{\mathcal{B}}\right\} \oslash \left\{\hat{\mathcal{B}}^2 + \sum_n \lambda_{A,n}\hat{\mathcal{L}}_n^2 + \eta_A\right\}, \tag{31}$$

$$\hat{\mathcal{B}} = \left\{\hat{\mathcal{Z}}^* \circledast \hat{\mathcal{A}} + \gamma\hat{\mathcal{W}}\right\} \oslash \left\{\hat{\mathcal{A}}^2 + \sum_n \lambda_{B,n}\hat{\mathcal{L}}_n^2 + \eta_B + \gamma\right\}, \tag{32}$$

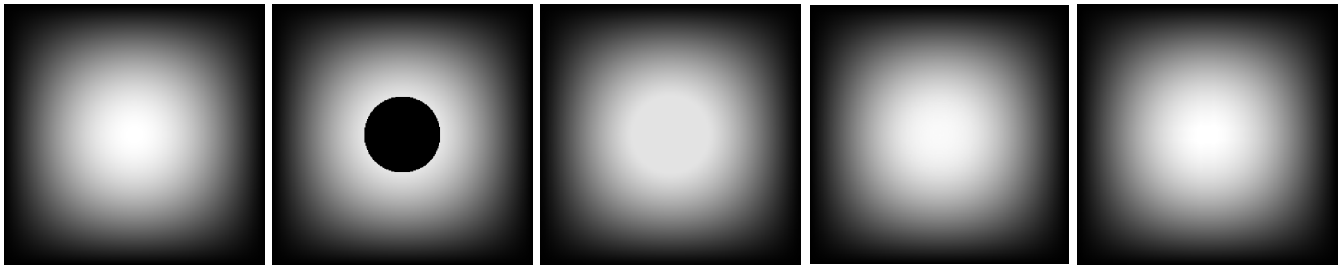
where $\hat{\mathcal{Z}}^*$ is the complex conjugate of $\hat{\mathcal{Z}}$. In summary, Equations (27) and (28) correspond to Step 1 of the MM algorithm, and Equations (31) and (32) correspond to Step 2. Algorithm 1 summarizes the proposed method.

D. COMPUTATIONAL COMPLEXITY

The algorithm consists of updated Equations (27), (28), (31), and (32). The time complexity of (28), (31), and (32) is $\mathcal{O}(T^N)$, and that of (27) is $\mathcal{O}(NT^N \log T)$. Since (27) is derived from the cyclic convolution by $\mathcal{A} * \mathcal{B} = \text{IFFT}(\text{FFT}(\mathcal{A}) \circledast \text{FFT}(\mathcal{B}))$, the time complexity $\mathcal{O}(NT^N \log T)$ of the FFT is dominant. Consequently, the overall time complexity of the update equation is $\mathcal{O}(NT^N \log T)$.

E. RELATION TO OUR EXISTING PAPER

This paper is an extended version based on our paper [38]. Therefore, the concept of the proposed data completion model is almost identical. However, existing methods discuss only vector data (one-dimensional data) and do not target tensor data (multidimensional data). The proposed method differs significantly from existing methods in extending the model design and algorithms to multiple dimensions. Furthermore, in addition to the 1D signal completion of the existing method, this paper also experiments with image completion for RGB and MRI images (see



(a) Original image (b) Clipped image (c) QV (d) Fast-MDT-Tucker (e) SCTF

FIGURE 9. Results of recoverd cliped image using QV (a), Fast-MDT-Tucker (b) and SCTF (c).

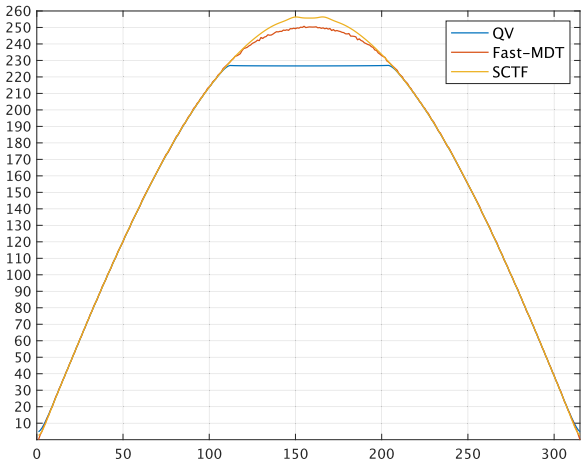


FIGURE 10. Cross-sectional view of the recoverd cliped image at height 157.

Sections V-B1 and V-B2) and hyper-parameter sensitivities experiments (see Section V-C2).

V. EXPERIMENT

A. COMPLETION OF CLIPPED DATA

This section presents the evaluation of the proposed method (SCTF) for reconstructing clipped data (declipping) as a type of completion. Clipping is an operation that uses a certain clipping level $c > 0$ to replace entries above c and below $-c$ with c and $-c$. The clipping operation on the entry of a tensor is given by

$$\mathcal{X}(t_1, t_2, \cdots, t_N) = \min(c, \max(-c, \mathcal{X}_0(t_1, \cdots, t_N))). \tag{33}$$

The value range of the clipped data was $[-c, c]$.

The indices of the clipped entries were recorded and treated as missing values for reconstruction. Thus, the set of observed entries can be expressed as

$$\mathcal{O}(t_1, \ldots, t_N) = \begin{cases} 1 & -c < \mathcal{X}(t_1, \ldots, t_N) < c \\ 0 & \text{otherwise.} \end{cases} \tag{34}$$

TABLE 1. Comparison of the peak signal-to-noise ratio (PSNR), the structural similarity (SSIM), and the computing time (sec) of recovery clipped images using TV, Fast-MDT-Tucker, and proposed method.

	QV	Fast-MDT-Tucker	Proposed
PSNR	34.9	50.8	**55.1**
SSIM	0.910	**0.987**	0.974
computing time	**3.29**	8.74	72.46

1) CLIPPED SIGNAL COMPLETION (1ST ORDER tensor)

In this experiment, we evaluated SCTF using signal completion. The signals to be completed were the sine and wavelet functions with a clip level of 0.2, and both had a maximum amplitude of 1. Examples of clipping are presented in Figures 6 and 7. The original signals are shown in Figures 6a and 7a, and the signals after clipping are shown in Figures 6b and 7b. We compared SCTF with QV regularization and spline interpolation. In SCTF, we set $\tau_1 = 32$, $\gamma = 1.0 \times 10^8$, and $\lambda_{A,1} = \lambda_{B,1} = \eta_A = \eta_B = 0.1$.

Figures 6c and 7c show the signals reconstructed using QV regularization, spline interpolation, and SCTF. The signal completed by SCTF had a maximum amplitude of approximately 1, which was the best among the three methods. The spline method reconstructs the waveform signals; however, its amplitude is smaller than that of the original signal. However, the QV method failed to restore the signal.

Figure 8 shows the signal-to-noise ratio (SNR) values of the declipping experiment for various clipping levels. The clipping levels were 0.8, 0.6, 0.4, and 0.2, and the parameters (λ, τ) were adjusted for all levels. SCTF achieved significantly higher values of SNR than QV regularization and spline interpolation.

2) COMPLETION OF CLIPPED DATA (2ND ORDER tensor)

In this experiment, we evaluated SCTF by completing a clipped image of 2D-sin. The maximum value of the image (amplitude of 2D-sin) was 255, and the clip threshold $c = 230$. The original image is shown in Figure 9a, and the image after clipping is shown in Figure 9b. We compared SCTF with the QV model and Fast-MDT-Tucker [45]. The QV model is the model without convolution in Equation (22), and is

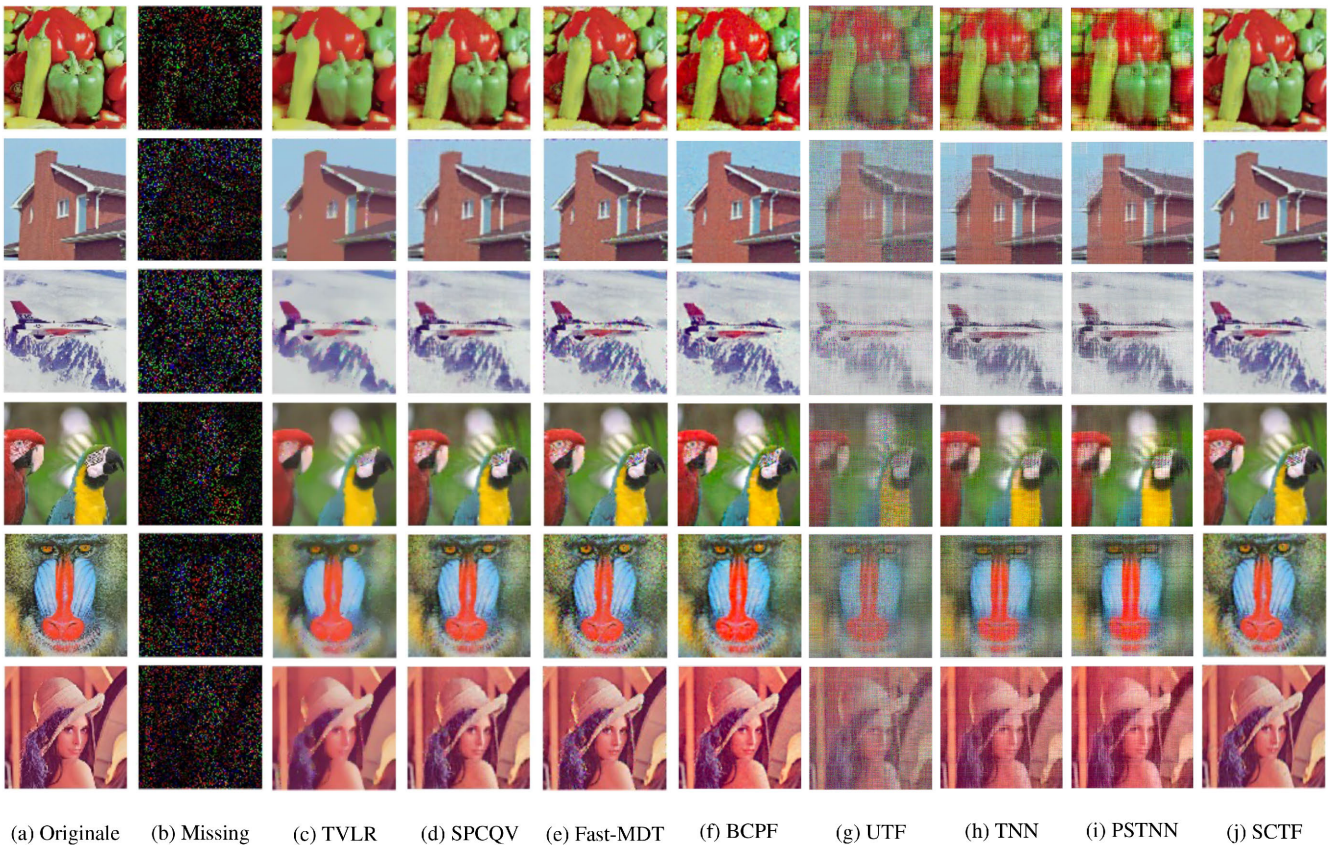


FIGURE 11. Results of recoverd RGB images completion using TVLR (a), SPCQV (b), Fast-MDT-Tucker (c), BCPF (d), UTF (e), TNN (f), PSTNN (g), and SCTF (h). The image types are listed in order from the top row: "pappers", "house", "airplane", "parrots", "mandrill", and "lena".

TABLE 2. Comparison of the peak signal-to-noise ratio (PSNR), the mean absolute error (MAE), the structural similarity (SSIM) and the computing time (sec) of recovery RGB images. Missing ratio is 85%.

Image name	evaluation	TVLR	SPCQV	Fast-MDT	BCPF	UTF	TNN	PSTNN	SCTF
peppers	PSNR	23.5	25.5	25.3	23.5	14.8	18.3	19.2	**26.5**
	MAE	1.80×10^6	1.52×10^6	1.36×10^6	1.87×10^6	6.92×10^6	4.21×10^6	3.62×10^6	$\mathbf{1.32 \times 10^6}$
	SSIM	0.931	0.950	0.951	0.770	0.547	0.786	0.825	**0.961**
	rumtime	37.07	54.19	3.02	88.59	**0.61**	19.52	21.2	38.20
house	PSNR	24.3	26.8	25.4	25.5	17.3	22.4	22.5	**27.3**
	MAE	1.42×10^6	1.27×10^6	1.35×10^6	1.34×10^6	5.04×10^6	2.44×10^6	2.36×10^6	$\mathbf{1.24 \times 10^6}$
	SSIM	0.892	0.913	0.897	0.797	0.463	0.766	0.771	**0.923**
	runtime	35.20	35.62	2.66	105.94	**0.45**	18.37	20.93	25.07
airplane	PSNR	22.2	24.3	22.8	23.2	17.6	20.6	20.8	**24.3**
	MAE	1.92×10^6	$\mathbf{1.62 \times 10^6}$	1.67×10^6	1.73×10^6	3.91×10^6	2.89^6	2.71×10^6	1.73×10^6
	SSIM	0.655	0.669	**0.706**	0.797	0.225	0.364	0.378	0.685
	runtime	42.95	38.99	2.85	59.28	**0.40**	18.37	20.77	14.69
parrots	PSNR	24.6	25.4	24.5	24.5	15.6	20.4	21.2	**25.9**
	MAE	1.48×10^6	1.33×10^6	1.21×10^6	1.50×10^6	5.72×10^6	2.99×10^6	2.63×10^6	$\mathbf{1.52 \times 10^6}$
	SSIM	0.899	0.902	0.906	0.820	0.375	0.721	0.748	**0.913**
	runtime	40.63	51.44	2.73	55.44	**0.39**	17.80	20.48	49.53
mandrill	PSNR	21.1	**21.8**	19.4	21.3	16.1	18.6	19.0	21.5
	MAE	3.04×10^6	$\mathbf{2.72 \times 10^6}$	3.40×10^6	2.93×10^6	5.81×10^6	4.24×10^6	4.02×10^6	2.92×10^6
	SSIM	0.667	**0.720**	0.648	0.583	0.375	0.545	0.567	0.706
	runtime	31.75	63.97	4.87	99.38	**0.42**	17.51	20.35	18.97
lena	PSNR	24.3	26.0	25.0	25.8	16.2	20.9	19.9	**26.2**
	MAE	1.70×10^6	$\mathbf{1.43 \times 10^6}$	1.44×10^6	1.52×10^6	5.81×10^6	3.01×10^6	2.77×10^6	1.52×10^6
	SSIM	0.938	0.950	0.939	0.783	0.566	0.856	0.876	**0.952**
	runtimes	36.94	45.99	4.61	72.94	**0.41**	17.44	19.95	23.93

formulated as

$$\hat{\mathcal{Z}} \leftarrow \arg\min_{\mathcal{Z}} \|\mathcal{O} \circledast (\mathcal{Y} - \mathcal{Z})\|_F^2 + \sum_n \lambda_n \|\mathcal{L}_n * \mathcal{Z}\|_F^2,$$

and $\hat{\mathcal{Z}}$ denotes the estimated completion tensor; In SCTF, we set $\tau_1 = \tau_2 = 151$, $\gamma = 1.0 \times 10^8$, $\lambda_{A,1} = \lambda_{A,2} = \lambda_{B,1} = \lambda_{B,1} = 200$, and $\eta_A = \eta_B = 400$.

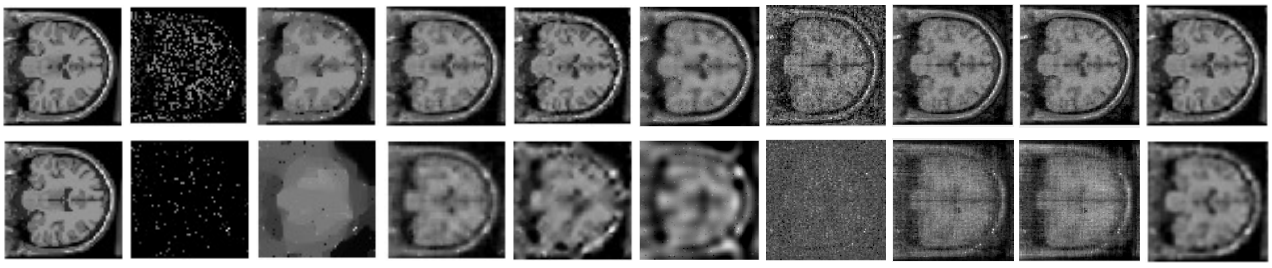


FIGURE 12. Results of recoverd MRI image completion using TVLR (a), SPCQV (b), Fast-MDT-Tucker (c), BCPF (d), UTF (e), TNN (f), PSTNN (g), and SCTF (h). The time slice t is 50. The 1st column is the image with 70% missing, the 2nd row is the image with 95% missing.

TABLE 3. Comparison of the peak signal-to-noise ratio (PSNR), the mean absolute error (MAE), the structural similarity (SSIM) and the computing time (sec) of recovery MRI images. Missing ratio are 70% and 95%. Note that SSIM is averaged over time.

missingrate	evaluation	TVLR	SPCQV	Fast-MDT	BCPF	UTF	TNN	PSTNN	SCTF
70%	PSNR	23.8	26.5	23.8	20.1	14.4	22.7	23.4	**27.0**
	MAE	6.76×10^6	5.70×10^6	6.52×10^6	1.26×10^7	2.86×10^6	9.84×10^6	8.76×10^6	$\mathbf{5.21 \times 10^6}$
	SSIM	0.619	0.633	0.638	0.330	0.360	0.521	0.528	**0.700**
	rumtime	39.24	230.91	7.67	176.43	**7.30**	77.64	90.86	48.12
95%	PSNR	16.3	20.6	18.0	10.5	12.3	17.1	17.8	**21.1**
	MAE	1.94×10^7	1.29×10^7	1.58×10^7	3.24×10^7	4.40×10^6	2.17×10^6	1.90×10^7	$\mathbf{1.13 \times 10^7}$
	SSIM	0.139	0.391	0.312	0.207	0.071	0.203	0.220	**0.459**
	runtime	99.94	187.58	11.68	130.48	**6.54**	77.03	87.95	70.42

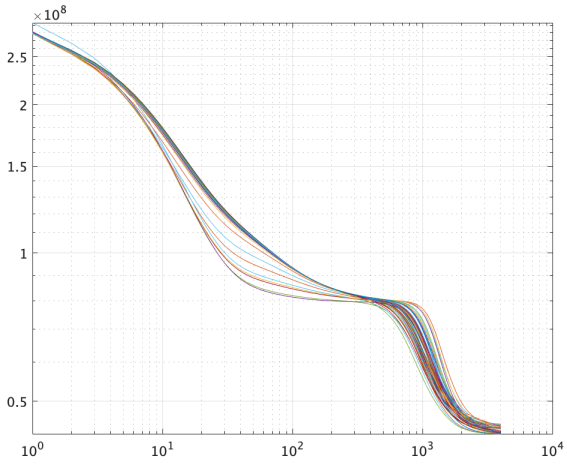


FIGURE 13. Optimization behavior: This figure shows 50 curves for 50 different initial values. The completion problem deals with the completion of clipped 2D-sin.

Figures 9 and 10 show the images reconstructed using the QV model, Fast-MDT-Tucker, and the proposed SCTF. In addition, Table 1 shows the numerical evaluation of the completion accuracy. The completion image by SCTF has a maximum amplitude close to 255 and shows an improvement in the oscillations that occurred in Fast-MDT-Tucker, indicating the effect of the smoothing term. In fact, SCTF had the best value in PSNR. However, SCTF had a flat shape that differed from that of sine at a maximum amplitude of 255. This affects the numerical evaluation of the completion accuracy, and SCTF is worse in SSIM than in Fast-MDT-Tucker. On the other hand, the QV model fails to recover 2D-sin because it flatly completes the missing parts.

B. COMPLETION OF RANDOM MISSING DATA

This section presents the results of the completion of random missing data.

1) COMPLETION OF RGB IMAGES

In this experiment, we evaluated SCTF using RGB images. Six images were tested, with a missing rate of 85%. Figures 11a and 11b show the original image and the image with missing data, respectively. We compared SCTF with TVLR [47], SPCQV [48], Fast-MDT-Tucker [45], BCPF [51], UTF [10], TNN [50], and PSTNN [19]. In SCTF, we set $\tau_1 = \tau_2 = \tau_3 = 9$, $\gamma = 1.0 \times 10^8$, $\lambda_{A,1} = \lambda_{A,2} = \lambda_{B,1} = \lambda_{B,2} = 50$, $\lambda_{A,3} = \lambda_{B,3} = 0$, and $\eta_A = \eta_B = 50$.

Figure 11 shows the experimental results. SCTF improves the image blur in TVLR, SPCQV, and BCPF and the jaggies in Fast-MDT-Tucker, UTF, TNN, and PSTNN. This may be because SCTF is based on the idea of both smoothness and MDT. Table 2 summarizes the recovery performance (PSNR and SSIM) and runtime. SCTF had the highest PSNR for five images and SSIM for four images. It is also slower than Fast-MDT-Tucker and UTF but has a faster computation time than SPCQV, which is the second most accurate method. In other words, SCTF is completed with high accuracy and at a modest computational cost.

2) COMPLETION OF MRI IMAGES

In this experiment, we evaluated SCTF using MRI images. We prepared MRI images with sizes of $(100 \times 91 \times 91)$

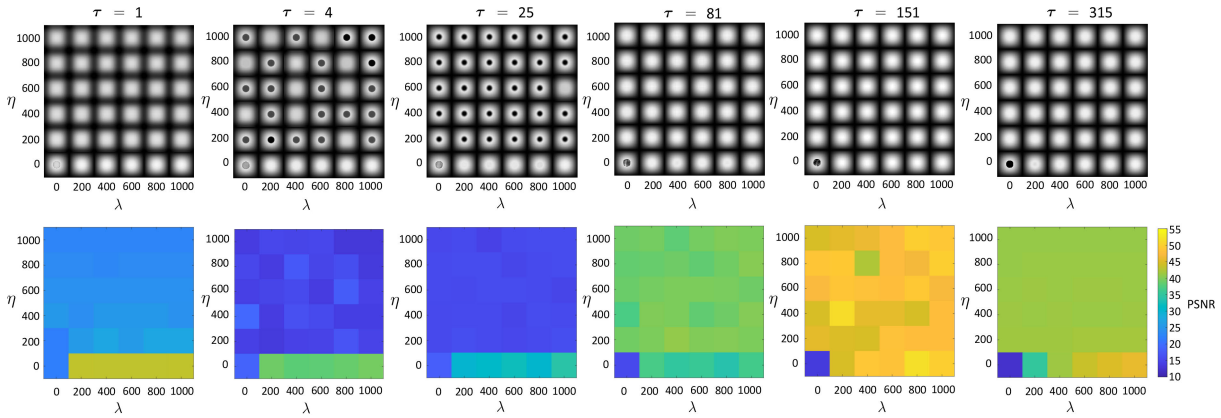


FIGURE 14. Effects of hyper-parameters in declipping sine function. Reconstruction quality was evaluated by the PSNR for various settings of hyper-parameters τ, λ, η (Average of 5 trials for each item.)

with 70% and 95% of the random voxel missing. Figures 12a and 12b show the original image and the image with missing data, respectively. We compare SCTF with TVLR [47], SPCQV [48], Fast-MDT-Tucker [45], BCPF [51], UTF [10], TNN [50], and PSTNN [19]. In SCTF, we set $\tau_1 = \tau_2 = \tau_3 = 4$, $\gamma = 1.0 \times 10^8$, $\lambda_{A,1} = \lambda_{A,2} = \lambda_{B,1} = \lambda_{B,2} = 50$, $\lambda_{A,3} = \lambda_{B,3} = 0$, and $\eta_A = \eta_B = 1000$.

Figure 12 shows the experimental results. SCTF showed successful completion in both the 70% and 95% missing cases. In the 95% case, TVLR, Fast-MDT-Tucker, BCPF, and UTF fail to recover, and SPCQV smoothes the image excessively. In addition, TNN and PSTNN do not restore smoothly compared to SCTF. Table 3 summarizes the recovery performance (PSNR and SSIM) and runtime. We can confirm that SCTF has the best completion accuracy compared to the other methods; it takes longer to execute than Fast-MDT-Tucker and UTF but more than half the time of SPCQV.

C. ANALYSIS OF ALGORITHM

1) CONVERGENCE OF ALGORITHM

Monotonic convergence is expected because the proposed algorithm uses the MM algorithm. Figure 13 shows that the objective function converges monotonically and that the algorithm works correctly.

2) HYPER-PARAMETER SENSITIVITIES

We investigated the effects of hyper-parameters of SCTF in the completion. SCTF has three hyperparameters: the delay window size $\tau_1, \ldots, \tau_N$, smoothing level $\lambda_{A,1}, \ldots, \lambda_{A,N}, \lambda_{B,1}, \ldots, \lambda_{B,N}$ and scale adjustment η_A, η_B (see Equations (23)). We redefine each of the three parameter types as $\tau := \tau_1 = \ldots = \tau_N$, $\lambda := \lambda_{A,1} = \ldots = \lambda_{A,N} = \lambda_{B,1} = \ldots = \lambda_{B,N}$, and $\eta := \eta_A = \eta_B$.

The experimental setup was the same as that in Section V-A2; we recovered the clipped 2D-sin image. Three hyper-parameters were varied in the range $\tau \in \{1, 9, 25, 81, 151, 315\}$ and $\lambda \in \{0, 200, 400, 600, 800, 1000\}$ and $\eta \in \{0, 200, 400, 600, 800, 1000\}$. A declipping experiment was performed for all the combinations to calculate the recovery accuracy.

Figure 14 shows the experiment's five-time average of the PSNR. Increasing τ improves the accuracy, whereas making it too large worsens the accuracy. For example, when $\tau = 1$, the algorithm matches the QV regularization and recovers smoothly; however, when the delay window is smaller than the clip range, such as when $\tau = 9$ or $\tau = 25$, it cannot recover at all. Therefore, it is important to set τ appropriately.

VI. CONCLUSION

In this study, we proposed a new model and algorithm for tensor completion using a convolution of smooth-factor tensors. Because the proposed method corresponds to a rank-1 decomposition in the delay-embedded space, it can achieve high completion accuracy in a short computation time. In the optimization formulation, we extended the existing mathematical model based on the inverse MDT by adding a penalty term for the factor tensor corresponding to the delay-embedding width. In addition, we set smoothing constraints for the factor to narrow down the candidate solutions. As for the algorithm for solving the optimization, we employed the MM algorithm with the expectation of monotonic convergence. Our experiments mainly completed clipped and random missing image data and confirmed that the proposed method achieves high completion accuracy with low computational cost. In the experiment, we also confirmed the effect of the completion accuracy on the variation in the delay-embedding width and monotone convergence of the algorithm.

REFERENCES

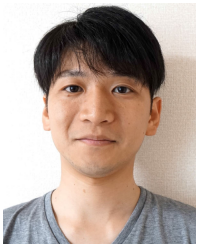
[1] I. Balazevic, C. Allen, and T. Hospedales, "TuckER: Tensor factorization for knowledge graph completion," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 5185–5194.

[2] H. Becker, L. Albera, P. Comon, M. Haardt, G. Birot, F. Wendling, M. Gavaret, C. G. Bénar, and I. Merlet, "EEG extended source localization: Tensor-based vs. conventional methods," *NeuroImage*, vol. 96, pp. 143–157, Aug. 2014.

[3] S. L. Brunton, B. W. Brunton, J. L. Proctor, E. Kaiser, and J. N. Kutz, "Chaos as an intermittently forced linear system," *Nature Commun.*, vol. 8, p. 19, May 2017.
[4] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of 'Eckart-Young' decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, Sep. 1970.
[5] Y. Chen, C. Hsu, and H. M. Liao, "Simultaneous tensor decomposition and completion using factor priors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 577–591, Mar. 2014.
[6] A. Cichocki and A.-H. Phan, "Fast local algorithms for large scale nonnegative matrix and tensor factorizations," *IEICE Trans. Fundamentals Electron., Commun. Comput. Sci.*, vols. E92–A, no. 3, pp. 708–721, 2009.
[7] A. Cichocki, R. Zdunek, S. Choi, R. Plemmons, and S.-I. Amari, "Non-negative tensor factorization using alpha and beta divergences," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2007, pp. 1393–1396.
[8] F. Cong, Q.-H. Lin, L.-D. Kuang, X.-F. Gong, P. Astikainen, and T. Ristaniemi, "Tensor decomposition of EEG signals: A brief review," *J. Neurosci. Methods*, vol. 248, pp. 59–69, Jun. 2015.
[9] S. Du, Y. Shi, W. Hu, W. Wang, and J. Lian, "Robust tensor factorization for color image and grayscale video recovery," *IEEE Access*, vol. 8, pp. 174410–174423, 2020.
[10] S. Du, Q. Xiao, Y. Shi, R. Cucchiara, and Y. Ma, "Unifying tensor factorization and tensor nuclear norm approaches for low-rank tensor completion," *Neurocomputing*, vol. 458, pp. 204–218, Oct. 2021.
[11] B. Erem, R. Martinez Orellana, D. E. Hyde, J. M. Peters, F. H. Duffy, P. Stovicek, S. K. Warfield, R. S. MacLeod, G. Tadmor, and D. H. Brooks, "Extensions to a manifold learning framework for time-series analysis on dynamic manifolds in bioelectric signals," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 93, no. 4, Apr. 2016, Art. no. 042218.
[12] Z. Fan, X. Song, and R. Shibasaki, "CitySpectrum: A non-negative tensor factorization approach," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Sep. 2014, pp. 213–223.
[13] M. Filipović and A. Jukić, "Tucker factorization with missing data with application to low-*n*-rank tensor completion," *Multidimensional Syst. Signal Process.*, vol. 26, no. 3, pp. 677–692, Jul. 2015.
[14] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, Feb. 2011, Art. no. 025010.
[15] X. Guo and Y. Ma, "Generalized tensor total variation minimization for visual data recovery?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3603–3611.
[16] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an 'explanatory' multimodal factor analysis," in *Proc. UCLA Work. Papers Phonetics*, vol. 16, 1970, pp. 1–84.
[17] K. Huang, N. D. Sidiropoulos, and A. P. Liavas, "A flexible and efficient algorithmic framework for constrained matrix and tensor factorization," *IEEE Trans. Signal Process.*, vol. 64, no. 19, pp. 5052–5065, Oct. 2016.
[18] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statistician*, vol. 58, no. 1, pp. 30–37, Feb. 2004.
[19] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, and L.-J. Deng, "Multi-dimensional imaging data recovery via minimizing the partial sum of tubal nuclear norm," *J. Comput. Appl. Math.*, vol. 372, Jul. 2020, Art. no. 112680.
[20] H. Lee, Y.-D. Kim, A. Cichocki, and S. Choi, "Nonnegative tensor factorization for continuous EEG classification," *Int. J. Neural Syst.*, vol. 17, no. 4, pp. 305–317, Aug. 2007.
[21] G. Liu, "Time series forecasting via learning convolutionally low-rank models," *IEEE Trans. Inf. Theory*, vol. 68, no. 5, pp. 3362–3380, May 2022.
[22] G. Liu and W. Zhang, "Recovery of future data via convolution nuclear norm minimization," *IEEE Trans. Inf. Theory*, vol. 69, no. 1, pp. 650–665, Jan. 2023.
[23] H. Liu, Y. Li, M. Tsang, and Y. Liu, "CoSTCo: A neural tensor completion model for sparse tensors," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 324–334.
[24] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 208–220, Jan. 2013.
[25] Z. Long, Y. Liu, L. Chen, and C. Zhu, "Low rank tensor completion for multiway visual data," *Signal Process.*, vol. 155, pp. 301–316, Feb. 2019.
[26] E. N. Lorenz, "Deterministic nonperiodic flow," *J. Atmos. Sci.*, vol. 20, no. 2, pp. 130–141, Mar. 1963.
[27] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis with a new tensor nuclear norm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 925–938, Apr. 2020.
[28] B. Madathil and S. N. George, "Twist tensor total variation regularized-reweighted nuclear norm based tensor completion for video missing area recovery," *Inf. Sci.*, vol. 423, pp. 376–397, Jan. 2018.
[29] I. Markovsky, "Structured low-rank approximation and its applications," *Automatica*, vol. 44, no. 4, pp. 891–909, Apr. 2008.
[30] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*. Philadelphia, PA, USA: SIAM, 2000.
[31] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.
[32] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *J. Fluid Mech.*, vol. 641, pp. 115–127, Dec. 2009.
[33] F. Sedighin and A. Cichocki, "Image completion in embedded space using multistage tensor ring decomposition," *Frontiers Artif. Intell.*, vol. 4, Aug. 2021, Art. no. 687176.
[34] F. Sedighin, A. Cichocki, T. Yokota, and Q. Shi, "Matrix and tensor completion in multiway delay embedded space using tensor train, with application to signal reconstruction," *IEEE Signal Process. Lett.*, vol. 27, pp. 810–814, 2020.
[35] A. Shashua and T. Hazan, "Non-negative tensor factorization with applications to statistics and computer vision," in *Proc. 22nd Int. Conf. Mach. Learn.*, 2005, pp. 792–799.
[36] Q. Shi, J. Yin, J. Cai, A. Cichocki, T. Yokota, L. Chen, M. Yuan, and J. Zeng, "Block Hankel tensor ARIMA for multiple short time series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 5758–5766.
[37] Q. Song, H. Ge, J. Caverlee, and X. Hu, "Tensor completion algorithms in big data analytics," *ACM Trans. Knowl. Discovery Data*, vol. 13, no. 1, pp. 1–48, Feb. 2019.
[38] H. Takayama and T. Yokota, "Fast signal completion algorithm with cyclic convolutional smoothing," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Nov. 2022, pp. 364–371.
[39] K. Takeuchi, Y. Kawahara, and T. Iwata, "Structurally regularized non-negative tensor factorization for spatio-temporal pattern discoveries," in *Proc. Mach. Learn. Knowl. Discovery Databases, Eur. Conf.*, 2017, pp. 582–598.
[40] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, Sep. 1966.
[41] P. Van Overschee and B. De Moor, "Subspace algorithms for the stochastic identification problem," *Automatica*, vol. 29, no. 3, pp. 649–660, May 1993.
[42] X. Wang, L. Miranda-Moreno, and L. Sun, "Hankel-structured tensor robust PCA for multivariate traffic time series anomaly detection," 2021, *arXiv:2110.04352*.
[43] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM J. Imag. Sci.*, vol. 6, no. 3, pp. 1758–1789, Jan. 2013.
[44] J. Xue, Y. Zhao, Y. Bu, W. Liao, J. C. Chan, and W. Philips, "Spatial-spectral structured sparse low-rank representation for hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 30, pp. 3084–3097, 2021.
[45] R. Yamamoto, H. Hontani, A. Imakura, and T. Yokota, "Fast algorithm for low-rank tensor completion in delay-embedded space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 2048–2056.
[46] T. Yokota, B. Erem, S. Guler, S. K. Warfield, and H. Hontani, "Missing slice recovery for tensors using a low-rank model in embedded space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8251–8259.
[47] T. Yokota and H. Hontani, "Simultaneous visual data completion and denoising based on tensor rank and total variation minimization and its primal-dual splitting algorithm," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3843–3851.
[48] T. Yokota, Q. Zhao, and A. Cichocki, "Smooth PARAFAC decomposition for tensor completion," *IEEE Trans. Signal Process.*, vol. 64, no. 20, pp. 5423–5436, Oct. 2016.

[49] K. Zhang, M. Wang, S. Yang, and L. Jiao, "Spatial–spectral-graph-regularized low-rank tensor decomposition for multispectral and hyperspectral image fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 4, pp. 1030–1040, Apr. 2018.
[50] Z. Zhang and S. Aeron, "Exact tensor completion using t-SVD," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1511–1526, Mar. 2017.
[51] Q. Zhao, L. Zhang, and A. Cichocki, "Bayesian CP factorization of incomplete tensors with automatic rank determination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1751–1763, Sep. 2015.
[52] P. Zhou, C. Lu, Z. Lin, and C. Zhang, "Tensor factorization for low-rank tensor completion," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1152–1163, Mar. 2018.
[53] M. Zhu and T. Chan, "An efficient primal-dual hybrid gradient algorithm for total variation image restoration," *UCLA Cam Rep.*, vol. 34, pp. 8–34, May 2008.

HIROMU TAKAYAMA received the B.E. and M.E. degrees in engineering from the Nagoya Institute of Technology, Nagoya, Japan, in 2019 and 2021, respectively, where he is currently pursuing the Ph.D. degree in engineering. His research interests include matrix/tensor factorization, signal/image processing, pattern recognition, and machine learning.



TATSUYA YOKOTA (Senior Member, IEEE) received the Ph.D. degree in engineering from the Tokyo Institute of Technology, Tokyo, Japan, in 2014. From 2011 to 2014, he was a Junior Researcher with the Laboratory for Advanced Brain Imaging Signal Processing (ABSP), RIKEN Brain Science Institute (BSI), Wako, Japan. From 2014 to 2016, he was a Research Scientist with the Laboratory for ABSP and a Visiting Research Scientist with the TOYOTA Collaboration Center, RIKEN BSI. He is currently an Associate Professor with the Department of Computer Science, Nagoya Institute of Technology, Nagoya, Japan, and a Visiting Scientist with the Tensor Learning Team, RIKEN Center for Advanced Intelligence Projects, Tokyo. His research interests include matrix/tensor factorization, signal/image processing, pattern recognition, and machine learning.