Smooth PARAFAC Decomposition for Tensor Completion

Tatsuya Yokota, *Member, IEEE*, Qibin Zhao, *Member, IEEE*, and Andrzej Cichocki, *Fellow, IEEE*

***Abstract*—In recent years, low-rank based tensor completion, which is a higher order extension of matrix completion, has received considerable attention. However, the low-rank assumption is not sufficient for the recovery of visual data, such as color and 3D images, when the ratio of missing data is extremely high. In this paper, we consider "smoothness" constraints as well as low-rank approximations and propose an efficient algorithm for performing tensor completion that is particularly powerful regarding visual data. The proposed method admits significant advantages, owing to the integration of smooth PARAFAC decomposition for incomplete tensors and the efficient selection of models in order to minimize the tensor rank. Thus, our proposed method is termed as "smooth PARAFAC tensor completion (SPC)." In order to impose the smoothness constraints, we employ two strategies, total variation (SPC-TV) and quadratic variation (SPC-QV), and invoke the corresponding algorithms for model learning. Extensive experimental evaluations on both synthetic and real-world visual data illustrate the significant improvements of our method, in terms of both prediction performance and efficiency, compared with many state-of-the-art tensor completion methods.**

***Index Terms*—Tensor completion for images, smoothness, low-rank tensor approximation, CP model, PARAFAC model, total variation (TV), quadratic variation.**

I. Introduction

COMPLETION is a procedure that facilitates the estimation of the values of missing elements of array data, using only the available elements and structural properties of the data. Clearly, if there is no relationship between the missing elements and the available elements, completion is not possible. However, real world data usually exhibits some correlations, latent factors, symmetry, continuity, or repetition, in which case completion is often possible. For example, when a vector consists of the sampled values of some continuous function that has several missing values, some interpolation methods can be used for completion, such as simple linear interpolation, spline interpolation, and polynomial interpolation. When a given matrix has several missing entries and the low-rank matrix factorization exists, then a low-rank structure can be used for completion, by approximating the given matrix using the low-rank factorization model. Such completion techniques are closely associated with computer vision, pattern recognition, and compressed sensing [16], [24], [38].

Techniques for vector/matrix completion have been comprehensively researched, and many sophisticated methods exist. Furthermore, techniques for "tensor" completion have attracted attention in recent years, because of their potential applications and flexibility. A tensor is a multi-dimensional array, vectors and matrices can be considered as first-order and second-order tensors, respectively. For example, the data for a color-image is a third-order tensor, because it consists of three color shading images of red, green, and blue. Similarly, the data for a color video is a fourth-order tensor, because it consists of multiple frames of color images. There have been several papers that have attempted to reconstruct color images using matrix completion [27], [30], [37]. In [27], the authors proposed to complete each color shading image separately, and subsequently concatenate them. However, such an approach ignores the natural multi-dimensional structure of tensors, and thus neglects some important information. In [30], [37], the authors applied a matrix completion technique to matrices that consist of set of similar patches selected by some patch matching algorithms. However, patch matching is generally time consuming, and it does not work well when there is an extremely high ratio of missing data.

On the other hand, recent tensor completion methods have shown significant progress in the completion of color images by exploiting the structural information of 3rd-order tensors. The state-of-the-art methods for tensor completion consist of approaches of two types. The first approach, called nuclear norm minimization, is based on the low-rank property [42], [23], [43]. The nuclear norm of a matrix is defined as the sum of all singular values, and is a convex envelope for the rank of matrix [52]. The nuclear norm of the tensor was also introduced and applied to tensor completion in [42]. It is defined as sum of all matrix nuclear norm for individual mode-matricization of tensor. Subsequently, some improved algorithms were proposed in [23], [43]. The minimization of tensor nuclear norms has also been applied in various contexts, such as error correction [39], [40], medical imaging [44], compression [36], and saliency detection [73]. The second approach involves the use of low-rank tensor

Manuscript received January 25, 2016; revised June 16, 2016; accepted June 19, 2016. Date of publication June 30, 2016; date of current version August 19, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Wei Liu. This work was supported by Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Number 15K16067 and 15K15955.

T. Yokota is with Nagoya Institute of Technology, 466-8555 Nagoya, Japan (e-mail: t.yokota@nitech.ac.jp).

Q. Zhao is with RIKEN Brain Science Institute, 351-0198 Saitama, Japan (e-mail: qbzhao@brain.riken.jp).

A. Cichocki is with RIKEN Brain Science Institute, 351-0198 Saitama, Japan and Systems research Institute, Polish Academy of Science, 01-447 Warsaw, Poland (e-mail: a.cichocki@riken.jp).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This includes MATLAB codes and test images for demonstrating the proposed tensor completion (named SPC) algorithm with synthetic data and natural color images. This material is 39.7 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2016.2586759

1053-587X © 2016 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

decomposition techniques, which have been proposed in [2], [12], [35], [50], [58], [62], [82]. Tensor factorization is a method for decomposing an Nth-order tensor into another Nth-order tensor of smaller size, termed as the "core tensor," and N factor matrices. There are two factorization models: Tucker decomposition [65] and polyadic decomposition [29]. The polyadic decomposition is also has several different names such as CANDECOMP (canonical decomposition) [9], PARAFAC (parallel factor) decomposition [28], and CP (CANDECOMP/PARAFAC) decomposition [32]. In the present paper, we usually refer to this model as PARAFAC/Polyadic Decomposition (PD) model. A core tensor of PD takes the form of a super-diagonal tensor, and a core tensor of Tucker decomposition takes the form of a general tensor. Thus, PD constitutes a special case of Tucker decomposition. Similarly to the singular value decomposition (SVD) of matrix factorization, the minimum size of a core tensor corresponds to the rank of the original tensor. In [12], the authors consider the low rank Tucker decomposition model for tensor completion and the minimization of the nuclear norms of individual factor matrices. In [82], the authors consider the low-rank PD model for tensor completion, by using a Bayesian framework to solve tuning parameter problems. Furthermore, the authors of [12] and [82] introduced a factor prior regularization, in order to yield improved results for visual data.

In this paper, we investigate smoothness constraints for PD model based tensor completion. Smoothness is an important property that is embedded in many examples of real world data, such as natural images/videos, some spectral signals, and biomedical data. For example, low-frequency (smooth) components of natural images computed by discrete cosine transform have higher magnitude than high-frequency components. This is a reason why the JPEG encoder is efficient and widely used in practice, and it indicates the usability of smoothness property for natural image processing. When we assume original images are smooth and additive noise are not smooth, a smoothness constraint helps to preserve a smooth original image and to remove non-smooth noise. In fact, matrix/tensor factorizations with smoothness constraints have many applications in which they are robust in the presence of noisy signals, such as blind source separation [79], [78], [80], video structuring [21], visual parts extraction [75], and genomic data analysis [71].

Total variation (TV) [54], which is defined as the l_1-norm of the difference of neighbor elements, is often used to impose piece-wise smoothness constraints. Many methods have applied the TV approach in image restoration and denoising [48], [26], [69], [6], [46], [10]. Furthermore, the TV constraint has been already applied to matrix completion in [17], [27], but until now it has not been investigated for tensors. Therefore, we apply the TV constraint to PD model based tensor completion. Furthermore, we investigate an additional smoothness constraint, which is defined as the "l_2-norm" of the difference between neighbor elements, and is a stronger constraint than the TV constraint. Our concept of smoothness is not same as in other existing smoothness based completion methods [54], [27], [12], [82]. The key point of our approach is to assume that "latent component vectors" of factor matrices are smooth, and different levels of smoothness are enforced on different components adaptively. The details of the novelty of our model are discussed in Section V-A.

In addition, we propose a new strategy for a rank determination for the smooth PD model. A general strategy of the above mentioned "state-of-the-arts" algorithms involves setting the upper-bound of the rank for a tensor and optimizing the rank by using some procedures to remove redundant components, such as singular value shrinkage. However, if we impose our smoothness constraint on a tensor, the number of required components increases, and we are not able to determine an upper-bound. Therefore, we propose a reverse strategy, estimating the rank of the tensor (i.e., the optimal number of rank-1 tensors) by increasing the rank step by step, starting with a rank-one tensor. Our algorithm will stop when the smooth PD model fits the observed data sufficiently well. We carried out experiments involving tensor completion problems by using several difficult benchmarks of color images with different types of missing pixels (e.g., text masked, scratched images, and random pixels missing), incomplete MRI data, and multi-way structured facial data.

The remainder of this paper is organized as follows. In Section I-A, we explain the notations used in this paper. Section II reviews several existing matrix/tensor completion methods. In Section III, we propose novel algorithms for tensor completion based on the smooth PD model. In Section IV, we investigate the performance and applications of our algorithms, and compare them with some state-of-the-art methods. In Section V, we discuss several aspects of our work. Finally, we present our conclusions in Section VI.

A. Preliminaries and Notations

A vector is denoted by a bold small letter $\boldsymbol{a} \in \mathbb{R}^I$. A matrix is denoted by a bold capital letter $\boldsymbol{A} \in \mathbb{R}^{I \times J}$. A higher-order ($N \geq 3$) tensor is denoted by a bold calligraphic letter $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. The ith entry of a vector $\boldsymbol{a} \in \mathbb{R}^I$ is denoted by $\boldsymbol{a}(i)$, and the (i,j)th entry of a matrix $\boldsymbol{A} \in \mathbb{R}^{I \times J}$ is denoted by $\boldsymbol{A}(i,j)$. The $(i_1, i_2, \ldots, i_N)$th entry of an Nth-order tensor $\boldsymbol{\mathcal{A}}$ is denoted by $\boldsymbol{\mathcal{A}}_{i_1 i_2 \cdots i_N}$, where $i_n \in \{1, 2, \ldots, I_n\}$ and $n \in \{1, 2, \ldots, N\}$. The Frobenius norm of an Nth-order tensor is defined by $||\boldsymbol{\mathcal{X}}||_F := \sqrt{\sum_{i_1, i_2, \ldots, i_N} \boldsymbol{\mathcal{X}}^2_{i_1 i_2 \cdots i_N}}$.

A mode-k unfolding (matricization) of a tensor $\boldsymbol{\mathcal{X}}$ is denoted as $\boldsymbol{X}_{(k)} \in \mathbb{R}^{I_k \times \Pi_{n \neq k} I_n}$. For example, a first mode unfolding of a third-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is given by $\boldsymbol{X}_{(1)}(i_1, (i_3 - 1)I_2 + i_2) = \boldsymbol{\mathcal{X}}_{i_1 i_2 i_3}$, where $i_n \in \{1, 2, \ldots, I_n\}$ and $n \in \{1, 2, 3\}$. A mode-k multiplication between a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ and a matrix/vector $\boldsymbol{A} \in \mathbb{R}^{R \times I_k}$ is denoted by $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_k \boldsymbol{A} \in \mathbb{R}^{I_1 \times \cdots \times I_{k-1} \times R \times I_{k+1} \times \cdots \times I_N}$, with entries given by $\boldsymbol{\mathcal{Y}}_{i_1 \cdots i_{k-1} r i_{k+1} \cdots i_N} = \sum_{i_k} \boldsymbol{\mathcal{X}}_{i_1 \cdots i_{k-1} i_k i_{k+1} \cdots i_N} \boldsymbol{A}(r, i_k)$, and we have $\boldsymbol{Y}_{(k)} = \boldsymbol{A}\boldsymbol{X}_{(k)}$.

II. Brief Review of Existing Methods

In this section, we explain the methodology for matrix/tensor completion based on the low-rank assumption, and review the several state-of-the-art methods.

Concerning matrix data, the most popular and basic approach for matrix completion is the minimization of the matrix rank, subject to fitting the available elements:

$$\underset{\boldsymbol{X}}{\text{minimize}} \ \ \text{rank}(\boldsymbol{X}), \ \ \text{s.t.} \ \ \boldsymbol{X}_\Omega = \boldsymbol{T}_\Omega, \tag{1}$$

where $\boldsymbol{X}$ is a completed output matrix, $\boldsymbol{T}$ is an incomplete input matrix, Ω denotes the indices of the available elements of $\boldsymbol{T}$, the equation $\boldsymbol{X}_\Omega = \boldsymbol{T}_\Omega$ means that $\boldsymbol{X}(i,j) = \boldsymbol{T}(i,j), \ \forall (i,j) \in \Omega$, and $\text{rank}(\boldsymbol{X})$ denotes the rank of matrix $\boldsymbol{X}$. Such a rank minimization approach has been applied in various research fields, such as machine learning [4], [5] and bioinformatics [64]. If an original matrix is structured as a low-rank matrix, then this rank minimization approach can be used to obtain an estimation of the ground truth matrix. However, rank is not convex function with respect to $\boldsymbol{X}$, and rank minimization is generally NP-hard [25]. Therefore, the nuclear-norm minimization method is widely used in practice:

$$\underset{\boldsymbol{X}}{\text{minimize}} \ \ ||\boldsymbol{X}||_*, \ \ \text{s.t.} \ \boldsymbol{X}_\Omega = \boldsymbol{T}_\Omega, \tag{2}$$

where $||\boldsymbol{X}||_* = \sum_i \sigma_i(\boldsymbol{X})$ denotes the nuclear norm, and $\sigma_i(\boldsymbol{X})$ denotes the ith largest singular value of the matrix $\boldsymbol{X}$. In order to solve the optimization problem (2), the following augmented Lagrange function can be applied:

$$\begin{aligned} &L_A(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{W}, \beta) \\ &\quad = ||\boldsymbol{X}||_* - \langle \boldsymbol{W}, \boldsymbol{X} - \boldsymbol{Y} \rangle + \frac{\beta}{2}||\boldsymbol{X} - \boldsymbol{Y}||_F^2, \end{aligned} \tag{3}$$

where $\boldsymbol{Y}_\Omega = \boldsymbol{T}_\Omega$, $\boldsymbol{W}$ is a Lagrangian multiplier matrix, $\beta > 0$ is a penalty (trade-off) parameter for the augmented term $||\boldsymbol{X} - \boldsymbol{Y}||_F^2$, and β is often increased in each iteration step for acceleration. This problem can be optimized by applying the alternating direction method of multipliers (ADMM) [8]. There are several existing papers that have applied the ADMM scheme to matrix or tensor completion problems [41], [23], [11], [43].

The smoothness property of data is often assumed for image completion. If we consider $\boldsymbol{X}$ to be a gray-scale image, and individual entries stand for the values of individual pixels, then the difference between neighbor pixels should typically be small. In [54], the minimization of the total variation (TV) was employed as a smoothness constraint, which was defined by

$$||\boldsymbol{X}||_{TV} := \sum_{i,j} \sqrt{\boldsymbol{X}_v(i,j)^2 + \boldsymbol{X}_h(i,j)^2}, \tag{4}$$

where $\boldsymbol{X}_v(i,j) := \boldsymbol{X}(i+1,j) - \boldsymbol{X}(i,j)$ and $\boldsymbol{X}_h(i,j) := \boldsymbol{X}(i,j+1) - \boldsymbol{X}(i,j)$. Image completion via TV minimization was proposed in [17], and formulated in the form of the following optimization problem:

$$\underset{\boldsymbol{X}}{\text{minimize}} \ \ ||\boldsymbol{X}||_{TV}, \ \ \text{s.t.} \ \boldsymbol{X}_\Omega = \boldsymbol{T}_\Omega.$$

This problem is convex, and can be solved using subgradient descent methods [17]. Since TV is not differentiable at $\boldsymbol{X}_v(i,j)^2 + \boldsymbol{X}_h(i,j)^2 = 0$, majorization-minimization (MM) approach [47], [55] was proposed recently. In MM approach, a quadratic upper-bound function of TV is minimized iteratively.

Furthermore, a combination of low-rank approximation and smoothness constraints was proposed in [27]. Usually, natural images yield both of these structural features, and this combination is quite efficient. The most straightforward approach for the combination of low-rank approximation and smoothness constraints is implemented by the following optimization problem:

$$\underset{\boldsymbol{X}}{\text{minimize}} \ \ ||\boldsymbol{X}||_* + \gamma||\boldsymbol{X}||_{TV}, \ \ \text{s.t.} \ \boldsymbol{X}_\Omega = \boldsymbol{T}_\Omega, \tag{5}$$

where $\gamma > 0$ is a trade-off parameter between nuclear norm minimization and TV minimization. In [27], a modified linear total variation was defined, as

$$\text{LTV}(\boldsymbol{X}) := \sum_{i,j} \{\boldsymbol{X}_v(i,j)^2 + \boldsymbol{X}_h(i,j)^2\}, \tag{6}$$

and a smooth low-rank matrix completion method was proposed in the form of

$$\underset{\boldsymbol{X}}{\text{minimize}} \ \ ||\boldsymbol{X}||_* + \gamma \text{LTV}(\boldsymbol{X}), \ \ \text{s.t.} \ \boldsymbol{X}_\Omega = \boldsymbol{T}_\Omega. \tag{7}$$

The optimization problem (7) was referred to as the linear total variation approximate regularized nuclear norm (LTVNN) minimization problem, and this problem can be solved using an ADMM like optimization scheme. This approach can be considered as the state-of-the-art method for low-rank image completion. However, LTVNN may be not useful for tensor completion. Concerning color images (third-order tensors), the authors separated an image into red, green, and blue frames, and applied the completion method to each color frame separately. However, when individual color frames are quite similar, such a separated method may not be effective.

Tensor completion is a natural extension of matrix completion with respect to the data structure, and it can use such structural information more effectively than matrix completion. A basic method for tensor completion was proposed using simple low-rank tensor completion (SiLRTC) in [43], and this was then formulated as the following constrained optimization problem:

$$\begin{aligned} &\underset{\boldsymbol{\mathcal{X}}, \boldsymbol{Y}^{(1)}, \ldots, \boldsymbol{Y}^{(N)}}{\text{minimize}} \ \ \sum_{i=1}^{N} \left\{ \alpha_i ||\boldsymbol{Y}^{(i)}||_* + \frac{\beta}{2}||\boldsymbol{X}_{(i)} - \boldsymbol{Y}^{(i)}||_F^2 \right\}, \\ &\qquad \text{s.t.} \ \ \boldsymbol{\mathcal{X}}_\Omega = \boldsymbol{\mathcal{T}}_\Omega, \end{aligned} \tag{8}$$

where $\boldsymbol{\mathcal{X}}$ is a completed output tensor, $\boldsymbol{\mathcal{T}}$ is an incomplete input tensor, $\boldsymbol{Y}^{(i)}$ is a low-rank matrix corresponding to the mode-i matricization form $\boldsymbol{X}_{(i)}$, and $\alpha_i > 0$ and $\beta > 0$ are weight parameters for individual cost functions. Essentially, this method attempts to minimize the tensor nuclear norm, defined as $\sum_i \alpha_i ||\boldsymbol{X}_{(i)}||_*$, which is a generalization of the matrix nuclear norm. Instead of the minimization of $\sum_i \alpha_i ||\boldsymbol{X}_{(i)}||_*$, we minimize the nuclear norm of alternate parameters $\boldsymbol{Y}^{(i)}$, the mean squared error between alternate parameters $\boldsymbol{Y}^{(i)}$, and the mode-i unfolding of $\boldsymbol{\mathcal{X}}$.

Note that when we strictly consider the minimization of $\sum_i \alpha_i ||\boldsymbol{X}_{(i)}||_*$, constraints $\boldsymbol{X}_{(i)} = \boldsymbol{Y}^{(i)}$ should be added. The high accuracy low rank tensor completion (HaLRTC) method [43] was proposed on the basis of this concept, by formulating

the following optimization problem:

$$\underset{\mathcal{X},\mathcal{Y}^{(1)},\ldots,\mathcal{Y}^{(N)}}{\text{minimize}} \quad \sum_{i=1}^{N} \alpha_i ||\boldsymbol{Y}^{(i)}||_*, \tag{9}$$
$$\text{s.t.} \quad \mathcal{X} = \mathcal{Y}^{(i)},\ \mathcal{X}_\Omega = \mathcal{T}_\Omega,$$

where $\mathcal{Y}^{(i)}$ is the tensor form of matrix $\boldsymbol{Y}^{(i)}$. The authors considered an augmented Lagrange function, which is an extension of (3), and can be minimized using the ADMM optimization scheme in similar manner.

Recently, an alternative and very efficient completion method has been proposed, called simultaneous tensor decomposition and completion (STDC) [12]. This method is based on Tucker decomposition, and minimizes the nuclear norm of individual factor matrices by solving the following optimization problem:

$$\underset{\mathcal{G},\boldsymbol{U}^{(1)},\ldots,\boldsymbol{U}^{(N)}}{\text{minimize}} \quad \sum_{i=1}^{N} \alpha_i ||\boldsymbol{U}^{(i)}||_* + \delta \text{tr}(\boldsymbol{\Phi}\boldsymbol{L}\boldsymbol{\Phi}^T) + \gamma ||\mathcal{G}||_F^2, \tag{10}$$
$$\text{s.t.} \quad \mathcal{X} = \mathcal{G} \times \{\boldsymbol{U}^{(i)}\}_{i=1}^N,\ \mathcal{X}_\Omega = \mathcal{T}_\Omega,$$

where $\mathcal{G}$ is a core tensor, $\boldsymbol{U}^{(i)}$ is a factor matrix, $\mathcal{G} \times \{\boldsymbol{U}^{(i)}\}_{i=1}^N := \mathcal{G} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \times_3 \ldots \times_N \boldsymbol{U}^{(N)}$ is the Tucker decomposition model of tensor, $\boldsymbol{\Phi} := (\boldsymbol{U}^{(1)} \otimes \cdots \otimes \boldsymbol{U}^{(N)})$ is a unified factor matrix that consists of all factor matrices, $\otimes$ is the Kronecker product, and $\boldsymbol{L}$ is a matrix designed by some prior information. The STDC method is based on the "low-rank Tucker decomposition" and "factor prior." The minimization of $\delta \text{tr}(\boldsymbol{\Phi}\boldsymbol{L}\boldsymbol{\Phi}^T)$ can be interpreted as enforcing the similarity between individual components, which plays the role of a regularization, and provides some smoothed results for the visual data. Thus, STDC can be considered as a tensor extension of low-rank and smooth matrix completion. The authors considered an augmented Lagrange function, in order to solve (10) in similar way.

HaLRTC [43] and STDC [12] can be characterized as "matricization based" tensor nuclear norm minimization. In contrast to this tensor nuclear norm, "PD based" tensor nuclear norm is defined as sum of super-diagonal entries of core tensor of PD model. It seems to be more natural extension of matrix nuclear norm, because PD model is formulated as a sum of rank-one tensors. There are several theoretical works of PD based tensor nuclear norm minimization for completion [56], [77]. However, calculation of PD based tensor nuclear norm is still an NP-hard problem [22].

To the best of our knowledge, the most efficient method for PD based tensor completion is fully Bayesian CANDECOMP/PARAFAC tensor completion with mixture prior (FBCP-MP) [82]. The FBCP-MP method is based on the "probabilistic low-rank PD" and "mixture prior." By using mixture prior, this probabilistic model enforces a kind of similarity between individual component vectors, similarly to STDC. Furthermore, FBCP-MP finds a PD with an appropriate tensor rank by using Bayesian inference, without need to tune or adjust any parameters.

Hence, we can categorize the state-of-the art tensor completion methods into three types: low-rank-nuclear-norm, low-rank-Tucker-decomposition, and low-rank-PD with optional smoothness constraints. The above mentioned papers on tensor completion did not discuss smoothness constraints in detail. However, smoothness is quite an important factor for visual data completion, which was explored in matrix completion problems [17], [51], [27].

III. Proposed New Method

In this section, we propose a new algorithm for PARAFAC decomposition (PD) based tensor completion with smoothness constraints. First, the unconstrained PD model can be formulated as

$$\mathcal{Z} = \sum_{r=1}^{R} g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}, \tag{11}$$

where $\circ$ stands for outer product of vectors, $\boldsymbol{u}_r^{(n)}$ are feature vectors, called "components", and g_r are scaling multipliers. By denoting factor matrices by $\boldsymbol{U}^{(n)} = [\boldsymbol{u}_1^{(n)}, \boldsymbol{u}_2^{(n)}, \ldots, \boldsymbol{u}_R^{(n)}]$ and a super-diagonal core tensor by $\mathcal{G}$, such that each super-diagonal element is $\mathcal{G}_{rr\ldots r} = g_r$, the PD model can be also denoted by $\mathcal{Z} = [\![\mathcal{G}; \boldsymbol{U}^{(1)}, \boldsymbol{U}^{(2)}, \ldots, \boldsymbol{U}^{(N)}]\!]$. The minimum number of components R needed to satisfy equation (11) is called the "tensor rank" of $\mathcal{Z}$. The exact PD with a minimal R is usually refereed to as canonical polyadic decomposition (CPD) or rank decomposition [33]. In general, when we impose the smoothness constraint on the feature vectors $\boldsymbol{u}_r^{(n)}$, the minimum number of components R increases, because the smoothness constraint decreases the flexibility of the decomposition model. For this reason, it is difficult to determine the upper bound of the tensor rank of the original tensor when the smoothness constraint is imposed. In the STDC [12] and FBCP-MP [82] approaches, the upper bound of the Tucker/tensor rank is first determined as some large value, and the subsequent Tucker/tensor rank is then estimated by decreasing the number of components in each iteration. In this paper, we propose a completely different approach, which does not require the determination of the upper bound of the tensor rank, because in the proposed method we increase the number of components R gradually from 1 up to its optimal value. We call the new method "smooth PARAFAC tensor completion" (SPC).

A. Fundamental Problem for the Fixed Rank SPC (FR-SPC)

In this section, we consider the fixed rank version of SPC (FR-SPC). The optimization problem for FR-SPC is formulated as

$$\underset{\mathcal{G},\boldsymbol{U}^{(1)},\ldots,\boldsymbol{U}^{(N)}}{\text{minimize}} \quad \frac{1}{2}||\mathcal{X} - \mathcal{Z}||_F^2 + \sum_{r=1}^{R} \frac{g_r^2}{2} \sum_{n=1}^{N} \rho^{(n)} ||\boldsymbol{L}^{(n)} \boldsymbol{u}_r^{(n)}||_p^p,$$
$$\text{s.t.} \quad \mathcal{Z} = \sum_{r=1}^{R} g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)},$$

$$\boldsymbol{\mathcal{X}}_\Omega = \boldsymbol{\mathcal{T}}_\Omega,\ \boldsymbol{\mathcal{X}}_{\bar{\Omega}} = \boldsymbol{\mathcal{Z}}_{\bar{\Omega}}, ||\boldsymbol{u}_r^{(n)}||_2 = 1,$$
$$\forall r \in \{1,\ldots,R\}, \forall n \in \{1,\ldots,N\}, \tag{12}$$

where $\boldsymbol{\mathcal{X}}$ is a completed output tensor in which the missing entries are filled using smooth PD approximation $\boldsymbol{\mathcal{Z}}$. Thus, the constraints $\boldsymbol{\mathcal{X}}_\Omega = \boldsymbol{\mathcal{T}}_\Omega$ and $\boldsymbol{\mathcal{X}}_{\bar{\Omega}} = \boldsymbol{\mathcal{Z}}_{\bar{\Omega}}$ stand for

$$\boldsymbol{\mathcal{X}}_{i_1 i_2 \ldots i_N} = \begin{cases} \boldsymbol{\mathcal{T}}_{i_1 i_2 \ldots i_N} & (i_1, i_2, \ldots, i_N) \in \Omega \\ \boldsymbol{\mathcal{Z}}_{i_1 i_2 \ldots i_N} & \text{otherwise} \end{cases}. \tag{13}$$

Next, $\boldsymbol{\rho} = [\rho^{(1)}, \rho^{(2)}, \ldots, \rho^{(N)}]^T$ is a smoothness parameter vector, $p \in \{1, 2\}$ is a parameter for selecting the types of smooth constraints, and the matrix $\boldsymbol{L}^{(n)} \in \mathbb{R}^{(I_n - 1) \times I_n}$ is a smoothness constraint matrix, typically defined as

$$\boldsymbol{L}^{(n)} := \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{pmatrix}. \tag{14}$$

The first term, $||\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{Z}}||_F^2$, of the objective function in (12) represents the mean squared error (MSE) between the values of the observed entries $\boldsymbol{\mathcal{T}}_\Omega$ and the PD model $\boldsymbol{\mathcal{Z}}_\Omega$, because $\boldsymbol{\mathcal{X}}_\Omega = \boldsymbol{\mathcal{T}}_\Omega$ and $\boldsymbol{\mathcal{X}}_{\bar{\Omega}} = \boldsymbol{\mathcal{Z}}_{\bar{\Omega}}$. Thus, the minimization of the first term of the objective function in (12) provides a PD of a given tensor $\boldsymbol{\mathcal{T}}$.

The second term of the objective function in (12) is a penalty term, which assures smooth component vectors $\boldsymbol{u}_r^{(n)}$. Note that we have $||\boldsymbol{L}^{(n)}\boldsymbol{u}_r^{(n)}||_p^p = \sum_{i=1}^{I_n - 1} |\boldsymbol{u}_r^{(n)}(i) - \boldsymbol{u}_r^{(n)}(i+1)|^p$, and the minimization of this non-smoothness measure enforces the smoothness of individual feature vectors $\boldsymbol{u}_r^{(n)}$. When $p = 1$, the constraint term becomes the total variation (TV), and when $p = 2$ it becomes the quadratic variation (QV). In contrast to the LTVNN [17], [27], which imposes the smoothness constraint into the surface (output matrix), our approach imposes the smoothness constraint on the vectors (latent components). The details of the difference between our approach and other existing smoothing methods are discussed in Section V-A.

Note that the adaptive factors g_r^2 are depending on penalty levels of the smoothness constraint terms in (12). This allows us to adaptively enforce the different levels of smoothness for different components. For image completion, the proposed method decomposes an image adaptively into a strong smooth background and a weaker smooth foreground. In other words, considering Fourier domain of image data, low-frequency components are generally stronger than high-frequency components. Therefore, we propose to use different penalty levels of individual rank-1 components depending on their scales g_r.

B. Derivation of the FR-SPC Algorithm

We solve the optimization problem (12) using the hierarchical alternating least squares (HALS) approach [13], [14]. The HALS algorithm, included in block coordinate descent scheme, considers 'feature-wise' update which allows us to treat each unit-norm constraint separately. According to the HALS approach, we consider the minimization of the following local cost functions:

$$\begin{aligned} \underset{g_r, \boldsymbol{u}_r^{(1)}, \ldots, \boldsymbol{u}_r^{(N)}}{\text{minimize}} \quad & \frac{1}{2}||\boldsymbol{\mathcal{Y}}_r - \boldsymbol{\mathcal{Z}}_r||_F^2 + \frac{g_r^2}{2}\sum_{n=1}^{N} \rho^{(n)} ||\boldsymbol{L}^{(n)}\boldsymbol{u}_r^{(n)}||_p^p, \\ \text{s.t.} \quad & \boldsymbol{\mathcal{Z}}_r = g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)} \\ & [\boldsymbol{\mathcal{Y}}_r]_\Omega = \boldsymbol{\mathcal{T}}_\Omega^{(r)},\ [\boldsymbol{\mathcal{Y}}_r]_{\bar{\Omega}} = [\boldsymbol{\mathcal{Z}}_r]_{\bar{\Omega}}, \\ & ||\boldsymbol{u}_r^{(n)}||_2 = 1, \forall n \in \{1, \ldots, N\}, \end{aligned} \tag{15}$$

where $\boldsymbol{\mathcal{Y}}_r = \boldsymbol{\mathcal{X}} - \sum_{i \neq r} g_i \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \circ \cdots \circ \boldsymbol{u}_i^{(N)}$ and $\boldsymbol{\mathcal{T}}_\Omega^{(r)} = \boldsymbol{\mathcal{T}}_\Omega - [\sum_{i \neq r} g_i \boldsymbol{u}_i^{(1)} \circ \boldsymbol{u}_i^{(2)} \circ \cdots \circ \boldsymbol{u}_i^{(N)}]_\Omega$. The local-problem (15) only involves the rth components of PD with a fixed $\boldsymbol{\mathcal{X}}$.

In order to solve optimization problem (15), we update $\boldsymbol{u}_r^{(1)}, \boldsymbol{u}_r^{(2)}, \ldots, \boldsymbol{u}_r^{(N)}$, g_r, and reset $\boldsymbol{\mathcal{Y}}_r$ sequentially as follows:

$$\boldsymbol{u}_r^{(n)} \leftarrow \underset{\boldsymbol{u} \in \mathbb{U}^{I_n}}{\text{argmin}}\, F_p^{(r,n)}(\boldsymbol{u}), \tag{16}$$

$$g_r \leftarrow \underset{g}{\text{argmin}}\, F_p^{(r)}(g), \tag{17}$$

$$[\boldsymbol{\mathcal{Y}}_r]_{\bar{\Omega}} \leftarrow [g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}]_{\bar{\Omega}}, \tag{18}$$

where $F_p^{(r)}(g_r) = F_p^{(r,n)}(\boldsymbol{u}_r^{(n)}) = \frac{1}{2}||\boldsymbol{\mathcal{Y}}_r - g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}||_F^2 + \frac{g_r^2}{2}\sum_{n=1}^{N} \rho^{(n)} ||\boldsymbol{L}^{(n)}\boldsymbol{u}_r^{(n)}||_p^p$ for all n and r, and $\mathbb{U}^{I_n} := \{\boldsymbol{u} \in \mathbb{R}^{I_n} \mid ||\boldsymbol{u}||_2 = 1\}$ is a sub-set of unit vectors.

1) Update Rule for $\boldsymbol{u}_r^{(n)}$: The problem (16) can be considered as a unit-norm constrained optimization. When $p = 2$, objective function is a quadratic function with respect to $\boldsymbol{u}$ and it can be formulated as spherical constrained quadratic optimization [61]. When $p = 1$, the objective function is non-differentiable. Fortunately, many methods for solving this problem have been studied, such as sub-gradient methods [7]. To treat both cases (i.e., $p = 1$ and 2) equally, in this paper, we consider the gradient (or sub-gradient) based coefficient normalization update method [49], [20]. Although it does not guarantee global convergence generally, sticking into local minima of sub-optimization problem is not critical issue because the objective of sub-optimization is to decrease global cost function rather than to obtain strict solution of itself. As other options, several alternative optimization schemes can be also applied such as Lagrange multiplier method [15], tangent gradient [34] and optimization method on manifolds [1]. In order the simplify the notation, we denote by $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$ the k-th updates of $\boldsymbol{u}_r^{(n)}$ and $\partial F_p^{(r,n)}(\boldsymbol{u}_r^{(n)})$ in the iterative algorithm, respectively, where $\partial F_p^{(r,n)}(\cdot)$ is a gradient (or sub-gradient) of the objective function. Thus, the update $\boldsymbol{u}_{k+1}$ is given by

$$\boldsymbol{u}_{k+1} = \frac{\boldsymbol{u}_k - \alpha \boldsymbol{v}_k}{\sqrt{1 - 2\alpha \boldsymbol{u}_k^T \boldsymbol{v}_k + \alpha^2 \boldsymbol{v}_k^T \boldsymbol{v}_k}}, \tag{19}$$

where α a step size parameter, which should be tuned so that $F_p^{(r,n)}(\boldsymbol{u}_{k+1}) \leq F_p^{(r,n)}(\boldsymbol{u}_k)$. This update rule can be considered as enforcing a sub-gradient descent on the hyper-spherical surface $||\boldsymbol{u}_k||_2 = ||\boldsymbol{u}_{k+1}||_2 = 1$. We iterate (19) until convergence is achieved.

Next, we will investigate the gradient (or sub-gradient) of the objective function. The objective function $F_p^{(r,n)}(\boldsymbol{u}_k)$ can be simplified as

$$\frac{g_r^2}{2}\rho^{(n)}||\boldsymbol{L}^{(n)}\boldsymbol{u}_k||_p^p - g_r\boldsymbol{u}_k^T\boldsymbol{y}_r^{(n)} + \frac{1}{2}g_r^2\boldsymbol{u}_k^T\boldsymbol{u}_k, \tag{20}$$

where $\boldsymbol{y}_r^{(n)} := \text{vec}(\boldsymbol{\mathcal{Y}}_r \times_1 \boldsymbol{u}_r^{(1)T} \times_2 \cdots \times_{n-1} \boldsymbol{u}_r^{(n-1)T} \times_{n+1} \boldsymbol{u}_r^{(n+1)T} \times_{n+2} \ldots \times_N \boldsymbol{u}_r^{(N)T})$. Thus, the gradient (or sub-gradient) of the objective function is given by

$$\partial F_p^{(r,n)}(\boldsymbol{u}_k) = \begin{cases} \frac{g_r^2}{2}\rho^{(n)}\boldsymbol{L}^{(n)T}\text{SGN}(\boldsymbol{L}^{(n)}\boldsymbol{u}_k) - g_r\boldsymbol{y}_r^{(n)} + g_r^2\boldsymbol{u}_k & (p=1) \\ g_r^2\rho^{(n)}\boldsymbol{L}^{(n)T}\boldsymbol{L}^{(n)}\boldsymbol{u}_k - g_r\boldsymbol{y}_r^{(n)} + g_r^2\boldsymbol{u}_k & (p=2) \end{cases}, \tag{21}$$

where the vector function SGN is defined by

$$\text{SGN}(\boldsymbol{x}) = [\text{sgn}(x_1), \text{sgn}(x_2), \ldots, \text{sgn}(x_J)]^T, \tag{22}$$

$$\text{sgn}(x_j) = \begin{cases} 1 & (x_j > 0) \\ 0 & (x_j = 0) \\ -1 & (x_j < 0) \end{cases}, \tag{23}$$

for any $\boldsymbol{x} \in \mathbb{R}^J$. This definition of $\text{SGN}(\boldsymbol{x})$ provides the most unbiased sub-gradient of $||\boldsymbol{x}||_1$.

2) Update Rule for g_r: Because the problem in (17) is an unconstrained quadratic optimization, the unique solution can be obtained analytically. The objective function can be simplified as

$$\frac{1}{2}g_r^2 - g_r\left\langle \boldsymbol{\mathcal{Y}}_r, \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)} \right\rangle + \frac{1}{2}g_r^2\sum_{n=1}^{N}\rho^{(n)}\left\|\boldsymbol{L}^{(n)}\boldsymbol{u}_r^{(n)}\right\|_p^p. \tag{24}$$

Therefore, the update rule is given by

$$g_r \leftarrow \frac{\left\langle \boldsymbol{\mathcal{Y}}_r, \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)} \right\rangle}{\left(1 + \sum_{n=1}^{N}\rho^{(n)}\left\|\boldsymbol{L}^{(n)}\boldsymbol{u}_r^{(n)}\right\|_p^p\right)}. \tag{25}$$

Finally, the FR-SPC optimization scheme can be summarized as shown in Algorithm 1. According to [76], several iterations of inner-loop of 10–13th lines in Algorithm 1 may accelerate convergence speed for $N = 2$, however it is not necessary for $N \geq 3$ [31].

Algorithm 1: FR-SPC Algorithm.

1: **input:** $\boldsymbol{\mathcal{T}}$, Ω, R, p and $\boldsymbol{\rho}$.
2: **initialize:** $\{g_r, \{\boldsymbol{u}_r^{(n)} \in \mathbb{U}^{I_n}\}_{n=1}^N\}_{r=1}^R$, randomly;
3: Construct matrix $\boldsymbol{L}^{(n)}$ by (14); $\forall n \in \{1, \ldots, N\}$
4: $\boldsymbol{\mathcal{X}}_\Omega \leftarrow \boldsymbol{\mathcal{T}}_\Omega$;
5: $\boldsymbol{\mathcal{X}}_{\bar{\Omega}} \leftarrow (\sum_{r=1}^R g_r\boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)})_{\bar{\Omega}}$;
6: $\boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{X}} - \sum_{r=1}^R g_r\boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}$;
7: **repeat**
8: **for** $r = 1, \ldots, R$ **do**
9: $\boldsymbol{\mathcal{Y}}_r \leftarrow \boldsymbol{\mathcal{E}} + g_r\boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}$;
10: **for** $n = 1, \ldots, N$ **do**
11: $\boldsymbol{u}_r^{(n)} \leftarrow \text{argmin}_{\boldsymbol{u} \in \mathbb{U}^{I_n}} F_p^{(r,n)}(\boldsymbol{u})$;
12: **end for**
13: $g_r \leftarrow \frac{\langle \boldsymbol{\mathcal{Y}}_r, \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)} \rangle}{(1 + \sum_{n=1}^N \rho^{(n)}||\boldsymbol{L}^{(n)}\boldsymbol{u}_r^{(n)}||_p^p)}$;
14: $\boldsymbol{\mathcal{E}} \leftarrow \boldsymbol{\mathcal{Y}}_r - g_r\boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}$;
15: $\boldsymbol{\mathcal{E}}_{\bar{\Omega}} \leftarrow 0$;
16: **end for**
17: **until** Change of $||\boldsymbol{\mathcal{E}}||_F^2$ between before and after update is sufficiently small.
18: $\boldsymbol{\mathcal{Z}} \leftarrow \sum_{r=1}^R g_r\boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}$;
19: $\boldsymbol{\mathcal{X}}_{\bar{\Omega}} \leftarrow \boldsymbol{\mathcal{Z}}_{\bar{\Omega}}$;
20: **output:** $\boldsymbol{\mathcal{X}}$, $\boldsymbol{\mathcal{Z}}$.

Algorithm 2: Algorithm for Estimation of Optimal Number of Components R.

1: **input:** $\boldsymbol{\mathcal{T}}$, Ω, p, $\boldsymbol{\rho}$, and SDR.
2: $\varepsilon \leftarrow 10^{(-\text{SDR}/10)}||\boldsymbol{\mathcal{T}}_\Omega||_F^2$;
3: $R \leftarrow 1$;
4: **repeat**
5: $[\boldsymbol{\mathcal{Z}}, \boldsymbol{\mathcal{X}}] \leftarrow$ FR-SPC($\boldsymbol{\mathcal{T}}, \Omega, R, p, \boldsymbol{\rho}$);
6: $R \leftarrow R + 1$;
7: **until** $||\boldsymbol{\mathcal{Z}}_\Omega - \boldsymbol{\mathcal{T}}_\Omega||_F^2 \leq \varepsilon$
8: **output:** $\boldsymbol{\mathcal{Z}}$, $\boldsymbol{\mathcal{X}}$.

C. Model Selection for the Number of Components R

The key problem of FR-SPC is choosing the optimal number of components R. The PD model with too small an R is not able to fit the data, and the PD model with too large an R may result in over-fitting problems.

In order to estimate an optimal value of R, we gradually increase R until we achieve the desired fit by formulating the following optimization problem:

$$\begin{aligned} \underset{R}{\text{minimize}} \quad & R, \\ \text{s.t.} \quad & ||\boldsymbol{\mathcal{Z}}_\Omega - \boldsymbol{\mathcal{T}}_\Omega||_F^2 \leq \varepsilon, \\ & \boldsymbol{\mathcal{Z}} \in \mathbb{S}(R, p, \boldsymbol{\rho}), \end{aligned} \tag{26}$$

where $\mathbb{S}(R, p, \boldsymbol{\rho})$ is defined as a set of all possible solution tensors via FR-SPC problem (15) with R, p, and $\boldsymbol{\rho}$ for any input tensors, in other words $\mathbb{S}(R, p, \boldsymbol{\rho})$ is a tensor space spanned by R smooth components, and ε is an error bound parameter. The criterion (26) allows us to finds a tensor $\boldsymbol{\mathcal{Z}}$ represented by the smooth PD model, with the minimum number of smooth components that guarantees a sufficient accuracy to fit the input tensor $\boldsymbol{\mathcal{T}}$. In order to guarantee that the signal to distortion ratio (SDR) is bounded within a specific threshold, we can define the error bound as $\varepsilon = 10^{(-\text{SDR}/10)}||\boldsymbol{\mathcal{T}}_\Omega||_F^2$. Note that $\min_{\boldsymbol{\mathcal{Z}} \in \mathbb{S}(R,p,\boldsymbol{\rho})}\{||\boldsymbol{\mathcal{Z}}_\Omega - \boldsymbol{\mathcal{T}}_\Omega||_F^2\}$ is a monotonically non-increasing function with respect to R. Algorithm 2 is an

Algorithm 3: SPC Algorithm (Accelerated Version of Algorithm 2).

1: **input:** $\boldsymbol{\mathcal{T}}$, Ω, p, $\boldsymbol{\rho}$, SDR, and ν.
2: $\varepsilon \leftarrow 10^{(-\mathrm{SDR}/10)}||\boldsymbol{\mathcal{T}}_\Omega||_F^2$;
3: $\boldsymbol{\mathcal{X}}_\Omega \leftarrow \boldsymbol{\mathcal{T}}_\Omega$; $\boldsymbol{\mathcal{X}}_{\bar{\Omega}} \leftarrow$ average of $\boldsymbol{\mathcal{T}}_\Omega$;
4: Construct matrix $\boldsymbol{L}^{(n)}$ by (14); $\forall n \in \{1, \ldots, N\}$
5: $R \leftarrow 1$;
6: Initialize $\{\boldsymbol{u}_R^{(n)} \in \mathbb{U}^{I_n}\}_{n=1}^N$, randomly;
7: $g_R \leftarrow \left\langle \boldsymbol{\mathcal{X}}, \boldsymbol{u}_R^{(1)} \circ \boldsymbol{u}_R^{(2)} \circ \cdots \circ \boldsymbol{u}_R^{(N)} \right\rangle$;
8: $\boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{X}} - \sum_{r=1}^R g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}$;
9: $\boldsymbol{\mathcal{E}}_{\bar{\Omega}} = 0$;
10: $t \leftarrow 0$;
11: $\mu_t \leftarrow ||\boldsymbol{\mathcal{E}}||_F^2$;
12: **repeat**
13: Update $\{\{\boldsymbol{u}_r^{(n)}\}_{n=1}^N, g_r\}_{r=1}^R$ by the FR-SPC algorithm (8th-16th lines);
14: $\mu_{t+1} \leftarrow ||\boldsymbol{\mathcal{E}}||_F^2$;
15: **if** $\frac{|\mu_t - \mu_{t+1}|}{|\mu_{t+1} - \varepsilon|} \leq \nu$ **then**
16: $R \leftarrow R + 1$;
17: Initialize $\{\boldsymbol{u}_R^{(n)} \in \mathbb{U}^{I_n}\}_{n=1}^N$, randomly;
18: $g_R \leftarrow \left\langle \boldsymbol{\mathcal{E}}, \boldsymbol{u}_R^{(1)} \circ \boldsymbol{u}_R^{(2)} \circ \cdots \circ \boldsymbol{u}_R^{(N)} \right\rangle$;
19: $\boldsymbol{\mathcal{E}} \leftarrow \boldsymbol{\mathcal{E}} - g_R \boldsymbol{u}_R^{(1)} \circ \boldsymbol{u}_R^{(2)} \circ \cdots \circ \boldsymbol{u}_R^{(N)}$;
20: $\boldsymbol{\mathcal{E}}_{\bar{\Omega}} \leftarrow 0$;
21: **end if**
22: $t \leftarrow t + 1$;
23: **until** $\mu_t \leq \varepsilon$
24: $\boldsymbol{\mathcal{Z}} \leftarrow \sum_{r=1}^R g_r \boldsymbol{u}_r^{(1)} \circ \boldsymbol{u}_r^{(2)} \circ \cdots \circ \boldsymbol{u}_r^{(N)}$;
25: $\boldsymbol{\mathcal{X}}_{\bar{\Omega}} \leftarrow \boldsymbol{\mathcal{Z}}_{\bar{\Omega}}$;
26: **output:** $\boldsymbol{\mathcal{X}}$, $\boldsymbol{\mathcal{Z}}$.

implementation to solve (26), and it allows us to obtain a good estimation for R. However, this algorithm is quite time consuming. If the current value of R is too small to fit the PD model sufficiently well to the given data tensor during the iteration process, then we can stop the algorithm for the current R, increase as $R \leftarrow R + 1$, and run the algorithm again for the new increased R. In this procedure, we propose switching to an increased R at an early stage of the iterations if the following condition is met:

$$\frac{|\mu_t - \mu_{t+1}|}{|\mu_{t+1} - \varepsilon|} < \nu, \tag{27}$$

where $\mu_t = ||\boldsymbol{\mathcal{Z}}_\Omega^t - \boldsymbol{\mathcal{T}}_\Omega||_F^2$, $\boldsymbol{\mathcal{Z}}^t$ denotes the PD model at the iteration step t, and $\nu > 0$ is a stopping threshold (typically, $\nu = 0.01$). The left hand side of (27) is a measure of the convergence speed, so this condition means that when the convergence speed becomes substantially slow, we stop the iteration procedure for the current R. By incorporating this simple stopping criterion, we finally arrive at an improved and considerably accelerated algorithm for the automatic determination of the number of components R of FR-SPC. We call this method simply "smooth PARAFAC tensor completion with TV and QV minimization" (SPC-TV and SPC-QV), and it is

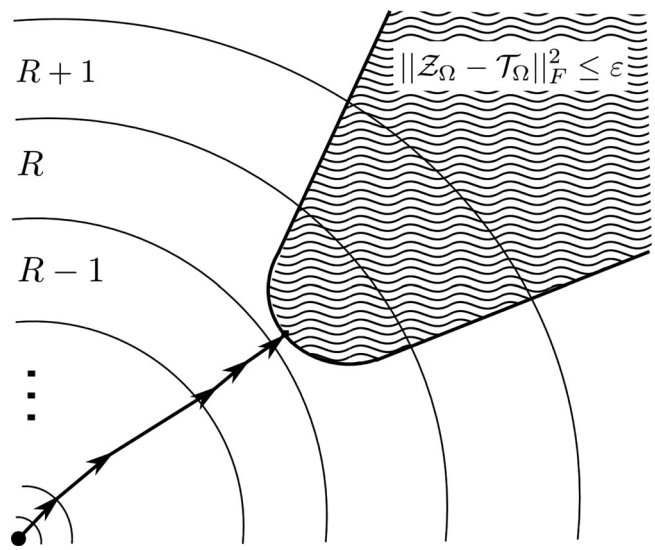


Fig. 1. Concept of optimization process in the SPC algorithm. An area R between arcs indicates the space of the tensor that is spanned by the components of R in the smooth PD model. An area that is filled with waves indicates the space of the tensor satisfying the condition $||\boldsymbol{\mathcal{Z}}_\Omega - \boldsymbol{\mathcal{T}}_\Omega||_F^2 \leq \varepsilon$. Concatenated arrows show the optimization process of the SPC algorithm, which gradually increases the number of components R from a rank-1 tensor to its end point, when the criterion is satisfied.

summarized in Algorithm 3. Fig. 1 presents the concept of the optimization process of the SPC algorithm, and its details are provided in Section V. Our algorithms have been implemented in MATLAB, and are available for readers at https://sites.google.com/site/yokotatsuya/home/software.

Because the best low-rank approximation problem for smooth PD model may be often ill-posed in the same way as for unconstrained PD model [18], our algorithm allows some redundant terms in smooth PD model. We focus on "good approximation" rather than "minimum rank decomposition", since estimation of number of components for smooth PD model is difficult, and its accuracy does not need to be very high under condition that the reconstructed tensor is sufficiently well approximated in practice.

IV. Experimental Results

In the experiments to be described here, we apply our proposed algorithms to one synthetic and several real-world visual datasets, to demonstrate the advantages of our proposed algorithms compared with the existing state-of-the-art methods LTVNN [27], HaLRTC [43], STDC [12], and FBCP-MP [82].

A. Convergence Properties Using a Synthetic Third-Order Tensor

Fig. 2 shows a visualization of the iso-surface of a synthetic 3rd-order tensor, and its completion results by achieved by applying the SPC algorithm with TV and QV constraints. The synthetic tensor was constructed using a combination of four multi-dimensional Gaussian functions, and 80% and 98% of the voxels were randomly removed in its incomplete tensor. We set SDR = 50 dB, $\nu = 0.01$, $\boldsymbol{\rho} = [0.01, 0.01, 0.01]$ for TV

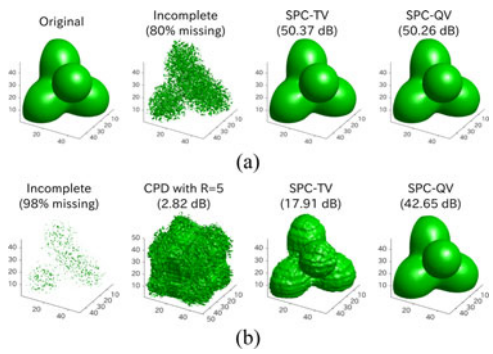


Fig. 2. Iso-surface visualization of synthetic data and completion results using SPC-TV and SPC-QV. (a) 80% missing, (b) 98% missing.

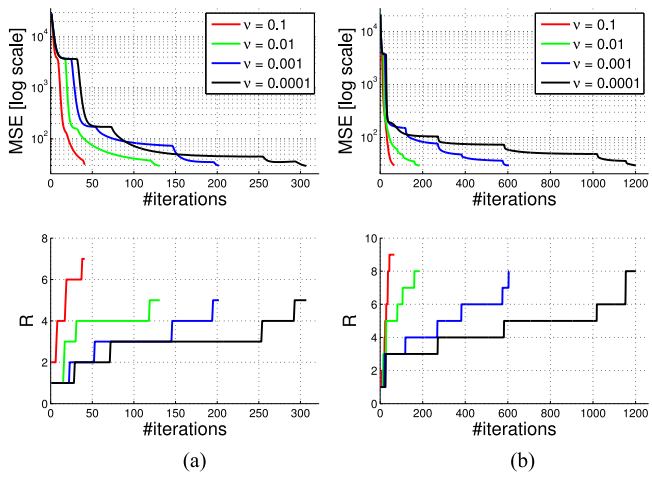


Fig. 3. Convergence of mean squared error between $\mathcal{Z}_\Omega$ and $\mathcal{T}_\Omega$, and number of components R as a function of the number of iterations in the SPC algorithm using the synthetic data with missing voxels rate of 80%. We performed simulations for various values of the stopping threshold $\nu \in \{0.1, 0.01, 0.001, 0.0001\}$. Other parameters were set as SDR = 30 dB, $\boldsymbol{\rho} = [0.01, 0.01, 0.01]$ for TV smoothing, and $\boldsymbol{\rho} = [1.0, 1.0, 1.0]$ for QV smoothing. (a) TV, (b) QV.

smoothing and $\boldsymbol{\rho} = [1.0, 1.0, 1.0]$ for QV smoothing. For reference, we applied unconstrained rank-5 CP decomposition (CPD) ('cpd.m' in tensorlab [59]) into 98% missing tensor. From the result, only low-rank structure assumption is not sufficient for highly missing data, and smoothness constraints dramatically improve the completion accuracy.

Fig. 3 shows the plots of the mean of squared errors (MSE) between the tensors $\mathcal{Z}_\Omega$ and $\mathcal{T}_\Omega$, and the number of estimated components R with respect to number of iterations of the SPC algorithm, with $\boldsymbol{\rho} = [0.01, 0.01, 0.01]$ for TV smoothing and $\boldsymbol{\rho} = [1.0, 1.0, 1.0]$ for QV smoothing. In addition, we set SDR = 30 dB and applied various switching threshold values of $\nu \in \{0.1, 0.01, 0.001, 0.0001\}$. We used the synthetic smooth third-order tensor with a random 80% of voxels missing. From Fig. 3, we can see the monotonic convergence of MSE and the change in the number of components of R are closely related to the change of the MSE. Note that the number of components, R, is updated when the convergence speed of the MSE becomes slow. Furthermore, the algorithm is seen to require a much larger

Fig. 4. Test images are 256×256 pixels color images, which consist of three layers of red, green, and blue color images.

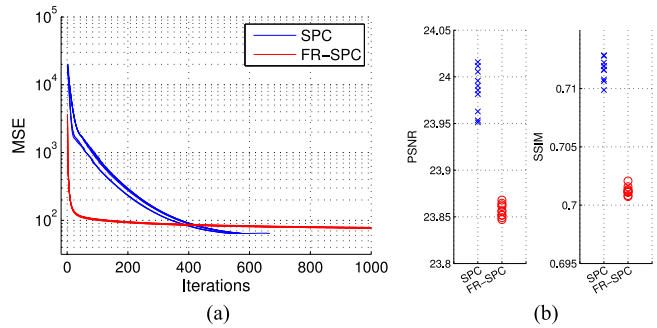


Fig. 5. Comparison of the SPC and FR-SPC algorithms. (a) Convergence curves, (b) PSNR and SSIM statistics.

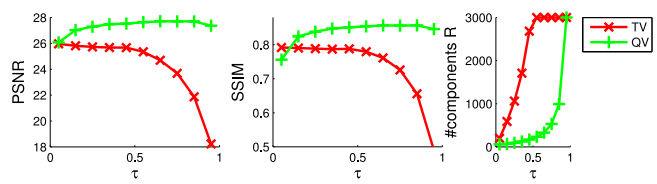


Fig. 6. Results of PSNR, SSIM, and the number of components of the SPC algorithms, using the image "Lena" with a missing ratio of 80%. We tested various values of $\boldsymbol{\rho} = [\rho, \rho, 0]^T$, where ρ is calculated by $\rho = \tau/(1-\tau)$. The algorithms were relatively insensitive to the wide variation of the parameters.

number of iterations for smaller values of ν. However, the results for the final number of components R were the same for different cases with $\nu \geq 0.01$. This means that for too small a value of ν the convergence of the algorithm is relatively slow. According to our extensive experiments, $\nu = 0.01$ is an acceptable default value.

B. Color Image Completion

In this experiment, we applied our proposed and existing methods to 10 color-images (Fig. 4). The size of all color images is $256 \times 256 \times 3$, and we generated incomplete data by deleting elements of the images randomly, with several different missing ratios $\in \{60, 70, 80, 90, 95\%\}$. For the images "Giant" and "Wasabi," all color elements of individual pixels (so called dead pixels) were deleted for all missing ratios. In order to evaluate the quality of completion, we used the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [70]. PSNR is defined as $10\ \log_{10}(255^2/\text{MSE})$, and SSIM is defined by using the average and variance parameters of the local small blocks in an image. As an improved evaluation measure, SSIM is usually considered as an improvement on PSNR, because SSIM evaluates both the error and a kind of "local smoothness" of images for visual quality. In this experiment, we generally set SDR = 25 dB and $\nu = 0.01$.

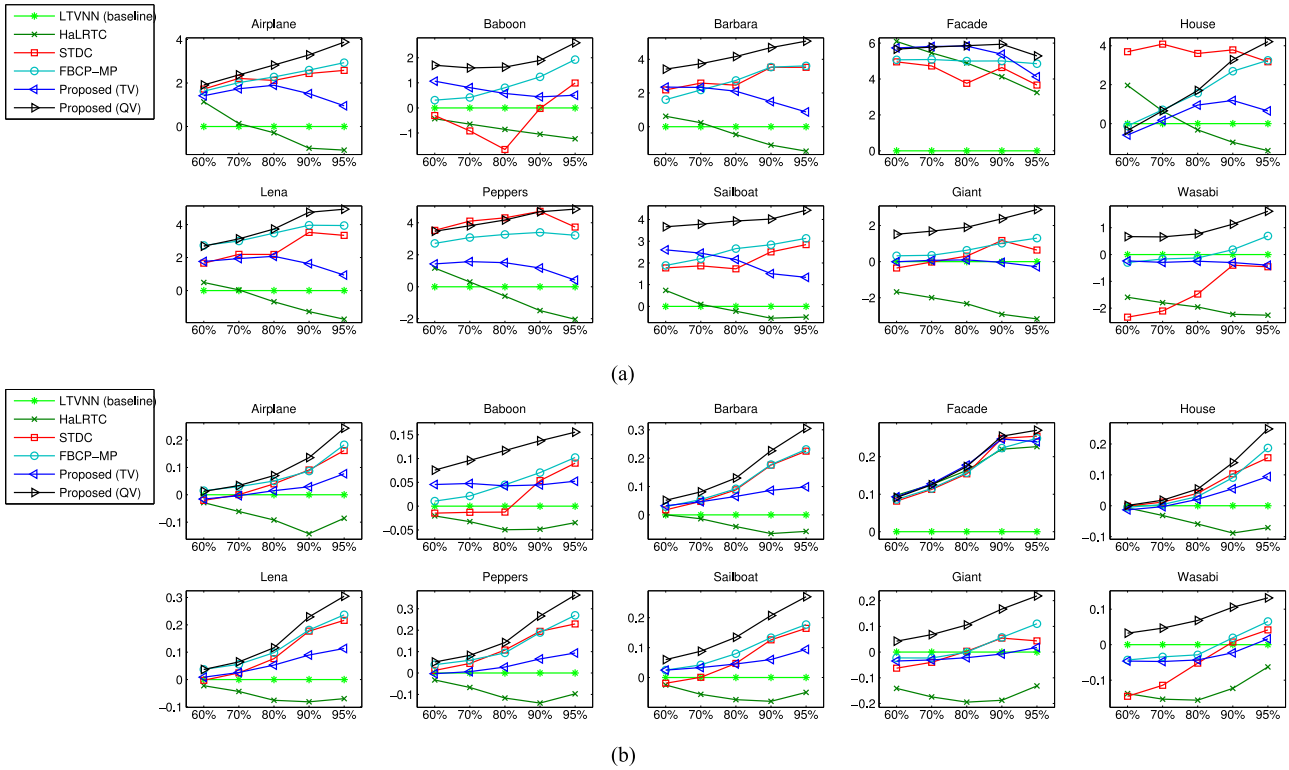


Fig. 7. Comparison of performances (PSNR and SSIM subtracted by baseline (LTVNN)) for all benchmark images and for various missing rates (from 60% to 95%) obtained using the proposed methods (SPC-TV and SPC-QV) and the state-of-the-art algorithms LTVNN, HaLRTC, STDC, and FBCP-MP. (a) Relative PSNR subtracted by LTVNN, (b) Relative SSIM subtracted by LTVNN.

1) Convergence Behavior of SPC and FR-SPC Algorithms: Fig. 5 shows the convergence curves, PSNR and SSIM for 10 runs with random initializations in the SPC and FR-SPC algorithms, using "Lena" with a random 95% of pixels missing, with $p = 2$ and $\boldsymbol{\rho} = [0.5, 0.5, 0]$. In both methods, the number of components is the same. Interestingly, FR-SPC converged to local minima that are larger than the convergence points for SPC. Furthermore, the PSNR and SSIM values achieved by SPC were significantly better than those for the FR-SPC algorithm. This implies that the "rank-increasing" approach results in an algorithm that converges to better local minima.

2) Performances for Various Values of ρ: Fig. 6 shows the PSNR, SSIM, and number of components R for different values of the smoothing parameter $\boldsymbol{\rho}$. A benchmark image, "Lena," with a random 80% of pixels missing, was used in this experiment. We applied the SPC-TV and SPC-QV algorithms, with various values of $\boldsymbol{\rho} = [\rho, \rho, 0]$. We defined $\rho := \tau/(1-\tau)$, and tested for a wide range of $\tau \in [0.05, 0.95]$. In this experiment, we set the maximum values for R as 3000, because of the limitations of the memory. We found that larger values of ρ resulted in considerable increases in PSNR and SSIM in QV/TV smoothing. However, too large a ρ substantially decreases the performances in TV smoothing. Furthermore, a larger ρ requires a larger number of components R to achieve a more precise fit, and also a larger computational cost. The SPC algorithm with TV smoothing required a larger number of components R than QV smoothing, and usually the performances, in terms of PSNR and SSIM, were inferior in comparison with QV smoothing. According to our results, QV smoothing performs, in general, better than TV smoothing for tensor completion. We discuss our interpretation of these results in Section V-B.

3) Comparison with the State-of-the-Art Methods: Fig. 7 shows the results of tensor completions using all 10 benchmark images with various random pixel missing ratios using the proposed method with TV and QV smoothing, and compares these with the performances of four state-of-the-art methods. The parameters for the SPC algorithms were set as $\boldsymbol{\rho} = [0.05, 0.05, 0]$ for TV smoothing, and $\boldsymbol{\rho} = [1.0, 1.0, 0]$ for QV smoothing. The hyper-parameters for the other methods were optimally tuned manually. Interestingly, LTVNN outperformed HaLRTC for several images with respect to SSIM. However, it is difficult to say whether LTVNN outperformed HaLRTC with respect to PSNR. This implies that the smoothness constraint improves the structural similarity of images. Thus, we can conclude that the smoothed methods (i.e., LTVNN, STDC, FBCP-MP, and the proposed method) are evaluated better by SSIM than PSNR. The proposed method with TV smoothing was inferior to STDC and FBCP-MP in most cases (e.g., "Peppers" and "House"). However, the proposed method with QV smoothing considerably outperformed the all of the existing methods for all of our benchmarks.

Fig. 8 presents several results for the completed images, using the existing methods and the proposed method. We prepared a text masked image of "Barbara," a scratched image of

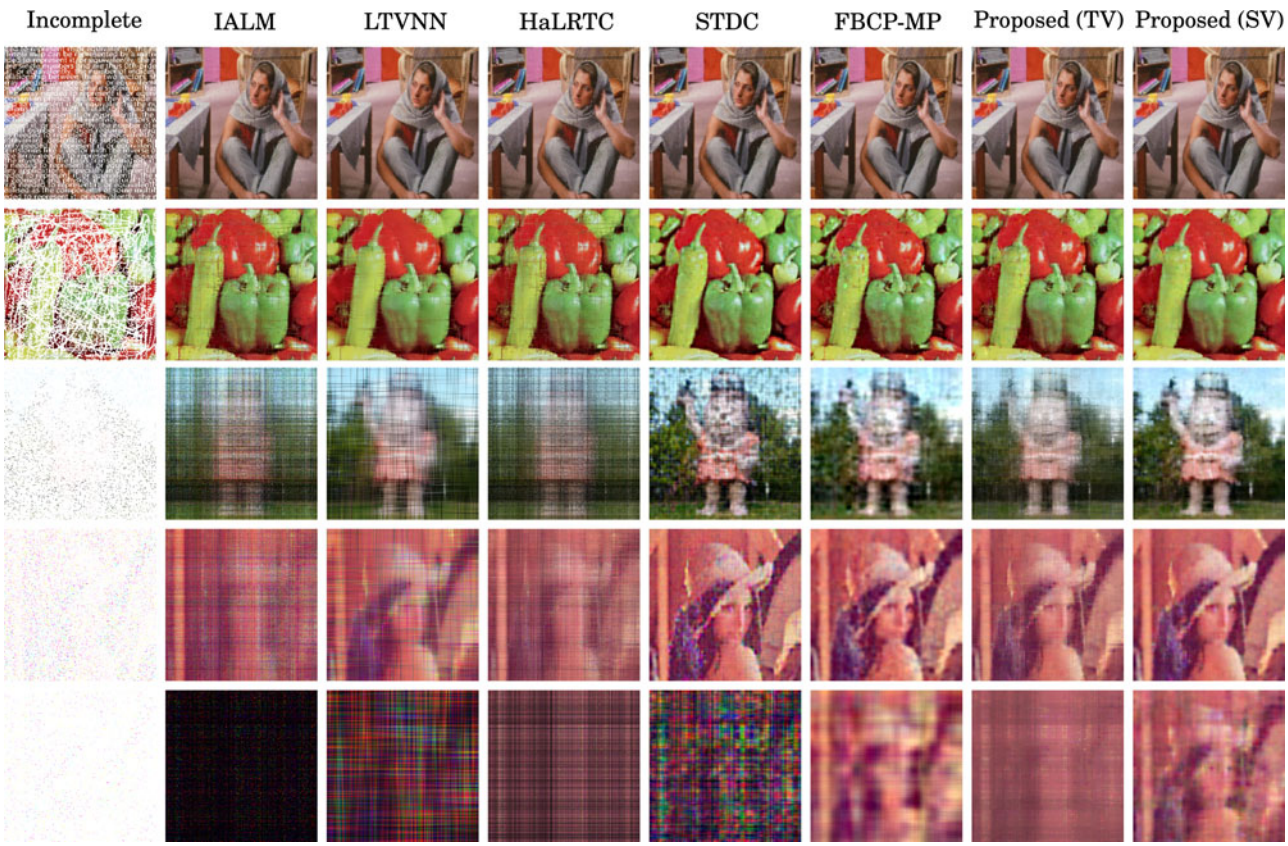


Fig. 8. Simulation results for various incomplete images (top to bottom): text masked image, scratched image, random dead pixels missing (95%), and random voxels missing (95% and 99%). Incomplete images are shown in the first column, and reconstructed images are shown in the other columns.

"Peppers," an incomplete image with random pixels missing (dead pixels) of "Giant," and two incomplete images with random voxels missing of "Lena." In addition, for the first time we tested an extreme case where the "99%" of pixels are missing, and such a large missing ratio may constitute an interesting challenge in this research field. There were rather small difference between the performances of most of the methods in the first and second rows. However, we observed significant differences for the images missing random pixels/voxels, especially when the ratio of missing pixels was high. In the reconstructed images in the last three rows for the non-smooth methods (i.e., IALM and HaLRTC), it was difficult to recognize the shapes of objects. In the reconstructed images in the second to last row, the smooth matrix completion method (i.e., LTVNN) was not able to reconstruct the woman's facial features, but it is not difficult to recognize the woman's facial features in the second to last row where STDC, FBCP-MP, and SPC were applied. For the last row with a 99% missing ratio, the shape of "Lena" could be recognized only for FBCP-MP and the proposed SPC-QV, but the performance of SPC-QV was better than for FBCP-MP.

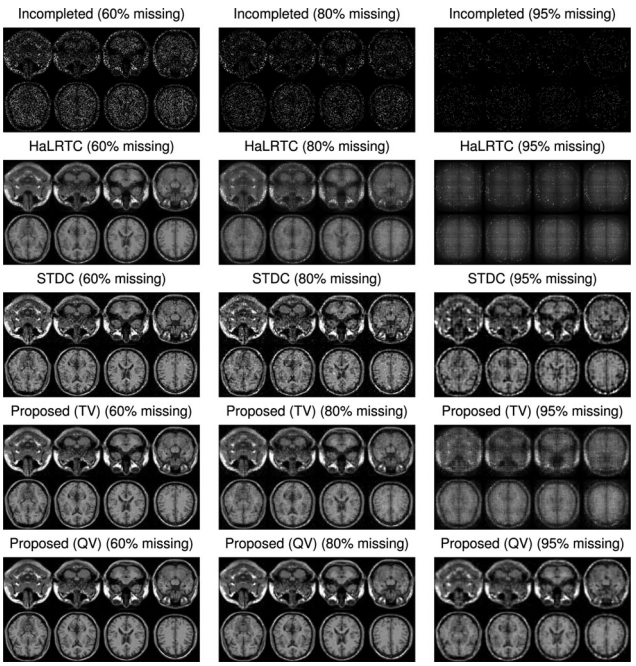


Fig. 9. Results of tensor completion for MRI data by using HaLRTC, STDC, and the proposed SPC -TV and SPC -QV methods for various missing rates.

C. MRI Image Completion

Fig. 9 shows the results for several slices of MRI 3D-images (of size $109 \times 91 \times 91$), with 60%–95% of voxels missing, obtained by HaLRTC, STDC, and SPC with TV and QV smoothing. Since the MRI image is smooth in all three dimensions (modes), the hyper-parameters of the SPC algorithm were set as $\boldsymbol{\rho} = [0.01, 0.01, 0.01]$ for TV smoothing, and $\boldsymbol{\rho} = [0.5, 0.5, 0.5]$

TABLE I
SIGNAL TO DISTORTION RATIO OF MRI COMPLETION

Methods	60%	70%	80%	90%	95%
HaLRTC	14.59	12.40	9.95	6.93	4.99
STDC	14.97	13.78	13.28	12.81	11.46
Proposed (TV)	18.13	16.23	14.16	11.20	8.36
Proposed (QV)	**20.65**	**19.09**	**17.29**	**14.73**	**12.79**

for QV smoothing. The hyper-parameters of other methods were tuned optimally and manually. Table I shows the performances (SDR) of the individual algorithms. From Fig. 9 and Table I, we can observe that the SPC algorithm succeeded in completing the incomplete MRI 3D-images, even with a 90%-95% missing ratio of voxels, and also significantly outperformed the other methods, with respect to the SDR.

D. 4th-Order Tensor Completion Using a CMU Faces Dataset

Next, we applied the three completion methods to a facial image database, provided by Carnegie Mellon University (CMU), called "CMU faces" [57]. The CMU dataset consists of 6930 facial images of size 32×32 pixels. The 6930 facial images represent 30 individual people, 11 poses, and 21 kinds of illumination (i.e., $6930 = 30 \times 11 \times 21$). Thus, the size of the CMU dataset is $(30 \times 11 \times 21 \times 1024)$. Since each facial image is generally smooth in this dataset, but the neighboring people, poses, and illuminations are not always similar, we enforced the smoothness constraint in only the final fourth dimension. Each facial image is described as a 1024-dimensional vector, which is a vectorized from a 32×32-gray-scaled-image. Therefore, in this experiment we employed a smoothness constraint matrix that applies a combination of vertical and horizontal smoothness:

$$\boldsymbol{L}^{(n)} = \begin{pmatrix} \boldsymbol{L}_v^{(n)} \\ \boldsymbol{L}_h^{(n)} \end{pmatrix}, \tag{28}$$

where $\boldsymbol{L}_v^{(n)}$ and $\boldsymbol{L}_h^{(n)}$ are the vertical and horizontal smoothness matrices, respectively. The smoothness parameters of the SPC were set as $\boldsymbol{\rho} = [0, 0, 0, 0.01]$ for TV smoothing and $\boldsymbol{\rho} = [0, 0, 0, 0.1]$ for QV smoothing. Fig. 10 shows a section of the original and incomplete CMU faces, with 80% of faces missing, and its completed results obtained by the HaLRTC, STDC, and SPC algorithms. In this case, HaLRTC failed to complete the faces, while STDC provided several broken faces. However SPC achieved excellent results.

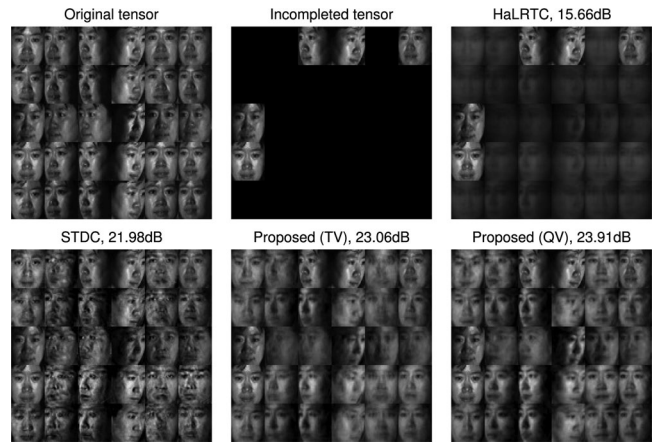


Fig. 10. Results of tensor completion for the CMU face dataset by using HaLRTC, STDC, and the proposed SPC -TV and SPC -QV methods. Values of SDR [dB] are described in each subfigure. Considerable improvements in performance were achieved.

V. DISCUSSIONS

A. Smooth PARAFAC Decomposition Model

In Section II, we introduced three methods for "low-rank" and "smooth" matrix/tensor completion: LTVNN [27], STDC [12], and FBCP-MP [82]. We will now expand on the differences between these models and our proposed model.

First, our smooth tensor decomposition model is based on enforcing the "local similarity between neighbor elements in each component," i.e., the vectors of the factor (component) matrices. In contrast, LTVNN [27] imposes the smoothness constraint on the "output matrix itself," but not on the "components." Furthermore, the smoothness of both STDC [12] and FBCP-MP [82] is designed to enforce the similarity (or smoothness) between "individual components." Thus our method is quite different from above methods [27], [12], [82]. In [60], a generalized framework for tensor factorization with missing entries is proposed and smooth factor constraint is included in this framework. However, it is considering fixed-rank optimization problem, and does not propose model selection method for R.

Second, our method is enforcing the different levels of smoothness for different components adaptively. Note that the smoothness constraint term in our objective function (12) is multiplied by g_r^2. Usually, the distribution of $|g_r|$ of the PARAFAC decomposition is similar to the distribution of the "singular values" of the singular value decomposition, which may decrease exponentially from largest to smallest especially for smooth image data. Thus, our model can be interpreted as constructing a smooth image by building up the various levels of smooth components, beginning with the smoothest one. It should be emphasized that the adaptive smoothness in our model is a key-technique to perform large improvements from state-of-the-art methods for image completion.

B. TV Smoothing and QV Smoothing

In this paper, we consider two types of smoothness constraints: TV and QV smoothing. In image recovery problems, such as denoising and restoration, it is generally considered that the TV constraint is better than QV. On the other hand, our result implies that the QV constraint performs better than TV for the image completion problem. This is a new and important discovery for the image completion research community, because many studies still consider the TV constraint for completion problems [17], [27].

The advantages of QV for the completion problem can be summarized as follows. If we consider the minimization of the

TV/QV term of a signal $\boldsymbol{z} = [0, \text{NaN}, 2, \text{NaN}, 0]^T$, the respective solutions are $\boldsymbol{z}_{TV} = [0, a, 2, b, 0]^T$, where $a, b \in [0, 2]$ and $\boldsymbol{z}_{QV} = [0, 1, 2, 1, 0]^T$. Since the QV term has a strong convexity compared with the TV term, the uniqueness of $\boldsymbol{z}_{QV}$ can be enforced. Furthermore, z_{TV} has strong possibility of performing bumps in a component such as $[0, 0, 2, 0, 0]^T$, and including $[0, 2, 2, 2, 0]^T$ in the solution set. In cases with high missing ratio, such as 90% or 95%, such bumps will be performed very often by TV minimization.

C. Rank Increasing Approach

In this section, we discuss the some advantages of rank increasing model selection which is employed in our method.

First, rank-increasing approach is more suitable than rank-decreasing approach for our smooth PD model. If there is no smoothness constraint, then the weak upper-bound of tensor rank is $\min_k(\prod_{n \neq k} I_n)$ for a general N-th order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, because any $\boldsymbol{\mathcal{X}}$ can be factorized by only one dense matrix $\boldsymbol{U}^{(k)} = \boldsymbol{X}_{(k)}$ and a unit matrix constructed as $\boldsymbol{I} = \boldsymbol{U}^{(1)} \odot \boldsymbol{U}^{(2)} \odot \ldots \odot \boldsymbol{U}^{(k-1)} \odot \boldsymbol{U}^{k+1} \odot \ldots \odot \boldsymbol{U}^{(N)} \in \mathbb{R}^{\prod_{n \neq k} I_n \times \prod_{n \neq k} I_n}$. However, such a unit matrix is not suitable for a smooth constraint, and it is difficult to estimate the upper-bound of the tensor rank, because it depends on the imposed level of smoothness. In contrast, the rank-increasing approach can easily be applied to various problems in practice since it does not require prior information for the tensor rank.

Second, the rank-increasing scheme is useful for the initialization. In general, even if we know the exact minimum (canonical) tensor rank of the smooth PARAFAC/Tucker decomposition for a completion problem, many local minima may exist for the optimization problem. Considering several local optimal solutions for the problem, then the results of optimization methods will depend on the initialization. Fig. 1 shows the concept of the optimization process for the SPC algorithm. The algorithm starts from a rank-one tensor ($R = 1$), the initialization of which has no critical meaning at this moment. The initialization of the rank-R tensor factorization is given by the rank-$(R-1)$ tensor factorization, which gives a better initialization for the completion problem, because this initialization arises from the lower-rank tensor space. When the algorithm is finally stopped, the SPC algorithm is able to find a good solution that is close to the lower-rank tensor space.

In recent years, rank-increasing approach is a relatively hot topic in matrix and tensor factorization models [45], [19], [63], [67], [74]. The main statement in these papers is that the rank-increasing (greedy) approach is more efficient for large scale and ill-posed setting than nuclear-norm based scheme. Our method and results are consistent with these statements.

D. Convergence Property

Unfortunately, unconstrained low-rank tensor approximation problem for higher ranks can be ill-posed or ill-conditioned in many cases [18]. However, some promissing approaches have been also established in recent years: Considering the unconstrained problem for low-rank approximation without missing values, a rank-one approximation by the ALS algorithm achieves a local convergence [68], [81]. PARAFAC decomposition with smooth spline components preserves the local convergence of the ALS algorithm [53]. The local convergence of the rank-one tensor approximation with penalized smoothing is proven in [3]. Moreover, very recent analysis indicates that rank-one ALS algorithm with regularization provides global convergence via Lojasiewicz inequality [66]. Please note that ALS and HALS are equivalent for rank-one tensor approximation $R = 1$.

For the case of $R \geq 2$, some results of block coordinate descent (BCD) scheme can be applied for HALS. In [72], local convergence of BCD scheme consisting of multiple convex sub-optimizations is obtained for non-negative tensor factorization model. In [31], a regularized PD based tensor completion algorithm is proposed and its local convergence is obtained. These results can be applied for our optimization problem when $p = 2$ except unit-norm constraints.

Unfortunately, theoretical result regarding convergence of our algorithm is not obtained. The difficulty of analysis for our model may be caused by "unit-norm constraint." This problem will be addressed in our future works.

VI. Conclusion

In this paper, we proposed a new low-rank smooth PARAFAC decomposition method for tensor completion problems. Our approach and algorithms are quite different from existing methods. Instead of setting the upper bound of the expected rank of the tensor, our algorithm increases the number of components gradually until the optimal rank is achieved. We considered two types of smoothness constraint in our SPC method: total variation (TV) and quadratic variation (QV). Our method was shown to outperform the state-of-the-art algorithms, in particular HaLRTC, STDC, and FBCP-MP. The method does not guarantee global convergence since tensor decomposition is in general non-convex optimization problem. Theoretical interpretations of local solution will be studied in our future works.

Acknowledgment

The authors would like to thank Chao Li for useful discussions.

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[2] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations for incomplete data," *Chemometr. Intell. Lab. Syst.*, vol. 106, no. 1, pp. 41–56, 2011.

[3] G. Allen, "Multi-way functional principal components analysis," in *Proc. IEEE 5th Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, pp. 220–223, 2013.

[4] Y. Amit, M. Fink, N. Srebro, and S. Ullman, "Uncovering shared structures in multiclass classification," in *Proc. 24th ACM Int. Conf. Mach. Learn.*, pp. 17–24, 2007.

[5] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," in *Advances in Neural Information Processing Systems 19*. Cambridge, MA, USA: MIT Press, 2007, pp. 41–48.

[6] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2419–2434, 2009.

[7] D. P. Bertsekas and A. Scientific, *Convex Optimization Algorithms*. Belmont, MA, USA: Athena Scientific, 2015.

[8] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Upper Saddle River, NJ, USA: Prentice-Hall, 1989.

[9] J. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of 'Eckart-Young' decomposition," *Psychometrika*, vol. 35, pp. 283–319, 1970.

[10] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented Lagrangian method for total variation video restoration," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3097–3111, 2011.

[11] C. Chen, B. He, and X. Yuan, "Matrix completion via an alternating direction method," *IMA J. Numer. Anal.*, vol. 32, no. 1, pp. 227–245, 2012.

[12] Y.-L. Chen, C.-T. Hsu, and H.-Y. Liao, "Simultaneous tensor decomposition and completion using factor priors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 577–591, 2014.

[13] A. Cichocki, R. Zdunek, and S. Amari, "Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization," in *Independent Component Analysis and Signal Separation*, ser. Lecture Notes in Computer Science, vol. 4666. New York NY, USA: Springer, 2007, pp. 169–176.

[14] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. New York, NY, USA: Wiley, 2009.

[15] R. Courant and D. Hilbert, *Methods of Mathematical Physics*, vol. 1, CUP Archive. New York, NY, USA: Wiley1966.

[16] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, 2004.

[17] Q. Dai and W. Sha, "The physics of compressive sensing and the gradient-based recovery algorithms," *arXiv preprint arXiv:0906.1487*, 2009.

[18] V. De Silva and L.-H. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1084–1127, 2008.

[19] S. V. Dolgov and D. V. Savostyanov, "Alternating minimal energy methods for linear systems in higher dimensions. Part I: SPD systems," *arXiv preprint arXiv:1301.6068*, 2013.

[20] S. C. Douglas, S. Amari, and S.-Y. Kung, "On gradient adaptation with unit-norm constraints," *IEEE Trans. Signal Process.*, vol. 48, no. 6, pp. 1843–1847, 2000.

[21] S. Essid and C. Fevotte, "Smooth nonnegative matrix factorization for unsupervised audiovisual document structuring," *IEEE Trans. Multimedia*, vol. 15, no. 2, pp. 415–425, 2013.

[22] S. Friedland and L.-H. Lim, "Nuclear norm of higher-order tensors," *arXiv preprint arXiv: 1410.6072 v3*, 2016.

[23] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, 2011.

[24] X. Geng, K. Smith-Miles, Z.-H. Zhou, and L. Wang, "Face image modeling by multilinear subspace analysis with missing values," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 3, pp. 881–892, 2011.

[25] N. Gillis and F. Glineur, "Low-rank matrix approximation with weights or missing data is NP-hard," *SIAM J. Matrix Anal. Appl.*, vol. 32, no. 4, pp. 1149–1165, 2011.

[26] D. Goldfarb and W. Yin, "Second-order cone programming methods for total variation-based image restoration," *SIAM J. Scientif. Comput.*, vol. 27, no. 2, pp. 622–645, 2005.

[27] X. Han, J. Wu, L. Wang, Y. Chen, L. Senhadji, and H. Shu, "Linear total variation approximate regularized nuclear norm optimization for matrix completion," *Abstract Appl. Anal.*, ID 765782, 2014.

[28] R. Harshman, "Foundations of the PARAFAC procedure: Model and conditions for an 'explanatory' multi-mode factor analysis," *UCLA Working Papers Phonetics*, vol. 16, no. 1–84, 1970.

[29] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *J. Math. Phys.*, vol. 6, no. 1, pp. 164–189, 1927.

[30] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn. (CVPR)*, pp. 1791–1798, 2010.

[31] L. Karlsson and A. Uschmajew, "Parallel algorithms for tensor completion in the CP format," *Parallel Comput.*, 2015, to be published.

[32] H. A. Kiers, "Towards a standardized notation and terminology in multiway analysis," *J. Chemometr.*, vol. 14, no. 3, pp. 105–122, 2000.

[33] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.

[34] T. Krasulina, "Method of stochastic approximation in the determination of the largest eigenvalue of the mathematical expectation of random matrices," *Autom. Remote Control*, pp. 50–56, 1970.

[35] D. Kressner, M. Steinlechner, and B. Vandereycken, "Low-rank tensor completion by Riemannian optimization," *BIT Numer. Math.*, vol. 54, no. 2, pp. 447–468, 2014.

[36] N. Li and B. Li, "Tensor completion for on-board compression of hyperspectral images," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, pp. 517–520, 2010.

[37] W. Li, L. Zhao, Z. Lin, D. Xu, and D. Lu, "Non-local image inpainting using low-rank matrix completion," *Comput. Graph. Forum*, vol. 34, no. 6, pp. 111–122, 2015.

[38] X. Li, "Compressed sensing and matrix completion with constant proportion of corruptions," *Construct. Approx.*, vol. 37, no. 1, pp. 73–99, 2013.

[39] Y. Li, J. Yan, Y. Zhou, and J. Yang, "Optimum subspace learning and error correction for tensors," in *Computer Vision–ECCV 2010*, Lecture Notes in Computer Science, vol. 6313. New York, NY, USA: Springer, 2010, pp. 790–803.

[40] Y. Li, Y. Zhou, J. Yan, J. Yang, and X. He, "Tensor error correction for corrupted values in visual data," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, pp. 2321–2324, 2010.

[41] Z. Lin, M. Chen, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," arXiv preprint arXiv:1009.5055, 2010.

[42] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pp. 2114–2121, 2009.

[43] 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, 2013.

[44] G. Lv, L. Yin, Y. Jie, and L. Li, "Hole-filling by rank sparsity tensor decomposition for medical imaging," *IEICE Trans. Inf. Syst.*, vol. 94, no. 2, pp. 396–399, 2011.

[45] B. Mishra, G. Meyer, F. Bach, and R. Sepulchre, "Low-rank optimization with trace norm penalty," *SIAM J. Optim.*, vol. 23, no. 4, pp. 2124–2149, 2013.

[46] M. K. Ng, P. Weiss, and X. Yuan, "Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods," *SIAM J. Scientif. Comput.*, vol. 32, no. 5, pp. 2710–2736, 2010.

[47] J. P. Oliveira, J. M. Bioucas-Dias, and M. A. Figueiredo, "Adaptive total variation image deblurring: a majorization–minimization approach," *Signal Process.*, vol. 89, no. 9, pp. 1683–1693, 2009.

[48] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Model. Simulat.*, vol. 4, no. 2, pp. 460–489, 2005.

[49] N. L. Owsley, "Adaptive data orthogonalization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 3, pp. 109–112, 1978.

[50] P. Rai, Y. Wang, S. Guo, G. Chen, D. Dunson, and L. Carin, "Scalable Bayesian low-rank decomposition of incomplete multiway tensors," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, pp. 1800–1808, 2014.

[51] I. Ram, M. Elad, and I. Cohen, "Image processing using smooth ordering of its patches," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2764–2774, 2013.

[52] 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.

[53] M. M. Reis and M. Ferreira, "PARAFAC with splines: A case study," *J. Chemometr.*, vol. 16, no. 8–10, pp. 444–450, 2002.

[54] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenom.*, vol. 60, no. 1, pp. 259–268, 1992.

[55] I. Selesnick, "Total variation denoising (an mm algorithm)," *NYU Polytechnic School of Eng. Lecture Notes*. New York, NY, USA, 2012.

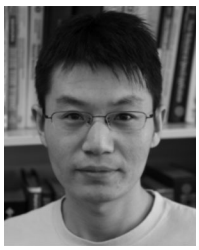
[56] P. Shah, N. Rao, and G. Tang, "Optimal low-rank tensor recovery from separable measurements: Four contractions suffice," arXiv preprint, arXiv:1505.04085, 2015.

[57] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, 2003.

[58] L. Sorber, M. Van Barel, and L. De Lathauwer, "Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-$(L_r, L_r, 1)$ terms, and a new generalization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 695–720, 2013.

[59] L. Sorber, M. Van Barel, and L. De Lathauwer, Tensorlab, v2. 0, Jan. 2014.
[60] L. Sorber, M. Van Barel, and L. De Lathauwer, "Structured data fusion," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 586–600, 2015.
[61] D. Sorensen, "Minimization of a large-scale quadratic function subject to a spherical constraint," *SIAM J. Opt.*, vol. 7, no. 1, pp. 141–161, 1997.
[62] H. Tan, B. Cheng, W. Wang, Y.-J. Zhang, and B. Ran, "Tensor completion via a multi-linear low-n-rank factorization model," *Neurocomput.*, vol. 133, 161–169, 2014.
[63] M. Tan, I. W. Tsang, L. Wang, B. Vandereycken, and S. J. Pan, "Riemannian pursuit for big matrix recovery," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, pp. 1539–1547, 2014.
[64] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani *et al.*, "Missing value estimation methods for DNA microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.
[65] L. R. Tucker, "Implications of factor analysis of three-way matrices for measurement of change," in *Problems in Measuring Change*, pp. 122–137. Madison, WI, USA: Univ. of Wisconsin Press, 1963.
[66] A. Uschmajew, "A new convergence proof for the high-order power method and generalizations," *Pac. J. Optim.*, vol. 11, no. 2, pp. 309–321, 2015.
[67] A. Uschmajew and B. Vandereycken, "Greedy rank updates combined with Riemannian descent methods for low-rank optimization," in *Proc. Int. Conf. Sampl. Theory Appl. (SampTA)*, pp. 420–424, 2015.
[68] L. Wang and M. T. Chu, "On the global convergence of the alternating least squares method for rank-one approximation to generic tensors," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 3, pp. 1058–1072, 2014.
[69] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, 2008.
[70] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.
[71] D. M. Witten, R. Tibshirani, and T. Hastie, "A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis," *Biostatistics*, 2009.
[72] 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, 2013.
[73] J. Yan, J. Liu, Y. Li, Z. Niu, and Y. Liu, "Visual saliency detection via rank-sparsity decomposition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, pp. 1089–1092, 2010.
[74] T. Yokota and A. Cichocki, "A fast automatic rank determination algorithm for noisy low-rank matrix completion," in *Proc. Asia-Pac. Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, pp. 43–46, 2015.
[75] T. Yokota, R. Zdunek, A. Cichocki, and Y. Yamashita, "Smooth nonnegative matrix and tensor factorizations for robust multi-way data analysis," *Signal Process.*, vol. 113: 234–249, 2015.
[76] H.-F. Yu, C.-J. Hsieh, S. Si, and I. S. Dhillon, "Parallel matrix factorization for recommender systems," *Knowl. Inf. Syst.*, vol. 41, no. 3, pp. 793–819, 2014.
[77] M. Yuan and C.-H. Zhang, "On tensor completion via nuclear norm minimization," *Found. Comput. Math.*, pp. 1–38, 2014.
[78] R. Zdunek, "Approximation of feature vectors in nonnegative matrix factorization with Gaussian radial basis functions," in *Neural Information Processing*, vol. 7663, LNCS, pp. 616–623. New York, NY, USA: Springer, 2012.
[79] R. Zdunek and A. Cichocki, "Blind image separation using nonnegative matrix factorization with Gibbs smoothing," in *Neural Information Processing*, vol. 4985, LNCS, pp. 519–528. New York, NY, USA: Springer, 2008.
[80] R. Zdunek, A. Cichocki, and T. Yokota, "B-spline smoothing of feature vectors in nonnegative matrix factorization," in *Artificial Intelligence and Soft Computing*, vol. 8468, LNCS, pp. 72–81. New York, NY, USA: Springer, 2014.
[81] T. Zhang and G. H. Golub, "Rank-one approximation to high order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 23, no. 2, pp. 534–550, 2001.
[82] 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, 2015.

Tatsuya Yokota (M'14) received the Ph.D. degree in engineering from Tokyo Institute of Technology, Tokyo, Japan, in 2013. From 2011 to 2013, he worked as a Junior Research Associate at the Laboratory for Advanced Brain Signal Processing (ABSP) in RIKEN Brain Science Institute (BSI), Japan. From 2013 to 2015, he worked as a Research Scientist at Laboratory for ABSP and a Visiting Research Scientist at TOYOTA Collaboration Center in RIKEN BSI, Japan. He is currently an Assistant Professor at Department of Computer Science and Engineering in Nagoya Institute of Technology, Japan. His research interests include matrix/tensor factorizations and signal/image processing.



Qibin Zhao (M'12) received the Ph.D. degree from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2009. He is currently a Research Scientist with the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Japan, and also a Visiting Professor with the Saitama Institute of Technology, Fukaya, Japan. He has authored over 60 papers in international journals and conferences. His current research interests include machine learning, tensor factorization, computer vision, and brain–computer interface.

Andrzej Cichocki (F'13) received the M.Sc. (with honors), Ph.D. and Dr.Sc. (Habilitation) degrees, all in electrical engineering from Warsaw University of Technology (Poland). He spent several years at University Erlangen (Germany) as an Alexander-von-Humboldt Research Fellow and Guest Professor. He is currently a Senior Team Leader and Head of the laboratory for Advanced Brain Signal Processing, at RIKEN Brain Science Institute (Japan). He is author of more than 400 technical journal papers and 4 monographs in English (two of them translated to Chinese). He served as Associated Editor of IEEE TRANSACTIONS ON SIGNALS PROCESSING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON CYBERNETICS, IEEE TRANSACTIONS ON SIGNAL PROCESSING, *Journal of Neurosciemce Methods* and he was as founding Editor-in-Chief for *Journal Computational Intelligence and Neuroscience*. Currently, his research focus on multiway blind source separation, ICA, NMF, tensor factorizations, tensor networks for big data analytics, and brain–computer interface. His publications currently report over 27,600 citations according to Google Scholar, with an h-index of 70.