

Robust Multilinear Tensor Rank Estimation Using Higher Order Singular Value Decomposition and Information Criteria

Tatsuya Yokota, *Member, IEEE*, Namgil Lee, and Andrzej Cichocki, *Fellow, IEEE*

Abstract

Model selection in tensor decomposition is important for real applications if the rank of the original data tensor is unknown and the observed tensor is noisy. In the Tucker model, the minimum description length (MDL) or Bayesian information criteria have been applied to tensors via matrix unfolding, but these methods are sensitive to noise when the tensors have a multilinear low rank structure given by the Tucker model. In this study, we propose new methods for improving the MDL so it is more robust to noise. The proposed methods are justified theoretically by analyzing the “multilinear low-rank structure” of tensors. Extensive experiments including numerical simulations and a real application to image denoising are provided to illustrate the advantages of the proposed methods.

Index Terms

Bayesian information criterion (BIC), higher order singular value decomposition (HOSVD), minimum description length (MDL), model selection, multilinear tensor rank, Tucker decomposition

I. INTRODUCTION

Rank estimation is an important problem when determining the appropriate model order from a given data matrix/tensor. We consider the following general model for matrix rank estimation:

$$\mathbf{Y} = \mathbf{Y}_0 + \mathbf{E}, \quad (1)$$

Copyright (c) 2016 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

This work was supported by Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Number 15K16067.

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

N. Lee is with RIKEN Brain Science Institute, 351-0198 Saitama, Wako-shi, Japan.

A. Cichocki is with RIKEN Brain Science Institute, 351-0198 Saitama, Wako-shi, Japan and Systems Research Institute, Polish Academy of Science, 01-447 Warsaw, Poland.

This paper has supplementary downloadable multimedia material available at <http://ieeexplore.ieee.org> provided by the authors. This includes MATLAB codes for demonstrating the proposed tensor rank estimation algorithm and its application in HOSVD denoising of gray-scale images. This material is 0.3 MB in size.

where $\mathbf{Y} \in \mathbb{R}^{I \times J}$ is a noisy data matrix, $\mathbf{Y}_0 \in \mathbb{R}^{I \times J}$ is a rank- R latent matrix, and $\mathbf{E} \in \mathbb{R}^{I \times J}$ is an additive noise matrix, the individual entries of which e_{ij} are i.i.d. based on a zero-mean normal distribution with variance σ^2 . The objective of rank estimation is to find the appropriate R from a noisy full rank matrix \mathbf{Y} . Rank estimation methods have several important applications depending on the objectives of the analysis. For example, the estimated rank may be the number of clusters in clustering tasks, the number of latent signals in blind source separations, and the appropriate dimension in dimensionality reduction (typically, principal component analysis) [36], [38], [37], [25], [22], [5], [12]. In order to solve this problem, many methods have been studied since the 1970s, such as Akaike's information criterion [1], [35], Bayesian information criterion (BIC) [30], minimum description length (MDL) [29], [35], generalized information criterion [20], the quotient of differences in additional values (QDA) measure [25], exponential fitting test [10], [27], Laplace's method (LAP) [24], cross-validation based method [4], and a method for simultaneously estimating the rank and noise level [21].

In this study, we consider how to extend the problem of matrix rank estimation to a "tensor" (multi-way array). In tensors, a straightforward and natural extension of the matrix rank is the *CANDECOMP/PARAFAC (CP) rank*, where the CP rank of a tensor is defined as the minimum number of rank-one tensors required to yield its exact CP decomposition [14], [11]. However, computing the CP rank of a tensor is known to be NP-hard [13]. The problem of estimating the CP rank of a latent low-rank tensor from an observed noisy tensor may be very difficult, but there are several methods for estimating the CP rank from noisy observations [3], [12], [8], [26], [23].

In the present study, we do not focus on the CP rank, but instead we consider another type of tensor rank: the *multilinear tensor rank*. In contrast to the CP rank, which is based on CP decomposition, the multilinear tensor rank is based on Tucker decomposition [34]. Tucker decomposition plays important roles in tensor data analysis [16], [17], [33], [39], [2], [15], [18] and it is described as

$$\mathcal{X}_0 = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \cdots \times_N \mathbf{U}^{(N)} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}, \quad (2)$$

where $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_N}$ is a core tensor, $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n}$ for $n \in \{1, 2, \dots, N\}$ are factor matrices with $R_n \leq I_n$ for all $n \in \{1, 2, \dots, N\}$, and \times_n denotes the n -th mode tensor-matrix product: $[\mathcal{G} \times_n \mathbf{U}^{(n)}]_{r_1 \cdots r_{n-1} i_n r_{n+1} \cdots r_N} = \sum_{r_n=1}^{R_n} g_{r_1 \cdots r_{n-1} r_n r_{n+1} \cdots r_N} u_{i_n r_n}^{(n)}$. A multilinear tensor rank of \mathcal{X}_0 , which is also referred to as the Tucker rank, can be defined as

$$\text{rank}_n(\mathcal{X}_0) := \min(R_n) = \text{rank}([\mathcal{X}_0]_{(n)}), \quad (3)$$

where $[\mathcal{X}_0]_{(n)} \in \mathbb{R}^{I_n \times \prod_{k \neq n} I_k}$ is a matrix that is unfolded from \mathcal{X}_0 with respect to n -th mode and it is referred to as " n -th mode unfolding matrix" [19]. If the observed tensor is contaminated by additive noise as: $\mathcal{X} = \mathcal{X}_0 + \mathcal{E}$, the multilinear tensor ranks of \mathcal{X} and \mathcal{X}_0 can be different. Estimating the multilinear tensor rank of the original tensor \mathcal{X}_0 from a noisy tensor \mathcal{X} is considered to be an important and challenging problem for model selection in tensor data analysis research.

Considering the definition of a multilinear tensor rank in (3), any matrix rank estimation method can be employed for multilinear tensor rank estimation via matrix rank estimation for the n -th mode unfolding matrix $\mathbf{X}_{(n)}$. For

example, MDL [29], [35] can be applied simply as

$$\widehat{R}_n = \text{MDL}(\mathbf{X}_{(n)}), \quad (4)$$

for each $n \in \{1, 2, \dots, N\}$.

Moreover, several methods specifically for multilinear tensor rank estimation have been proposed such as DIFFIT [32], TDA-SORTE [40], and MLREST [31]. In [32], all of the N -tuple (R_1, R_2, \dots, R_N) candidates were considered and evaluated based on the fitting rate, and an appropriate rank was selected from these candidates. In the tensor decomposition toolbox ‘‘TDALAB’’ [40], an estimation strategy is implemented using the SORT algorithm [12], which was originally proposed for CP rank estimation. In TDALAB, the SORT algorithm is applied to the n -th mode unfolding matrix $\mathbf{X}_{(n)}$. In another tensor decomposition toolbox called ‘‘Tensorlab’’ [31], an estimation function is available as ‘‘mlrankst,’’ which uses an L-curve to evaluate the multilinear rank candidates. According to [31], an L-curve is useful for finding a good point with the optimal trade-off between accuracy and compression.

Roughly speaking, the number of multilinear tensor rank combinations is $\prod_{n=1}^N I_n$ for N -th order tensors. It is computationally demanding to select one optimal value from all of these combinations, so DIFFIT and MLREST require a relatively long time to obtain estimates. Cross-validation-based methods also require a long computational time for their training and testing procedures. By contrast, matrix-based ‘‘mode-wise’’ rank estimation methods that use an information criterion (such as MDL, QDA, and SORT) are quite convenient and practical from a computational perspective. However, when we consider how to apply matrix-based rank estimation methods to low multilinear rank tensors, the gap between low-rank matrix and low-rank tensor models causes a weakness in the presence of noise. The objective of this study is to investigate the difference between low-rank matrix and low-rank tensor models by theoretically analyzing the covariance matrices and eigenvalues for both models. Based on this analysis, we propose two new multilinear tensor (Tucker) rank estimation methods, which use the informative part of the core tensor selectively to modify eigenvalues. The proposed algorithms are robust to noise, but they are also relatively simple and computationally fast. The proposed algorithms are justified and compared competitively in numerical simulations with state-of-the-art methods.

The remainder of this paper is organized as follows. In Section II, we review a matrix-based rank estimation method and its direct application to tensors. Section III considers the difference in the covariance matrices and eigenvalues between matrix and tensor models. In Sections IV and V, we propose two methods for robust multilinear tensor rank estimation. In Section VI, we investigate the performance and applications of our algorithms, as well as comparing them with some state-of-the-art methods. Finally, we present our conclusions in Section VII.

II. PRELIMINARIES: APPLYING A MATRIX MODEL TO TENSORS

A. Review of the MDL method

In this section, we first review a (basic) matrix rank estimation method called MDL. We assume that: (a) a noisy observed matrix is given by $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_J] = \mathbf{Y}_0 + \mathbf{E} \in \mathbb{R}^{I \times J}$, where the matrix \mathbf{Y}_0 is rank R and $I \leq J$;

and (b) each element in a noise matrix \mathbf{E} follows an i.i.d. Gaussian distribution $N(0, \sigma^2)$ and it is independent of \mathbf{Y}_0 .

The eigenvalues of its covariance matrix $\Sigma_{\mathbf{Y}}$ provide useful information about the original rank estimation because it can be decomposed as

$$\begin{aligned}\Sigma_{\mathbf{Y}} &= \Sigma_{\mathbf{Y}_0} + \sigma^2 \mathbf{I}_I, \\ &= \left(\sum_{r=1}^R \lambda_r^{(0)} \mathbf{u}_r \mathbf{u}_r^T \right) + \sigma^2 \mathbf{I}_I,\end{aligned}\quad (5)$$

where $\lambda_1^{(0)}, \lambda_2^{(0)}, \dots, \lambda_R^{(0)}$ are the eigenvalues of $\Sigma_{\mathbf{Y}_0}$, and \mathbf{u}_r are the corresponding eigenvectors. Thus, the eigenvalues of $\Sigma_{\mathbf{Y}}$ can be expressed as

$$\lambda_i = \begin{cases} \lambda_i^{(0)} + \sigma^2 & (1 \leq i \leq R) \\ \sigma^2 & (R+1 \leq i \leq I) \end{cases} . \quad (6)$$

Plotting the eigenvalues with an ordering of $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_I$ obtains an ‘‘L’’-type curve, which allows us to recognize some threshold between the signal and noise spaces. In practice, when J is a finite number, the covariance matrix $\Sigma_{\mathbf{Y}}$ can be estimated by the sample covariance matrix $\mathbf{S}_{\mathbf{Y}} := \frac{1}{J} \mathbf{Y} \mathbf{Y}^T \rightarrow \Sigma_{\mathbf{Y}}$, and its eigenvalue decomposition gives the estimators for the eigenvalues λ_i . Finally, the matrix rank of \mathbf{Y}_0 can be estimated by the MDL (which is equivalent to the BIC) criterion [29], [35]:

$$\begin{aligned}\hat{R} = \operatorname{argmin}_r & -2 \log \left\{ \frac{\prod_{i=r+1}^I \lambda_i^{1/(I-r)}}{\frac{1}{I-r} \sum_{i=r+1}^I \lambda_i} \right\}^{J(I-r)} \\ & + r(2I - r) \log(J).\end{aligned}\quad (7)$$

The MDL criterion usually works well [29], [35], [36], [22] when the number of samples J is sufficiently large, even if the variance σ^2 is unknown and the noise level is relatively high.

B. Application of MDL to tensors

Now, we discuss how to apply the MDL to tensors. In the form of the n -th mode unfolding, we assume that: (a) the observed tensor can be decomposed by $\mathbf{X}_{(n)} = [\mathbf{X}_0]_{(n)} + \mathbf{E}_{(n)}$ ($n = 1, 2, \dots, N$); and (b) each element of a noise matrix $\mathbf{E}_{(n)}$ follows an i.i.d. Gaussian distribution $N(0, \sigma^2)$ and it is independent of $[\mathbf{X}_0]_{(n)}$. Based only on the assumptions given above, the MDL can be applied to tensors, almost directly.

The n -th mode empirical covariance matrix and its eigenvalues can be derived by

$$\mathbf{S}_{\mathbf{X}_{(n)}} = \frac{1}{\prod_{n \neq k} I_k} \mathbf{X}_{(n)} \mathbf{X}_{(n)}^T, \quad (8)$$

$$\mathbf{\Lambda}^{(n)} = \mathbf{V}^{(n)T} \mathbf{S}_{\mathbf{X}_{(n)}} \mathbf{V}^{(n)}, \quad (9)$$

where $\mathbf{\Lambda}^{(n)} = \operatorname{diag}(\lambda_1^{(n)}, \lambda_2^{(n)}, \dots, \lambda_{I_n}^{(n)}) \in \mathbb{R}^{I_n \times I_n}$ is a diagonal matrix of eigenvalues in descending ordering and $\mathbf{V}^{(n)} \in \mathbb{R}^{I_n \times I_n}$ is a factor matrix comprising the orthonormal column eigenvectors of $\mathbf{S}_{\mathbf{X}_{(n)}}$. Using all of the

mode factor matrices, a data tensor \mathcal{X} can be decomposed exactly by

$$\mathcal{X} = \mathcal{H} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \cdots \times_N \mathbf{V}^{(N)}, \quad (10)$$

where a core tensor is given by $\mathcal{H} = \mathcal{X} \times_1 \mathbf{V}^{(1)T} \times_2 \mathbf{V}^{(2)T} \times_3 \cdots \times_N \mathbf{V}^{(N)T}$. This decomposition (10) is referred to as higher order singular value decomposition (HOSVD) [9]. The diagonal matrix $\mathbf{\Lambda}^{(n)}$ of eigenvalues can be expressed as follows:

$$\mathbf{\Lambda}^{(n)} = \frac{I_n}{\prod_{n=1}^N I_n} \mathbf{H}_{(n)} \mathbf{H}_{(n)}^T. \quad (11)$$

The n -th mode multilinear tensor rank R_n can be estimated by using the MDL criterion (7) and the distribution of the n -th mode eigenvalues $\mathbf{\Lambda}^{(n)}$.

In a similar manner, most of the matrix-based rank estimation methods can be applied to tensors via n -th mode unfolding. MDL and other matrix-based methods often work well for tensors, and they are competitive with some methods designed specifically for tensors when the noise level is sufficiently low. However, these methods are quite sensitive to strong noise, especially because this direct approach completely ignores the multilinear low-rank structure of tensors. In this study, we derive some modified estimators of eigenvalues by considering the multilinear low-rank structure of tensors and we propose new robust methods for multilinear tensor rank estimation.

III. MODIFIED EIGENVALUES FOR THE TUCKER MODEL

In this section, we consider a third order tensor to simplify the notation, but our method can be applied to any order of tensor.

A. Modified form of tensor decomposition

The Tucker decomposition model of a noisy tensor is given by

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} + \mathcal{E}, \quad (12)$$

where $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is a noisy observed data tensor, $\mathcal{E} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is an additive Gaussian noise tensor, $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ is a core tensor, and $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n}$ is an n -th mode factor matrix of orthonormal columns. It should be noted that the Tucker decomposition is not unique because we have $\mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \mathbf{Q}^{(1)T} \mathbf{Q}^{(1)} \times_2 \mathbf{U}^{(2)} \mathbf{Q}^{(2)T} \mathbf{Q}^{(2)} \times_3 \mathbf{U}^{(3)} \mathbf{Q}^{(3)T} \mathbf{Q}^{(3)} = \mathcal{G}_Q \times_1 \mathbf{U}_Q^{(1)} \times_2 \mathbf{U}_Q^{(2)} \times_3 \mathbf{U}_Q^{(3)}$ for any orthogonal matrices $\mathbf{Q}^{(n)} \in \mathbb{R}^{R_n \times R_n}$, where $\mathbf{U}_Q^{(n)} = \mathbf{U}^{(n)} \mathbf{Q}^{(n)T}$ is a new orthogonal factor matrix and $\mathcal{G}_Q = \mathcal{G} \times_1 \mathbf{Q}^{(1)} \times_2 \mathbf{Q}^{(2)} \times_3 \mathbf{Q}^{(3)}$ is a new core tensor. However, the minimum size of the core tensor (R_1, R_2, R_3) is unique. The minimum size of the individual modes of the core tensor (R_1, R_2, R_3) is equivalent to the multilinear tensor rank. A matrix representation of the Tucker model obtained via the n -th mode unfolding of (12) is given by

$$\mathbf{X}_{(1)} = \mathbf{U}^{(1)} \mathbf{G}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T + \mathbf{E}_{(1)} \in \mathbb{R}^{I_1 \times I_2 I_3}, \quad (13)$$

where \otimes denotes the Kronecker product. Now, we assume that: (a) the observed tensor is generated by $\mathcal{X} = \mathcal{X}_0 + \mathcal{E}$; (b) each element in \mathcal{E} follows an i.i.d. Gaussian distribution $N(0, \sigma^2)$ and it is independent of $\mathcal{X}_0 = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times_2$

$\mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$; and (c) the rows of $\mathbf{G}_{(1)}$ are orthogonal, i.e., $\mathbf{\Lambda}_0^{(1)} = \frac{1}{R_2 R_3} \mathbf{G}_{(1)} \mathbf{G}_{(1)}^T$ is a diagonal matrix. It should be noted that any core tensor \mathcal{G} can be transformed into a core tensor with the property in (c) via singular value decomposition, $(\mathbf{U}_G, \mathbf{D}_G, \mathbf{V}_G) = \text{svd}(\mathbf{G}_{(1)})$, and by modifying the factors as $\mathbf{U}^{(1)} \leftarrow \mathbf{U}^{(1)} \mathbf{U}_G$, and $\mathbf{G}_{(1)} \leftarrow \mathbf{D}_G \mathbf{V}_G^T$. We note that assumption (c) does not impose any special restriction on tensor decomposition and it is only used to simplify some of the formulations in this study.

In our approach, we consider a transformation of Eq. (13) by multiplying the orthogonal matrix $(\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})$, as follows:

$$\begin{aligned} \mathbf{Y}_{(1)} &:= \mathbf{X}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)}) \\ &= \mathbf{U}^{(1)} \mathbf{G}_{(1)} + \mathbf{F}_{(1)} \in \mathbb{R}^{I_1 \times R_2 R_3}, \end{aligned} \quad (14)$$

where $\mathbf{F}_{(1)} := \mathbf{E}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})$. Eq. (14) can be rewritten in equivalent vectorized form as

$$\mathbf{y}_i^{(1)} = \mathbf{U}^{(1)} \mathbf{g}_i^{(1)} + \mathbf{f}_i^{(1)} \text{ for all } i \in \{1, 2, \dots, R_2 R_3\}, \quad (15)$$

where $\mathbf{Y}_{(1)} = [\mathbf{y}_1^{(1)}, \mathbf{y}_2^{(1)}, \dots, \mathbf{y}_{R_2 R_3}^{(1)}]$, $\mathbf{G}_{(1)} = [\mathbf{g}_1^{(1)}, \mathbf{g}_2^{(1)}, \dots, \mathbf{g}_{R_2 R_3}^{(1)}] \in \mathbb{R}^{R_1 \times R_2 R_3}$, and $\mathbf{F}_{(1)} = [\mathbf{f}_1^{(1)}, \mathbf{f}_2^{(1)}, \dots, \mathbf{f}_{R_2 R_3}^{(1)}]$. It should be remarked that $(\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T$ is a mapping operator onto a signal subspace spanned by orthonormal bases with respect to the second and third modes of the tensor. Thus, we can see that Eq. (15) is equivalent to the matrix rank estimation problem (1), which is the problem of separating the whole space into a signal subspace (spanned by basis matrix $\mathbf{U}^{(1)}$) and a noise subspace (orthogonal complement of the signal space) with respect to only the first mode of the tensor. $\mathbf{E}_{(1)}$ follows an i.i.d. Gaussian distribution, so the matrix $\mathbf{F}_{(1)}$ obtained by mapping $\mathbf{E}_{(1)}$ onto its subspace spanned by orthonormal bases also follows an i.i.d. Gaussian distribution. Hence, $\mathbf{F}_{(1)}$ and $\mathbf{G}_{(1)}$ are assumed to be statistically independent.

B. Modified eigenvalues

In this section, we analyze the covariance matrices of the signals in two different forms: (13) and (14). First, the empirical covariance matrix for $\mathbf{X}_{(1)} = [\mathbf{x}_1^{(1)}, \mathbf{x}_2^{(1)}, \dots, \mathbf{x}_{I_2 I_3}^{(1)}]$ is given by

$$\begin{aligned} \mathbf{S}_{\mathbf{X}_{(1)}} &= \frac{1}{I_2 I_3} \sum_{i=1}^{I_2 I_3} \mathbf{x}_i^{(1)} \mathbf{x}_i^{(1)T} \\ &= \frac{1}{I_2 I_3} \left(\mathbf{U}^{(1)} \mathbf{G}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T + \mathbf{E}_{(1)} \right) \\ &\quad \cdot \left(\mathbf{U}^{(1)} \mathbf{G}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T + \mathbf{E}_{(1)} \right)^T \\ &= \frac{1}{I_2 I_3} \left(\mathbf{U}^{(1)} \mathbf{G}_{(1)} \mathbf{G}_{(1)}^T \mathbf{U}^{(1)T} \right. \\ &\quad + \mathbf{U}^{(1)} \mathbf{G}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T \mathbf{E}_{(1)}^T \\ &\quad + \mathbf{E}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)}) \mathbf{G}_{(1)}^T \mathbf{U}^{(1)T} \\ &\quad \left. + \mathbf{E}_{(1)} \mathbf{E}_{(1)}^T \right) \\ &\rightarrow \frac{R_2 R_3}{I_2 I_3} \mathbf{U}^{(1)} \mathbf{\Lambda}_0^{(1)} \mathbf{U}^{(1)T} + \sigma^2 \mathbf{I}_{I_1}, \end{aligned} \quad (16)$$

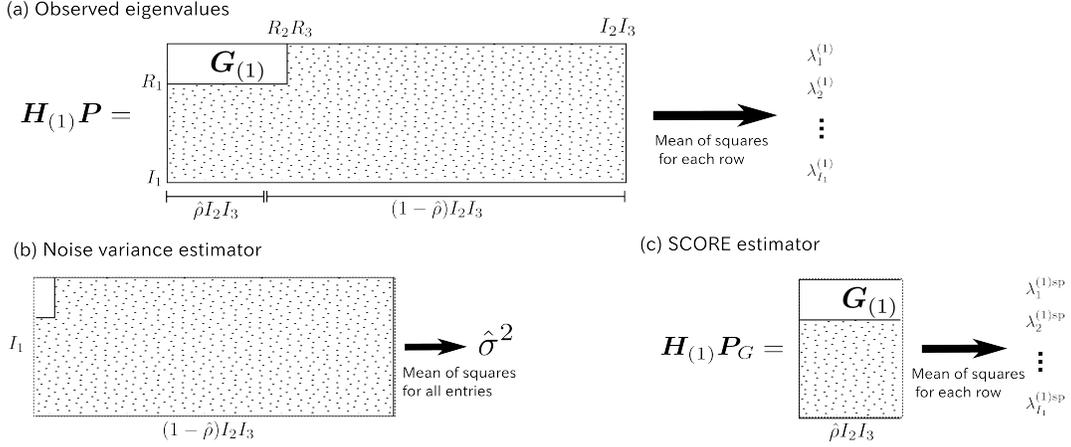


Fig. 1. Visual illustrations of the computation of $\lambda_i^{(1)}$, $\hat{\sigma}^2$, and $\lambda_i^{(1)sp}$. (a) The eigenvalues $\lambda_i^{(1)}$ for $i \in \{1, 2, \dots, I_1\}$ are computed by the MDL-based matrix rank estimation methods using the whole unfolding matrix $\mathbf{H}_{(1)}$ of the HOSVD core tensor, as described in (11). Using the permutation matrix $\mathbf{P} = [\mathbf{P}_G, \mathbf{P}_E]$, (b) the noise variance estimator $\hat{\sigma}^2$ is computed by the modified eigenvalues estimator for Tucker rank determination (MEET) method using the selected columns $\mathbf{H}_{(1)}\mathbf{P}_E$ of $\mathbf{H}_{(1)}$, and (c) the modified eigenvalues $\lambda_i^{(1)sp}$ for $i \in \{1, 2, \dots, I_1\}$ are computed by the SCORE method using the selected columns $\mathbf{H}_{(1)}\mathbf{P}_G$ of $\mathbf{H}_{(1)}$.

where \rightarrow denotes the convergence in probability based on the weak law of large numbers. In addition, the covariance matrix of $\mathbf{Y}_{(1)}$ can be expressed as

$$\begin{aligned}
 \mathbf{S}_{\mathbf{Y}_{(1)}} &= \frac{1}{R_2 R_3} \sum_{i=1}^{R_2 R_3} \mathbf{y}_i^{(1)} \mathbf{y}_i^{(1)T} \\
 &= \frac{1}{R_2 R_3} \left(\mathbf{U}^{(1)} \mathbf{G}_{(1)} + \mathbf{F}_{(1)} \right) \left(\mathbf{U}^{(1)} \mathbf{G}_{(1)} + \mathbf{F}_{(1)} \right)^T \\
 &= \frac{1}{R_2 R_3} \left(\mathbf{U}^{(1)} \mathbf{G}_{(1)} \mathbf{G}_{(1)}^T \mathbf{U}^{(1)T} + \mathbf{U}^{(1)} \mathbf{G}_{(1)} \mathbf{F}_{(1)}^T \right. \\
 &\quad \left. + \mathbf{F}_{(1)} \mathbf{G}_{(1)}^T \mathbf{U}^{(1)T} + \mathbf{F}_{(1)} \mathbf{F}_{(1)}^T \right) \\
 &\rightarrow \mathbf{U}^{(1)} \mathbf{\Lambda}_0^{(1)} \mathbf{U}^{(1)T} + \sigma^2 \mathbf{I}_{I_1}.
 \end{aligned} \tag{17}$$

A derivation of $\frac{1}{R_2 R_3} \mathbf{F}_{(1)} \mathbf{F}_{(1)}^T \rightarrow \sigma^2 \mathbf{I}_{I_1}$ was partly provided by [7], [6]. Let us put $\rho := \frac{R_2 R_3}{I_2 I_3}$, and considering that $I_2 I_3$ and $R_2 R_3$ tend to infinity while keeping the ratio ρ constant, we obtain the following relationship between both covariance matrices:

$$\mathbf{S}_{\mathbf{X}_{(1)}} \simeq \rho \mathbf{S}_{\mathbf{Y}_{(1)}} + (1 - \rho) \sigma^2 \mathbf{I}_{I_1}. \tag{18}$$

In a strict sense, (18) holds at infinity. Thus, it does not hold generally in practice, however, it is meaningful to understand the behavior of mode-eigenvalues of Tucker model, and design the approach for improvement. It should be noted that there is a special case of $\mathbf{S}_{\mathbf{X}_{(1)}} = \mathbf{S}_{\mathbf{Y}_{(1)}}$ for $\rho = 1$ and $\mathbf{S}_{\mathbf{X}_{(1)}}$ is usually more noisy than $\mathbf{S}_{\mathbf{Y}_{(1)}}$ for $\rho < 1$. In the case of $R_2 R_3 < I_2 I_3$ (i.e., $\rho < 1$), the dimensions $(I_2 I_3 - R_2 R_3)$ corresponds to the noise subspaces. The model (13) includes the noise subspaces and the model (14) does not. Thus, the MDL obtained via $\mathbf{S}_{\mathbf{X}_{(1)}}$ based on the model (13), as introduced in Section II-B, would be more sensitive to noise than that based on

the model (14) if ρ is small. Therefore, when we apply MDL to the unfolded and matricized Tucker model with a small ρ , the model (14) would be better. In general, $\mathbf{S}_{\mathbf{Y}_{(1)}}$ is unknown but the eigenvalue decomposition of $\mathbf{S}_{\mathbf{Y}_{(1)}}$ can be approximated from (18) as $\mathbf{S}_{\mathbf{Y}_{(1)}} \simeq \frac{1}{\rho} \mathbf{S}_{\mathbf{X}_{(1)}} - \frac{1-\rho}{\rho} \sigma^2 \mathbf{I}_{I_1} = \mathbf{V}^{(1)} \left(\frac{1}{\rho} \mathbf{\Lambda}^{(1)} - \frac{1-\rho}{\rho} \sigma^2 \mathbf{I}_{I_1} \right) \mathbf{V}^{(1)T}$. Thus, the effect of ρ can be reduced by using the following modification:

$$\begin{aligned} \mathbf{\Lambda}_{\text{mod}}^{(1)} &= \text{diag} \left(\lambda_1^{(1)\text{mod}}, \lambda_2^{(1)\text{mod}}, \dots, \lambda_{I_1}^{(1)\text{mod}} \right) \\ &= \frac{1}{\hat{\rho}} \mathbf{\Lambda}^{(1)} - \frac{1-\hat{\rho}}{\hat{\rho}} \hat{\sigma}^2 \mathbf{I}_{I_1}, \end{aligned} \quad (19)$$

where $\mathbf{\Lambda}^{(1)} \in \mathbb{R}^{I_1 \times I_1}$ is the eigenvalue matrix of $\mathbf{S}_{\mathbf{X}_{(1)}}$, $\hat{\sigma}^2$ is an estimator of noise variance, and $\hat{\rho}$ is an estimator of ρ . The remaining problem comprises how to choose the values of $\hat{\sigma}^2$ and $\hat{\rho}$. In Section IV, we discuss an estimation method for $\hat{\sigma}^2$ and $\hat{\rho}$ by exploiting the core tensor in HOSVD.

IV. NOISE VARIANCE ESTIMATOR

Next, we discuss how to estimate the noise variance $\hat{\sigma}^2$ by considering a Tucker model with noise (12). Since i.i.d. Gaussian noise does not change the eigenvectors of the covariance matrix, then we have $\mathbf{V}^{(n)} \simeq [\mathbf{U}^{(n)}, \tilde{\mathbf{U}}^{(n)}]$, where $\tilde{\mathbf{U}}^{(n)}$ is a set of basis vectors spanning the orthogonal complement space of $\mathbf{U}^{(n)}$. A permutation matrix $\mathbf{P} \in \mathbb{R}^{I_2 I_3 \times I_2 I_3}$ exists such that $(\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T (\mathbf{V}^{(3)} \otimes \mathbf{V}^{(2)}) \mathbf{P} = [\mathbf{I}, \mathbf{0}]$. Hence, we have

$$\begin{aligned} \mathbf{H}_{(1)} \mathbf{P} &= \mathbf{V}^{(1)T} \mathbf{X}_{(1)} (\mathbf{V}^{(3)} \otimes \mathbf{V}^{(2)}) \mathbf{P} \\ &= \mathbf{V}^{(1)T} \left\{ \mathbf{U}^{(1)} \mathbf{G}_{(1)} (\mathbf{U}^{(3)} \otimes \mathbf{U}^{(2)})^T + \mathbf{E}_{(1)} \right\} \\ &\quad \cdot (\mathbf{V}^{(3)} \otimes \mathbf{V}^{(2)}) \mathbf{P} \\ &= \begin{pmatrix} \mathbf{G}_{(1)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} + \mathbf{E}'_{(1)} = \begin{pmatrix} \mathbf{G}_{(1)} + \mathbf{E}_1 & \mathbf{E}_2 \\ \mathbf{E}_3 & \mathbf{E}_4 \end{pmatrix}, \end{aligned} \quad (20)$$

where $\mathbf{E}'_{(1)} := \mathbf{V}^{(1)T} \mathbf{E}_{(1)} (\mathbf{V}^{(3)} \otimes \mathbf{V}^{(2)}) \mathbf{P}$, and $\mathbf{E}_1 \in \mathbb{R}^{R_1 \times R_2 R_3}$, $\mathbf{E}_2 \in \mathbb{R}^{R_1 \times (I_2 I_3 - R_2 R_3)}$, $\mathbf{E}_3 \in \mathbb{R}^{(I_1 - R_1) \times R_2 R_3}$, and $\mathbf{E}_4 \in \mathbb{R}^{(I_1 - R_1) \times (I_2 I_3 - R_2 R_3)}$ are block matrices of $\mathbf{E}'_{(1)}$. We note the orthonormal transform and permutation preservation properties of i.i.d. Gaussian noise: $[\mathbf{E}'_{(1)}]_{ij} \sim N(0, \sigma^2)$. Let $\mathbf{H}_{(1)} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_{I_1}]^T$, \mathbf{h}_i for $i = 1, \dots, R_1$ correspond to the block $[\mathbf{G}_{(1)} + \mathbf{E}_1, \mathbf{E}_2]$, and \mathbf{h}_i for $i = R_1 + 1, \dots, I_1$ correspond to the noise block $[\mathbf{E}_3, \mathbf{E}_4]$. The relationship between an estimated core tensor $\mathbf{H}_{(1)}$ and a true core tensor $\mathbf{G}_{(1)} = [\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_{R_1}]^T$ can be expressed as

$$\langle \mathbf{h}_i, \mathbf{h}_i \rangle \approx (\langle \mathbf{g}_i, \mathbf{g}_i \rangle + R_2 R_3 \sigma^2) + (I_2 I_3 - R_2 R_3) \sigma^2, \quad (21)$$

where $\langle \mathbf{g}_i, \mathbf{g}_i \rangle = 0$ for $i \in \{R_1 + 1, R_1 + 2, \dots, I_1\}$. The noise terms $R_2 R_3 \sigma^2$ and $(I_2 I_3 - R_2 R_3) \sigma^2$ have different interpretations. The former noise term corresponds to $\mathbf{F}_{(1)}$ in Eq. (14) and it is the same noise level in the low rank matrix model. By contrast, the latter noise term is removed in Eq. (14) and it appears only in the multilinear low-rank tensor model. The noise in Eq. (18) also corresponds to the latter noise $(I_2 I_3 - R_2 R_3) \sigma^2$, so we focus on the latter noise term $(I_2 I_3 - R_2 R_3) \sigma^2$ to estimate σ^2 . Note that the noise term $(I_2 I_3 - R_2 R_3) \sigma^2$ corresponds

Algorithm 1 Modified eigenvalues estimator for Tucker rank determination (MEET)

-
- 1: **input:** $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$
 - 2: $\bar{I} = \prod_{n=1}^N I_n$;
 - 3: $I_{\bar{n}} = \prod_{k \neq n} I_k$ for $n \in \{1, 2, \dots, N\}$;
 - 4: **for** $n = 1, 2, \dots, N$ **do**
 - 5: $\mathbf{V}^{(n)} \leftarrow$ a set of all eigenvectors of $I_{\bar{n}}^{-1} \mathbf{X}_{(n)} \mathbf{X}_{(n)}^T$;
 - 6: **end for**
 - 7: $\mathcal{H} = \mathcal{X} \times_1 \mathbf{V}^{(1)T} \times_2 \mathbf{V}^{(2)T} \times_3 \dots \times_N \mathbf{V}^{(N)T}$;
 - 8: **for** $n = 1, 2, \dots, N$ **do**
 - 9: $\lambda_i^{(n)} \leftarrow \frac{1}{I_{\bar{n}}} (\mathbf{H}_{(n)} \mathbf{H}_{(n)}^T)_{ii}$ for $i \in \{1, 2, \dots, I_n\}$;
 - 10: $\mu_j \leftarrow (\mathbf{H}_{(n)}^T \mathbf{H}_{(n)})_{jj}$ for $j \in \{1, 2, \dots, I_{\bar{n}}\}$;
 - 11: $\boldsymbol{\nu} \leftarrow$ sorted array of $\boldsymbol{\mu}$ in descending order;
 - 12: $\hat{\rho} \leftarrow \operatorname{argmin}_{\rho} \left\{ \rho \in S_{\rho} \mid \lambda_{I_n}^{(n)\text{mod}} > 0 \right\}$;
 - 13: $\hat{\sigma}^2 = \frac{1}{I(1-\hat{\rho})} \sum_{k=\hat{\rho}I_{\bar{n}+1}}^{I_{\bar{n}}} \nu_k$;
 - 14: $\lambda_i^{(n)\text{mod}} \leftarrow \frac{1}{\hat{\rho}} \lambda_i^{(n)} - \frac{(1-\hat{\rho})}{\hat{\rho}} \hat{\sigma}^2$ for $i \in \{1, 2, \dots, I_n\}$;
 - 15: Estimate R_n via the MDL criterion with $\lambda_i^{(n)\text{mod}}$;
 - 16: **end for**
 - 17: **output:** (R_1, R_2, \dots, R_N)
-

to \mathbf{E}_2 and \mathbf{E}_4 . The permutation matrix can be split into two matrices: $\mathbf{P} = [\mathbf{P}_G, \mathbf{P}_E]$, where $\mathbf{P}_G \in \mathbb{R}^{I_2 I_3 \times R_2 R_3}$, and $\mathbf{P}_E \in \mathbb{R}^{I_2 I_3 \times (I_2 I_3 - R_2 R_3)}$; hence from (20), we have

$$\begin{aligned}
\sigma^2 &\approx \frac{1}{I_1(I_2 I_3 - R_2 R_3)} \left\{ \operatorname{tr}(\mathbf{E}_2 \mathbf{E}_2^T) + \operatorname{tr}(\mathbf{E}_4 \mathbf{E}_4^T) \right\} \\
&= \frac{1}{I_1(I_2 I_3 - R_2 R_3)} \operatorname{tr}(\mathbf{H}_{(1)} \mathbf{P}_E \mathbf{P}_E^T \mathbf{H}_{(1)}^T) \\
&= \frac{1}{I_1(I_2 I_3 - R_2 R_3)} \sum_{k=R_2 R_3+1}^{I_2 I_3} (\boldsymbol{\mu}^T \mathbf{P})_k,
\end{aligned} \tag{22}$$

where $\mu_k := (\mathbf{H}_{(1)}^T \mathbf{H}_{(1)})_{kk}$ for $k \in \{1, 2, \dots, I_2 I_3\}$. $R_2 R_3 = \rho I_2 I_3$, so the problem can be converted into estimating \mathbf{P} and ρ .

We have

$$(\boldsymbol{\mu}^T \mathbf{P})_k \approx \begin{cases} \sum_{i=1}^{R_1} [\mathbf{G}_{(1)}]_{ik}^2 + I_1 \sigma^2 & k \leq R_2 R_3 \\ I_1 \sigma^2 & \text{otherwise} \end{cases}, \tag{23}$$

and thus the latent factor entries are approximately larger than the noise factor entries in $\boldsymbol{\mu}$. Therefore, we propose that \mathbf{P} can be obtained by sorting $\boldsymbol{\mu}$. The sorting procedure for $\boldsymbol{\mu}$ in descending order outputs the sorted array $\boldsymbol{\nu}$, where $\nu_1 \geq \nu_2 \geq \dots \geq \nu_{I_2 I_3}$, and $\boldsymbol{\mu}^T \hat{\mathbf{P}} = \boldsymbol{\nu}^T$. Finally, we can estimate ρ and σ^2 by applying the following

criterion:

$$\hat{\sigma}^2 = \frac{1}{I_1 I_2 I_3 (1 - \hat{\rho})} \sum_{k=\hat{\rho} I_2 I_3 + 1}^{I_2 I_3} \nu_k, \quad (24)$$

$$\hat{\rho} = \operatorname{argmin}_{\rho} \left\{ \rho \in S_{\rho} \mid \lambda_{I_1}^{(1)\text{mod}} > 0 \right\}, \quad (25)$$

where S_{ρ} is a set of candidates for ρ with $0 < \rho < 1$. Figure 1(b) shows a visual illustration of the computation of the noise variance estimator $\hat{\sigma}^2$. The parameter $\hat{\rho}$ indicates how the whole matrix $\mathbf{H}_{(1)}\mathbf{P}$ is divided into left-hand side and right-hand side matrices. The noise variance estimator $\hat{\sigma}^2$ is computed based on the average of the squares for all entries in the right-hand side matrix. According to (19), the modified eigenvalues could be negative when the noise variance estimator is excessively large. $\lambda_{I_1}^{(1)\text{mod}}$ is the minimum of the modified eigenvalues, so we constrain it to being positive in (25). By substituting (24) into (19), the condition that $\lambda_{I_1}^{(1)\text{mod}} > 0$ can be transformed into

$$\begin{aligned} \lambda_{I_1}^{(1)\text{mod}} &= \frac{1}{\hat{\rho}} \lambda_{I_1}^{(1)} - \frac{1}{\hat{\rho} I_1 I_2 I_3} \sum_{k=\hat{\rho} I_2 I_3 + 1}^{I_2 I_3} \nu_k > 0, \\ \iff I_1 I_2 I_3 \lambda_{I_1}^{(1)} &> \sum_{k=\hat{\rho} I_2 I_3 + 1}^{I_2 I_3} \nu_k. \end{aligned} \quad (26)$$

(25) can be solved easily by line-search because $\sum_{k=\hat{\rho} I_2 I_3 + 1}^{I_2 I_3} \nu_k$ increases monotonically as $\hat{\rho}$ decreases.

Finally, the rank is estimated by the MDL:

$$\begin{aligned} \operatorname{argmin}_r - 2 \log &\left\{ \frac{\prod_{i=r+1}^{I_n} (\lambda_i^{(n)\text{mod}})^{1/(I_n-r)}}{\frac{1}{I_n-r} \sum_{i=r+1}^{I_n} \lambda_i^{(n)\text{mod}}} \right\}^{\hat{\rho} I_{\bar{n}} (I_n-r)} \\ &+ r(2I_n - r) \log(\hat{\rho} I_{\bar{n}}), \end{aligned} \quad (27)$$

where $I_{\bar{n}} = \prod_{k \neq n} I_k$.

We summarize the generalized procedure of the proposed method in Algorithm 1 for any $N \geq 3$. We refer to the proposed method as MEET.

V. SCORE ALGORITHM

In the previous section, we proposed an improved method for modified eigenvalue estimation given by Eq. (19). The proposed modified eigenvalue estimator is based on the selective estimation of the noise level using $\hat{\mathbf{P}}_E$. In this section, we consider another efficient approach for reconstructing eigenvalues from $\mathbf{H}_{(n)}$ using $\hat{\mathbf{P}}_G$, as follows:

$$\lambda_i^{(1)\text{sp}} \leftarrow \frac{1}{\hat{\rho} I_2 I_3} (\mathbf{H}_{(1)} \hat{\mathbf{P}}_G \hat{\mathbf{P}}_G^T \mathbf{H}_{(1)}^T)_{ii}, \quad (28)$$

for $i \in \{1, 2, \dots, I_n\}$, where $\hat{\mathbf{P}}_G \in \mathbb{R}^{I_2 I_3 \times \hat{\rho} I_2 I_3}$ and $\lambda_1^{\text{sp}} \geq \lambda_2^{\text{sp}} \geq \dots \geq \lambda_{I_n}^{\text{sp}}$. Figure 1(c) shows a visual illustration of the computation of $\lambda_i^{(1)\text{sp}}$. The use of several columns in the unfolding matrix $\mathbf{H}_{(1)}$ of the core tensor can be regarded as an approximation of Tucker decomposition by HOSVD with a block-sparse core tensor. Thus, we refer to the proposed method as ‘‘SCORE’’ (sparse core). We note that the permutation matrix $\hat{\mathbf{P}}$ plays a key role for both the MEET and SCORE algorithms (e.g., see Eqs. (23), (24), (25), and (28)). In contrast to MEET, the SCORE

Algorithm 2 SCORE algorithm

-
- 1: **input:** $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$, and $\hat{\rho}$ (typically, 0.0001-0.01);
 - 2: $\bar{I} = \prod_{n=1}^N I_n$;
 - 3: $I_{\bar{n}} = \prod_{k \neq n} I_k$ for $n \in \{1, 2, \dots, N\}$;
 - 4: **for** $n = 1, 2, \dots, N$ **do**
 - 5: $\mathbf{V}^{(n)} \leftarrow$ a set of all eigenvectors of $I_{\bar{n}}^{-1} \mathbf{X}_{(n)} \mathbf{X}_{(n)}^T$;
 - 6: **end for**
 - 7: $\mathcal{H} = \mathcal{X} \times_1 \mathbf{V}^{(1)T} \times_2 \mathbf{V}^{(2)T} \times_3 \dots \times_N \mathbf{V}^{(N)T}$;
 - 8: **for** $n = 1, 2, \dots, N$ **do**
 - 9: $\mu_j \leftarrow (\mathbf{H}_{(n)}^T \mathbf{H}_{(n)})_{jj}$ for $j \in \{1, 2, \dots, I_{\bar{n}}\}$;
 - 10: Obtain the permutation matrix $\hat{\mathbf{P}} = [\hat{\mathbf{P}}_G, \hat{\mathbf{P}}_E]$ by sorting $\boldsymbol{\mu}$ in descending order;
 - 11: $\lambda_i^{(n)\text{sp}} \leftarrow \frac{1}{\hat{\rho} I_{\bar{n}}} (\mathbf{H}_{(n)} \hat{\mathbf{P}}_G \hat{\mathbf{P}}_G^T \mathbf{H}_{(n)}^T)_{ii}$ for $i \in \{1, 2, \dots, I_{\bar{n}}\}$;
 - 12: Sort $\lambda_i^{(n)\text{sp}}$ to satisfy $\lambda_1^{(n)\text{sp}} \geq \lambda_2^{(n)\text{sp}} \geq \dots \geq \lambda_{I_{\bar{n}}}^{(n)\text{sp}}$;
 - 13: Estimate R_n via the MDL criterion with $\lambda_i^{(n)\text{sp}}$;
 - 14: **end for**
 - 15: **output:** (R_1, R_2, \dots, R_N)
-

TABLE I
ESTIMATORS OF ρ AND σ .

ρ	σ	$\hat{\rho}$	$\hat{\sigma}$
0.04	0.2	0.14	0.196
0.04	0.4	0.14	0.393
0.04	0.6	0.14	0.587
0.25	0.6	0.19	0.616
0.25	0.8	0.18	0.810
0.25	1.0	0.17	1.006
0.49	2.0	0.16	2.008
0.49	2.5	0.14	2.489
0.49	3.0	0.15	2.959

estimator is always positive for any $\hat{\rho} > 0$. The SCORE algorithm is simply developed by replacing $\lambda_i^{(1)\text{mod}}$ by $\lambda_i^{(1)\text{sp}}$ in Algorithm 1. In contrast to MEET, SCORE does not need to estimate the unknown noise variance σ^2 and the ratio parameter $\hat{\rho}$ can be chosen from a wider range of $0 < \hat{\rho} < 1$. Empirically, the SCORE algorithm is quite robust to high noise by choosing a smaller value of $\hat{\rho}$ and thus $\hat{\rho}$ is typically in the range of 0.0001–0.01. In Section VI-A3, we discuss the relationship between $\hat{\rho}$ and the noise level based on extensive experiments. We demonstrate that a sufficiently small value of $\hat{\rho}$ provides very good performance with almost any noise level. The SCORE algorithm for general $N \geq 3$ is summarized in Algorithm 2.

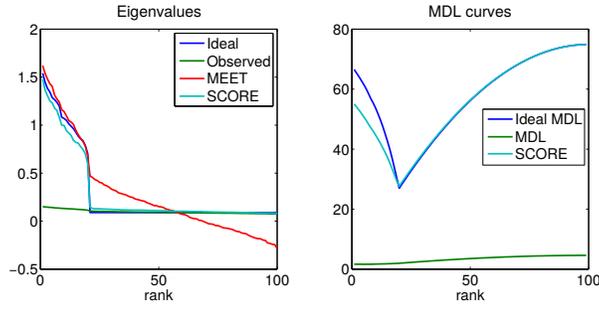


Fig. 2. Eigenvalues and MDL curves of the ideal, observed, MEET, and SCORE estimators with the noisy Tucker model (SNR = -10.7 dB) obtained using the true values for $\hat{\rho} = \rho = 0.04$ and $\sigma = 0.3$. The true rank is 20.

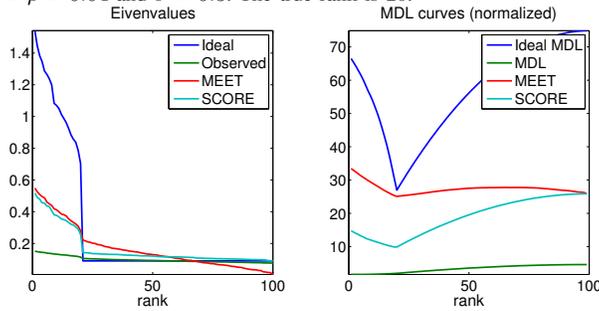


Fig. 3. Eigenvalues and MDL curves of the ideal, observed, MEET, and SCORE estimators with the noisy Tucker model (SNR = -10.7 dB) obtained using the estimators $\hat{\rho} = 0.15$ and $\hat{\sigma} = 0.29$. The true rank is 20, $\rho = 0.04$, and $\sigma = 0.3$.

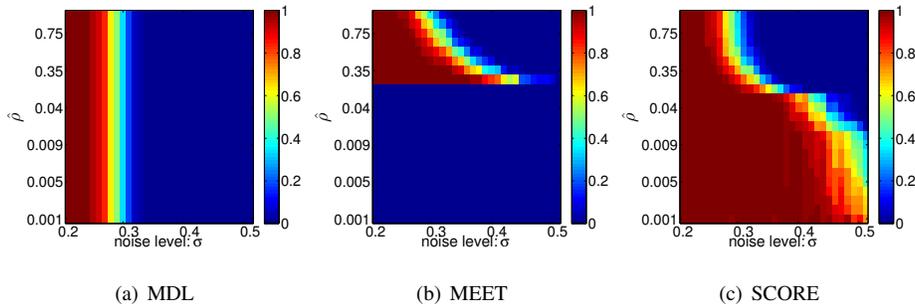


Fig. 4. Accuracy of multilinear rank estimation for various noise levels σ and the parameter $\hat{\rho}$. The true ratio ρ is 0.04.

VI. EXPERIMENTAL RESULTS

A. Synthetic simulations

First, we generated a data tensor for the Tucker model (12) with using several different settings of (I_1, I_2, I_3) and (R_1, R_2, R_3) . All of the elements of $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ and $U^{(n)} \in \mathbb{R}^{I_n \times R_n}$ were generated according to a normal distribution. The noise tensor $\mathcal{E} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ was generated based on a normal distribution with a mean of zero and variance of σ^2 . A noise-free tensor was given by $\mathcal{X}_0 = \mathcal{G} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}$, and a noisy tensor by $\mathcal{X} = \mathcal{X}_0 + \mathcal{E}$. We calculated the noise level as the signal-to-noise ratio (SNR). The HOSVDs for \mathcal{X}_0 and \mathcal{X} were given by $\mathcal{X}_0 = \mathcal{H}_0 \times_1 V_0^{(1)} \times_2 V_0^{(2)} \times_3 V_0^{(3)}$, and $\mathcal{X} = \mathcal{H} \times_1 V^{(1)} \times_2 V^{(2)} \times_3 V^{(3)}$. Each entry of \mathcal{E} follows

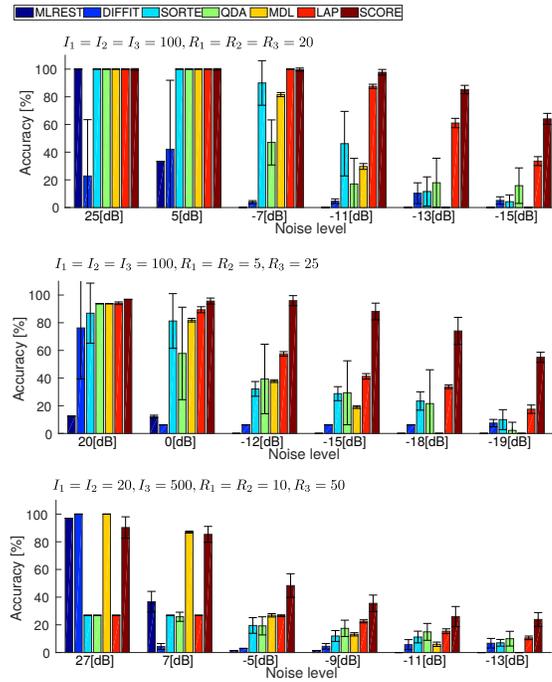


Fig. 5. Comparison of the proposed algorithm with other state-of-the-art methods.

TABLE II
AVERAGE COMPUTATIONAL TIME REQUIREMENTS

Method	MLREST	DIFFIT	SORTE	QDA	MDL	LAP	SCORE
Time[sec]	7.3	63	0.013	0.010	0.033	0.13	0.11

an i.i.d. normal distribution, so $\mathbf{V}_0^{(n)}$ and $\mathbf{V}^{(n)}$ are theoretically equivalent. We considered \mathcal{H}_0 and \mathcal{H} as the true and observed core tensors, respectively. We note that \mathcal{H}_0 does not contain any corresponding noise components. According to Eq. (17), the true eigenvalues can be simulated by $(\lambda_{\text{ideal}})_i = \frac{1}{R_2 R_3} ([\mathbf{H}_0]_{(1)} [\mathbf{H}_0]_{(1)}^T)_{ii} + \sigma^2$. In addition, the eigenvalues of the observed data $\mathbf{X}_{(1)}$ were calculated by $(\lambda_{\text{obs}})_i = \frac{1}{I_2 I_3} (\mathbf{H}_{(1)} \mathbf{H}_{(1)}^T)_{ii}$.

1) *Evaluations of the estimates of ρ and σ* : For the first simulation, we evaluated the accuracy of the estimators of ρ and σ^2 by Eqs. (25) and (24). We fixed $I_1 = I_2 = I_3 = 100$ and varied the ranks (R_1, R_2, R_3) in $\{(20, 20, 20), (50, 50, 50), (70, 70, 70)\}$, which corresponded to variations of $\rho \in \{0.04, 0.25, 0.49\}$. The noise variance was also changed based on the SNR. Table I shows the results obtained for the estimators of ρ and σ . We can see that $\hat{\rho}$ estimators were not very accurate, whereas the $\hat{\sigma}$ estimators were accurate.

The robustness of the estimated noise variance is explained based on Figure 1(b). We note that the accuracy of the estimation of $\hat{\sigma}^2$ depends on the number of entries and the proportion of entries corresponding to noise in the matrix $\mathbf{H}_{(1)} \mathbf{P}_E$. When $\hat{\rho} < \rho$, the proportion of noise entries decreases but the number of total entries

in $\mathbf{H}_{(1)}\mathbf{P}_E$ increases. When $\hat{\rho} > \rho$, the proportion of noise entries is 100% but the total number of entries in $\mathbf{H}_{(1)}\mathbf{P}_E$ decreases. Thus, the accuracy of $\hat{\sigma}^2$ is relatively robust with respect to the estimation error in $\hat{\rho}$. Precisely estimating $\hat{\rho}$ as the ratio (R_2R_3/I_2I_3) is considered to be as difficult as rank estimation.

2) *Evaluations of the estimated eigenvalues and multilinear ranks:* In this experiment, the sizes of the data tensor and core tensor were set at $I_1 = I_2 = I_3 = 100$, and $R_1 = R_2 = R_3 = 20$, respectively, and we applied the MDL, MEET, and SCORE algorithms with two different values of $\hat{\rho}$. Figure 2 shows the results for the eigenvalue estimators and the MDL curves for MEET and SCORE obtained using the true ρ and σ . When $\hat{\rho} = \rho$, the eigenvalues estimated by SCORE were quite similar to the ground truth. Using the MEET algorithm, the estimated eigenvalues were accurate for the first to 20th values, but not accurate for the 21st to 100th and they included negative values. Thus, the MDL curve obtained by SCORE matched well with the ground truth, whereas the MDL curve obtained by MEET was not suitable because of the negative eigenvalues. Figure 3 shows the results for the eigenvalue estimators and the MDL curves for MEET and SCORE with the values of $\hat{\rho}$ and $\hat{\sigma}$ estimated by Eqs. (25) and (24). The eigenvalues were not estimated accurately, but the ranks selected from the MDL curves obtained by MEET and SCORE were much more accurate than those produced by standard MDL.

3) *Evaluation of the robustness for σ and $\hat{\rho}$:* In this simulation, we evaluated the accuracy of multilinear rank estimation using various settings for the noise level σ . The basic settings were the same as those used in the experiment above: $I_n = 100$, $R_n = 20$ for $n \in \{1, 2, 3\}$, etc. The rank estimation accuracy is defined by

$$\alpha(\hat{R}_1, \hat{R}_2, \dots, \hat{R}_N) := 1 - \frac{\sum_{n=1}^N \min(\Delta R_n, |R_n - \hat{R}_n|)}{\sum_{n=1}^N \Delta R_n}, \quad (29)$$

where $\Delta R_n = \min(I_n - R_n, R_n - 1)$. Obviously, $\alpha(\hat{R}_1, \hat{R}_2, \dots, \hat{R}_N) = 1$ if $R_n = \hat{R}_n$ for $n \in \{1, 2, \dots, N\}$, and thus we have $0 \leq \alpha \leq 1$.

We tested the rank estimation performance of the MDL, MEET, and SCORE algorithms using various values of $\hat{\rho}$. Figure 4 shows the rank estimation accuracy with various values of σ and $\hat{\rho}$ for each method. MDL failed to estimate the multilinear ranks for larger values of σ , regardless of $\hat{\rho}$. MEET performed very well for small values of $\hat{\rho}$, but it could not obtain good results for $\hat{\rho}$ smaller than a threshold due to the negative eigenvalues. SCORE performed very well for smaller values of $\hat{\rho}$. Furthermore, SCORE usually performed very well with smaller values of $\hat{\rho}$ regardless of the noise level. These results suggest that the precise selection of $\hat{\rho}$ is not important for the SCORE algorithm if we choose a sufficiently small value for $\hat{\rho}$. In addition, accurate estimation of the noise variance might not be necessary for the rank estimation problem. Essentially, the rank estimation problem may be related to the behavior as the eigenvalue changes rather than the exact values. The problem is that the behavior of the eigenvalue changes was often deformed by noise and the multilinear low-rank structure. Thus, the SCORE algorithm can be interpreted as a type of restoration for the behavior of the eigenvalue changes.

4) *Comparison of the proposed algorithm with state-of-the-art methods:* In these simulations, we compared the proposed SCORE algorithm with existing state-of-the-art methods: DIFFIT [32], MLREST [31], TDA-SORTE [40], BIC/MDL [30], [29], [35], QDA [25], and LAP [24]. To evaluate the performance of these methods, we calculated the accuracy (29) by changing the noise variance σ^2 over a very wide range. The basic settings were

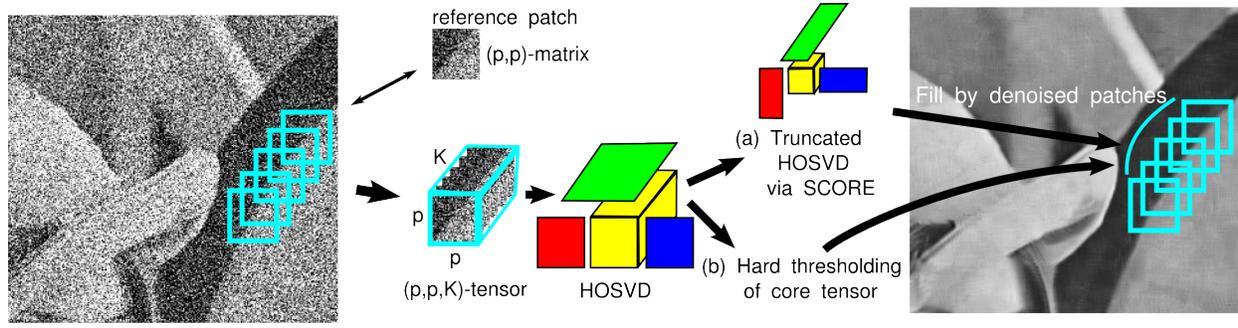


Fig. 6. Scheme for HOSVD denoising: (a) to estimate the multilinear rank by SCORE and to reconstruct patches using the truncated HOSVD; and (b) to remove some entries of the core tensor by hard thresholding (HT) and to reconstruct patches by HOSVD.



(a) Noisy image ($\sigma = 40$) (b) HOSVD with HT ($\hat{\sigma} = 20$) (c) HOSVD with HT ($\hat{\sigma} = 40$) (d) Truncated HOSVD via MDL



(e) Truncated HOSVD via SCORE (f) Truncated HOSVD with under-estimate ($\hat{R}_1 = \hat{R}_2 = \hat{R}_3 = 1$) (g) Truncated HOSVD with over-estimate ($\hat{R}_1 = \hat{R}_2 = 5$ and $\hat{R}_3 = 10$)

Fig. 7. Selected images in denoising experiments.

the same as those used in the experiments described above. DIFFIT and MLREST can be regarded as tensor-based methods because these methods evaluate each combination of multilinear ranks (R_1, R_2, \dots, R_N) as one unit. By contrast, SORTe, QDA, MDL, LAP, and SCORE are matrix-based methods because these methods estimate each R_n independently. In computational terms, matrix based methods are more practical because the number of all combinations of multilinear ranks (R_1, R_2, \dots, R_N) increases exponentially with N . In this set of experiments, we set $\hat{\rho} = 0.001$ for the SCORE algorithm.

TABLE III
RESULTS OBTAINED AFTER DENOISING “LENA” USING THREE METHODS.

	σ	PSNR				SSIM			
		noisy	HOSVD	MDL	SCORE	noisy	HOSVD	MDL	SCORE
Lena	20	22.1312	32.7440	31.9589	32.9154	0.4484	0.8888	0.8796	0.8927
	40	16.3679	29.1801	28.7732	29.3390	0.2410	0.7903	0.7992	0.8002
	60	13.2679	26.8078	26.6925	26.8932	0.1484	0.7012	0.7256	0.7083
Mandrill	20	22.1248	26.0259	23.0561	25.2466	0.6772	0.8172	0.6639	0.7889
	40	16.2889	23.0134	21.6267	22.7610	0.4188	0.6601	0.5420	0.6384
	60	13.1732	21.5674	20.8570	21.4663	0.2713	0.5396	0.4659	0.5271
Peppers	20	22.2198	31.5794	30.7761	31.7540	0.4122	0.8862	0.8838	0.8921
	40	16.3859	27.6211	27.0221	27.7665	0.2063	0.7784	0.7970	0.7915
	60	13.2662	25.1401	24.8017	25.2001	0.1255	0.6840	0.7179	0.6931

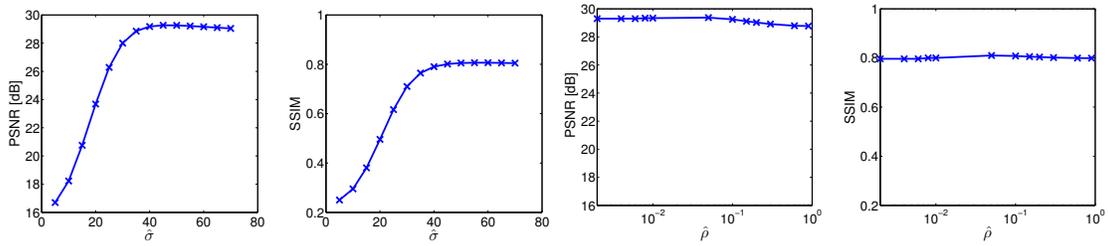
(a) HOSVD denoiser with various values of $\hat{\sigma}$ (b) SCORE denoiser with various values of $\hat{\rho}$

Fig. 8. Comparison of the HOSVD denoising and SCORE denoising performance with various values for the parameters $\hat{\sigma}$ and $\hat{\rho}$. The true noise variance was $\sigma = 40$.

We compared three sets of tensor and core tensor sizes: $(I_1 = I_2 = I_3 = 100, R_1 = R_2 = R_3 = 20)$, $(I_1 = I_2 = I_3 = 100, R_1 = R_2 = 5, R_3 = 25)$, and $(I_1 = I_2 = 20, I_3 = 500, R_1 = R_2 = 10, R_3 = 50)$. Furthermore, we varied the noise variance σ^2 based on the SNR. We evaluated the computational time and estimated accuracy of rank based on the average and standard deviation over 10 trials. A noisy tensor was generated randomly for each trial.

Table II shows the average computational times for 10 trials in the first setting (i.e., $I_1 = I_2 = I_3 = 100$ and $R_1 = R_2 = R_3 = 20$). The matrix-based methods were obviously faster than the tensor-based methods. Figure 5 shows the results in terms of accuracy for all of the methods, where the lengths of the bars denote the average accuracy and the error bars represent the standard deviation of accuracy. We can see that MLREST and DIFFIT worked well only for very small noise levels, but SORTE and QDA had low accuracy for high noise levels. LAP and SCORE were quite robust with high noise levels. In particular, the SCORE algorithm outperformed all of the other methods for almost any noise level.

B. Real-world application: Image denoising

In this experiment, we applied the proposed multilinear tensor rank estimation method to a denoising problem as an illustrative example. The denoising scheme based on HOSVD was proposed by [28], so we refer to this method as the ‘‘HOSVD denoiser’’ and the concept is illustrated in Figure 6. The HOSVD denoiser is regarded as a patch-based image processing method. In this algorithm, we selected a patch image, (p, p) -matrix, as a reference patch from the noisy image and we then found K similar patches in the peripheral regions. A (p, p, K) -tensor $\mathcal{Z} \in \mathbb{R}^{p \times p \times K}$ was constructed using the K similar patches and we factorized this tensor with the HOSVD: $\mathcal{Z} = \mathcal{H} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \mathbf{V}^{(3)}$, where $\mathcal{H} \in \mathbb{R}^{p \times p \times K}$, $\mathbf{V}^{(1)} \in \mathbb{R}^{p \times p}$, $\mathbf{V}^{(2)} \in \mathbb{R}^{p \times p}$, and $\mathbf{V}^{(3)} \in \mathbb{R}^{K \times K}$. Using hard thresholding [28], we obtained a modified core tensor by

$$\widehat{\mathcal{H}}(i, j, k) \leftarrow \begin{cases} \mathcal{H}(i, j, k) & \mathcal{H}(i, j, k) > \tau \\ 0 & \text{otherwise} \end{cases}, \quad (30)$$

where $\tau = \hat{\sigma} \sqrt{2 \log p^2 K}$. Hence, $\widehat{\mathcal{Z}}$ was reconstructed by $\widehat{\mathcal{H}} \times_1 \mathbf{V}^{(1)} \times_2 \mathbf{V}^{(2)} \times_3 \mathbf{V}^{(3)}$, and the denoised patches are returned to their original positions. We repeated these procedures for all of the reference patches and the overlapping pixels were filled based on their average. A key procedure in HOSVD denoising is hard thresholding of the core tensor, where its threshold τ depends on the noise variance parameter $\hat{\sigma}$. The noise variance parameter $\hat{\sigma}$ is considered to be a known parameter in [28], whereas we note that it is an unknown parameter in practical applications. The estimation of $\hat{\sigma}$ is a challenging problem for the HOSVD.

Next, we propose an alternative HOSVD denoising method. We estimate the ranks (R_1, R_2, R_3) using SCORE and reconstruct a patch tensor $\widehat{\mathcal{Z}}$ with the truncated HOSVD model: $\widehat{\mathcal{Z}} = \mathcal{G} \times_1 \mathbf{U}^{(1)} \times \mathbf{U}^{(2)} \times \mathbf{U}^{(3)}$, where $\mathcal{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$, $\mathbf{U}^{(1)} \in \mathbb{R}^{p \times R_1}$, $\mathbf{U}^{(2)} \in \mathbb{R}^{p \times R_2}$, and $\mathbf{U}^{(3)} \in \mathbb{R}^{K \times R_3}$. Core tensor is given by $\mathcal{G}(i, j, k) = \mathcal{H}(i, j, k)$ for $i \in \{1, 2, \dots, R_1\}$, $j \in \{1, 2, \dots, R_2\}$, and $k \in \{1, 2, \dots, R_3\}$. The factor matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$ are given as the left R_1 , R_2 , and R_3 column vectors of $\mathbf{V}^{(1)}$, $\mathbf{V}^{(2)}$, and $\mathbf{V}^{(3)}$, respectively. We refer to this method as the ‘‘SCORE denoiser’’. During the denoising of image data, overestimating the rank still provides noisy signals (Fig. 7(g)) and underestimating the rank yields distorted signals (Fig. 7(f)). Thus, the appropriate rank estimation is required in its denoising task.

In fact, both hard thresholding with HOSVD and low rank approximation via SCORE produce sparse core tensors. However, the SCORE denoiser performs slice-wise thresholding in contrast to the element-wise thresholding with the HOSVD denoiser. Thus, the low rank approximation in the SCORE denoiser can be interpreted as the hard thresholding of ‘‘factor matrices.’’

In this set of experiments, we used gray-scale images of ‘‘Lena’’ (512×512), ‘‘Mandrill’’ (512×512), and ‘‘Peppers’’ (256×256) for comparison, which were corrupted by additive Gaussian noise with $\sigma \in \{20, 40, 60\}$. The general parameters were set as $p = 8$ and $K = 30$, where the correct value of $\hat{\sigma}$ was used for the HOSVD denoiser and $\hat{\rho} = 0.01$ was used for the SCORE denoiser. Table III shows the peak SNR (PSNR) results and structural similarity (SSIM) measures for the images denoised by the HOSVD, MDL, and SCORE denoisers, where the MDL denoiser used the low-rank HOSVD with rank estimation by MDL. The performance of the SCORE denoiser was

very similar to that of the HOSVD denoiser for all images and noise levels although the correct $\hat{\sigma}$ (which is unknown in practice) was used for the HOSVD denoiser. This suggests that the rank estimation obtained by SCORE was accurate and suitable for a wide range of images and noise levels. Figure 7 shows a noisy image obtained with $\sigma = 40$ as well as the results obtained by the HOSVD denoiser with incorrect and correct values of $\hat{\sigma}$, the MDL denoiser, and the SCORE denoiser with $\hat{\rho} = 0.01$. We can see that the results obtained by the HOSVD denoiser with incorrect $\hat{\sigma}$ were still quite noisy, the results produced by the MDL denoiser were blurred, but the results were very clear using the HOSVD denoiser with the correct $\hat{\sigma}$ and the SCORE denoiser. Figure 8 shows the PSNR and SSIM results obtained by the HOSVD and SCORE denoisers for various values of $\hat{\sigma}$ and $\hat{\rho}$. The HOSVD denoiser was quite sensitive to the hyper-parameter $\hat{\sigma}$, but SCORE denoiser was quite robust to the parameter $\hat{\rho}$. Therefore, the SCORE denoiser is more suitable in practice, especially when $\hat{\sigma}$ is unknown.

VII. CONCLUSIONS

Multilinear tensor rank estimation is an important problem that affects the practical applications of tensor decomposition techniques. In this study, we greatly improved the performance of matrix-based multilinear tensor rank estimation methods for large noise by considering the multilinear low-rank structure of tensors from both theoretical and practical perspectives. We demonstrated the robustness of the SCORE algorithm in terms of the noise level based on extensive experiments. As a real-world application, we considered a denoising problem in this study, but the SCORE algorithm can also be applied to a wide range of applications such as multi-way blind source separation, dimensionality reduction, and the clustering of low-rank tensor data. The extension or improvement of the proposed methods to CP rank selection or other information theoretic criteria can be investigated in future research.

REFERENCES

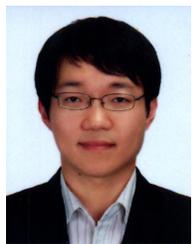
- [1] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.
- [2] R. Ballester-Ripoll and R. Pajarola. Lossy volume compression using Tucker truncation and thresholding. *The Visual Computer*, pages 1–14, 2015.
- [3] R. Bro and H. A. Kiers. A new efficient method for determining the number of components in PARAFAC models. *Journal of Chemometrics*, 17(5):274–286, 2003.
- [4] R. Bro, K. Kjeldahl, A. Smilde, and H. Kiers. Cross-validation of component models: A critical look at current methods. *Analytical and Bioanalytical Chemistry*, 390(5):1241–1251, 2008.
- [5] W. Chen, K. M. Wong, and J. P. Reilly. Detection of the number of signals: A predicted eigen-threshold approach. *IEEE Transactions on Signal Processing*, 39(5):1088–1098, 1991.
- [6] J. P. C. L. da Costa, K. Liu, H. C. So, S. Schwarz, M. Haardt, and F. Römer. Multidimensional prewhitening for enhanced signal reconstruction and parameter estimation in colored noise with kronecker correlation structure. *Signal Processing*, 93(11):3209–3226, 2013.
- [7] J. P. C. L. da Costa, F. Roemer, and M. Haardt. Sequential GSVD based prewhitening for multidimensional HOSVD based subspace estimation. In *Proc. ITG Workshop on Smart Antennas*, pages 205–212, 2009.
- [8] J. P. C. L. da Costa, F. Roemer, M. Haardt, and R. T. de Sousa. Multi-dimensional model order selection. *EURASIP Journal on Advances in Signal Processing*, 2011(1):1–13, 2011.
- [9] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

- [10] J. Grouffaud, P. Larzabal, and H. Clergeot. Some properties of ordered eigenvalues of a Wishart matrix: application in detection test and model order selection. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 1996 (ICASSP-96)*, volume 5, pages 2463–2466. IEEE, 1996.
- [11] R. A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics 16*, 1970.
- [12] Z. He, A. Cichocki, S. Xie, and K. Choi. Detecting the number of clusters in n-way probabilistic clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11):2006–2021, 2010.
- [13] C. J. Hillar and L.-H. Lim. Most tensor problems are NP-hard. *Journal of the ACM (JACM)*, 60(6):45, 2013.
- [14] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1):164–189, 1927.
- [15] P. D. Hoff. Equivariant and scale-free Tucker decomposition models. *Bayesian Analysis*, 11(3):627–648, 2016.
- [16] M. Ishteva. *Numerical methods for the best low multilinear rank approximation of higher-order tensors*. PhD thesis, Department of Electrical Engineering, Katholieke Universiteit Leuven, 2009.
- [17] M. Ishteva, P.-A. Absil, S. Van Huffel, and L. De Lathauwer. Best low multilinear rank approximation of higher-order tensors, based on the Riemannian trust-region scheme. *SIAM Journal on Matrix Analysis and Applications*, 32(1):115–135, 2011.
- [18] B. Jiang, F. Yang, and S. Zhang. Tensor and its Tucker core: The invariance relationships. *arXiv preprint arXiv:1601.01469*, 2016.
- [19] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.
- [20] S. Konishi and G. Kitagawa. Generalised information criteria in model selection. *Biometrika*, 83(4):875–890, 1996.
- [21] S. Kritchman and B. Nadler. Determining the number of components in a factor model from limited noisy data. *Chemometrics and Intelligent Laboratory Systems*, 94(1):19–32, 2008.
- [22] A. P. Liavas, P. A. Regalia, and J.-P. Delmas. Blind channel approximation: Effective channel order determination. *IEEE Transactions on Signal Processing*, 47(12):3336–3344, 1999.
- [23] K. Liu, J. P. C. L. da Costa, H. C. So, L. Huang, and J. Ye. Detection of number of components in CANDECOMP/PARAFAC models via minimum description length. *Digital Signal Processing*, 51:110–123, 2016.
- [24] T. P. Minka. Automatic choice of dimensionality for PCA. In *Advances in Neural Information Processing Systems 13*, pages 598–604. MIT Press, 2001.
- [25] J. Niesing. *Simultaneous Component and Factor Analysis Methods for Two or More Groups: A Comparative Study*. DSWO Press, Leiden University, 1997.
- [26] S. Pouryazdian, S. Beheshti, and S. Krishnan. CANDECOMP/PARAFAC model order selection based on reconstruction error in the presence of kronecker structured colored noise. *Digital Signal Processing*, 48:12–26, 2016.
- [27] A. Quinlan, J.-P. Barbot, P. Larzabal, and M. Haardt. Model order selection for short data: An exponential fitting test (EFT). *EURASIP Journal on Applied Signal Processing*, 2007(1):201–201, 2007.
- [28] A. Rajwade, A. Rangarajan, and A. Banerjee. Image denoising using the higher order singular value decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):849–862, 2013.
- [29] J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.
- [30] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.
- [31] L. Sorber, M. Van Barel, and L. De Lathauwer. Tensorlab v2.0. Available online, URL: www.tensorlab.net, 2014.
- [32] M. E. Timmerman and H. A. Kiers. Three-mode principal components analysis: Choosing the numbers of components and sensitivity to local optima. *British Journal of Mathematical and Statistical Psychology*, 53(1):1–16, 2000.
- [33] R. Tomioka, K. Hayashi, and H. Kashima. Estimation of low-rank tensors via convex optimization. *arXiv preprint, arXiv:1010.0789*, 2010.
- [34] L. R. Tucker. Implications of factor analysis of three-way matrices for measurement of change. In *Problems in Measuring Change*, pages 122–137. University of Wisconsin Press, 1963.
- [35] M. Wax and T. Kailath. Detection of signals by information theoretic criteria. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 33(2):387–392, 1985.
- [36] M. Wax and I. Ziskind. Detection of the number of coherent signals by the MDL principle. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(8):1190–1196, 1989.
- [37] H.-T. Wu, J.-F. Yang, and F.-K. Chen. Source number estimators using transformed Gerschgorin radii. *IEEE Transactions on Signal Processing*, 43(6):1325–1333, 1995.

- [38] Q. Wu and D. R. Fuhrmann. A parametric method for determining the number of signals in narrow-band direction finding. *IEEE Transactions on Signal Processing*, 39(8):1848–1857, 1991.
- [39] T. Yokota and A. Cichocki. Multilinear tensor rank estimation via sparse Tucker decomposition. In *Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*, pages 478–483. IEEE, 2014.
- [40] G. Zhou and A. Cichocki. TDALAB: Tensor decomposition laboratory. *LABSP, Wako-shi, Japan*, URL: <http://www.bsp.brain.riken.jp/TDALAB/>, 2013.



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.



Namgil Lee received his Ph.D. degree in Mathematics from the Korea Advanced Institute of Science and Technology in 2013. Currently, he is a research scientist of the RIKEN Brain Science Institute, Japan. His research interests lie in multivariate statistical modeling and machine learning. His research currently focuses on developing large-scale optimization algorithms using low-rank tensor networks for time series analysis and signal processing of neuroscience data.



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 Trans. on Signals Processing*, *IEEE Trans. on Neural Networks and Learning Systems*, *IEEE Trans on Cybernetics*, *Journal of Neuroscience 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.