

GENERALIZED VISUAL INFORMATION ANALYSIS VIA TENSORIAL ALGEBRA

LIANG LIAO AND STEPHEN JOHN MAYBANK

LIAOLIANGIS@126.COM, SJMAYBANK@DCS.BBK.AC.UK

ABSTRACT. High order data is modeled using matrices whose entries are numerical arrays of a fixed size. These arrays, called t-scalars, form a commutative ring under the convolution product. Matrices with elements in the ring of t-scalars are referred to as t-matrices. The t-matrices can be scaled, added and multiplied in the usual way. There are t-matrix generalizations of positive matrices, orthogonal matrices and Hermitian symmetric matrices. With the t-matrix model, it is possible to generalize many well-known matrix algorithms. In particular, the t-matrices are used to generalize the SVD (Singular Value Decomposition), HOSVD (High Order SVD), PCA (Principal Component Analysis), 2DPCA (Two Dimensional PCA) and GCA (Grassmannian Component Analysis). The generalized t-matrix algorithms, namely TSVD, THOSVD, TPCA, T2DPCA and TGCA, are applied to low-rank approximation, reconstruction, and supervised classification of images. Experiments show that the t-matrix algorithms compare favorably with standard matrix algorithms.

KEYWORDS. Commutative ring, Generalized scalars, Grassmannian manifold, Image Analysis, Tensor singular value decomposition, Tensors

1. INTRODUCTION

In data analysis, machine learning and computer vision, the data are often given in the form of multi-dimensional arrays of numbers. For example, an RGB image has three dimensions, namely two for the pixel array and a third dimension for the values of the pixels. An RGB image is said to be an array of order three. Alternatively, the RGB image is said to have three modes or to be three-way. A video sequence of images is of order four, with two dimensions for the pixel array, one dimension for time and a fourth dimension for the pixel values.

One way of analyzing multi-dimensional data is to remove the array structure by flattening, to obtain a vector. A set of vectors obtained in this way can be analyzed using standard matrix-vector algorithms such as the singular value decomposition (SVD) and principal components analysis (PCA). An alternative to flattening is to use algorithms that preserve the multi-dimensional structure. In these algorithms, the elements of matrices and vectors are entire arrays rather than real numbers in \mathbb{R} or complex numbers in \mathbb{C} . Multi-dimensional arrays with the same dimensions can be added in the usual way, but there is no definition of multiplication which satisfies the requirements for a field such as \mathbb{R} or \mathbb{C} . However, multiplication based on the convolution product has

many but not all of the properties of a field. Convolution multiplication differs from the multiplication in a field in that many elements have no multiplicative inverse. The multi-dimensional arrays with given dimensions form a commutative ring under the convolution product. The elements of this ring are referred to as t-scalars.

An application of the Fourier transform shows that each ring of t-scalars under the convolution product is isomorphic to a ring of arrays in which the Hadamard product defines the multiplication. In effect, the ring obtained by applying the Fourier transform splits into a product of copies of \mathbb{C} . It is this splitting which allows the construction of new algorithms for analyzing tensorial data without flattening. The so-called t-matrices with t-scalar entries have many of the properties of matrices with elements in \mathbb{R} or \mathbb{C} . In particular, t-matrices can be scaled, added and multiplied. There is an additive identity and a multiplicative identity. The determinant of a t-matrix is defined and a given t-matrix is invertible if and only if it has an invertible determinant. The t-matrices include generalizations of positive matrices, orthogonal matrices and symmetric matrices.

A tensorial version, TSVD, of the SVD is described in [18] and [41]. The TSVD expresses a t-matrix as the product of three t-matrices, of which two are generalizations of the orthogonal matrices and one is a diagonal matrix with positive t-scalars on the diagonal. The TSVD is used to define tensorial versions of principal components analysis (PCA) and two dimensional PCA (2DPCA). A tensorial version of Grassmannian components analysis is also defined. These tensorial algorithms are tested by experiments that include low-rank approximations to tensors, reconstruction of tensors and terrain classification using hyperspectral images. The different algorithms are compared using the peak signal to noise ratio and Cohen's kappa.

The t-scalars are described in Section 2 and the t-matrices are described in Section 3. The TSVD is described in Section 4. A tensorial version of principal components analysis (TPCA) is obtained from the TSVD in Section 5 and then generalized to tensorial two dimensional PCA (T2DPCA). A tensorial version of Grassmannian components analysis is also defined. The tensorial algorithms are tested experimentally in Section 6. Some concluding remarks are made in section 7.

1.1. Related Work. A tensor of order two or more can be simplified using the so-called N mode singular value decomposition (SVD). The three mode case is described by Tucker in [30]. The multi-modal case is discussed in detail by De Lathauwer et al. in [6]. Each mode of the tensor has an associated set of vectors, each one of which is obtained by varying the index for the given mode while keeping the indices of the other modes fixed. In the N mode SVD, an orthonormal basis is obtained for the space spanned by these vectors. In the 2-mode case, the result is the usual SVD. The resulting decomposition of a tensor is referred to as the higher-order SVD (HOSVD). Surveys of tensor decompositions can be found in Kolda and Bader [19] and Sidiropoulos et al. [27]. De Lathauwer et al. [6] describe a higher-order eigenvalue decomposition. Vasilescu and Terzopoulos [34] use the N mode SVD to simplify a fifth-order tensor constructed from face images taken under varying conditions and with varying expressions. A tensor version of the singular value decomposition is described in [18], [41], and [17].

He et al. [15] sample a hyperspectral data cube to yield tensors of order three of which two orders are for the pixel array and one order is for the hyperspectral bands. A training set of samples is used to produce a dictionary for sparse classification. Lu et al. [23] use N -mode analysis to obtain projections of tensors to a lower-dimensional space. The resulting multilinear PCA is applied to the classification of gait images. Vannieuwenhoven et al. [32] describe a new method for truncating the higher-order SVD, to obtain low-rank multilinear approximations to tensors. The method is tested on the classification of handwritten digits and the compression of a database of face images.

Many authors have studied algebras of matrices in which the elements are tensors of order one, equipped with a convolution multiplication, under which they form a commutative ring R with a multiplicative identity. In particular, Gleich et al. [11] describe the generalized eigenvalues and eigenvectors of matrices with elements in R and show how the standard power method for finding an eigenvector and the standard Arnoldi method for constructing an orthogonal basis for a Krylov subspace can both be generalized. Braman [3] shows that the t -vectors with a given dimension form a free module over R . Kilmer and Martin [18] show that many of the properties and structures of canonical matrices and vectors can be generalized. Their examples include transposition, orthogonality and the singular value decomposition (SVD). The tensor SVD is used to compress tensors. A tensor-based method for image de-blurring is also described. Kilmer et al. [17] generalize the inner product of two vectors, suggest a notion of the angle between two vectors with elements in R , and define a notion of orthogonality for two vectors. A generalization of the Gram-Schmidt method for generating an orthonormal set of vectors is also described in [17].

Zhang et al. [41] use the tensor SVD to store video sequences efficiently and also to fill in missing entries in video sequences. Zhang et al. [39] use a randomized version of the tensor SVD to produce low-rank approximations to matrices. Ren et al. [28] define a tensor version of principal component analysis and use it to extract features from hyperspectral images. The features are classified using standard methods such as support vector machines and nearest neighbors. Liao et al. [20] generalize a sparse representation classifier to tensor data and apply the generalized classifier to image data such as numerals and faces. Chen et al. [4] use a four-dimensional HOSVD to detect changes in a time sequence of hyperspectral images. The K-means clustering algorithm is used to classify the pixel values as changed or unchanged. Fan et al. [8] model a hyperspectral image as the sum of an ideal image, a sparse noise term and a Gaussian noise term. A product of two low-rank tensors models the ideal image. The low-rank tensors are estimated by minimizing a penalty function obtained by adding the squared errors in a fit of the hyperspectral image to penalty terms for the sparse noise and the sizes of the two low-rank tensors. Lu et al. [22] approximate a third-order tensor using the sum of a low-rank tensor and a sparse tensor. Under suitable conditions, the low-rank tensor and the sparse tensor are recovered exactly.

2. T-SCALARS

The notations for t-scalars are summarized in Section 2.1. Basic definitions are given in Section 2.2. The Fourier transform of a t-scalar is defined in Section 2.3. Properties of t-scalars and the Fourier transform of a t-scalar are described in Section 2.4. A generalization of the t-scalars is described in Section 3.5.

2.1. Notations and Preliminaries. An array of order N over the complex numbers \mathbb{C} is an element of the set C defined by $C \equiv \mathbb{C}^{I_1 \times \dots \times I_N}$, where the I_n for $1 \leq n \leq N$ are strictly positive integers. Similarly, an array of order N over the real numbers is an element of the set R defined by $R \equiv \mathbb{R}^{I_1 \times \dots \times I_N}$. The sets R and C have the structure of commutative rings, in which the product is defined by circular convolution. The elements of C and R are referred to as t-scalars.

Elements of \mathbb{R} and \mathbb{C} are denoted by lower case letters and tensorial data are denoted by upper case letters. The t-scalars are identified using the subscript T , for example, X_T . Lower case subscripts such as i, j, α, β are indices or lists of indices.

All indices begin from 1 rather than 0. Given an array of any order N , namely $X \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$ ($N \geq 1$), X_{i_1, i_2, \dots, i_N} or $(X)_{i_1, i_2, \dots, i_N}$ denote its (i_1, i_2, \dots, i_N) -th entry in \mathbb{C} . The notation X_i , or $(X)_i$, is also used, where i is a multi-index defined by $i = (i_1, \dots, i_N)$. Let $I = (I_1, I_2, \dots, I_N)$ and let i be a multi-index. The notation $1 \leq i \leq I$ specifies the range of values of i such that $1 \leq i_n \leq I_n$ for $1 \leq n \leq N$. It is often convenient to extend the indexing beyond the range specified by I . Let j be a general multi-index. Then X_j is defined by $X_j = X_i$, where i is the multi-index such that each component i_n is in the range $1 \leq i_n \leq I_n$ and $i_n - j_n$ is divisible by I_n . A multi-index such as $i - j + 1$ has components $i_n - j_n + 1$ for $1 \leq n \leq N$. The sum $\sum_{i=1}^I(\cdot)$ is an abbreviation for $\sum_{i_1=1}^{I_1} \dots \sum_{i_N=1}^{I_N}(\cdot)$.

2.2. Definitions. The following definitions are for t-scalars in C . Similar definitions can be made for t-scalars in R .

Definition 2.1. T-scalar addition. Given t-scalars X_T and Y_T in C , the addition of X_T and Y_T denoted by $D_T \doteq X_T + Y_T$ is element-wise,

$$D_{T,i} = X_{T,i} + Y_{T,i}, \quad 1 \leq i \leq I. \quad (2.1)$$

Definition 2.2. T-scalar multiplication. Given t-scalars X_T and Y_T in C , their product, denoted by $D_T = X_T \circ Y_T$ is a t-scalar in C defined by the circular convolution

$$D_{T,i} = \sum_{j=1}^I X_{T,i-j+1} Y_{T,j}, \quad 1 \leq i \leq I. \quad (2.2)$$

Definitions 2.1 and 2.2 reduce to complex number addition and multiplication when $N = 1$ and $I_1 = 1$.

Definition 2.3. Zero t-scalar. The zero t-scalar Z_T is the array in C defined by

$$Z_{T,i} = 0, \quad 1 \leq i \leq I. \quad (2.3)$$

For all t-scalars X_T , $X_T + Z_T = X_T$ and $X_T \circ Z_T = Z_T$.

Definition 2.4. Identity t-scalar. The identity t-scalar E_T in C has the first entry equal to 1 and all other entries equal to 0, namely, $E_{T,i} = 1$ if $i = (1, \dots, 1)$ and $E_{T,i} = 0$ otherwise.

For all t-scalars $X_T \in C$, $X_T \circ E_T \equiv X_T$.

The set of t-scalars satisfies the axioms of a commutative ring with Z_T as an additive identity and E_T as a multiplicative identity. This ring of t-scalars is denoted by $(C, +, \circ)$. The ring $(C, +, \circ)$ is a generalization of the field $(\mathbb{C}, +, \cdot)$ of complex numbers. If the t-scalars are restricted to have real number elements, then the ring $(R, +, \circ)$ is obtained.

2.3. Fourier Transform of a T-scalar. Let ζ_n be a primitive I_n -th root of unity, for example,

$$\zeta_n = \exp(2\pi\sqrt{-1}/I_n), \quad 1 \leq n \leq N.$$

Let $\bar{\zeta}_n$ be the complex conjugate of ζ_n and let X_T be a t-scalar in the ring C . The Fourier transform $F(X_T)$ of X_T is defined by

$$F(X_T)_i = \sum_{j=1}^I X_{T,j} \cdot \zeta_1^{(i_1-1)(j_1-1)} \dots \zeta_N^{(i_N-1)(j_N-1)}$$

for all indices $1 \leq i \leq I$.

The inverse of the Fourier transform is defined by

$$X_{T,i} = \frac{\sum_{j=1}^I F(X_T)_j \cdot \bar{\zeta}_1^{(i_1-1)(j_1-1)} \dots \bar{\zeta}_N^{(i_N-1)(j_N-1)}}{I_1 \dots I_N}$$

for all indices $1 \leq i \leq I$.

Given t-scalars $X_T \in C$ and $Y_T \in C$ and their t-scalar product $D_T = X_T \circ Y_T$, it follows that

$$F(D_T) = F(X_T) * F(Y_T), \quad (2.4)$$

where $*$ denotes the Hadamard product in C . Equation (2.4) is an extension of the convolution theorem [2]. The equation can be equivalently rewritten as

$$F(D_T)_i = F(X_T)_i \cdot F(Y_T)_i, \quad 1 \leq i \leq I, \quad (2.5)$$

where \cdot is multiplication in \mathbb{C} .

An equivalent definition of the Fourier transform of a high-order array in the form of multi-mode tensor multiplication and a diagram of the multiplication of two t-scalars, computed in the Fourier domain, is given in a supplementary file.

It is not difficult to prove that C is a commutative ring, $(C, +, *)$, under the Hadamard product. The Fourier transform is a ring isomorphism from $(C, +, \circ)$ to $(C, +, *)$. The identity element of $(C, +, *)$ is $J_T = F(E_T)$. All the entries of J_T are equal to 1.

2.4. Properties of t-scalars. The invertible t-scalars are defined as follows.

Definition 2.5. Invertible t-scalar: Given a t-scalar X_T , if there exists a t-scalar Y_T satisfying $X_T \circ Y_T = E_T$, then X_T is said to be invertible. The t-scalar Y_T is the inverse of X_T and denoted by $Y_T \doteq X_T^{-1} \doteq E_T/X_T$.

The zero t-scalar Z_T is non-invertible. In addition, there is an infinite number of t-scalars that are non-invertible. For example, given a t-scalar $X_T \in C$, if the entries of X_T are all equal, then X_T is non-invertible. The existence of more than one non-invertible element shows that C is not a field.

Definition 2.6. Scalar multiplication of a t-scalar. Given a scalar $\lambda \in \mathbb{C}$ and a t-scalar $X_T \in C$, their product, denoted by $Y_T = \lambda \cdot X_T \equiv X_T \cdot \lambda$ is the t-scalar given by

$$Y_{T,i} = \lambda \cdot X_{T,i}, \quad 1 \leq i \leq I. \quad (2.6)$$

It can be shown that the set of t-scalars is a vector space over \mathbb{C} .

The following definition of the conjugate of a t-scalar generalizes the conjugate of a complex number.

Definition 2.7. Conjugate of a t-scalar. Given a t-scalar X_T in C , its conjugate, denoted by $\text{conj}(X_T)$, is the t-scalar in C such that

$$\text{conj}(X_T)_i = \overline{X_{T,2-i}}, \quad 1 \leq i \leq I, \quad (2.7)$$

where $\overline{X_{T,2-i}}$ is the complex conjugate of $X_{T,2-i}$ in \mathbb{C} .

The conjugate of a t-scalar reduces to the conjugate of a complex number when $N = 1$, $I_1 = 1$. The relationship of $\text{conj}(X_T)$ and X_T is much clearer if they are mapped to the Fourier domain – each entry of $F(\text{conj}(X_T))$ is the complex conjugate of the corresponding entry of $F(X_T)$, namely

$$F(\text{conj}(X_T))_i = \overline{F(X_T)_i}, \quad 1 \leq i \leq I. \quad (2.8)$$

It follows from equation (2.7) that $\text{conj}(\text{conj}(X_T)) = X_T$ for any $X_T \in C$.

Definition 2.8. Self-conjugate t-scalar: Given a t-scalar $X_T \in C$, if $X_T = \text{conj}(X_T)$, then X_T is said to be a self-conjugate t-scalar.

If X_T is self conjugate, then

$$\overline{F(X_T)_i} = F(\text{conj}(X_T))_i = F(X_T)_i \in \mathbb{C}, \quad 1 \leq i \leq I. \quad (2.9)$$

It follows from equation (2.9) that X_T is self-conjugate if and only if all the elements of $F(X_T)$ are real numbers.

The t-scalars Z_T and E_T are both self-conjugate. Furthermore, the self-conjugate t-scalars form a ring denoted by C^{sc} . This ring is a subring of C .

Given any t-scalar $X_T \in C$, let $\Re(X_T)$ and $\Im(X_T)$ be defined by

$$\Re(X_T) = 2^{-1}(X_T + \text{conj}(X_T)), \quad (2.10)$$

$$\Im(X_T) = (2\sqrt{-1})^{-1}(X_T - \text{conj}(X_T)). \quad (2.11)$$

It follows from equation (2.9) that $\Re(X_T)$ and $\Im(X_T)$ are self-conjugate. The t-scalars $X_T \in C$ and $\text{conj}(X_T) \in C$ can be expressed in the form

$$X_T = \Re(X_T) + \sqrt{-1}\Im(X_T), \quad (2.12)$$

$$\text{conj}(X_T) = \Re(X_T) - \sqrt{-1}\Im(X_T). \quad (2.13)$$

In an analogy with the real and imaginary parts of a complex number, $\Re(X_T)$ is called the real part of X_T and $\Im(X_T)$ is called the imaginary part of X_T .

Given two t-scalars X_T and Y_T , the equations (2.14) hold true and are backward compatible with the corresponding equations for complex numbers.

$$\begin{aligned} X_T + Y_T &\equiv \left(\Re(X_T) + \Re(Y_T) \right) + \sqrt{-1} \cdot \left(\Im(X_T) + \Im(Y_T) \right) \\ X_T \circ Y_T &\equiv \left(\Re(X_T) \circ \Re(Y_T) - \Im(X_T) \circ \Im(Y_T) \right) + \sqrt{-1} \left(\Im(X_T) \circ \Re(Y_T) + \Re(X_T) \circ \Im(Y_T) \right) \\ \text{conj}(X_T) \circ X_T &\equiv X_T \circ \text{conj}(X_T) \equiv \Re(X_T)^2 + \Im(X_T)^2 \end{aligned} \quad (2.14)$$

Definition 2.9. Nonnegative t-scalar: The t-scalar X_T is said to be nonnegative if there exists a self-conjugate t-scalar Y_T such that $X_T = Y_T \circ Y_T \doteq Y_T^2$.

If a t-scalar X_T is nonnegative, it is also self-conjugate, because the multiplication of any two self-conjugate t-scalars is also a self-conjugate t-scalar. Thus, both Z_T and E_T are nonnegative, since Z_T and E_T are self-conjugate t-scalars and satisfy $Z_T = Z_T^2$ and $E_T = E_T^2$. Furthermore, for all $X_T \in C$, the ring element $\Re(X_T)^2 + \Im(X_T)^2$ is nonnegative.

The set S^{nonneg} of nonnegative t-scalars is closed under the t-scalar addition and multiplication. Since a nonnegative t-scalar is also a self-conjugate t-scalar, $S^{nonneg} \subset C^{sc} \subset C$.

Theorem 2.10. For all t-scalars $X_T \in S^{nonneg}$, there exists a unique t-scalar $S_T \in S^{nonneg}$ satisfying $X_T = S_T \circ S_T \doteq S_T^2$. We call the nonnegative t-scalar S_T the arithmetic square root of the nonnegative t-scalar X_T and denote it by

$$S_T \doteq \sqrt{X_T} \doteq X_T^{1/2} \quad (2.15)$$

Proof. Let $X_T = Y_T \circ Y_T$, such that Y_T is self-conjugate. On applying the Fourier transform, it follows that

$$F(X_T)_i = F(Y_T)_i^2 \geq 0, \quad 1 \leq i \leq I.$$

Let S_T be defined such that

$$F(S_T)_i = (F(X_T)_i)^{1/2}, \quad 1 \leq i \leq I,$$

where the nonnegative square root is chosen for each value of i . The Fourier components $F(S_T)_i$ are real valued, thus S_T is self conjugate. The equation $X_T = S_T \circ S_T$ holds because the Fourier transform is injective. \square

Definition 2.11. A nonnegative t-scalar that is invertible under multiplication is called a positive t-scalar. The set of positive t-scalars is denoted by S^{pos} .

The following inclusions are strict, $S^{pos} \subset S^{nonneg} \subset C^{sc} \subset C$. The inverse and the arithmetic square root of a positive t-scalar are positive.

The absolute t-value $r(X_T)$ of X_T is defined by

$$r(X_T) = \sqrt{\Re(X_T)^2 + \Im(X_T)^2}. \quad (2.16)$$

The t-scalars $\Re(X_T)$ and $\Im(X_T)$ are both self-conjugate, therefore $\Re(X_T)^2$ and $\Im(X_T)^2$ are both nonnegative. The sum $\Re(X_T)^2 + \Im(X_T)^2$ is nonnegative and it has a nonnegative arithmetical square root, namely $r(X_T)$.

If $r(X_T)$ is invertible, then let $\phi(X_T)$ be defined by

$$\phi(X_T) \doteq r(X_T)^{-1} \circ X_T. \quad (2.17)$$

The ring element $\phi(X_T)$ is a generalized angle. The order 1 version of $\phi(X_T)$ is obtained by Gleich et al. in [11]. Equation (2.17) generalizes the polar form of a complex number. It can be shown that

$$\phi(X_T) \circ \text{conj}(\phi(X_T)) = E_T.$$

The absolute t-value $r(X_T)$ is used in Section 3 to define a generalization of the Frobenius norm for t-matrices.

3. MATRICES WITH T-SCALAR ELEMENTS

It is shown that t-matrices, i.e. matrices with elements in the rings C or R , are in many ways analogous to matrices with elements in \mathbb{C} or \mathbb{R} .

3.1. Indexing. The t-matrices are order-two arrays of t-scalars. Since the t-scalars are arrays of complex numbers, it is convenient to organize t-matrices as hierarchical arrays of complex numbers.

Let X_{TM} be a t-matrix with D_1 rows and D_2 columns. Then X_{TM} is an element of $C^{D_1 \times D_2}$. The (α, β) entry of X_{TM} is the element of C denoted by $X_{TM, \alpha, \beta}$ for $1 \leq \alpha \leq D_1$ and $1 \leq \beta \leq D_2$. Let i be a multi-index for elements of C . Then $X_{TM, i, \alpha, \beta}$ is the element of \mathbb{C} given as the i -th entry of the ring element $X_{TM, \alpha, \beta}$.

The t-matrix X_{TM} can be interpreted as an element in $\mathbb{C}^{I_1 \times \dots \times I_N \times D_1 \times D_2}$ or alternatively it can be interpreted as an element in $\mathbb{C}^{D_1 \times D_2 \times I_1 \times \dots \times I_N}$. The only thing needed to switch

from one data structure to the other is a permutation of indices. The data structure $\mathbb{C}^{I_1 \times \dots \times I_N \times D_1 \times D_2}$ is chosen unless otherwise indicated.

3.2. Properties of t-matrices. (1) **T-matrix addition:** Given any t-matrices $A_{TM} \in C^{D_1 \times D_2}$ and $B_{TM} \in C^{D_1 \times D_2}$, the addition, denoted by $C_{TM} \doteq A_{TM} + B_{TM} \in C^{D_1 \times D_2}$, is entry-wise, such that $C_{TM,\alpha,\beta} = A_{TM,\alpha,\beta} + B_{TM,\alpha,\beta}$, for $1 \leq \alpha \leq D_1$ and $1 \leq \beta \leq D_2$.

(2) **T-matrix multiplication:** Given any t-matrices $A_{TM} \in C^{D_1 \times Q}$ and $B_{TM} \in C^{Q \times D_2}$, their product, denoted by $C_{TM} \doteq A_{TM} \circ B_{TM}$, is the t-matrix in $C^{D_1 \times D_2}$ defined by

$$C_{TM,\alpha,\beta} = \sum_{\gamma=1}^Q A_{TM,\alpha,\gamma} \circ B_{TM,\gamma,\beta}$$

for all indices $1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2$.

An example of t-matrix multiplication $C_{TM} = A_{TM} \circ B_{TM} \in C^{2 \times 1} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 1}$ where $A_{TM} \in C^{2 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 2}$ and $B_{TM} \in C^{2 \times 1} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 1}$ is given in a supplementary file.

(3) **Identity t-matrix :** The identity t-matrix is the diagonal t-matrix, in which each diagonal entry is equal to the identity t-scalar E_T in Definition 2.4. The $D \times D$ identity t-matrix is denoted by $I_{TM}^{(D)} \doteq \underbrace{\text{diag}(E_T, \dots, E_T)}_D$.

Given any $X_{TM} \in C^{D_1 \times D_2}$, it follows that $I_{TM}^{(D_1)} \circ X_{TM} = X_{TM} \circ I_{TM}^{(D_2)} = X_{TM}$. The identity t-matrix $I_{TM}^{(D)}$ is also denoted by I_{TM} if the value of D can be inferred from context.

(4) **Scalar multiplication:** Given any $A_{TM} \in C^{D_1 \times D_2}$ and $\lambda \in \mathbb{C}$, their multiplication, denoted by $B_{TM} \doteq \lambda \cdot A_{TM}$, is the t-matrix in $C^{D_1 \times D_2}$ defined by

$$B_{TM,\alpha,\beta} = \lambda \cdot A_{TM,\alpha,\beta}, \quad 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2.$$

where the products with λ are computed as in Definition 2.6.

(5) **T-scalar multiplication:** Given any $A_{TM} \in C^{D_1 \times D_2}$ and $\lambda_T \in C$, their product, denoted by $B_{TM} \doteq \lambda_T \circ A_{TM}$, is the t-matrix in $C^{D_1 \times D_2}$ defined by

$$B_{TM,\alpha,\beta} = \lambda_T \circ A_{TM,\alpha,\beta}, \quad 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2.$$

(6) **Conjugate transpose of a t-matrix :** Given any t-matrix $X_{TM} \in C^{D_1 \times D_2}$, its conjugate transpose, denoted by X_{TM}^H is the t-matrix in $C^{D_2 \times D_1}$ given by

$$X_{TM,\beta,\alpha}^H = \text{conj}(X_{TM,\alpha,\beta}) \in C, \quad 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2.$$

A square matrix U_{TM} is said to orthogonal if U_{TM}^H is the inverse t-matrix of U_{TM} , i.e., $U_{TM}^H \circ U_{TM} = U_{TM} \circ U_{TM}^H = I_{TM}$. The Fourier transform F is extended to t-matrices element-wise, i.e. $F(X_{TM})$ is the $D_1 \times D_2$ t-matrix defined by

$$F(X_{TM})_{\alpha,\beta} = F(X_{TM,\alpha,\beta}) \quad . \tag{3.1}$$

for all indices $1 \leq \alpha \leq D_1$ and $1 \leq \beta \leq D_2$.

It is not difficult to prove that

$$F(X_{TM}^H)_{i,\beta,\alpha} = \overline{F(X_{TM})_{i,\alpha,\beta}} \in \mathbb{C}$$

for all indices $1 \leq i \leq I, 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2$.

(7) T-vector dot product and the Frobenius norm: Given any two t-vectors (i.e., two t-matrices, each having only one column) X_{TV} and Y_{TV} of the same length D , their dot product is the t-scalar defined by

$$\langle X_{TV}, Y_{TV} \rangle \doteq \sum_{\alpha=1}^D \text{conj}(X_{TV,\alpha}) \circ Y_{TV,\alpha} .$$

If $\langle X_{TV}, Y_{TV} \rangle = Z_T$, then X_{TV} and Y_{TV} are said to be orthogonal. The nonnegative t-scalar $\sqrt{\langle X_{TV}, X_{TV} \rangle}$ is called the generalized norm of X_{TV} and denoted by

$$\|X_{TV}\|_F \doteq \sqrt{\langle X_{TV}, X_{TV} \rangle} \equiv \left(\sum_{\alpha=1}^D r(X_{TV,\alpha})^2 \right)^{1/2} \quad (3.2)$$

where $r(\cdot)$ is the absolute t-value as defined by equation (2.16). The generalized Frobenius norm of a $D_1 \times D_2$ t-matrix W_{TM} is defined by

$$\|W_{TM}\|_F \doteq \left(\sum_{\alpha=1}^{D_1} \sum_{\beta=1}^{D_2} r(W_{TM,\alpha,\beta})^2 \right)^{1/2} . \quad (3.3)$$

In order to have a mechanism to connect t-matrices with matrices with elements in \mathbb{C} or \mathbb{R} , the slices of a t-matrix are defined as follows.

(8) Slice of a t-matrix : Any t-matrix $X_{TM} \in C^{D_1 \times D_2}$, organized as an array in $\mathbb{C}^{I_1 \times \dots \times I_N \times D_1 \times D_2}$, can be sliced into $\prod_{n=1}^N I_n$ matrices in $\mathbb{C}^{D_1 \times D_2}$, indexed by the multi-index i . Let $X_{TM}(i) \in \mathbb{C}^{D_1 \times D_2}$ be the i -th slice. The entries of $X_{TM}(i)$ are complex numbers in \mathbb{C} given by

$$(X_{TM}(i))_{\alpha,\beta} = X_{TM,i,\alpha,\beta} \in \mathbb{C}$$

for all indices $1 \leq i \leq I, 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2$.

The t-vectors with a given dimension form an algebraic structure called a module over the ring C [16]. Modules are generalizations of vector spaces [17]. The t-vector whose entries are all equal to Z_T is denoted by Z_{TV} , and called the zero t-vector. The next step is to define what is meant by a set of linearly independent t-vectors and what is meant by a full column rank t-matrix.

(9) Linear independence in t-vector module: The t-vectors in a subset $\{X_{TV,1}, X_{TV,2}, \dots, X_{TV,K}\}$ of a t-vector module are said to be linearly independent if the equation $\sum_{k=1}^K \lambda_{T,k} \circ X_{TV,k} = Z_{TV}$ holds true if and only if $\lambda_{T,k} = Z_T, 1 \leq k \leq K$.

If the t-vectors $X_{TV,i}, 1 \leq i \leq K$, are linearly independent then they are said to have a rank of K . If the t-vectors $Y_{TV,i}$ for $1 \leq i \leq K'$ are linearly independent and span the same sub-module as the $X_{TM,i}$ then $K = K'$. For further information see [16].

(10) **Full column rank t-matrix:** A t-matrix is said to be of full column rank if all its column t-vectors are linearly independent.

3.3. T-matrix Analysis via the Fourier Transform. The Fourier transform of the t-matrix $X_{TM} \in C^{D_1 \times D_2}$ is the t-matrix in $C^{D_1 \times D_2}$ given by equation (3.1).

Many t-matrix computations can be carried out efficiently using the Fourier transform. For example, any multiplication $C_{TM} = X_{TM} \circ Y_{TM} \in \mathbb{C}^{I_1 \times \dots \times I_N \times D_1 \times D_2}$, where $X_{TM} \in \mathbb{C}^{I_1 \times \dots \times I_N \times D_1 \times Q}$, $Y_{TM} \in \mathbb{C}^{I_1 \times \dots \times I_N \times Q \times D_2}$, can be decomposed to $\prod_{n=1}^N I_n$ matrix multiplications over the complex numbers, namely

$$F(C_{TM})_{i,\alpha,\beta} = \sum_{\gamma=1}^Q F(X_{TM})_{i,\alpha,\gamma} \cdot F(Y_{TM})_{i,\gamma,\beta} \quad (3.4)$$

for all indices $1 \leq i \leq I, 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2$.

The conjugate transpose $X_{TM}^H \in \mathbb{C}^{I_1 \times \dots \times I_N \times D_2 \times D_1}$ of any t-matrix $X_{TM} \in \mathbb{C}^{I_1 \times \dots \times I_N \times D_1 \times D_2}$ can be decomposed to $\prod_{n=1}^N I_n$ canonical conjugate transposes of matrices,

$$F(X_{TM}^H)_{i,\beta,\alpha} = \overline{F(X_{TM})_{i,\alpha,\beta}} \quad (3.5)$$

for all indices $1 \leq i \leq I, 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2$. Each slice of $F(I_{TM}^{(D)})$ is the canonical identity matrix with elements in \mathbb{C} .

The Fourier transform decomposes a t-matrix computation such as multiplication to $\prod_{n=1}^N I_n$ independent complex matrix computations in the Fourier domain. The i -th ($1 \leq i \leq I$) computation involves only the i -th slices of the associated t-matrices. This fact underlies an approach for speeding-up t-matrix algorithms using parallel computations. This independence of the data in the Fourier domain makes it possible to implement parallel computing using the so-called vectorization programming (also known as array programming), which is supported by many programming languages including MATLAB, R, NumPy, Julia, and Fortran.

3.4. Pooling. Sometimes, it is necessary to have a pooling mechanism to transform t-scalars to scalars in \mathbb{R} or \mathbb{C} . Given any t-scalar $X_T \in C$, its pooling result $P(X_T) \in \mathbb{C}$ is defined by

$$P(X_T) = (I_1 \cdots I_N)^{-1} \sum_{i=1}^I X_{T,i}. \quad (3.6)$$

The pooling operation for t-matrices transforms each t-scalar entry to a scalar. More formally, given any t-matrix $Y_{TM} \in C^{D_1 \times D_2}$, its pooling result $P(Y_{TM})$ is by definition the matrix in $\mathbb{C}^{D_1 \times D_2}$ given by

$$P(Y_{TM})_{\alpha,\beta} = P(Y_{TM,\alpha,\beta}), \quad 1 \leq \alpha \leq D_1, 1 \leq \beta \leq D_2. \quad (3.7)$$

The pooling of t-vectors is a special case of equation (3.7).

3.5. Generalized tensors. Generalized tensors, called g-tensors, generalize t-matrices and canonical tensors. The generalized tensors defined in this section are used to construct the higher order TSVD in Section 4.2. A g-tensor, denoted by $X_{GT} \in C^{D_1 \times D_2 \times \dots \times D_M}$, is a generalized tensor with t-scalar entries (i.e., an order- M array of t-scalars). Its t-scalar entries are indexed by $(X_{GT})_{\alpha_1, \dots, \alpha_M}$. Then, a generalized mode- k multiplication of X_{GT} , denoted by $M_{GT} \doteq X_{GT} \circ_k Y_{TM}$ where $Y_{TM} \in C^{J \times D_k}$ and $1 \leq k \leq M$, is a g-tensor in $C^{D_1 \times \dots \times D_{k-1} \times J \times D_{k+1} \times \dots \times D_M}$ defined as follows.

$$\begin{aligned} & (M_{GT})_{\alpha_1, \dots, \alpha_{k-1}, \beta, \alpha_{k+1}, \dots, \alpha_M} \\ &= \sum_{\alpha_k=1}^{D_k} (X_{GT})_{\alpha_1, \dots, \alpha_{k-1}, \alpha_k, \alpha_{k+1}, \dots, \alpha_M} \circ (Y_{TM})_{\beta, \alpha_k} \end{aligned} \quad (3.8)$$

The **generalized mode- k flattening** of a g-tensor $X_{GT} \in C^{D_1 \times D_2 \times \dots \times D_M}$ is an (K_1, K_2) -reshaping where $K_1 = \{k\}$ and $K_2 = \{1, \dots, M\} \setminus \{k\}$. The result is a t-matrix in $C^{D_k \times D_k^{-1} \cdot \prod_{m=1}^M D_m}$. Each column of the matrix is obtained by holding the indices in K_2 fixed and varying the index in K_1 .

The generalized mode- k multiplication defined in equation (3.8) can also be expressed in terms of unfolded g-tensors:

$$M_{GT} = X_{GT} \circ_k Y_{TM} \Leftrightarrow M_{GT(k)} = Y_{TM} \circ X_{GT(k)}$$

where $M_{GT(k)} \in C^{J \times (D_1 \dots D_{k-1} D_{k+1} \dots D_M)}$ and $X_{GT(k)} \in C^{D_k \times (D_1 \dots D_{k-1} D_{k+1} \dots D_M)}$ are respectively the generalized mode- k flattening of the g-tensors M_{GT} and X_{GT} .

An example of a generalized tensor (g-tensor) $X_{GT} \in C^{2 \times 3 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3 \times 2}$, its mode- k flattening, and its mode-2 multiplication with a t-matrix $Y_{TM} \in C^{2 \times 3} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3}$ are given in a supplementary file.

4. TENSOR SINGULAR VALUE DECOMPOSITION

The singular value decomposition (SVD) is a well known factorization of real or complex matrices [12]. It generalizes the eigen-decomposition of positive semi-definite normal matrices to non-square and non-normal matrices. The SVD has a wide range of applications in data analytics, including computing the pseudo-inverse of a matrix, solving linear least squares problems, low-rank approximation and linear and multi-linear component analysis. A tensor version TSVD of the SVD is described in Section 4.1, and then applied in Section 4.2 to obtain a tensor version, THOSVD, of the Higher Order SVD (HOSVD). Further information about the TSVD can be found in [18] and [41].

4.1. TSVD: Tensorial SVD. *Algorithm.* A tensor version, TSVD, of the singular value decomposition is described in this section and then applied in Section 4.2 to obtain a tensor version of the High Order SVD (HOSVD). See [41] and [18].

Given a t-matrix $X_{TM} \in C^{D_1 \times D_2}$, let $Q \doteq \min(D_1, D_2)$. The TSVD of X_{TM} yields the following three t-matrices $U_{TM} \in C^{D_1 \times Q}$, $S_{TM} \in C^{Q \times Q}$ and $V_{TM} \in C^{D_2 \times Q}$, such that

$$X_{TM} = U_{TM} \circ S_{TM} \circ V_{TM}^H \quad (4.1)$$

where $U_{TM}^H \circ U_{TM} = V_{TM}^H \circ V_{TM} = I_{TM}^{(Q)}$, $S_{TM} = \text{diag}(\lambda_{T,1}, \dots, \lambda_{T,Q})$ and $\lambda_{T,1}, \dots, \lambda_{T,Q} \in C$ are nonnegative, and satisfy $F(\lambda_{T,1})_i \geq \dots \geq F(\lambda_{T,Q})_i \geq 0$, $1 \leq i \leq I$. The t-matrices U_{TM} and V_{TM} are generalizations of the orthogonal matrices in the SVD of a matrix with elements in \mathbb{R} or \mathbb{C} .

Although it is possible to compute U_{TM} , S_{TM} and V_{TM} in the spatial domain, it is preferable to organize the TSVD algorithm in the Fourier domain, because of the observation in Section 2.3 that the Fourier transform converts the convolution product to the Hadamard product. The TSVD of X_{TM} can be decomposed into $\prod_{n=1}^N I_n$ SVDs of complex number matrices given by the slices of the Fourier transform $F(X_{TM})$. The t-matrices U_{TM} , S_{TM} and V_{TM} in equation (4.1) are obtained in Algorithm 1.

Algorithm 1 $(U_{TM}, S_{TM}, V_{TM}) = \text{tsvd}(X_{TM})$

Inputs: A t-matrix $X_{TM} \in C^{D_1 \times D_2}$ as in equation (4.1) and the t-scalar dimensions I .

Outputs: The t-matrices U_{TM} , S_{TM} and V_{TM} as in equation (4.1)

- 1: Compute the Fourier transformation $\tilde{X}_{TM} \leftarrow F(X_{TM})$
- 2: **for all** $1 \leq i \leq I$ **do**
- 3: Compute the canonical SVD of $\tilde{X}_{TM}(i) \in C^{D_1 \times D_2}$ such that

$$\tilde{X}_{TM}(i) = U_{mat} \cdot S_{mat} \cdot V_{mat}^H$$

where $U_{mat} \in C^{D_1 \times Q}$, $S_{mat} \in C^{Q \times Q}$, $V_{mat} \in C^{D_2 \times Q}$, $Q \doteq \min(D_1, D_2)$ and V_{mat}^H denotes the conjugate transpose of the complex matrix V_{mat} .

- 4: Assign the i -th slices $\tilde{U}_{TM} \in C^{D_1 \times Q}$, $\tilde{S}_{TM} \in C^{Q \times Q}$ and $\tilde{V}_{TM} \in C^{D_2 \times Q}$ in the Fourier domain

$$\tilde{U}_{TM}(i) \leftarrow U_{mat}, \quad \tilde{S}_{TM}(i) \leftarrow S_{mat}, \quad \tilde{V}_{TM}(i) \leftarrow V_{mat}.$$

- 5: **end for**

- 6: Compute the inverse transforms to obtain U_{TM} , S_{TM} and V_{TM}

$$U_{TM} \leftarrow F^{-1}(\tilde{U}_{TM}), \quad S_{TM} \leftarrow F^{-1}(\tilde{S}_{TM}), \quad V_{TM} \leftarrow F^{-1}(\tilde{V}_{TM}).$$

- 7: **return** U_{TM} , S_{TM} , V_{TM} .
-

If X_{TM} is defined over \mathbb{R} , then U_{TM} , S_{TM} and V_{TM} can be chosen such that they are defined over \mathbb{R} . It is sufficient to choose the slices $\tilde{U}_{TM}(i)$, $\tilde{S}_{TM}(i)$ and $\tilde{V}_{TM}(i)$ such that $\tilde{U}_{TM}(i) = \overline{\tilde{U}_{TM}(2-i)}$, $\tilde{S}_{TM}(i) = \overline{\tilde{S}_{TM}(2-i)}$ and $\tilde{V}_{TM}(i) = \overline{\tilde{V}_{TM}(2-i)}$. When the t-scalar dimensions are given by $N = 1$, $I_1 = 1$, TSVD reduces to the canonical SVD of a matrix in $C^{D_1 \times D_2}$. The properties of the SVD can be used to show that the t-matrix S_{TM} in Algorithm 1 is unique. The t-matrices U_{TM} and V_{TM} are not unique.

TSVD Approximation. TSVD can be used to approximate data. Given a t-matrix $X_{TM} \in C^{D_1 \times D_2}$, let $Q \doteq \min(D_1, D_2)$ and let the TSVD of X_{TM} be computed as in equation (4.1). The low-rank approximation \hat{X}_{TM} of X_{TM} with rank of r ($1 \leq r \leq Q$) is defined by

$$\hat{X}_{TM} = U_{TM} \circ \hat{S}_{TM} \circ V_{TM}^H \quad (4.2)$$

where $\hat{S}_{TM} = \text{diag}(\lambda_{T,1}, \dots, \lambda_{T,r}, \underbrace{Z_T, \dots, Z_T}_{Q-r})$ and $\lambda_{T,1}, \dots, \lambda_{T,r} \neq Z_T$.

When the t-scalar dimensions are given by $N = 1, I_1 = 1$, equation (4.2) reduces to the SVD low-rank approximation to a matrix in $\mathbb{C}^{D_1 \times D_2}$.

Furthermore, we contend that the approximation \hat{X}_{TM} computed as in equation (4.2) is the solution of the following optimization problem

$$\begin{aligned} X_{TM}^{approx} = \operatorname{argmin}_{Y_{TM} \in \mathbb{C}^{D_1 \times D_2}} & \|X_{TM} - Y_{TM}\|_F \\ & \text{subject to } \operatorname{rank}(Y_{TM}) \leq r \cdot E_T \end{aligned} \quad (4.3)$$

where $\|\cdot\|_F$ denotes the generalized Frobenius norm of a t-matrix, which is a nonnegative t-scalar, as defined in equation (3.3). The result X_{TM}^{approx} generalizes the Eckart-Young-Mirsky theorem [7].

To have an optimization problem in the form of (4.3), the notation $\operatorname{rank}(\cdot)$, i.e., the rank of a t-matrix, and $\min(\cdot)$, i.e., the minimization of a nonnegative t-scalar variable belonging to a subset of S^{nonneg} , and the ordering relationship \leq between two nonnegative t-scalars need to be defined.

These definitions generalize their canonical counterparts. The definitions and the generalized Eckart-Young-Mirsky theorem are discussed in an appendix.

4.2. THOSVD: Tensor Higher Order SVD. In multilinear algebra, the higher order singular value decomposition (HOSVD), also known as the orthogonal Tucker decomposition of a tensor, is a generalization of the SVD. It is commonly used to extract directional information from multi-way arrays [30, 6]. The applications of HOSVD include data analytics [32, 29], machine learning [33, 34, 23], DNA and RNA analysis [26, 25] and texture mapping in computer graphics [35].

On using the t-scalar algebra, the HOSVD can be generalized further to obtain a tensorial HOSVD, called THOSVD. The THOSVD is obtained by replacing the complex number elements of each multi-way array by t-scalar elements. Based on the definitions of g-tensors in Section 3.5, the THOSVD of $X_{GT} \in \mathbb{C}^{D_1 \times D_2 \times \dots \times D_M}$ is given by the following generalized mode- k multiplications.

$$X_{GT} = S_{GT} \circ_1 U_{TM,1} \circ_2 U_{TM,2} \cdots \circ_M U_{TM,M} \quad (4.4)$$

where $S_{GT} \in \mathbb{C}^{Q_1 \times Q_2 \times \dots \times Q_M}$ is called the core g-tensor, $U_{TM,k} \in \mathbb{C}^{D_k \times Q_k}$ is the mode- k factor t-matrix and $Q_k \doteq \min(D_k, D_k^{-1} \prod_{m=1}^M D_m)$ for $1 \leq k \leq M$.

Given a g-tensor $X_{GT} \in \mathbb{C}^{D_1 \times D_2 \times \dots \times D_M}$, the THOSVD of X_{GT} , as in equation (4.4), is obtained in Algorithm 2, using a strategy analogous to that of Tucker [30] and De Lathauwer et al. [6] for computing the HOSVD of a tensor with elements in \mathbb{R} or \mathbb{C} .

Algorithm 2 THOSVD

Input: $X_{GT} \in C^{D_1 \times D_2 \times \dots \times D_M}$.
Outputs: $U_{TM,1}, U_{TM,2}, \dots, U_{TM,M}$ and S_{GT} as in equation (4.4)
1: **for all** $1 \leq k \leq M$ **do**
2: Construct the generalized mode- k flattening $X_{GT(k)} \in C^{D_k \times D_k^{-1} \prod_{m=1}^M D_m}$.
3: $(U_{TM,k}, \sim, \sim) \leftarrow \text{tsvd}(X_{GT(k)})$
4: **end for**
5: $S_{GT} \leftarrow X_{GT} \circ_1 U_{TM,1}^{\mathcal{H}} \circ_2 U_{TM,2}^{\mathcal{H}} \dots \circ_M U_{TM,M}^{\mathcal{H}}$
6: **return** $U_{TM,1}, U_{TM,2}, \dots, U_{TM,M}$ and S_{GT} .

Note that THOSVD generalizes the HOSVD for canonical tensors, TSVD for t-matrices, and SVD for canonical matrices. Many SVD and HOSVD based algorithms can be generalized by TSVD and THOSVD, respectively.

5. TENSOR BASED ALGORITHMS

Three tensor based algorithms are proposed. They are Tensorial Principal Component Analysis (TPCA), Tensorial Two-Dimensional Principal Component Analysis (T2DPCA) and Tensorial Grassmannian Component Analysis (TGCA). TPCA and T2DPCA are generalizations of the well-known algorithms PCA and 2DPCA [37]. TGCA is a generalization of the recent GCA algorithm [14, 13]. It is possible to generalize many other linear or multi-linear algorithms using similar methods.

5.1. TPCA: Tensorial Principal Component Analysis. Principal Component Analysis (PCA) is a well known algorithm for extracting the prominent components of observed vectors. PCA is generalized to TPCA in a straightforward manner. Let $X_{TV,1}, \dots, X_{TV,K} \in C^D$ be K given t-vectors. Then, the covariance-like t-matrix $G_{TM} \in C^{D \times D}$ is defined by

$$G_{TM} = \frac{1}{K-1} \sum_{k=1}^K (X_{TV,k} - \bar{X}_{TV}) \circ (X_{TV,k} - \bar{X}_{TV})^{\mathcal{H}} \quad (5.1)$$

where $\bar{X}_{TV} = (1/K) \sum_{k=1}^K X_{TV,k}$. It is not difficult to verify that G_{TM} is Hermitian, namely $G_{TM}^{\mathcal{H}} = G_{TM}$.

The t-matrix $U_{TM} \in C^{D \times D}$ is computed from the TSVD of G_{TM} as in Algorithm 1. Then, given any t-vector $Y_{TV} \in C^D$, its feature t-vector $Y_{TV}^{feat} \in C^D$ is defined by

$$Y_{TV}^{feat} = U_{TM}^{\mathcal{H}} \circ (Y_{TV} - \bar{X}_{TV}) . \quad (5.2)$$

To reduce Y_{TV}^{feat} from a t-vector in C^D to a t-vector in C^d ($D > d$), simply discard the last $(D-d)$ t-scalar entries of Y_{TV}^{feat} .

In algebraic terminology, the column t-vectors of U_{TM} span a linear sub-module of t-vectors, which is a generalization of a vector subspace [3]. In this sense, each t-scalar entry of Y_{TV}^{feat} is a generalized coordinate of the projection of the t-vector $(Y_{TV} - \bar{X}_{TV})$

onto the sub-module. The low-rank reconstruction $Y_{TV}^{rec} \in C^D$ with the parameter d is given by

$$Y_{TV}^{rec} = (U_{TM})_{:,1:d} \circ (Y_{TV}^{feat})_{1:d} + \bar{X}_{TV} \quad (5.3)$$

where $(U_{TM})_{:,1:d} \in C^{D \times d}$ denotes the t-matrix containing the first d t-vector columns of $U_{TM} \in C^{D \times D}$ and $(Y_{TV}^{feat})_{1:d} \in C^d$ denotes the t-vector containing the first d t-scalar entries of $Y_{TV}^{feat} \in C^D$.

Note that PCA is a special case of TPCA. When the t-scalar dimensions are given by $N = 1$, $I_1 = 1$, TPCA reduces to PCA.

5.2. T2DPCA: Tensorial Two-dimensional Principal Component Analysis.

The algorithm 2DPCA is an extension of PCA proposed by Yang et al. [37] for analysing the principal components of matrices. Although 2DPA is written in a non-centred row-vector oriented form in the original paper [37], it is rewritten here in a centred column-vector oriented form, which is consistent with the formulation of PCA. The centred column-vector oriented form of 2DPCA is chosen for discussing its generalization to T2DPCA (Tensorial 2DPCA).

Similar to TPCA, T2DPCA also finds sub-modules, but they are obtained by analysing t-matrices. Let $X_{TM,1}, \dots, X_{TM,K} \in C^{D_1 \times D_2}$ be the K observed t-matrices. Then, the Hermitian covariance-like t-matrix $G_{TM} \in C^{D_1 \times D_1}$ is given by

$$G_{TM} = \frac{1}{K-1} \sum_{k=1}^K (X_{TM,k} - \bar{X}_{TM}) \circ (X_{TM,k} - \bar{X}_{TM})^{\mathcal{H}} \quad (5.4)$$

where $\bar{X}_{TM} = (1/K) \sum_{k=1}^K X_{TM,k}$.

Then, the t-matrix $U_{TM} \in C^{D_1 \times D_1}$ is computed from the TSVD of G_{TM} as in Algorithm 1. Given any t-matrix $Y_{TM} \in C^{D_1 \times D_2}$, its feature t-matrix $Y_{TM}^{feat} \in C^{D_1 \times D_2}$ is a centred t-matrix projection (i.e., a collection of centred column t-vector projections) on the module spanned by U_{TM} , namely

$$Y_{TM}^{feat} = U_{TM}^{\mathcal{H}} \circ (Y_{TM} - \bar{X}_{TM}). \quad (5.5)$$

To reduce Y_{TM}^{feat} from a t-matrix in $C^{D_1 \times D_2}$ to a t-matrix in $C^{d \times D_2}$ ($D_1 > d$), simply discard the last $(D_1 - d)$ row t-vectors of Y_{TM}^{feat} .

The T2DPCA reconstruction with the parameter d is given by $Y_{TM}^{rec} \in C^{D_1 \times D_2}$ as follows.

$$Y_{TM}^{rec} = U_{TM, :, 1:d} \circ (Y_{TM}^{feat})_{1:d, :} + \bar{X}_{TM} \quad (5.6)$$

where $(U_{TM})_{:, 1:d} \in C^{D_1 \times d}$ denotes the t-matrix containing the first d column t-vectors of U_{TM} and $(Y_{TM}^{feat})_{1:d, :} \in C^{d \times D_2}$ denotes the t-matrix containing the first d row t-vectors of Y_{TM}^{feat} .

Algorithm 3 Generalized Gram-Schmidt orthogonalization**Input:** $Y_{TM} \in C^{D \times d}$.**Outputs:** $\dot{Y}_{TM} \in C^{D \times d}$ satisfying $\dot{Y}_{TM}^{\mathcal{H}} \circ \dot{Y}_{TM} = I_{TM}^{(d)}$.

- 1: $(\dot{Y}_{TM})_{:,1} \leftarrow \|(Y_{TM})_{:,1}\|_F^{-1} \circ (Y_{TM})_{:,1}$
- 2: **for all** $2 \leq k \leq d$ **do**
- 3: $(\dot{Y}_{TM})_{:,k} \leftarrow (Y_{TM})_{:,k}$
- 4: **for all** $1 \leq j \leq k-1$ **do**
- 5: $(\dot{Y}_{TM})_{:,k} \leftarrow (\dot{Y}_{TM})_{:,k} - \langle (\dot{Y}_{TM})_{:,k}, (\dot{Y}_{TM})_{:,j} \rangle \circ (\dot{Y}_{TM})_{:,j}$
- 6: **end for**
- 7: $(\dot{Y}_{TM})_{:,k} \leftarrow \|(\dot{Y}_{TM})_{:,k}\|_F^{-1} \circ (\dot{Y}_{TM})_{:,k}$
- 8: **end for**
- 9: **return** \dot{Y}_{TM}

When the t-scalar dimensions are given by $N = 1$, $I_1 = 1$, T2DPCA reduces to 2DPCA. In addition, TPCA is a special case of T2DPCA. When $D_2 = 1$, T2DPCA reduces to TPCA. Furthermore, when $N = 1$, $I_1 = 1$ and $D_2 = 1$, T2DPCA reduces to PCA.

5.3. TGCA: Tensorial Grassmannian Component Analysis. A t-matrix algorithm which generalizes the recent algorithm for Grassmannian Component Analysis (GCA) is proposed. An example of GCA can be found in [13], where it forms part of an algorithm for sparse coding on Grassmann manifolds. In this section GCA is extended to its generalized version called TGCA (Tensorial GCA).

In TGCA, each measurement is a set of t-vectors organized into a “thin” t-matrix, with the number of rows larger than the number of columns. Let $X_{TM,1}, \dots, X_{TM,K} \in C^{D \times d}$ ($D > d$) be the observed t-matrices. Then, the t-vector columns of each t-matrix are first orthogonalized. Using the t-scalar algebra, it is straightforward to generalize the classical Gram-Schmidt orthogonalization process for t-vectors. The TSVD can also be used to orthogonalise a set of t-vectors. In GCA and TGCA, the choice of orthogonalization algorithm doesn’t matter as long as the algorithm is consistent for all sets of vectors and t-vectors.

Given a t-matrix $Y_{TM} \in C^{D \times d}$, let $\dot{Y}_{TM} \in C^{D \times d}$ be the corresponding unitary orthogonalized t-matrix (namely, $\dot{Y}_{TM}^{\mathcal{H}} \circ \dot{Y}_{TM} = I_{TM}^{(d)}$) computed from Y_{TM} . Let $(Y_{TM})_{:,k}$ be the k -th column t-vector of Y_{TM} and let $(\dot{Y}_{TM})_{:,k}$ be the k -th column t-vector of \dot{Y}_{TM} for $1 \leq k \leq d$. The generalized Gram-Schmidt orthogonalization is given by Algorithm 3.

Let $\dot{X}_{TM,k} \in C^{D \times d}$ be the unitary orthogonalized t-matrices computed from $X_{TM,k}$ for $1 \leq k \leq K$. Then, for $1 \leq k, k' \leq K$, the (k, k') t-scalar entry of the symmetric t-matrix $G_{TM} \in C^{K \times K}$ is nonnegative and given by

$$(G_{TM})_{k,k'} = \|\dot{X}_{TM,k}^{\mathcal{H}} \circ \dot{X}_{TM,k'}\|_F^2, \quad 1 \leq k, k' \leq K \quad (5.7)$$

where $\|\cdot\|_F$ is the generalized Frobenius norm of a t-matrix, as defined by equation (3.3).

Given any query t-matrix sample $Y_{TM} \in C^{D \times d}$, let $\dot{Y}_{TM} \in C^{D \times d}$ be the corresponding unitary orthogonalized t-matrix computed from Y_{TM} . Then, the k -th t-scalar entry of $K_{TV} \in C^K$ is computed as follows.

$$(K_{TV})_k = \|\dot{Y}_{TM}^{\mathcal{H}} \circ \dot{X}_{TM,k}\|_F^2, \quad 1 \leq k \leq K. \quad (5.8)$$

Since G_{TM} , computed as in equation (5.7), is symmetric, the TSVD of G_{TM} has the following form

$$G_{TM} = U_{TM} \circ S_{TM} \circ U_{TM}^{\mathcal{H}}. \quad (5.9)$$

Furthermore, if it is assumed that the diagonal entries $S_{TM} \doteq \text{diag}(\lambda_{T,1}, \dots, \lambda_{T,K})$ are all strictly positive, then the multiplicative inverse of $\lambda_{T,k}$ exists for $1 \leq k \leq K$. The t-matrix $S_{TM}^{1/2} \doteq \text{diag}(\sqrt{\lambda_{T,1}}, \dots, \sqrt{\lambda_{T,K}})$ is called the t-matrix square root of S_{TM} and the t-matrix $S_{TM}^{-1/2} \doteq \text{diag}(\frac{E_T}{\sqrt{\lambda_{T,1}}}, \dots, \frac{E_T}{\sqrt{\lambda_{T,K}}})$ is called the inverse t-matrix of $S_{TM}^{1/2}$.

Thus, the features of the t-matrix sample $Y_{TM} \in C^{D \times d}$ are given by the t-vector $Y_{TV}^{feat} \in C^K$ as

$$Y_{TV}^{feat} = S_{TM}^{-1/2} \circ U_{TM}^{\mathcal{H}} \circ K_{TV} \quad (5.10)$$

and the features of the k -th measurement $X_{TM,k}$ are given by the t-vector $X_{TV,k}^{feat}$ as follows.

$$X_{TV,k}^{feat} = S_{TM}^{-1/2} \circ U_{TM}^{\mathcal{H}} \circ (G_{TM})_{:,k}, \quad 1 \leq k \leq K \quad (5.11)$$

where $(G_{TM})_{:,k}$ denotes k -th t-vector column of G_{TM} . It is not difficult to verify that $S_{TM}^{-1/2} \circ U_{TM}^{\mathcal{H}} \circ G_{TM} \equiv S_{TM}^{1/2} \circ U_{TM}^{\mathcal{H}}$. This yields the following compact form for $X_{TV,k}^{feat}$.

$$X_{TV,k}^{feat} = (S_{TM}^{1/2} \circ U_{TM}^{\mathcal{H}})_{:,k}, \quad 1 \leq k \leq K \quad (5.12)$$

where $(S_{TM}^{1/2} \circ U_{TM}^{\mathcal{H}})_{:,k}$ denotes the k -th t-vector column of the t-matrix $(S_{TM}^{1/2} \circ U_{TM}^{\mathcal{H}})$. The equation (5.12) is more efficient in computations than equation (5.11).

The dimension of a TGCA feature t-vector is reduced from K to K' ($K > K'$) by discarding the last $(K - K')$ t-scalar entries. It is noted that GCA is a special case of TGCA when the dimensions of the t-scalars are given by $N = 1, I_1 = 1$.

6. EXPERIMENTS

The results obtained from TSVD, THOSVD, TPCA, T2DPCA, TGCA and their precursors are compared in applications to low-rank approximation in Section 6.1, reconstruction in Section 6.2 and supervised classification of images in Section 6.3.

In these experiments “vertical” and “horizontal” comparisons between generalised algorithms and the corresponding canonical algorithms are made.

In a “vertical” experiment, tensorized data is obtained from the canonical data in 3×3 neighborhoods. The associated t-scalar is a 3×3 array. To make the vertical comparison

fair, we put the central slices of a generalized result into the original canonical form and then compare it with the result of the associated canonical algorithm.

In a “horizontal” comparison, a generalized order- N array of order-two t-scalars is equivalent to a canonical order- $(N + 2)$ array of scalars. Therefore, a generalized algorithm based on order- N arrays of order-two t-scalars is compared with a canonical algorithm based on order- $(N + 2)$ arrays of scalars.

6.1. Low-rank Approximation. TSVD approximation is computed as in equation (4.2). THOSVD approximation generalizes low-rank approximation by TSVD and low-rank approximation by HOSVD. To simplify the calculations, the approximation is obtained for a g-tensor X_{GT} in $C^{D_1 \times D_2 \times D_3}$. Let $Q_k \doteq \min(D_k, D_k^{-1} D_1 D_2 D_3)$ for $k = 1, 2, 3$. The THOSVD of X_{GT} yields

$$X_{GT} = S_{GT} \circ_1 U_{TM,1} \circ_2 U_{TM,2} \circ_3 U_{TM,3} \tag{6.1}$$

where $U_{TM,k} \in C^{D_k \times Q_k}$ for $k = 1, 2, 3$ and $S_{GT} \in C^{Q_1 \times Q_2 \times Q_3}$.

The low-rank approximation $\hat{X}_{GT} \in C^{D_1 \times D_2 \times D_3}$ to X_{GT} and with multilinear rank tuple (r_1, r_2, r_3) , ($1 \leq r_k \leq Q_k$ for all $k = 1, 2, 3$), is computed as in equation (6.2), where $(U_{TM,k})_{:,1:r_k}$ denotes the t-matrix containing the first r_k t-vector columns of $U_{TM,k}$ for $k = 1, 2, 3$ and $(S_{GT})_{1:r_1,1:r_2,1:r_3} \in C^{r_1 \times r_2 \times r_3}$ denotes the g-tensor containing the first $r_1 \times r_2 \times r_3$ t-scalar entries of S_{GT} .

$$\hat{X}_{GT} = (S_{GT})_{1:r_1,1:r_2,1:r_3} \circ_1 (U_{TM,1})_{:,1:r_1} \circ_2 (U_{TM,2})_{:,1:r_2} \circ_3 (U_{TM,3})_{:,1:r_3} . \tag{6.2}$$

When the t-scalar dimensions are given by $N = 1$, $I_1 = 1$, equation (6.2) reduces to the HOSVD low-rank approximation of a tensor in $C^{D_1 \times D_2 \times D_3}$. When the g-tensor dimension $D_3 = 1$, equation (6.2) reduces to the SVD low-rank approximation of a canonical matrix in $C^{D_1 \times D_2}$.

6.1.1. TSVD versus SVD — A “Vertical” Comparison. The low-rank approximation performances of TSVD and SVD are compared. In the experiment, the test sample is the $512 \times 512 \times 3$ RBG Lena image downloaded from Wikipedia.¹

For the SVD low-rank approximations, the RGB Lena image is split into three 512×512 monochrome images. Each monochrome image is analyzed using the SVD. The three extracted monochrome Lena images are order-two arrays in $\mathbb{R}^{512 \times 512}$. Each monochrome Lena image is tensorized to produce a t-image (a generalized monochrome image) in $\mathbb{R}^{512 \times 512} \equiv \mathbb{R}^{3 \times 3 \times 512 \times 512}$. In the tensorized version of the image each pixel value is replaced by a 3×3 square of values obtained from the 3×3 neighborhood of the pixel. Padding with 0 is used where necessary at the boundary of the image.

¹<https://en.wikipedia.org/wiki/Lenna>

To evaluate the TSVD approximations in a manner relevant to the SVD approximations, upon obtaining a t-image approximation $\hat{X}_{TM} \in \mathbb{R}^{3 \times 3 \times 512 \times 512}$, the part $\hat{X}_{MT}(i)|_{i=(2,2)} \in \mathbb{R}^{512 \times 512}$, i.e. the central slice of the TSVD approximation, is used for comparisons.

Given an array X of any order over the real numbers \mathbb{R} , let \hat{X} be an approximation to X . Then, the PSNR (Peak Signal-to-Noise Ratio) for \hat{X} is defined as in [1] by

$$PSNR = 20 \log_{10} \frac{MAX \cdot \sqrt{N^{entry}}}{\|X - \hat{X}\|_F} \quad (6.3)$$

where N^{entry} denotes the number of real number entries of X , $\|X - \hat{X}\|_F$ is the canonical Frobenius norm of the array $(X - \hat{X})$ and MAX is the maximum possible value of the entries of X . In all the experiments, $MAX = 255$.

Figure 1 shows the PSNR curves of the SVD and TSVD approximations as functions of the rank of \hat{X} . It is clear that the PSNR of the TSVD approximation is consistently higher than that of SVD approximation. When the rank $r = 500$, the PSNRs of TSVD and SVD differ by more than 37 dBs.

6.1.2. TSVD versus HOSVD — A “Horizontal” Comparison. Given a monochrome Lena image as an order-two array in $\mathbb{R}^{512 \times 512}$ and its tensorized form as an order-four array in $\mathbb{R}^{3 \times 3 \times 512 \times 512}$, TSVD yields an approximation array in $\mathbb{R}^{3 \times 3 \times 512 \times 512}$. Since the HOSVD is applicable to order-four arrays in $\mathbb{R}^{3 \times 3 \times 512 \times 512}$, we give a “horizontal” comparison of the performances of TSVD and HOSVD.

More specifically, given a generalized monochrome Lena image $X_{TM} \equiv X \in C^{512 \times 512} \equiv \mathbb{R}^{3 \times 3 \times 512 \times 512}$ and a specified rank r , the TSVD approximation yields a t-matrix $\hat{X}_{TM} \in C^{512 \times 512} \equiv \mathbb{R}^{3 \times 3 \times 512 \times 512}$, which is computed as in equation (4.2) with $D_1 = 512$ and $D_2 = 512$.

Let the HOSVD of $X \in \mathbb{R}^{3 \times 3 \times 512 \times 512}$ be $X = S \times_1 U_1 \times_2 U_2 \times_3 U_3 \times_4 U_4$ where $S \in \mathbb{R}^{3 \times 3 \times 512 \times 512}$ denotes the core tensor, and $U_1 \in \mathbb{R}^{3 \times 3}$, $U_2 \in \mathbb{R}^{3 \times 3}$, $U_3 \in \mathbb{R}^{512 \times 512}$, $U_4 \in \mathbb{R}^{512 \times 512}$ are all orthogonal matrices. Then, to give a “horizontal” comparison with the TSVD approximation \hat{X}_{TM} with rank r , the HOSVD approximation $\hat{X} \in \mathbb{R}^{3 \times 3 \times 512 \times 512}$ is given by the multi-mode product

$$\hat{X} = (S)_{:, :, 1:r, 1:r} \times_1 U_1 \times_2 U_2 \times_3 (U_3)_{:, 1:r} \times_4 (U_4)_{:, 1:r} . \quad (6.4)$$

The PSNRs TSVD and HOSVD are computed as in equation (6.3) with $MAX = 255$ and $N^{entry} = 3 \times 3 \times 512 \times 512 = 2359296$.

For each of the generalized monochrome Lena images (respectively marked by the channel type “red”, “green” and “blue”), as a $3 \times 3 \times 512 \times 512$ real number array, the PSNRs of TSVD and HOSVD are given in Figure 2.

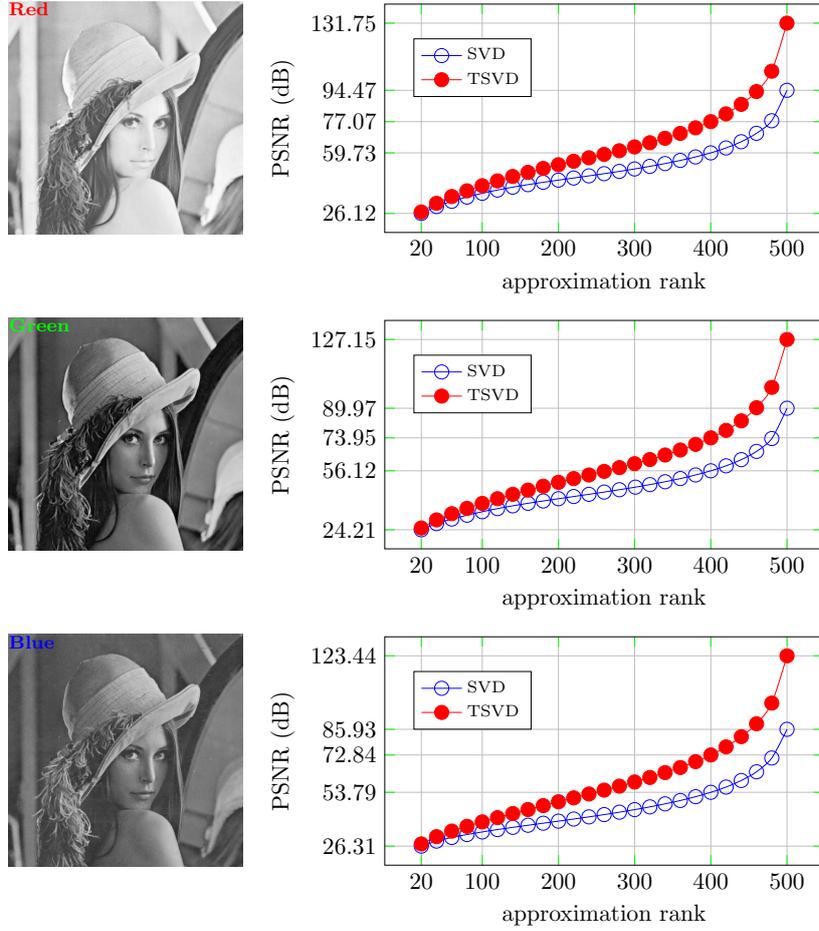


FIGURE 1. A “vertical” comparison of low-rank approximations by SVD and TSVD for each monochrome Lena image. First column: Monochrome images extracted from the RGB Lena image. Second column: PSNR curves of SVD/TSVD approximation on/for each monochrome image.

As rank r is varied, the PSNR of TSVD approximation is always higher than that of the corresponding HOSVD approximation. When rank r is equal to 500, the PSNRs of TSVD and HOSVD approximations differ significantly.

6.1.3. *THOSVD versus HOSVD — A “Vertical” Comparison.* The low-rank approximation performances of THOSVD and HOSVD are compared. For the HOSVD approximations the RGB Lena image, which is a tensor in $\mathbb{R}^{512 \times 512 \times 3}$, is used as the test sample. For the THOSVD the 3×3 neighborhood (with zero-padding) strategy is used to tensorize each real number entry of the RGB Lena image. The obtained t-image X_{GT} is a g-tensor in $\mathbb{R}^{512 \times 512 \times 3}$, i.e., an order-five array in $\mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$.

To give a “vertical” comparison, on obtaining an approximation $\hat{X}_{GT} \in \mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$, we compare $\hat{X}_{GT}(i)|_{i=(2,2)} \in \mathbb{R}^{512 \times 512 \times 3}$, i.e., the central slice of the THOSVD approximation, with the HOSVD approximation on the RGB Lena image.

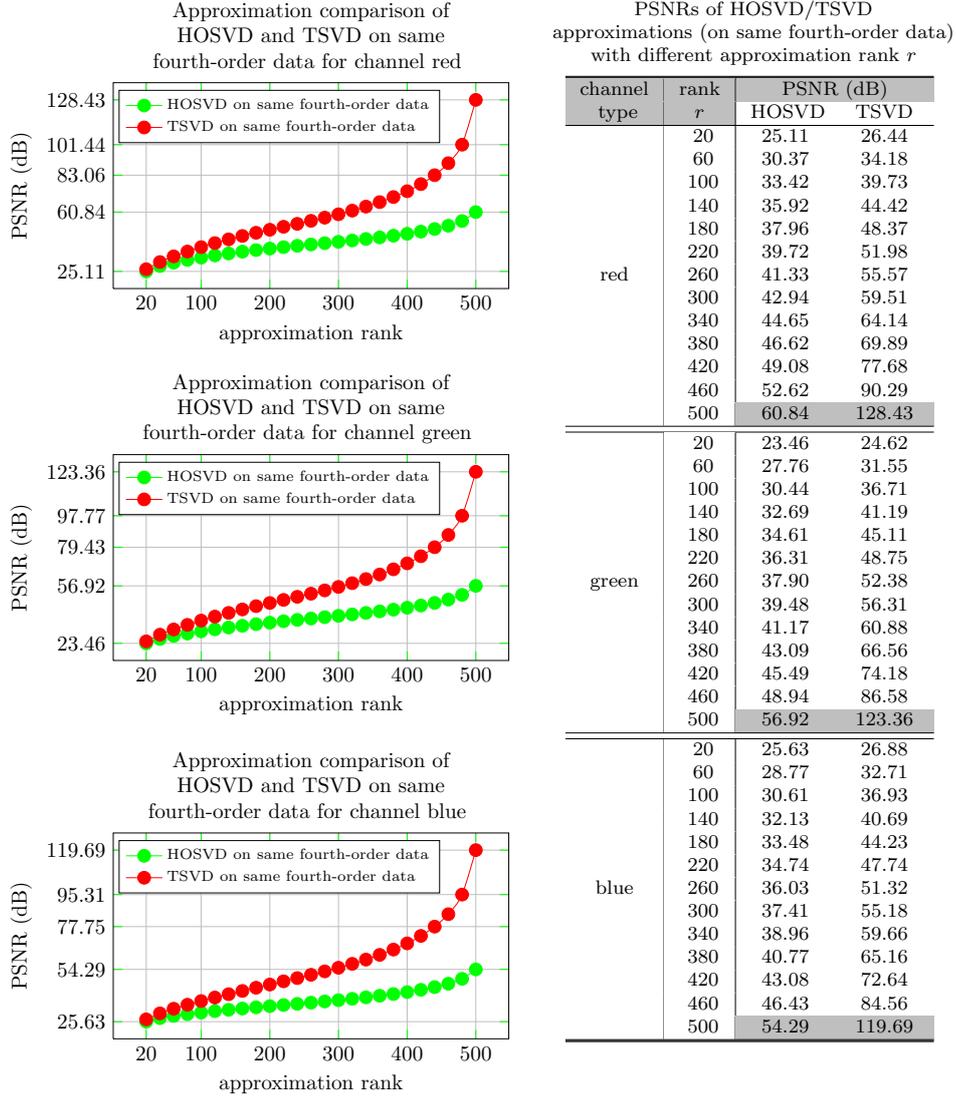


FIGURE 2. A “horizontal” comparison of low-rank approximations by HOSVD and TSVD on each generalized monochrome Lena image, as an fourth-order real number array in $\mathbb{R}^{3 \times 3 \times 512 \times 512}$. First column: PSNR curves, over rank r , of HOSVD/TSVD approximations on each generalized monochrome Lena image. Second column: Some quantitative PSNRs of HOSVD/TSVD approximations with rank r .

Figure 3 gives a “vertical” comparison of the PSNR maps of THOSVD and HOSVD approximations and the tabulated PSNRs for some representative multilinear rank tuples (r_1, r_2, r_3) . It shows the PSNR of the THOSVD approximation is consistently higher than the PSNR of the HOSVD approximation. When $(r_1, r_2, r_3) = (500, 500, 3)$, the approximations obtained by THOSVD and HOSVD differ by 30.29 dB in their PSNR values.

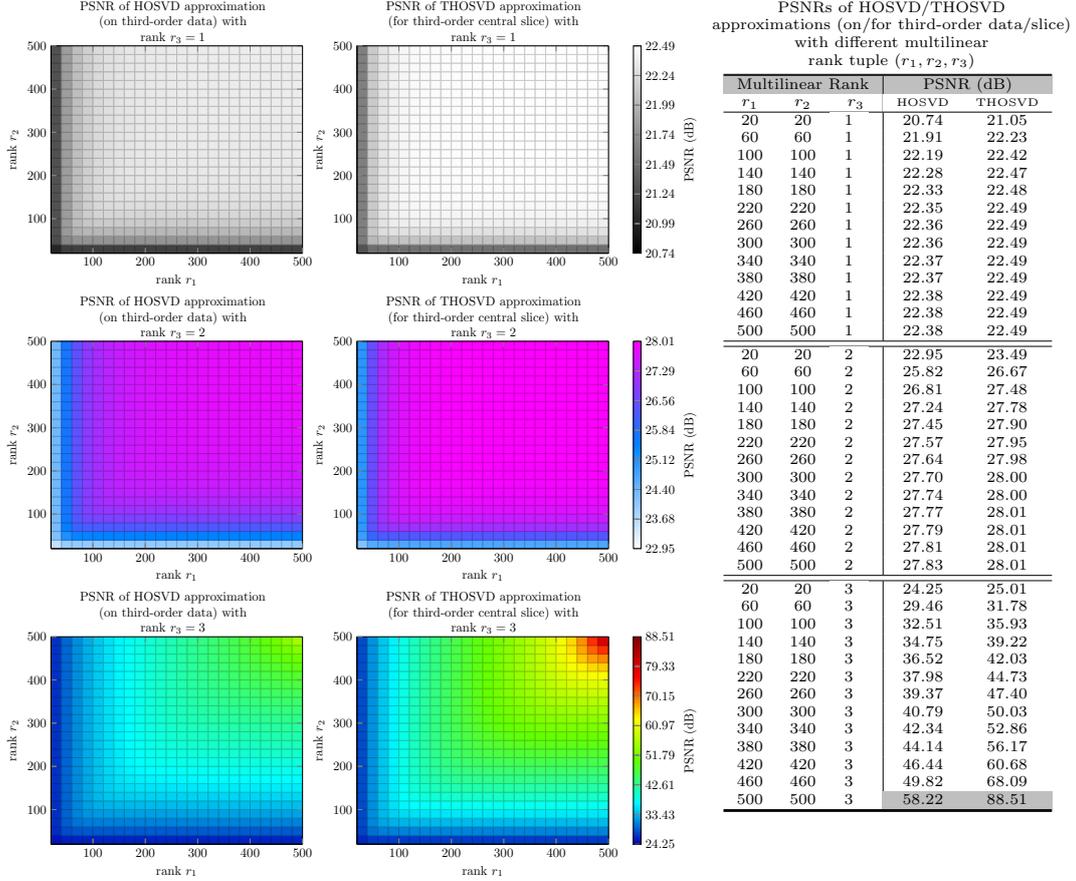


FIGURE 3. A “vertical” comparison of THOSVD approximations and HOSVD approximations with the multilinear rank tuple (r_1, r_2, r_3) . First column: PSNR maps of HOSVD approximation on the RGB Lena image. Second column: PSNR maps of THOSVD approximation for the RGB Lena image (i.e., third-order central slice of THOSVD approximation). Third column: Some quantitative PSNRs of HOSVD/THOSVD approximations with representative multilinear rank tuples.

6.1.4. *THOSVD versus HOSVD — A “Horizontal Comparison”.* Given a fifth-order array $X \in \mathbb{R}^{3 \times 3 \times 512 \times 512 \times 5}$ tensorized from the RGB Lena image, which is a third-order array in $\mathbb{R}^{512 \times 512 \times 3}$, both THOSVD and HOSVD can be applied to the same data X .

THOSVD takes X as a g-tensor $X_{GT} \in C^{512 \times 512 \times 3} \equiv \mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$ while HOSVD takes X merely as a canonical fifth-order array in $\mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$.

Then, given a rank tuple (r_1, r_2, r_3) subject to $1 \leq r_1 \leq 512$, $1 \leq r_2 \leq 512$ and $1 \leq r_3 \leq 3$, the THOSVD approximation $\hat{X}_{GT} \in C^{512 \times 512 \times 3}$ is computed as in equation (6.2).

Let the HOSVD of $X \in \mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$ be $X = S \times_1 U_1 \times_2 U_2 \times_3 U_3 \times_4 U_4 \times_5 U_5$ where $S \in \mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$ is the core tensor and $U_1 \in \mathbb{R}^{3 \times 3}$, $U_2 \in \mathbb{R}^{3 \times 3}$, $U_3 \in \mathbb{R}^{512 \times 512}$, $U_4 \in \mathbb{R}^{512 \times 512}$, $U_5 \in \mathbb{R}^{3 \times 3}$ are all orthogonal matrices.

Then, to give a “horizontal” comparison with the THOSVD approximation $\hat{X}_{GT} \in \mathbb{C}^{512 \times 512 \times 3}$ with a rank tuple (r_1, r_2, r_3) , the HOSVD approximation $\hat{X} \in \mathbb{R}^{3 \times 3 \times 512 \times 512 \times 3}$ is given by the following multi-mode product

$$\hat{X} = (S)_{:, :, 1:r_1, 1:r_2, 1:r_3} \times_1 U_1 \times_2 U_2 \times_3 (U_3)_{:, :, 1:r_1} \times_4 (U_4)_{:, :, 1:r_2} \times_5 (U_5)_{:, :, 1:r_3}. \quad (6.5)$$

Figure 4 gives the “horizontal” comparison of THOSV approximations and HOSVD approximations on the same array with different rank tuples (r_1, r_2, r_3) . Albeit somewhat smaller in PSNRs, the results in Figure 4 are similar to the results in Figure 3 (a “vertical” comparison), corroborating the claim that a THOSV approximation outperforms, in terms of PSNR, the corresponding HOSV approximation on the same data.

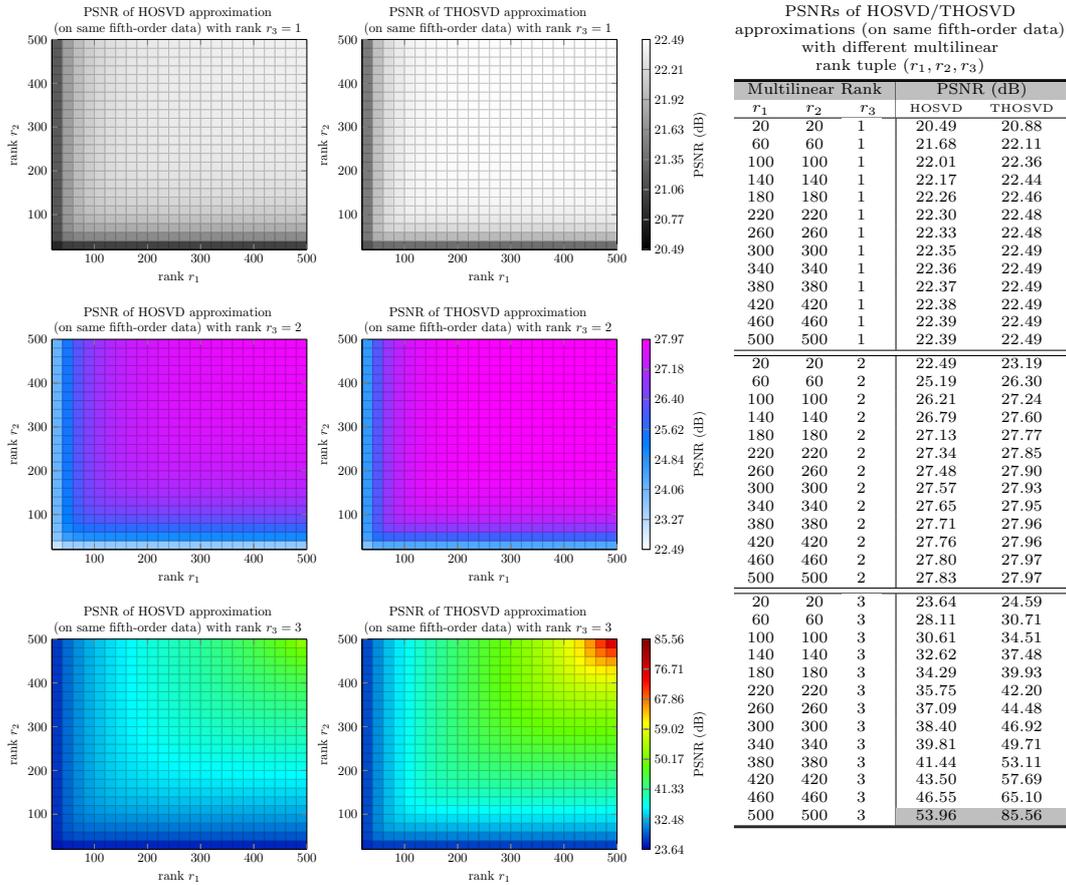


FIGURE 4. A “horizontal” comparison of THOSVD approximations and HOSVD approximations with multilinear rank tuple (r_1, r_2, r_3) . First column: PSNR maps of HOSVD approximation. Second column: PSNR maps of THOSVD approximation. Third column: Some PSNRs of HOSVD/THOSVD approximations on same fifth-order data with representative multilinear rank tuples (r_1, r_2, r_3) .

6.2. Reconstruction. The qualities of the low-rank reconstructions produced by TPCA and PCA and by T2DPCA and 2DPCA, as described by the equations (5.3) and (5.6), are compared.

The effectiveness of PCA, 2DPCA, TPCA and T2DPCA for reconstruction is assessed using the ORL dataset. The data set contains 400 face images in 40 classes, i.e., 10 images/class \times 40 classes. Each image has 112×92 pixels². The first 200 images (5 images/class \times 40 classes) are used as the observed images and the remaining 200 images are the query images.

For the experiments with TPCA/T2DPCA, all ORL images are tensorized to t-images in $\mathbb{R}^{112 \times 92}$, namely, order-four arrays in $\mathbb{R}^{3 \times 3 \times 112 \times 92}$. Eigendecompositions and t-eigendecompositions are computed on the observed images and t-images, respectively. Reconstructions are computed for the query images and t-images respectively. The number of PSNRs for the reconstructed images and t-images is 200. It is convenient to use the average of the PSNRs (denoted by A), the standard deviation of PSNRs (denoted by S), and the ratio, A/S . A larger value of A with a smaller value of S , indicates a better quality of reconstruction.

6.2.1. *TPCA versus PCA — A “Vertical” Comparison.* To make the TPCA and PCA reconstructions computationally tractable, each image is resized to 56×46 pixels by bi-cubic interpolation. The resized images are also tensorized to t-images, i.e., order-four arrays in $\mathbb{R}^{3 \times 3 \times 56 \times 46}$. The obtained images and t-images are then transformed to vectors and t-vectors, respectively, by stacking their columns. The central slices of the TPCA reconstructions are compared with the PCA reconstructions.

Figure 5 shows graphs and some tabulated values of A , S and A/S for a number of eigen-vectors and eigen-t-vectors. Note that K linearly independent observed vectors or t-vectors yield at most $(K - 1)$ eigen-vectors or eigen-t-vectors. Thus, the maximum number of eigen-vectors and eigen-t-vectors in Figure 5 is 199 ($K = 200$).

The average PSNR for TPCA is consistently higher than the average PSNR for PCA. The PSNR standard deviation for TPCA is slightly larger than the PSNR standard deviation for PCA, but the ratio A/S for TPCA is generally smaller than the ratio A/S for PCA. This indicates that TPCA outperforms PCA in terms of reconstruction quality.

6.2.2. *T2DPCA versus 2DPCA — A “Vertical” Comparison.* The same observed samples from the ORL dataset (the first 200 images, 5 images/class \times 40 classes) and query samples (the remaining 200 images) are used to compare the reconstruction performances of T2DPCA and 2DPCA. The central slices of the T2DPCA are compared with the 2DPCA reconstructions.

Figure 6 shows the reconstruction curves and some tabulated values yielded by T2PCA and 2DPCA as functions of the number d of eigenvectors or eigen-t-vectors. The average PSNR obtained by T2DPCA is consistently higher than the average PSNR obtained by 2DPCA. When the parameter d equals 111, the gap between the two average PSNRs

²<https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>

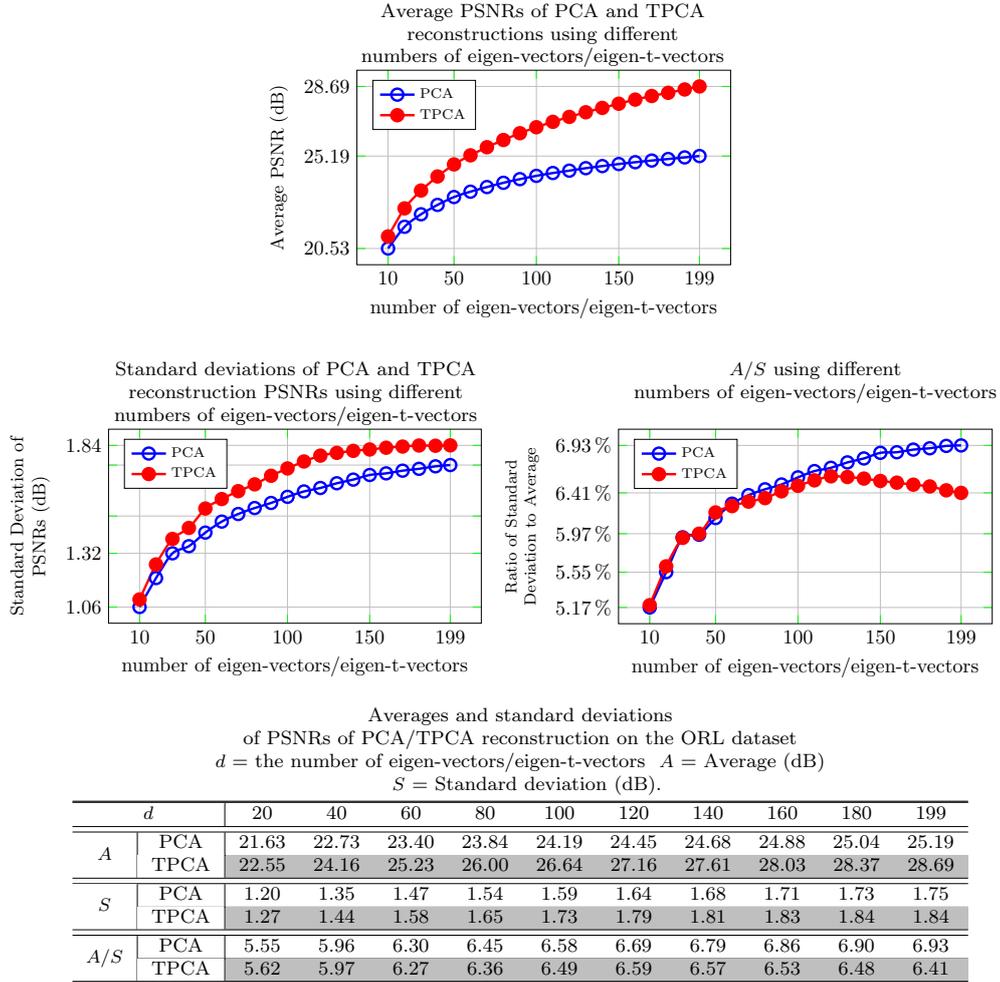


FIGURE 5. A “vertical” comparison of PSNR averages and standard deviations for PCA and TPCA reconstructions.

is 31.98 dBs. Furthermore, the PSNR standard deviation for T2DPCA is also generally smaller than the PSNR standard deviation for 2DPCA. In terms of reconstruction quality, T2DPCA outperforms 2DPCA.

6.3. Classification. TGCA and GCA are applied to the classification of the pixel values in hyperspectral images. Hyperspectral images have hundreds of spectral bands, in contrast with RGB images which have only three spectral bands. The multiple spectral bands and high resolution make hyperspectral imagery essential in remote sensing, target analysis, classification and identification [21, 15, 38, 10, 36, 24, 40]. Two publicly available data sets are used to evaluate the effectiveness of TGCA and GCA for supervised classification.

6.3.1. Datasets. The first hyperspectral image dataset is the Indian Pines cube (Indian cube for short), which consists of 145×145 hyperspectral pixels (hyperpixels for short)

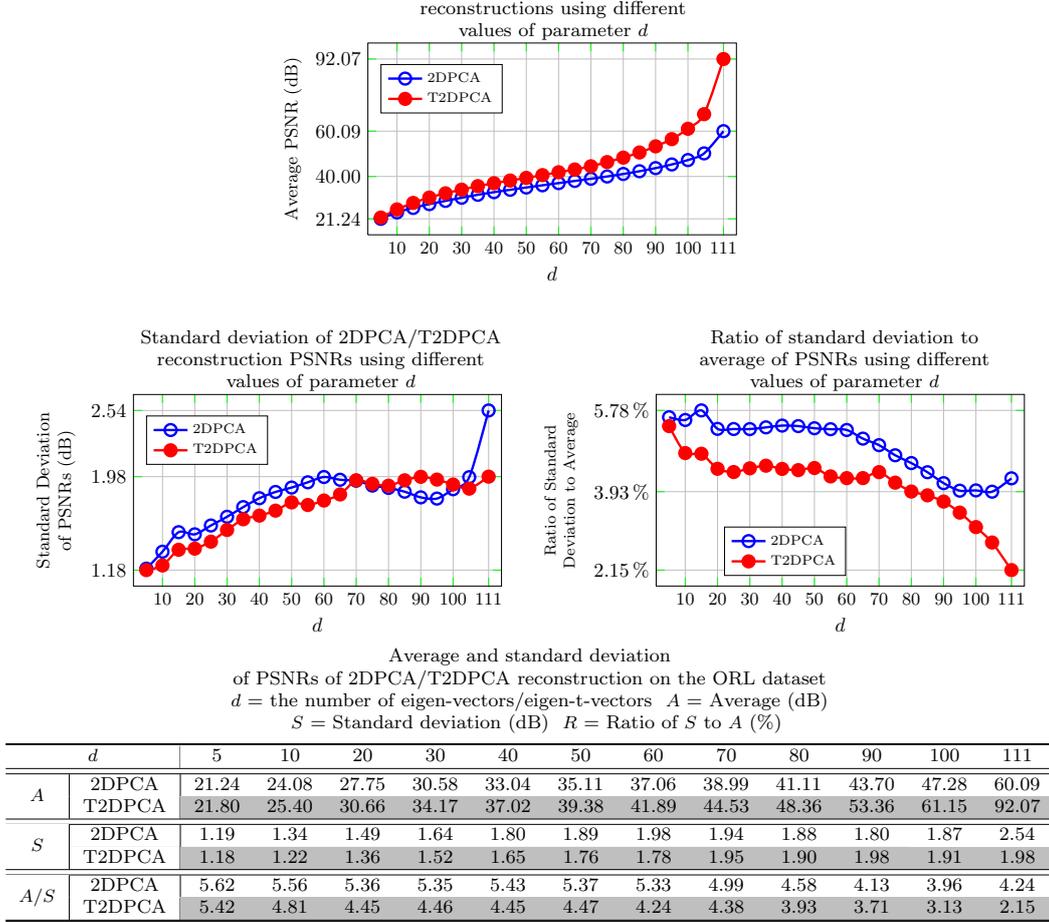


FIGURE 6. A “vertical” comparison of PSNR averages and standard deviations for the 2DPCA and T2DPCA reconstructions.

and has 220 spectral bands, yielding an array of order-three in $\mathbb{R}^{145 \times 145 \times 220}$. The Indian cube comes with ground-truth labels for 16 classes [31]. The second hyperspectral image dataset is the Pavia University cube (Pavia cube for short), which consists of 610×340 hyperpixels with 103 spectral bands, yielding an array of order three in $\mathbb{R}^{610 \times 340 \times 103}$. The ground-truth contains 9 classes [31].

6.3.2. Tensorization. Given a hyperspectral cube, let D_1 be the number of rows, D_2 the number of columns and D the number of spectral bands. A hyperpixel is represented by a vector in \mathbb{R}^D . Each pixel is tensorized by its 3×3 neighborhood. The tensorized hyperspectral cube is represented by an array in $\mathbb{R}^{3 \times 3 \times D_1 \times D_2 \times D}$. Each tensorized hyperpixel, called t-hyperpixel in this paper, is represented by a t-vector in \mathbb{R}^D , i.e., an array in $\mathbb{R}^{3 \times 3 \times D}$.

Figure 7 shows the tensorization of a canonical vector extracted from a hyperspectral cube. The tensorization of all vectors yields a tensorized hyperspectral cube in $\mathbb{R}^{3 \times 3 \times D_1 \times D_2 \times D}$.

6.3.3. *Input Matrices and T-matrices.* To classify a query hyperpixel, it is necessary to extract features from the hyperpixel. A t-hyperpixel in TGCA is represented by a set of t-vectors in the 5×5 neighborhood of the t-hyperpixel. These t-vectors are used to construct a t-matrix. A similar construction is used for GCA.

In GCA for example, let the vectors in the 5×5 neighborhood of a hyperpixel be $X_{vec,1}, \dots, X_{vec,25}$. The ordering of the vectors should be the same for all hyperpixels. The raw matrix X_{mat} representing the hyperpixel is given by marshalling these vectors as the columns of X_{mat} , namely $X_{mat} \doteq [X_{vec,1}, \dots, X_{vec,25}] \in \mathbb{R}^{D \times 25}$. The associated t-matrix $X_{TM} \in \mathbb{C}^{D \times 25}$ in TGCA is obtained by marshalling the associated 25 t-vectors.

After obtaining each matrix and t-matrix, the columns are orthogonalized. The resulting matrices and t-matrices are input samples for GCA and TGCA respectively.

6.3.4. *Classification.* To evaluate GCA, TGCA and the competing methods, the overall accuracies (OA) and the Cohen’s κ indices of the supervised classification of hyperpixels (i.e., prediction of class labels of hyperpixels) are used. The overall accuracies and κ indices are obtained for different component analysers and classifiers. Higher values of OA or κ indicate a higher component analyzer performance [9]. Let K be the number of query samples, let K' be the number of correctly classified samples. The overall accuracy is simply defined by $OA = K'/K$. The κ index is defined by [5]

$$\kappa = \frac{K \cdot K' - \sum_{j=1}^{N^{class}} a_j b_j}{K^2 - \sum_{j=1}^{N^{class}} a_j b_j} \quad (6.6)$$

where N^{class} is the number of classes, a_j is the number of samples belonging to the j -th class and b_j is the number of samples classified to the j -th class.

Two classical component analyzers, namely PCA and LDA, and four state-of-the-art component analyzers, namely TDLA [40], LTDA [42], GCA [13] and TPCA (ours) are evaluated against TGCA. As an evaluation baseline, the results obtained with the original raw canonical vectors for hyperpixels are given. These raw vectors are denoted as the “original” (ORI for short) vectors. Three vector-oriented classifiers, NN (Nearest

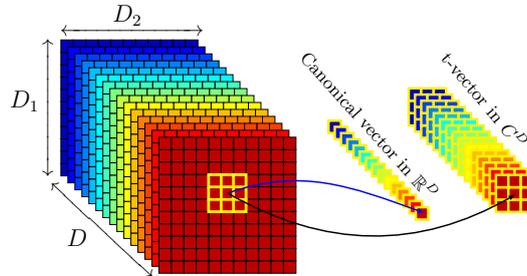


FIGURE 7. Tensorization of a canonical vector extracted from a hyper-spectral cube

Neighbor), SVM (Support Vector Machine), and RF (Random Forest), are employed to evaluate the effectiveness of the features extracted by these component analyzers.

In the experiments, the background hyperpixels are excluded, because they do not have labels in the ground-truth. A total of 10% of the foreground hyperpixels are randomly and uniformly chosen without replacement as the observed samples (i.e., samples whose class labels are known in advance). The rest of the foreground hyperpixels are chosen as the query samples, that is samples with the class labels to be determined.

In order to use the vector-oriented classifiers NN, SVM and RF, the t-vector results, generated by TGCA or TPCA, are transformed by pooling them to yield canonical vectors. For TGCA, the canonical vectors obtained by pooling are referred to as TGCA-I features and the t-vectors without pooling are referred to as the TGCA-II features.

To assess the effectiveness of the TGCA-II features, a generalized classifier which deals with t-vectors is needed. It is possible to generalize many canonical classifiers from vector-oriented to t-vector-oriented, however a comprehensive discussion of these generalizations is outside the scope of this paper. Nevertheless, it is very straightforward to generalize NN. The d -dimensional t-vectors are not only elements of the module C^d , but also the elements in the vector space $\mathbb{C}^{3 \times 3 \times d}$. This enables the use of the canonical Frobenius norm to measure the distance between two t-vectors, as the elements in $\mathbb{C}^{3 \times 3 \times d}$. The canonical Frobenius norm should not be confused with the generalized Frobenius norm defined in equation (3.3).

Figure 8 gives the highest classification accuracies obtained by each pair of component analyzer and classifier on the two hyperspectral cubes. The highest accuracies are obtained by traversing the set of feature dimensions $d \in \{5, 10, \dots, D_m\}$ where D_m is the maximum dimension valid for the associated component analyzer. Figure 8, shows that the results obtained by the algorithms TPCA, TGCA-I and TGCA-II, are consistently better than those obtained by their canonical counterparts. Even working with a relatively weak classifier NN, TGCA achieves the highest accuracies and highest κ indices in the experiments. Further results are shown in Figures 9 and 10. It is clear that the pair TGCA and NN yield the best results, outperforming any other pair of analyzer and classifier.

6.3.5. *TGCA versus GCA*. It is noted that the maximum dimension of the TGCA and GCA features is equal to the number of observed training samples, and therefore is much higher than the original dimension, which is equal to the number of spectral bands. Thus, taking the original dimension as the baseline, one can employ TGCA or GCA either for dimension reduction or dimension increase. When the so-called ‘‘curse of dimension’’ is the concern, one can discard the insignificant entries of the TGCA and GCA features. When the accuracy is the primary concern, one can use higher dimensional features.

The performances of TGCA and GCA for varying feature dimension are compared using accuracy curves generated by TGCA (ie., TGCA-I and TGCA-II) and GCA, as shown

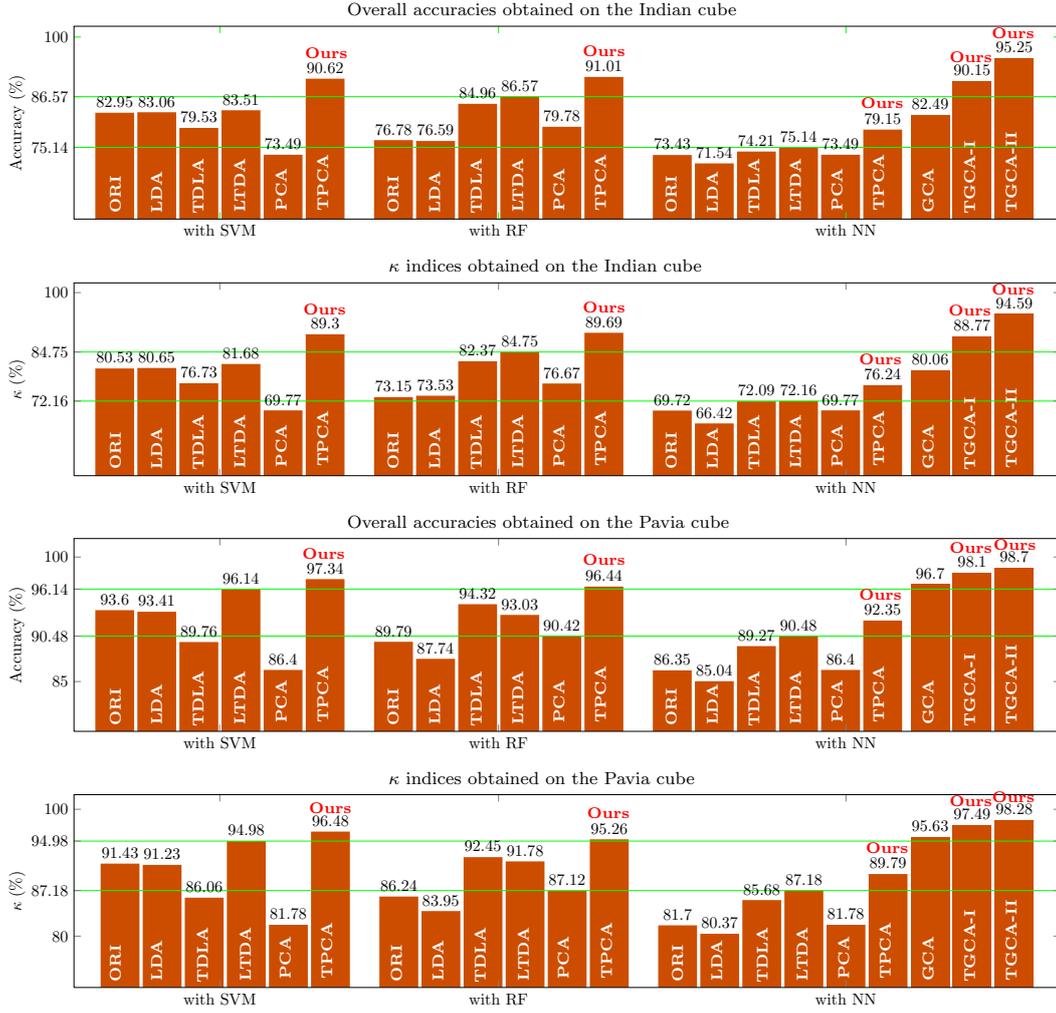


FIGURE 8. Classification accuracies obtained on two hyperspectral cubes

in Figure 11. The results are obtained for low feature dimensions and for high feature dimensions. It is clear that the classification accuracies obtained using TGCA and TGCAII are consistently higher than the accuracies obtained using GCA.

6.3.6. *TPCA versus PCA*. The classification accuracies of TPCA and PCA are compared, although the highest classification accuracies are not obtained from TPCA or PCA. The classification accuracy curves obtained by TPCA and PCA (with classifiers NN, SVM and RF) are given in Figure 12. It is clear that, no matter which classifier and feature dimension are chosen, the accuracy using TPCA is consistently higher than the accuracy using PCA.³

6.4. **Computational Cost.** The run times of t-matrix manipulations with different t-scalar sizes $I_1 \times I_2$ are given in Figure 13. The size of t-scalars ranges from $1 \leq I_1, I_2 \leq 32$. The evaluated t-matrix manipulations include addition, conjugate transposition,

³To use the same classifiers, pooling is used to transform the t-vectors by TPCA to canonical vectors.

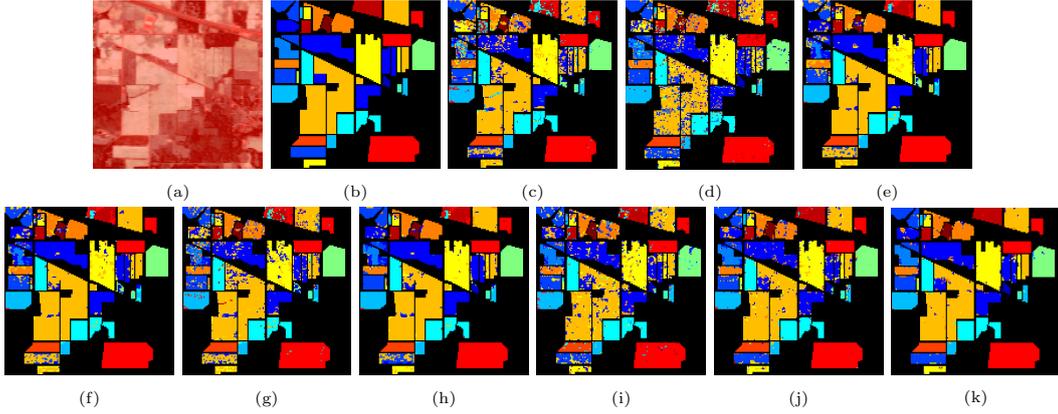


FIGURE 9. Some visual results obtained on the Indian cube. (a) Pseudo-colored 2D scene of Indian Pines (b) class ground-truth of hyperpixels (c) ORI with RF (d) LDA with RF (e) TDLA with RF (f) LTDA with RF (g) PCA with RF (h) TPCA with RF (i) GCA with NN (j) TGCA-I with NN (k) TGCA-II with NN

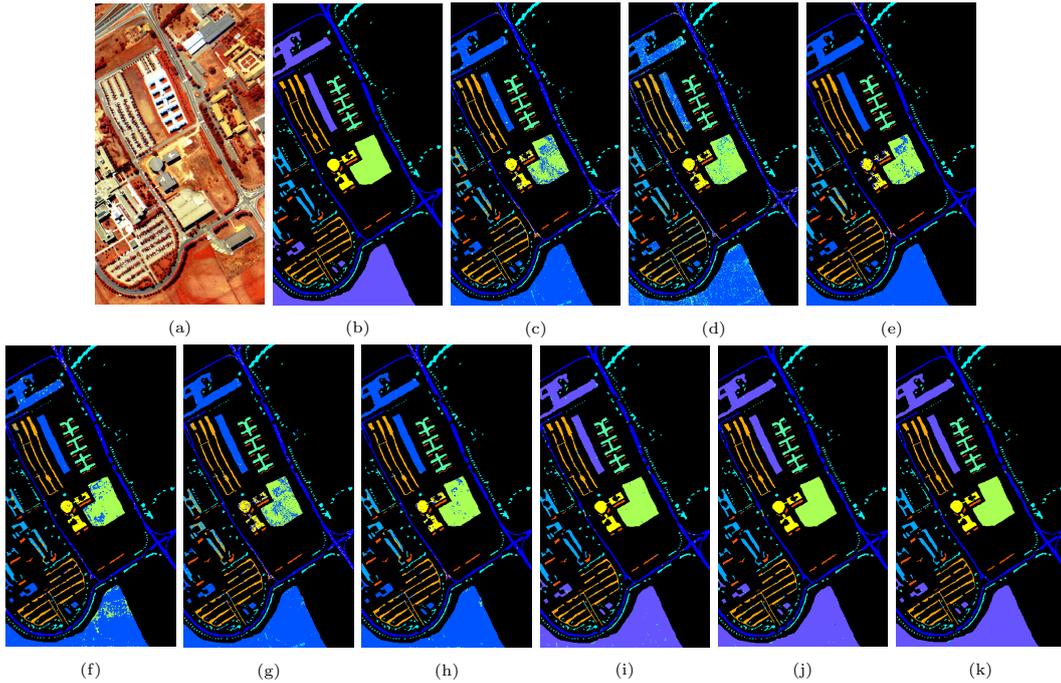


FIGURE 10. Some visual results obtained on the Pavia cube. (a) Pseudo-colored 2D scene of the Pavia University (b) class ground-truth of hyperpixels (c) ORI with RF (d) LDA with RF (e) TDLA with RF (f) LTDA with RF (g) PCA with RF (h) TPCA with RF (i) GCA with NN (j) TGCA-I with NN (k) TGCA-II with NN

multiplication and TSVD. The run time is evaluated using MATLAB R2018B on a notebook PC with Intel i7-4700MQ CPU at 2.40GHz and 16G GB memory.

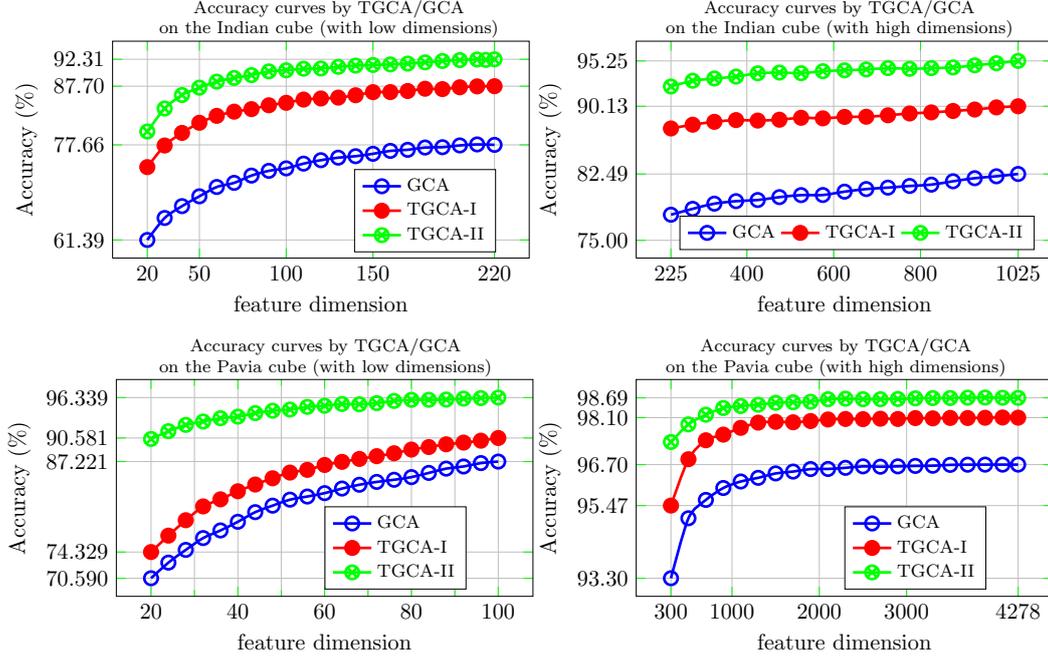


FIGURE 11. Accuracy curves obtained by TGCA/GCA (with NN) on the Indian/Pavia cube

Each time point in the figure is obtained by averaging 100 manipulations on random t -matrices in $\mathbb{R}^{I_1 \times I_2 \times 64 \times 64}$. Each t -matrix with $(I_1, I_2) \neq (1, 1)$ is transformed to the Fourier domain and manipulated via its $I_1 \cdot I_2$ slices. The results are transferred back to the original domain by the inverse Fourier transform. Note that when $(I_1, I_2) = (1, 1)$, a t -matrix manipulation is reduced to canonical matrix manipulation. The reported run time of a canonical matrix manipulation does not include the time spent on the Fourier transform and its inverse transform.

From Figure 13, it can be seen that the run time is essentially an increasing linear function of the number of slices, i.e., $I_1 \cdot I_2$.

7. CONCLUSION

An algebraic framework of tensorial matrices is proposed for generalized visual information analysis. The algebraic framework generalizes the canonical matrix algebra, combining the “multi-way” merits of high-order arrays and the “two-way” intuition of matrices. In the algebraic framework, scalars are extended to t -scalars, which are implemented as high-order numerical arrays of a fixed-size. With appropriate operations, the t -scalars are trinitarian in the following sense. (1) T -scalars are generalized complex numbers. (2) T -scalars are elements of an algebraic ring. (3) T -scalars are elements of a linear space.

Tensorial matrices, called t -matrices, are constructed with t -scalar elements. The resulting t -matrix algebra is backward compatible with the canonical matrix algebra. Using

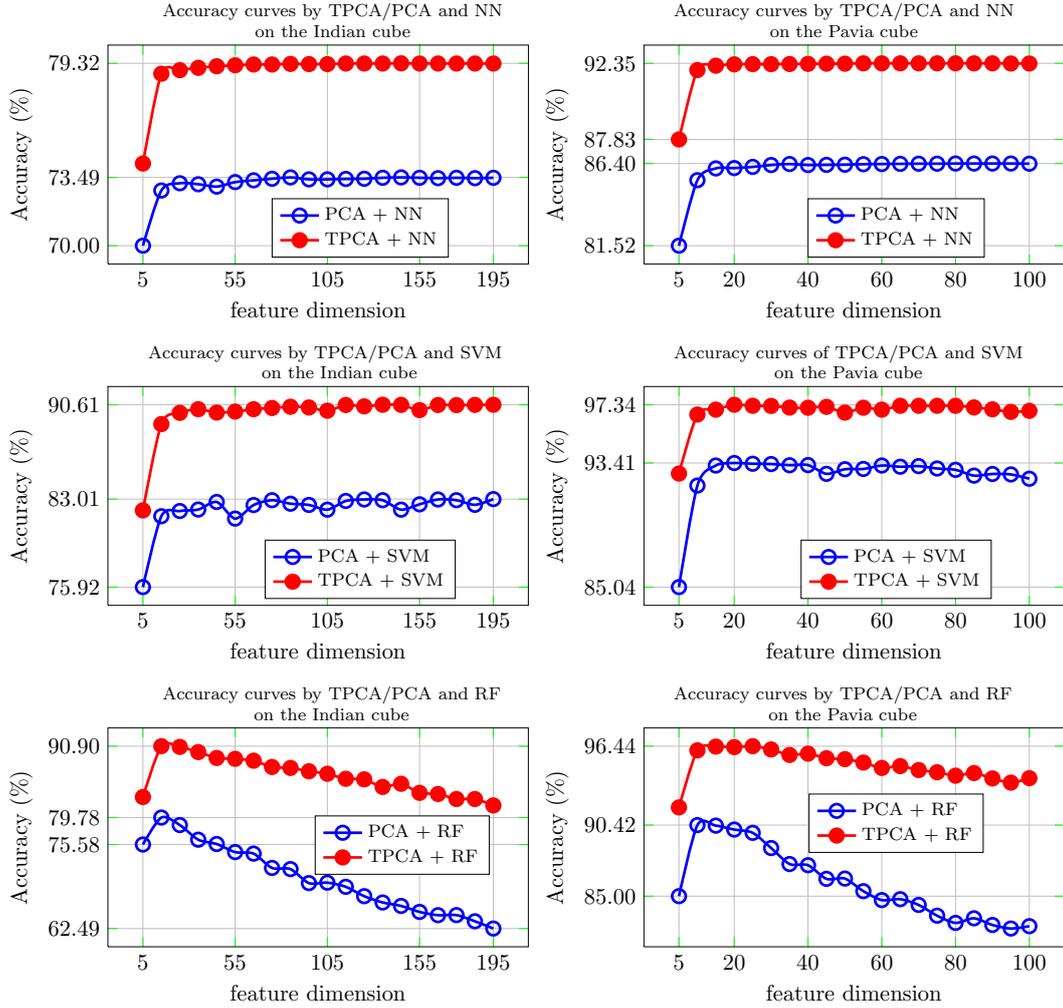


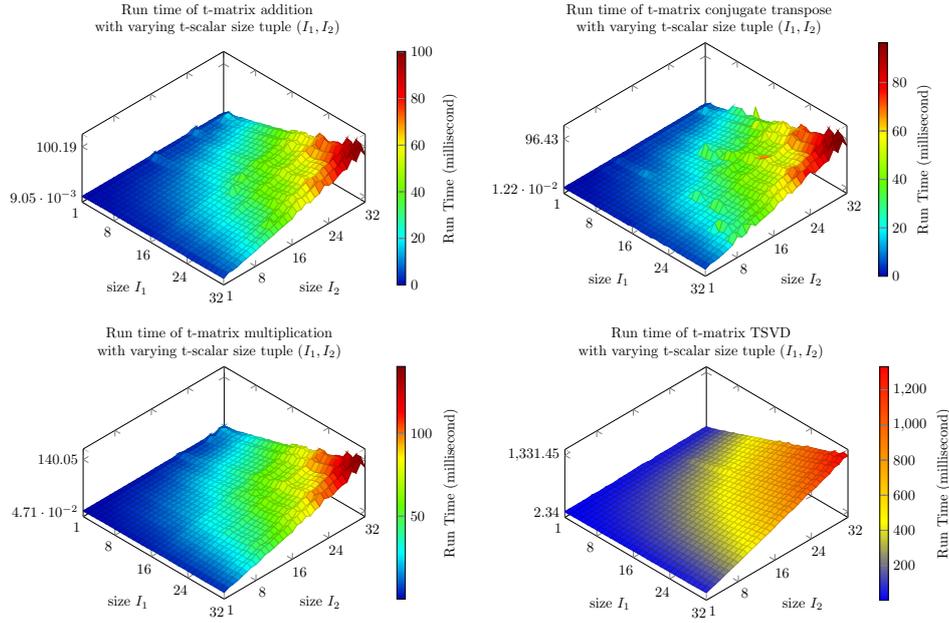
FIGURE 12. Accuracy curves obtained by TPCA/PCA and different classifiers on the Indian/Pavia cube. First column: results on the Indian cube. Second column: results on the Pavia cube.

this t-algebra framework, it is possible to generalize many canonical matrix and vector constructions and algorithms.

To demonstrate the “multi-way” merits and “two-way” matrix intuition of the proposed tensorial algebra and its applications to generalized visual information analysis, the canonical matrix algorithms SVD, HOSVD, PCA, 2DPCA and GCA are generalized. Experiments with low-rank approximation, reconstruction, and supervised classification show that the generalized algorithms compare favorably with their canonical counterparts on visual information analysis.

ACKNOWLEDGEMENTS

Liang Liao would like to thank professor Pinzhi Fan (Southwestern Jiaotong University, China) for his support and some insightful suggestions to this work. Liang Liao also



Run time (milliseconds) of t-matrix addition/conjugate transposition/multiplication/TSVD with some different t-scalar size duples

(I_1, I_2)	(1, 1)	(4, 4)	(8, 8)	(12, 12)	(16, 16)	(20, 20)	(24, 24)	(28, 28)	(32, 32)
addition	0.009	1.278	4.371	11.075	19.906	32.610	51.206	67.237	83.592
conjugate transpose	0.012	1.434	4.462	11.297	19.630	33.012	51.696	65.216	82.747
multiplication	0.047	1.846	6.730	15.786	28.253	59.813	72.874	95.064	118.535
TSVD	2.341	18.837	77.587	177.804	321.547	503.543	740.465	1002.679	1314.483

[†] When $(I_1, I_2) = (1, 1)$, the manipulation is equivalent to a canonical matrix manipulation and the reported time does not include the time spent on the Fourier transform and its inverse transform.

FIGURE 13. Run time of some t-matrix manipulations with different t-scalar sizes

would like to thank Yuemei Ren, Chengkai Yang, Haichang Ye, Jie Yang and Xuechun Zhang for their supports to some early stage experiments of this work.

All prospective supports and collaborations to this research are welcome. Contact email: liaolangis@126.com or liaoliang2018@gmail.com.

REFERENCES

- [1] Almohammad, A., Ghinea, G.: Stego image quality and the reliability of PSNR. In: 2010 2nd International Conference on Image Processing Theory, Tools and Applications, pp. 215–220 (2010)
- [2] Bracewell, R.N., Bracewell, R.N.: The Fourier transform and its applications, 3rd edn. pp. 108–112. McGraw-Hill, New York (1999)
- [3] Braman, K.: Third-order tensors as linear operators on a space of matrices. *Linear Algebra & Its Applications* **433**(7), 1241–1253 (2010)
- [4] Chen, Z., Wang, B., Niu, Y., Xia, W., Zhang, J.Q., Hu, B.: Change detection for hyperspectral images based on tensor analysis. In: Geoscience and Remote Sensing Symposium, pp. 1662–1665 (2015)
- [5] Cohen, J.: A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* **20**(1), 37–46 (1960)

- [6] De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications* **21**(4), 1253–1278 (2000)
- [7] Eckart, C., Young, G.: The approximation of one matrix by another of lower rank. *Psychometrika* **1**(3), 211–218 (1936)
- [8] Fan, H., Li, C., Guo, Y., Kuang, G., Ma, J.: Spatial-spectral total variation regularized low-rank tensor decomposition for hyperspectral image denoising. *IEEE Transactions on Geoscience & Remote Sensing* **56**(10), 6196–6213 (2018)
- [9] Fitzgerald, R.W., Lees, B.G.: Assessing the classification accuracy of multisource remote sensing data. *Remote Sensing of Environment* **47**(3), 362–368 (1994)
- [10] Fu, W., Li, S., Fang, L., Kang, X., Benediktsson, J.A.: Hyperspectral image classification via shape-adaptive joint sparse representation. *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing* **9**(2), 556–567 (2016)
- [11] Gleich, D.F., Chen, G., Varah, J.M.: The power and Arnoldi methods in an algebra of circulants. *Numerical Linear Algebra with Applications* **20**(5), 809–831 (2013)
- [12] Golub, G., Loan, C.V.: *Matrix Computations*, chap. 2. North Oxford Academic, Oxford (1983)
- [13] Harandi, M., Hartley, R., Shen, C., Lovell, B., Sanderson, C.: Extrinsic methods for coding and dictionary learning on Grassmann manifolds. *International Journal of Computer Vision* **114**(2), 113–136 (2015)
- [14] Harandi, M.T., Hartley, R., Lovell, B., Sanderson, C.: Sparse coding on symmetric positive definite manifolds using Bregman divergences. *IEEE Transactions on Neural Networks & Learning Systems* **27**(6), 1294–1306 (2015)
- [15] He, Z., Li, J., Liu, L.: Tensor block-sparsity based representation for spectral-spatial hyperspectral image classification. *Remote Sensing* **8**(8), 636 (2016)
- [16] Hungerford, T.: *Algebra, Graduate Texts in Mathematics*, vol. 73, chap. IV. Springer, New York (1974)
- [17] Kilmer, M.E., Braman, K., Hao, N., Hoover, R.C.: Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *SIAM Journal on Matrix Analysis & Applications* **34**(1), 148–172 (2013)
- [18] Kilmer, M.E., Martin, C.D.: Factorization strategies for third-order tensors. *Linear Algebra & Its Applications* **435**(3), 641–658 (2011)
- [19] Kolda, T., Bader, B.W.: Tensor decompositions and applications. *SIAM Review* **51**(3), 455–500 (2009)
- [20] Liao, L., Maybank, S.J., Zhang, Y., Liu, X.: Supervised classification via constrained subspace and tensor sparse representation. In: *International Joint Conference on Neural Networks*, pp. 2306–2313 (2017)
- [21] Liu, Z., Tang, B., He, X., Qiu, Q., Wang, H.: Sparse tensor-based dimensionality reduction for hyperspectral spectral-spatial discriminant feature extraction. *IEEE Geoscience & Remote Sensing Letters* **1775-1779**(99), 1–5 (2017)
- [22] Lu, C., Feng, Y., Liu, W., Lin, Z., Yan, S.: Tensor robust principal component analysis: exact recovery of corrupted low rank tensors via convex optimization. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5249–5257 (2016)
- [23] Lu, H., Plataniotis, K.N., Venetsanopoulos, A.N.: MPCA: Multilinear principal component analysis of tensor objects. *IEEE transactions on Neural Networks* **19**(1), 18–39 (2008)
- [24] Ma, X., Wang, H., Geng, J.: Spectral-spatial classification of hyperspectral image based on deep auto-encoder. *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing* **9**(9), 4073–4085 (2016)
- [25] Muralidhara, C., Gross, A.M., Gutell, R.R., Alter, O.: Tensor decomposition reveals concurrent evolutionary convergences and divergences and correlations with structural motifs in ribosomal RNA. *PloS one* **6**(4), e18768 (2011)
- [26] Omberg, L., Golub, G.H., Alter, O.: A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies. *Proceedings of the National Academy of Sciences of the United States of America* **104**, 18371–18376 (2007)
- [27] Papalexakis, N.S.L.D.X.F.K.H.E., Faloutsos, C.: Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing* **65**(13), 3551–3582 (2017)

- [28] Ren, Y., Liao, L., Maybank, S.J., Zhang, Y., Liu, X.: Hyperspectral image spectral-spatial feature extraction via tensor principal component analysis. *IEEE Geoscience & Remote Sensing Letters* **14**(9), 1431–1435 (2017)
- [29] Taguchi, Y.H.: Tensor decomposition-based unsupervised feature extraction applied to matrix products for multi-view data processing. *Plos One* **12**(8), e0183933 (2017)
- [30] Tucker, L.R.: Some mathematical notes on three-mode factor analysis. *Psychometrika* **31**(3), 279–311 (1966)
- [31] University of the Basque country: Hyperspectral remote sensing scenes. http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes
- [32] Vannieuwenhoven, N., Vandebril, R., Meerbergen, K.: A new truncation strategy for the higher-order singular value decomposition. *SIAM Journal on Scientific Computing* **34**(2), A1027–A1052 (2012)
- [33] Vasilescu, M.A.O.: Human motion signatures: Analysis, synthesis, recognition. In: *Proceedings of International Conference on Pattern Recognition*, vol. 3, pp. 456–460 (2002)
- [34] Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: TensorFaces. In: *European Conference on Computer Vision*, pp. 447–460. Springer (2002)
- [35] Vasilescu, M.A.O., Terzopoulos, D.: TensorTextures: Multilinear image-based rendering. *ACM Transactions on Graphics* **23**(3), 336–342 (2004)
- [36] Wei, Y., Zhou, Y., Li, H.: Spectral-spatial response for hyperspectral image classification. *Remote Sensing* **9**(3), 203–233 (2017)
- [37] Yang, J., Zhang, D., Frangi, A.F., Yang, J.Y.: Two-dimensional PCA: a new approach to appearance-based face representation and recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence* **26**(1), 131–7 (2004)
- [38] Zhang, E., Zhang, X., Jiao, L., Li, L., Hou, B.: Spectral-spatial hyperspectral image ensemble classification via joint sparse representation. *Pattern Recognition* **59**, 42–54 (2016)
- [39] Zhang, J., Saibaba, A.K., Kilmer, M., Aeron, S.: A randomized tensor singular value decomposition based on the t-product. *Numerical Linear Algebra with Applications* **25**(5) (2018)
- [40] Zhang, L., Zhang, L., Tao, D., Huang, X.: Tensor discriminative locality alignment for hyperspectral image spectral-spatial feature extraction. *IEEE Transactions on Geoscience & Remote Sensing* **51**(1), 242–256 (2013)
- [41] Zhang, Z., Ely, G., Aeron, S., Hao, N., Kilmer, M.: Novel methods for multilinear data completion and de-noising based on tensor-SVD. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3842–3849 (2014)
- [42] Zhong, Z., Fan, B., Duan, J., Wang, L., Ding, K., Xiang, S., Pan, C.: Discriminant tensor spectral-spatial feature extraction for hyperspectral image classification. *IEEE Geoscience & Remote Sensing Letters* **12**(5), 1028–1032 (2015)

APPENDIX I

Before giving a proof of the equivalence of equations (4.2) and (4.3), namely, the generalized Eckart-Young-Mirsky theorem, some notations need to be defined.

First, $\text{rank}(\cdot)$ denotes the rank of a t-matrix, which generalizes the rank of a canonical matrix and is defined as follows.

Definition I, rank of a t-matrix. Given a t-matrix, the rank $Y_T \doteq \text{rank}(X_{TM})$ is a nonnegative t-scalar such that

$$F(Y_T)_i = \text{rank}(F(X_{TM})(i)) \geq 0, \quad 1 \leq i \leq I. \quad (7.1)$$

where $F(X_{TM})(i)$ denotes the i -th slice of the Fourier transform $F(X_{TM})$.

Definition II, partial ordering of nonnegative t-scalars. Given two nonnegative t-scalars X_T and Y_T , the notation $X_T \leq Y_T$ is equivalent to the following condition

$$F(X_T)_i \leq F(Y_T)_i, \quad 1 \leq i \leq I. \quad (7.2)$$

Definition III, minimization of nonnegative t-scalar variable. For a nonnegative t-scalar variable X_T varying in a subset of S^{nonneg} , $Y_T \doteq \min(X_T)$ is the nonnegative t-scalar infimum of the subset, satisfying the following condition.

$$F(Y_T)_i = \min(F(X_T)_i) \geq 0, \quad 1 \leq i \leq I \quad (7.3)$$

where $F(Y_T)$ and $F(X_T)$ respectively denote the Fourier transforms of Y_T and X_T .

Given two nonnegative t-scalars X_T and Y_T , let M_T be the nonnegative t-scalar defined by $M_T = \min(X_T, Y_T)$, namely

$$F(M_T)_i = \min(F(X_T)_i, F(Y_T)_i) \geq 0, \quad i \leq i \leq I. \quad (7.4)$$

The above definitions are not casual ones. Following the above definitions, it is not difficult to verify that many generalized rank properties hold in the analogous form of their canonical counterparts.

For examples, given any t-matrices $X_{TM} \in C^{D_1 \times D_2}$ and $Y_{TM} \in C^{D_2 \times D_3}$, the following inequalities hold.

$$Z_T \leq \text{rank}(X_{TM}) \leq \min(D_1, D_2) \cdot E_T. \quad (7.5)$$

$$\begin{aligned} Z_T &\leq \text{rank}(X_{TM} + Y_{TM}) \\ &\leq \text{rank}(X_{TM}) + \text{rank}(Y_{TM}). \end{aligned} \quad (7.6)$$

$$\begin{aligned}
& \text{rank}(X_{TM}) + \text{rank}(Y_{TM}) - D_2 \cdot E_T \\
& \leq \text{rank}(X_{TM} \circ Y_{TM}) \\
& \leq \min(\text{rank}(X_{TM}), \text{rank}(Y_{TM}))
\end{aligned} \tag{7.7}$$

Since a t-scalar is a t-matrix of one row and one column, the rank of a t-scalar can be obtained.

Given any t-scalar X_T , let $G_T \doteq \text{rank}(X_T)$ be the rank of X_T . Then, following equation (7.1), it is not difficult to prove that the i -th entry of the Fourier transform $F(G_T)$ is given as follows.

$$F(G_T)_i = \begin{cases} 1, & \text{if } X_{T,i} \neq 0 \\ 0, & \text{otherwise} \end{cases}, \quad 1 \leq i \leq I. \tag{7.8}$$

Following the partial ordering given as in (7.2) and equation (7.1), it is not difficult to prove that the following propositions hold.

$$\begin{aligned}
Z_T & \leq \text{rank}(X_T) \leq E_T, \text{ for all t-scalars} \\
Z_T & = \text{rank}(X_T) \text{ iff } X_T = Z_T \\
E_T & = \text{rank}(X_T) \text{ iff } X_T \text{ is invertible.}
\end{aligned} \tag{7.9}$$

It follows from (7.9) that $Z_T < \text{rank}(X_T) < E_T$ iff the t-scalar X_T is non-zero and non-invertible.⁴

Generalized rank from a TSVD perspective. Given any t-matrix $X_{TM} = U_{TM} \circ S_{TM} \circ V_{TM}^H$ where $S_{TM} \doteq \text{diag}(\lambda_{T,1}, \dots, \lambda_{T,k}, \dots, \lambda_{T,Q})$ and $\lambda_{T,k} \in C$ is a t-scalar for all k , then the following equation holds and generalizes its canonical counterpart.

$$Z_T \leq \text{rank}(X_{TM}) \equiv \sum_{k=1}^Q \text{rank}(\lambda_{T,k}) \leq Q \cdot E_T. \tag{7.10}$$

Let the approximation be $\hat{X}_{TM} = U_{TM} \circ \hat{S}_{TM} \circ V_{TM}^H$ where $\hat{S}_{TM} = \text{diag}(\lambda_{T,1}, \dots, \lambda_{T,r}, \underbrace{Z_T, \dots, Z_T}_{Q-r})$.

Then, \hat{X}_{TM} is a low-rank approximation to X_{TM} since the following rank inequality holds.

$$\text{rank}(\hat{X}_{TM}) \equiv \sum_{k=1}^r \text{rank}(\lambda_{T,k}) \leq \text{rank}(X_{TM}). \tag{7.11}$$

Furthermore, it is not difficult to verify that equation (4.3) is equivalent to the following equation in the form of canonical matrices (i.e., slices of Fourier transformed t-matrices).

$$\begin{aligned}
\tilde{X}_{TM}^{approx}(i) & = \text{argmin}_{Y_{mat} \in C^{D_1 \times D_2}} \|\tilde{X}_{TM}(i) - Y_{mat}\|_F \\
& \text{subject to } \text{rank}(Y_{mat}) \leq r, \quad 1 \leq i \leq I
\end{aligned} \tag{7.12}$$

⁴ The partial order “<” is defined between nonnegative t-scalars. The inequality $Z_T < \text{rank}(X_T) < E_T$ means $Z_T \leq \text{rank}(X_T) \leq E_T$ and $\text{rank}(X_T) \neq Z_T$ and $\text{rank}(X_T) \neq E_T$.

where \tilde{X}_{TM}^{approx} and \tilde{X}_{TM} respectively denote the Fourier transform of X_{TM}^{approx} and X_{TM} in equation (4.3), and $\text{rank}(Y_{mat})$ is the rank of a complex matrix Y_{mat} in $\mathbb{C}^{D_1 \times D_2}$.

On the other hand, by applying the Fourier transforms to both sides of equation (4.2), equation (4.2) is transformed to the following equation in the form of canonical matrices (i.e., slices of Fourier transformed t-matrices).

$$\tilde{X}_{TM}^{svd}(i) = \tilde{U}_{TM}(i) \cdot \tilde{S}_{TM}^{approx}(i) \cdot \left(\tilde{V}_{TM}(i)\right)^H, \quad 1 \leq i \leq I \quad (7.13)$$

where \tilde{X}_{TM}^{svd} , \tilde{U}_{TM} , \tilde{S}_{TM}^{approx} and \tilde{V}_{TM}^{approx} respectively denote the Fourier transforms of \hat{X}_{TM} , U_{TM} , \hat{S}_{TM} and V_{TM} in equation (4.2).

The canonical Eckart-Young-Mirsky theorem guarantees the equivalence of equations (7.12) and (7.13).

APPENDIX II

This appendix contains an equivalent definition of the Fourier transform and its inverse transform of a multi-way array, and numerical examples to illustrate some of the definitions in the article.

1. Given a multi-way array $X \in \mathbb{C}^{I_1 \times I_2 \times \dots \times I_N}$, the order- N Fourier transform of X is given by the following multi-mode tensor multiplication, which is equivalent to the Fourier transform definition given in Section 2.3.

$$\tilde{X} \doteq F(X) \doteq X \times_1 W_{mat}^{(I_1)} \dots \times_k W_{mat}^{(I_k)} \dots \times_N W_{mat}^{(I_N)}$$

where $W_{mat}^{(I_k)}$ is the Fourier matrix of size $(I_k \times I_k)$. The (k_1, k_2) -th entry of $W_{mat}^{(I_k)}$ is defined as the following complex number

$$\left(W_{mat}^{(I_k)} \right)_{k_1, k_2} \doteq I_k^{-\frac{1}{2}} \cdot e^{2\pi\sqrt{-1} \cdot (k_1-1) \cdot (k_2-1) \cdot I_k^{-1}}$$

for all $1 \leq k_1, k_2 \leq I_k$.

The inverse Fourier transform is also defined using multi-mode tensor multiplication

$$X \doteq \tilde{X} \times_1 \left(W_{mat}^{(I_1)} \right)^{-1} \dots \times_k \left(W_{mat}^{(I_k)} \right)^{-1} \dots \times_N \left(W_{mat}^{(I_N)} \right)^{-1}$$

where $\left(W_{mat}^{(I_k)} \right)^{-1}$ is the the inverse matrix of $W_{mat}^{(I_k)}$ for all $1 \leq k \leq N$.

2. A diagram of the multiplication of two t-scalars, either in the spacial domain or the Fourier domain, is given in Figure 14.

3. An illustrative example of t-matrix multiplication is given in Figure 15. The t-matrices are $C_{TM} = A_{TM} \circ B_{TM} \in C^{2 \times 1} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 1}$, where $A_{TM} \in C^{2 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 2}$ and $B_{TM} \in C^{2 \times 1} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 1}$.

4. An example of a generalized tensor (g-tensor) $X_{GT} \in C^{2 \times 3 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3 \times 2}$ (the size of t-scalars is 3×3) and its generalized mode- k flattening are given in Figure 16.

In the Figure 16, the notation $(X_{GT})_{::,1} \in C^{2 \times 3}$ denotes the first frame of X_{GT} and the notation $(X_{GT})_{::,2} \in C^{2 \times 3}$ denotes the second frame of X_{GT} . The notations $X_{GT(1)} \in C^{2 \times 6}$, $X_{GT(2)} \in C^{3 \times 4}$, $X_{GT(3)} \in C^{2 \times 6}$ respectively denote the generalized mode-1, mode-2 and mode-3 flattening t-matrix forms of X_{GT} .

5. An example result of generalized mode-2 multiplication $M_{GT} \doteq X_{GT} \circ_2 Y_{TM} \in C^{2 \times 2 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 2 \times 2}$, where $X_{GT} \in C^{2 \times 3 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3 \times 2}$ and $Y_{TM} \in C^{2 \times 3} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3}$ is shown in Figure 17.

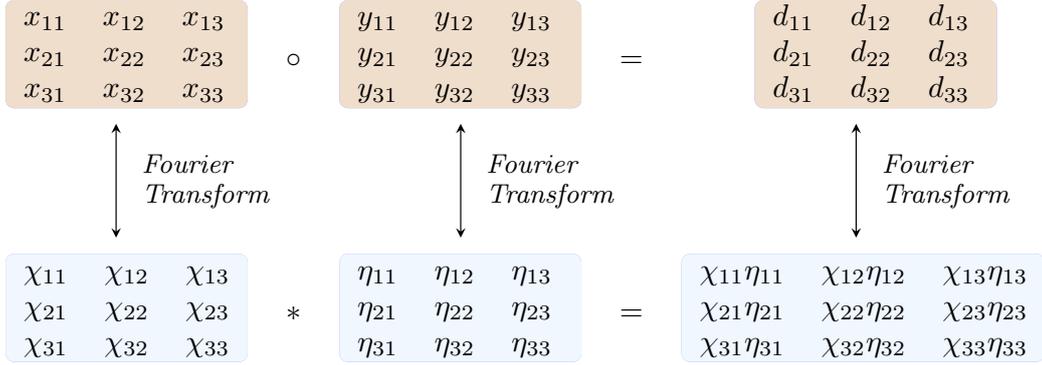


FIGURE 14. A diagram of t-scalar multiplication where the size of t-scalars is 3×3 .

$$\begin{array}{ccc}
 A_{TM} & & B_{TM} & & C_{TM} \\
 \left[\begin{array}{|c|c|c|} \hline 1 & 4 & 7 \\ \hline 2 & 5 & 8 \\ \hline 3 & 6 & 9 \\ \hline \end{array} & \begin{array}{|c|c|c|} \hline 19 & 22 & 25 \\ \hline 20 & 23 & 26 \\ \hline 21 & 24 & 27 \\ \hline \end{array} & \circ & \left[\begin{array}{|c|c|c|} \hline 37 & 40 & 43 \\ \hline 38 & 41 & 44 \\ \hline 39 & 42 & 45 \\ \hline \end{array} & \begin{array}{|c|c|c|} \hline 46 & 49 & 52 \\ \hline 47 & 50 & 53 \\ \hline 48 & 51 & 54 \\ \hline \end{array} \right] & = & \left[\begin{array}{|c|c|c|} \hline 12255 & 12255 & 12093 \\ \hline 12255 & 12255 & 12093 \\ \hline 12237 & 12237 & 12075 \\ \hline \end{array} & \begin{array}{|c|c|c|} \hline 19626 & 19626 & 19464 \\ \hline 19626 & 19626 & 19464 \\ \hline 19608 & 19608 & 19446 \\ \hline \end{array} \right]
 \end{array}$$

FIGURE 15. An illustrative example of t-matrix multiplication where the size of t-scalars is 3×3 .

$$\begin{array}{l}
 (X_{GT})_{\dots,1} = \left[\begin{array}{|c|c|c|} \hline 10 & 13 & 16 \\ \hline 11 & 14 & 17 \\ \hline 12 & 15 & 18 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 28 & 31 & 34 \\ \hline 29 & 32 & 35 \\ \hline 30 & 33 & 36 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 46 & 49 & 52 \\ \hline 47 & 50 & 53 \\ \hline 48 & 51 & 54 \\ \hline \end{array} \right] \quad (X_{GT})_{\dots,2} = \left[\begin{array}{|c|c|c|} \hline 64 & 67 & 70 \\ \hline 65 & 68 & 71 \\ \hline 66 & 69 & 72 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 82 & 85 & 88 \\ \hline 83 & 86 & 89 \\ \hline 84 & 87 & 90 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 98 & 95 & 92 \\ \hline 97 & 94 & 91 \\ \hline 96 & 93 & 90 \\ \hline \end{array} \right] \\
 X_{GT(1)} = \left[\begin{array}{|c|c|c|} \hline 10 & 13 & 16 \\ \hline 11 & 14 & 17 \\ \hline 12 & 15 & 18 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 28 & 31 & 34 \\ \hline 29 & 32 & 35 \\ \hline 30 & 33 & 36 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 46 & 49 & 52 \\ \hline 47 & 50 & 53 \\ \hline 48 & 51 & 54 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 64 & 67 & 70 \\ \hline 65 & 68 & 71 \\ \hline 66 & 69 & 72 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 82 & 85 & 88 \\ \hline 83 & 86 & 89 \\ \hline 84 & 87 & 90 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 98 & 95 & 92 \\ \hline 97 & 94 & 91 \\ \hline 96 & 93 & 90 \\ \hline \end{array} \right] \\
 X_{GT(2)} = \left[\begin{array}{|c|c|c|} \hline 10 & 13 & 16 \\ \hline 11 & 14 & 17 \\ \hline 12 & 15 & 18 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 19 & 22 & 25 \\ \hline 20 & 23 & 26 \\ \hline 21 & 24 & 27 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 64 & 67 & 70 \\ \hline 65 & 68 & 71 \\ \hline 66 & 69 & 72 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 73 & 76 & 79 \\ \hline 74 & 77 & 80 \\ \hline 75 & 78 & 81 \\ \hline \end{array} \right] \\
 X_{GT(3)} = \left[\begin{array}{|c|c|c|} \hline 10 & 13 & 16 \\ \hline 11 & 14 & 17 \\ \hline 12 & 15 & 18 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 19 & 22 & 25 \\ \hline 20 & 23 & 26 \\ \hline 21 & 24 & 27 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 28 & 31 & 34 \\ \hline 29 & 32 & 35 \\ \hline 30 & 33 & 36 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 37 & 40 & 43 \\ \hline 38 & 41 & 44 \\ \hline 39 & 42 & 45 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 46 & 49 & 52 \\ \hline 47 & 50 & 53 \\ \hline 48 & 51 & 54 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 64 & 67 & 70 \\ \hline 65 & 68 & 71 \\ \hline 66 & 69 & 72 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 73 & 76 & 79 \\ \hline 74 & 77 & 80 \\ \hline 75 & 78 & 81 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 82 & 85 & 88 \\ \hline 83 & 86 & 89 \\ \hline 84 & 87 & 90 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline 98 & 95 & 92 \\ \hline 97 & 94 & 91 \\ \hline 96 & 93 & 90 \\ \hline \end{array} \right]
 \end{array}$$

FIGURE 16. An illustrative example of a generalized tensor in $C^{2 \times 3 \times 2}$ and the mode- k ($k = 1, 2, 3$) flattened form of the generalized tensor.

$$\begin{aligned}
(X_{GT})_{::,1} &= \begin{bmatrix} \begin{matrix} 10 & 13 & 16 \\ 11 & 14 & 17 \\ 12 & 15 & 18 \end{matrix} & \begin{matrix} 28 & 31 & 34 \\ 29 & 32 & 35 \\ 30 & 33 & 36 \end{matrix} & \begin{matrix} 46 & 49 & 52 \\ 47 & 50 & 53 \\ 48 & 51 & 54 \end{matrix} \\ \begin{matrix} 19 & 22 & 25 \\ 20 & 23 & 26 \\ 21 & 24 & 27 \end{matrix} & \begin{matrix} 37 & 40 & 43 \\ 38 & 41 & 44 \\ 39 & 42 & 45 \end{matrix} & \begin{matrix} 55 & 58 & 61 \\ 56 & 59 & 62 \\ 57 & 60 & 63 \end{matrix} \end{bmatrix} \\
(X_{GT})_{::,2} &= \begin{bmatrix} \begin{matrix} 64 & 67 & 70 \\ 65 & 68 & 71 \\ 66 & 69 & 72 \end{matrix} & \begin{matrix} 82 & 85 & 88 \\ 83 & 86 & 89 \\ 84 & 87 & 90 \end{matrix} & \begin{matrix} 98 & 95 & 92 \\ 97 & 94 & 91 \\ 96 & 93 & 90 \end{matrix} \\ \begin{matrix} 73 & 76 & 79 \\ 74 & 77 & 80 \\ 75 & 78 & 81 \end{matrix} & \begin{matrix} 91 & 94 & 97 \\ 92 & 95 & 98 \\ 93 & 96 & 99 \end{matrix} & \begin{matrix} 89 & 86 & 83 \\ 88 & 85 & 82 \\ 87 & 84 & 81 \end{matrix} \end{bmatrix} \\
Y_{TM} &= \begin{bmatrix} \begin{matrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{matrix} & \begin{matrix} 19 & 22 & 25 \\ 20 & 23 & 26 \\ 21 & 24 & 27 \end{matrix} & \begin{matrix} 37 & 40 & 43 \\ 38 & 41 & 44 \\ 39 & 42 & 45 \end{matrix} \\ \begin{matrix} 10 & 13 & 16 \\ 11 & 14 & 17 \\ 12 & 15 & 18 \end{matrix} & \begin{matrix} 28 & 31 & 34 \\ 29 & 32 & 35 \\ 30 & 33 & 36 \end{matrix} & \begin{matrix} 46 & 49 & 52 \\ 47 & 50 & 53 \\ 48 & 51 & 54 \end{matrix} \end{bmatrix} \\
(M_{GT})_{::,1} &= \begin{bmatrix} \begin{matrix} 25794 & 25794 & 25551 \\ 25794 & 25794 & 25551 \\ 25767 & 25767 & 25524 \end{matrix} & \begin{matrix} 33570 & 33570 & 33327 \\ 33570 & 33570 & 33327 \\ 33543 & 33543 & 33300 \end{matrix} \\ \begin{matrix} 31383 & 31383 & 31140 \\ 31383 & 31383 & 31140 \\ 31356 & 31356 & 31113 \end{matrix} & \begin{matrix} 41346 & 41346 & 41103 \\ 41346 & 41346 & 41103 \\ 41319 & 41319 & 41076 \end{matrix} \end{bmatrix} \\
(M_{GT})_{::,2} &= \begin{bmatrix} \begin{matrix} 55578 & 55578 & 55497 \\ 55578 & 55578 & 55497 \\ 55569 & 55569 & 55488 \end{matrix} & \begin{matrix} 75666 & 75666 & 75585 \\ 75666 & 75666 & 75585 \\ 75657 & 75657 & 75576 \end{matrix} \\ \begin{matrix} 54525 & 54525 & 54444 \\ 54525 & 54525 & 54444 \\ 54516 & 54516 & 54435 \end{matrix} & \begin{matrix} 75342 & 75342 & 75261 \\ 75342 & 75342 & 75261 \\ 75333 & 75333 & 75252 \end{matrix} \end{bmatrix}
\end{aligned}$$

FIGURE 17. An illustrative example of mode-2 generalized tensor multiplication $M_{GT} = X_{GT} \circ_2 Y_{TM} \in C^{2 \times 2 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 2 \times 2}$ where $X_{GT} \in C^{2 \times 3 \times 2} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3 \times 2}$, $Y_{TM} \in C^{2 \times 3} \equiv \mathbb{C}^{3 \times 3 \times 2 \times 3}$.