SIAM J. Sci. Comput.
Vol. 34, No. 2, pp. A1027–A1052

A NEW TRUNCATION STRATEGY FOR THE HIGHER-ORDER SINGULAR VALUE DECOMPOSITION*

NICK VANNIEUWENHOVEN†, RAF VANDEBRIL†, AND KARL MEERBERGEN†

Abstract. We present an alternative strategy for truncating the higher-order singular value decomposition (T-HOSVD). An error expression for an approximate Tucker decomposition with orthogonal factor matrices is presented, leading us to propose a novel truncation strategy for the HOSVD, which we refer to as the sequentially truncated higher-order singular value decomposition (ST-HOSVD). This decomposition retains several favorable properties of the T-HOSVD, while reducing the number of operations required to compute the decomposition and practically always improving the approximation error. Three applications are presented, demonstrating the effectiveness of ST-HOSVD. In the first application, ST-HOSVD, T-HOSVD, and higher-order orthogonal iteration (HOOI) are employed to compress a database of images of faces. On average, the ST-HOSVD approximation was only 0.1% worse than the optimum computed by HOOI, while cutting the execution time by a factor of 20. In the second application, classification of handwritten digits, ST-HOSVD achieved a speedup factor of 50 over T-HOSVD during the training phase, and reduced the classification time and storage costs, while not significantly affecting the classification error. The third application demonstrates the effectiveness of ST-HOSVD in compressing results from a numerical simulation of a partial differential equation. In such problems, ST-HOSVD inevitably can greatly improve the running time. We present an example wherein the 2 hour 45 minute calculation of T-HOSVD was reduced to just over one minute by ST-HOSVD, representing a speedup factor of 133, while even improving the memory consumption.

Key words. tensor, sequentially truncated higher-order singular value decomposition, higher-order singular value decomposition, multilinear singular value decomposition, multilinear orthogonal projection

AMS subject classifications. 15A03, 15A69, 15A72, 65F99, 65Y20

DOI. 10.1137/110836067

1. Introduction.

In this paper, we seek a *good* multilinear rank-$(r_1, r_2, \ldots, r_d)$ approximation to the order-d tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$, using only numerical linear algebra tools. Formally, we are interested in an *approximate solution* to

$$\min_{\mathcal{B}} \|\mathcal{A} - \mathcal{B}\|_F^2 \tag{1.1}$$

with $\mathcal{B} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ restricted to be of rank $(r_1, r_2, \ldots, r_d)$ and $r_i \leq n_i$. Here, we deal with the *multilinear rank*, as defined by Hitchcock [22]. The above approximation problem is well-posed [13, Corollary 4.5], but does not exhibit a known closed solution. This motivated the search for reliable numerical algorithms for solving problem (1.1).

*Submitted to the journal's Methods and Algorithms for Scientific Computing section June 1, 2011; accepted for publication (in revised form) January 26, 2012; published electronically April 3, 2012. This work presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization) and was supported by the Interuniversity Attraction Poles Programme, initiated by the Belgian State Science Policy Office, and partially supported by the Research Council KU Leuven grants CoE OPTEC (optimization in engineering), OT/10/038 (multi-parameter model order reduction and its applications), OT/11/055 (spectral properties of perturbed normal matrices and their applications), and the Research Foundation—Flanders (Belgium) project G034212N (reestablishing smoothness for matrix manifold optimization via resolution of singularities). The scientific responsibility rests with its author(s).

http://www.siam.org/journals/sisc/34-2/83606.html

†Numerical Approximation and Linear Algebra Group, Department of Computer Science, KU Leuven, Leuven, Belgium (nick.vannieuwenhoven@cs.kleuven.be, raf.vandebril@cs.kleuven.be, karl.meerbergen@cs.kleuven.be). The first author's research was supported by a Ph.D. fellowship of the Research Foundation—Flanders (FWO).

Early approaches to this problem, as well as applications, originated in psychometrics. Tucker considered a decomposition of a third-order tensor [54, 55, 56], now known as the Tucker decomposition, into a set of three *factor matrices* and a third-order *core tensor*. Formally, we write

$$a_{i,j,k} = \sum_{i'=1}^{n_1} \sum_{j'=1}^{n_2} \sum_{k'=1}^{n_3} s_{i',j',k'} x_{i,i'} y_{j,j'} z_{k,k'} \tag{1.2}$$

for a third-order tensor $\mathcal{A} = [\![a_{i,j,k}]\!] \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Here $X = [\![x_{i,i'}]\!] \in \mathbb{R}^{n_1 \times n_1}$, $Y = [\![y_{j,j'}]\!] \in \mathbb{R}^{n_2 \times n_2}$, and $Z = [\![z_{k,k'}]\!] \in \mathbb{R}^{n_3 \times n_3}$ are the factor matrices, and $\mathcal{S} = [\![s_{i',j',k'}]\!] \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ is the core tensor. In [56], Tucker proposed algorithms for computing a Tucker decomposition (1.2). Of particular interest is the Tucker1 algorithm [56], which was later refined by De Lathauwer, De Moor, and Vandewalle [10]. The refined algorithm is called the *higher-order singular value decomposition* (HOSVD) [10, 56], and is particularly useful for constructing an approximate solution to problem (1.1). A rank-(r_1, r_2, r_3) approximation can be obtained simply by restricting the factor matrices X, Y, and Z to the first r_1, r_2, and r_3 columns, respectively, and by restricting the core tensor to $\mathcal{S}' = [\![s_{i,j,k}]\!]_{i,j,k=1}^{r_1,r_2,r_3}$. This truncated HOSVD (T-HOSVD) has traditionally been the method of choice for obtaining a cheap approximate solution to problem (1.1). In this paper, we investigate a new technique for truncating the HOSVD, which is cheaper still to compute and often improves the approximation error with respect to the T-HOSVD.

The problem we consider, in this paper, is different from the problem of determining the *best* approximation of a given multilinear rank to $\mathcal{A}$. This problem has been subjected to extensive research. In general, only locally optimal solutions [23] can be computed efficiently, and there is a wide variety of algorithms to accomplish this. Kroonenberg and de Leeuw were the first to propose such an (iterative) algorithm in [30]. It is the popular *alternating least squares* (ALS) algorithm, called TUCKALS3 in [30] but now more commonly known as the higher-order orthogonal iteration (HOOI) algorithm [2, 3, 11, 30]. HOOI alternates between the modes, computing the factor matrix in one mode by fixing the other factor matrices and solving a least squares problem. A new iteration is started if all factor matrices have been updated in this manner. This is repeated until convergence. Optimization-based algorithms to tackle problem (1.1) have been investigated only very recently and include Newton–Grassmann [15, 26], quasi–Newton–Grassmann [51], and trust-region [24, 25] methods on manifolds.

The HOSVD has been applied in numerous application domains [29], such as image processing [9, 32, 39, 61], pattern recognition [49, 50, 59, 60, 62], data mining [33, 34, 52, 53], signal processing [12, 19, 36, 37, 38, 45], psychometrics [54, 55, 56], chemometrics [5], and biomedicine [16, 40, 41]. Aside from its use in applications, the HOSVD is also of considerable theoretical importance. For instance, the T-HOSVD is used to initialize iterative algorithms to compute the best rank-1 approximation [7, 28, 63] or best approximation of a specified multilinear rank [11, 15, 24, 26, 51]. It is a building block for other tensor decompositions, such as the hierarchical Tucker decomposition [18, 21], the Tucker tree decomposition [43, 44], and the tensor-train decomposition [42]. Another use is dimensionality reduction, in order to compute the canonical polyadic decomposition, or CANDECOMP/PARAFAC [29], more efficiently [6].

In this paper, we present a new truncation strategy for the HOSVD. Our investigation was spurred by an interesting remark made by Andersson and Bro in [2].

In proposing techniques to improve the computational efficiency of the HOOI algorithm, they briefly point out a different initialization scheme. Instead of initializing the factor matrices by the T-HOSVD, they essentially propose initializing it with the sequentially truncated HOSVD—the decomposition we study in this paper. However, the decomposition was not formalized, nor were its properties investigated in [2].

The main contribution of this paper is a new truncation strategy for the HOSVD which always reduces the number of floating operations to compute the decomposition, and simultaneously improves the approximation error in many cases. Furthermore, the approximation error can be expressed in terms of the singular values that are computed as a byproduct of the approximation process. This allows for accurate numerical thresholding techniques without first computing the full HOSVD.

The paper is structured as follows. In the next section, we state some basic definitions. Special emphasis is put in section 3 on multilinear orthogonal projections. In section 4, we briefly present the HOSVD [10]. Thereafter, in section 5, we present an expression for the error of any approximate orthogonal Tucker decomposition. Based on this error expression, a new truncation strategy is proposed in section 6. The relationship between the error of the T-HOSVD and that of the ST-HOSVD is investigated in section 7. In section 8, we illustrate numerically that the ST-HOSVD often outperforms T-HOSVD in terms of approximation error and computation time. Finally, in section 9, we summarize our main results.

2. Preliminaries.

In this section, some necessary preliminaries are discussed. First, some notational conventions are established. Tensors are typeset in an uppercase calligraphic font ($\mathcal{A}, \mathcal{S}$), matrices as uppercase letters (A, U), vectors as boldface lowercase letters ($\mathbf{u}, \mathbf{v}$), and scalars as lowercase letters (a, b). I denotes the identity matrix of suitable dimensions. The scalar d denotes the order of the tensor. The scalar k denotes an integer between 1 and d. A multilinear orthogonal projector that projects along mode k is denoted as π_k; see also section 3. Projectors, matrices, and tensors which are typeset with a bar ($\bar{\pi}_1$, $\bar{U}_1$, $\bar{\mathcal{A}}$) are related to the T-HOSVD, with a hat ($\hat{\pi}_1$, $\hat{U}_1$, $\hat{\mathcal{A}}$) they are related to the ST-HOSVD, and with a breve ($\breve{\pi}_1$, $\breve{U}_1$, $\breve{\mathcal{A}}$) they are related to any orthogonal Tucker approximation (including T-HOSVD and ST-HOSVD). A permutation vector is denoted by square brackets, $\mathbf{p} = [1, 2, 3]$, and a set is denoted by curly brackets, $p = \{1, 2, 3\}$. The tensor product is denoted by $\circledast$, and the Kronecker product by $\otimes$.

Tensor algebra. This paragraph is based on [8, 13]; for more details, the reader is referred to those references. A tensor is an element of the tensor product of a set of vector spaces. We are interested in tensor products of real vector spaces. That is,

$$\mathcal{A} \in \mathbb{R}^{n_1} \circledast \mathbb{R}^{n_2} \circledast \cdots \circledast \mathbb{R}^{n_d}$$

is a tensor of *order* d over the real numbers. More practically, such an object can be represented as a d-array of numbers with respect to a given tensor basis. That is,

$$\mathcal{A} = [\![a_{i_1,i_2,\ldots,i_d}]\!]_{i_1,i_2,\ldots,i_d=1}^{n_1,n_2,\ldots,n_d} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}.$$

In the above, $\mathcal{A}$ is specified with respect to the *standard tensor basis of order* d,

$$\mathcal{E}_d = \{\mathbf{e}_{i_1} \circledast \mathbf{e}_{i_2} \circledast \cdots \circledast \mathbf{e}_{i_d}\}_{i_1,i_2,\ldots,i_d=1}^{n_1,n_2,\ldots,n_d},$$

where $\mathbf{e}_i$ is the ith standard basis vector of suitable dimension. We may thus write $\mathcal{A}$ as a multilinear combination of these basis vectors, as follows:

$$\mathcal{A} = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_d=1}^{n_d} a_{i_1,i_2,\ldots,i_d} \mathbf{e}_{i_1} \circledast \mathbf{e}_{i_2} \circledast \cdots \circledast \mathbf{e}_{i_d}. \tag{2.1}$$

In this paper, we will identify the tensor with the d-array representing its coordinates with respect to a suitable basis. A tensor may be multiplied in each of its modes with a (different) matrix. Let

$$\mathcal{A} = [\![a_{i_1,i_2,\ldots,i_d}]\!] \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d} \qquad \text{and} \qquad H^{(k)} = [\![h_{p,q}^{(k)}]\!] \in \mathbb{R}^{m_k \times n_k}.$$

Then, $\mathcal{A}$ may be transformed into a tensor $\mathcal{B} = [\![b_{j_1,j_2,\ldots,j_d}]\!] \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_d}$ via the *multilinear multiplication* of $\mathcal{A}$ by $H^{(k)}$, $k = 1, \ldots, d$,

$$b_{j_1,j_2,\ldots,j_d} = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_d=1}^{n_d} a_{i_1,i_2,\ldots,i_d} h_{j_1,i_1}^{(1)} h_{j_2,i_2}^{(2)} \cdots h_{j_d,i_d}^{(d)}.$$

We will write this more concisely [13] as

$$\mathcal{B} = (H^{(1)}, H^{(2)}, \ldots, H^{(d)}) \cdot \mathcal{A}.$$

Unfolding. Given a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$, a *mode-$k$ vector* $\mathbf{v}$ is defined as the vector that is obtained by fixing all indices of $\mathcal{A}$ and varying the mode-k index: $\mathbf{v} = \mathcal{A}_{i_1,\ldots,i_{k-1},:,i_{k+1},\ldots,i_d}$ with i_j a fixed value. We refer to the set of all mode-k vectors of $\mathcal{A}$ as the *mode-k vector space.* The *mode-k unfolding*, or matricization [15], of $\mathcal{A}$, denoted by $\mathcal{A}_{(k)}$, is an $n_k \times \Pi_{i \neq k} n_i$ matrix whose columns are all possible mode-k vectors. The specific order of the mode-k vectors within this unfolding is usually not important, as long as it is consistent. We assume the canonical order, as presented in [15]. The column space of $\mathcal{A}_{(k)}$ is the mode-k vector space; hence its name.

Multilinear rank. The multilinear rank [22] of a tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ is a d-tuple $(r_1, r_2, \ldots, r_d)$, wherein r_k is the dimension of the mode-k vector space. In other words, r_k is the column rank of $\mathcal{A}_{(k)}$.

Inner product and norm. The *Frobenius inner product* of two tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ is defined as

$$\langle \mathcal{A}, \mathcal{B} \rangle_F := \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} a_{i_1,\ldots,i_d} b_{i_1,\ldots,i_d},$$

and furthermore [15, Lemma 2.1], for every $1 \le k \le d$,

$$\langle \mathcal{A}, \mathcal{B} \rangle_F = \operatorname{trace}\left(\mathcal{B}_{(k)}^T \mathcal{A}_{(k)} \right) = \operatorname{trace}\left(\mathcal{A}_{(k)}^T \mathcal{B}_{(k)} \right) = \langle \mathcal{B}, \mathcal{A} \rangle_F, \tag{2.2}$$

where $\operatorname{trace}(A) := \sum_i a_{ii}$ is the trace of A. The induced *Frobenius norm* is

$$\|\mathcal{A}\|_F^2 := \langle \mathcal{A}, \mathcal{A} \rangle_F = \|\mathcal{A}_{(k)}\|_F^2,$$

for any mode-k unfolding of $\mathcal{A}$. The Frobenius norm for matrices is unitarily invariant. That is, if $A \in \mathbb{R}^{m \times n}$ and $U \in \mathbb{R}^{r \times m}$, with $r \ge m$, is a matrix with orthonormal columns, and $V \in \mathbb{R}^{s \times n}$, with $s \ge n$, is a matrix with orthonormal columns, then $\|UAV^T\|_F^2 = \|A\|_F^2$; see, e.g., [58].

Multilinear multiplication. In this paragraph, we assume that the dimensions of the matrices involved in the multilinear multiplications are compatible. Multilinear multiplication in one mode, say $1 \leq k \leq d$, with a matrix M can be interpreted as multiplying the mode-k vectors by M. That is, if M is at position k in the tuple, then $[(I, \ldots, I, M, I, \ldots, I) \cdot \mathcal{A}]_{(k)} = M\mathcal{A}_{(k)}$. In general, the unfolding of a multilinear multiplication is given by (see [10, 15])

$$[(M_1, M_2, \ldots, M_d) \cdot \mathcal{A}]_{(k)} = M_k \mathcal{A}_{(k)} \left(M_1 \otimes M_2 \otimes \cdots \otimes M_{k-1} \otimes M_{k+1} \otimes \cdots \otimes M_d\right)^T.$$

Two multilinear multiplications can be transformed into one, as follows [13]:

$$(L_1, L_2, \ldots, L_d) \cdot [(M_1, M_2, \ldots, M_d) \cdot \mathcal{A}] = (L_1 M_1, L_2 M_2, \ldots, L_d M_d) \cdot \mathcal{A}.$$

3. Multilinear orthogonal projections.

3. Multilinear orthogonal projections. Key to this paper is the use of multilinear orthogonal projections. An *orthogonal projector* [46, 47] is a linear transformation P that projects a vector $\mathbf{u} \in \mathbb{R}^n$ onto a vector space $\mathcal{U} \subseteq \mathbb{R}^n$ such that the residual $\mathbf{u} - P\mathbf{u}$ is orthogonal to $\mathcal{U}$. Such a projector can always be represented in matrix form as $P = UU^T$, assuming that the columns of U form an orthonormal basis for the vector space $\mathcal{U}$. De Silva and Lim [13] state that if ϕ_k is an orthogonal projector from the vector space $\mathcal{V}_k \subseteq \mathbb{R}^{n_k}$ onto $\mathcal{U}_k \subseteq \mathcal{V}_k$, then $\Phi = (\phi_1, \phi_2, \ldots, \phi_d)$ is a multilinear orthogonal projection from the tensor space $\mathcal{V} := \mathcal{V}_1 \circledast \mathcal{V}_2 \circledast \cdots \circledast \mathcal{V}_d$ onto the tensor subspace $\mathcal{U}_1 \circledast \mathcal{U}_2 \circledast \cdots \circledast \mathcal{U}_d \subseteq \mathcal{V}$. In this paper, we deal with an orthogonal projector from $\mathbb{R}^{n_1} \circledast \cdots \circledast \mathbb{R}^{n_{k-1}} \circledast \mathbb{R}^{n_k} \circledast \mathbb{R}^{n_{k+1}} \circledast \cdots \circledast \mathbb{R}^{n_d}$ onto the subspace $\mathbb{R}^{n_1} \circledast \cdots \circledast \mathbb{R}^{n_{k-1}} \circledast \mathcal{U}_k \circledast \mathbb{R}^{n_{k+1}} \circledast \cdots \circledast \mathbb{R}^{n_d}$ exclusively. This multilinear orthogonal projection is given by

$$\pi_k \mathcal{A} := (\underbrace{I, \ldots, I}_{\substack{k-1 \\ \text{matrices}}}, U_k U_k^T, I, \ldots, I) \cdot \mathcal{A} \qquad \text{with} \qquad \mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}, \tag{3.1}$$

where we assume that the columns of U_k form an orthonormal basis of the vector space $\mathcal{U}_k$. The subscript of the projector π_k indicates that it projects orthogonally along mode k. The projector satisfies the following properties. Every projector π_k is idempotent, $\pi_k \pi_k \mathcal{A} = \pi_k \mathcal{A}$, and any two projectors commute, $\pi_i \pi_j \mathcal{A} = \pi_j \pi_i \mathcal{A}$. The orthogonal complement of π_k can be characterized explicitly by

$$(1 - \pi_k)\mathcal{A} = (I, \ldots, I, I - U_k U_k^T, I, \ldots, I) \cdot \mathcal{A},$$

due to the multilinearity of the multilinear multiplication [13, eq. (2.9)]. Finally, the projector is orthogonal with respect to the Frobenius norm [13, eq. (2.20)],

$$\|\mathcal{A}\|_F^2 = \|\pi_k \mathcal{A}\|_F^2 + \|(1 - \pi_k)\mathcal{A}\|_F^2. \tag{3.2}$$

The above projector is used extensively in this paper.

4. T-HOSVD. De Lathauwer, De Moor, and Vandewalle proved the following theorem in [10, section 3].

Theorem 4.1 (HOSVD [10]). *Every tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ admits a* higher-order singular value decomposition:

$$\mathcal{A} = (U_1, U_2, \ldots, U_d) \cdot \mathcal{S}, \tag{4.1}$$

where the factor matrix U_k *is an orthogonal* $n_k \times n_k$ *matrix, obtained from the SVD of the mode-*k *unfolding of* $\mathcal{A}$*,*

$$\mathcal{A}_{(k)} = U_k \Sigma_k V_k^T, \tag{4.2}$$

and the core tensor $\mathcal{S} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ *can be obtained from*

$$\mathcal{S} = (U_1^T, U_2^T, \ldots, U_d^T) \cdot \mathcal{A}.$$

The HOSVD can be employed to construct a low multilinear rank approximation to a tensor. Suppose that we want to approximate $\mathcal{A}$ by a rank-$(r_1, r_2, \ldots, r_d)$ tensor $\bar{\mathcal{A}}$, with $r_k \leq n_k$ for all $1 \leq k \leq d$. The factor matrix $\bar{U}_k$ of the truncated HOSVD is obtained from a truncated SVD of the mode-k unfolding of the tensor [10, 29],

$$\mathcal{A}_{(k)} = U_k \Sigma_k V_k^T = \begin{bmatrix} \bar{U}_k & \tilde{U}_k \end{bmatrix} \begin{bmatrix} \bar{\Sigma}_k & \\ & \tilde{\Sigma}_k \end{bmatrix} \begin{bmatrix} \bar{V}_k^T \\ \tilde{V}_k^T \end{bmatrix}, \qquad \text{with} \quad \bar{U}_k \in \mathbb{R}^{n_k \times r_k}. \tag{4.3}$$

The approximation is then obtained by an *orthogonal projection* onto the tensor basis, represented by these factor matrices. That is,

$$\bar{\mathcal{A}} := \bar{\pi}_1 \bar{\pi}_2 \cdots \bar{\pi}_d \mathcal{A} := (\bar{U}_1 \bar{U}_1^T, \bar{U}_2 \bar{U}_2^T, \ldots, \bar{U}_d \bar{U}_d^T) \cdot \mathcal{A} =: (\bar{U}_1, \bar{U}_2, \ldots, \bar{U}_d) \cdot \bar{\mathcal{S}} \approx \mathcal{A},$$

wherein the *truncated core tensor* is defined as

$$(\bar{U}_1^T, \bar{U}_2^T, \ldots, \bar{U}_d^T) \cdot \mathcal{A} =: \bar{\mathcal{S}} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_d}.$$

In the remainder, we will always denote the T-HOSVD projector onto mode k by $\bar{\pi}_k \mathcal{A} := (I, \ldots, I, \bar{U}_k \bar{U}_k^T, I, \ldots, I) \cdot \mathcal{A}$.

5. Error of a truncated orthogonal Tucker decomposition.

In this section, we present an explicit formula for the error of an approximate Tucker decomposition with orthogonal factor matrices. This formula provides insights into the structure of the optimization problem. In the next section, we propose a new greedy optimization algorithm, based on the error expansion presented in the next theorem.

It is well known (see, e.g., [29]) that problem (1.1) may be rewritten as

$$\min_{\substack{\breve{\mathcal{S}} \in \mathbb{R}^{r_1 \times \cdots \times r_d} \\ \breve{U}_i \in \mathbb{R}^{n_i \times r_i}}} \|\mathcal{A} - (\breve{U}_1, \breve{U}_2, \ldots, \breve{U}_d) \cdot \breve{\mathcal{S}}\|_F,$$

with $\breve{U}_i$, $i = 1, 2, \ldots, d$, a matrix with orthonormal columns. Furthermore, if the factor matrices $\{\breve{U}_k\}_k$ are fixed, then the core tensor $\breve{\mathcal{S}}$ that minimizes the approximation error is given by $\breve{\mathcal{S}} = (\breve{U}_1^T, \breve{U}_2^T, \ldots, \breve{U}_d^T) \cdot \mathcal{A}$, as proved in [11]. Therefore, given fixed factor matrices, the optimal approximation to $\mathcal{A}$ is obtained by *a multilinear orthogonal projection onto the tensor basis spanned by the columns of the factor matrices.* The error of this optimal approximation is investigated in the next theorem.

THEOREM 5.1 (error of a truncated orthogonal Tucker decomposition). *Let* $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$. *Let* $\mathcal{A}$ *be approximated by*

$$\breve{\mathcal{A}} := \breve{\pi}_1 \breve{\pi}_2 \cdots \breve{\pi}_d \mathcal{A} := (\breve{U}_1 \breve{U}_1^T, \breve{U}_2 \breve{U}_2^T, \ldots, \breve{U}_d \breve{U}_d^T) \cdot \mathcal{A} \approx \mathcal{A},$$

in which the factor matrices $\breve{U}_k \in \mathbb{R}^{n_k \times r_k}$ *have orthonormal columns. The approximation error is then given, for any permutation* $\mathbf{p}$ *of* $\{1, 2, \ldots, d\}$, *by*

$$\|\mathcal{A} - \breve{\mathcal{A}}\|_F^2 = \|\breve{\pi}_{p_1}^{\perp} \mathcal{A}\|_F^2 + \|\breve{\pi}_{p_2}^{\perp} \breve{\pi}_{p_1} \mathcal{A}\|_F^2 + \|\breve{\pi}_{p_3}^{\perp} \breve{\pi}_{p_1} \breve{\pi}_{p_2} \mathcal{A}\|_F^2 + \cdots + \|\breve{\pi}_{p_d}^{\perp} \breve{\pi}_{p_1} \cdots \breve{\pi}_{p_{d-1}} \mathcal{A}\|_F^2,$$

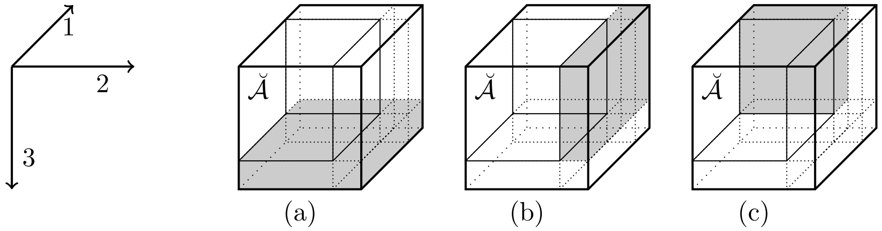


FIG. 5.1. *Theorem* 5.1 *states that the squared approximation error of the truncated orthogonal Tucker decomposition is given by the sum of the squared norms of the areas shaded in grey in the three tensor representations. This particular shading of the areas corresponds to the permutation* $[3, 2, 1]$.

and it is bounded by

$$\|\mathcal{A} - \breve{\mathcal{A}}\|_F^2 \leq \sum_{k=1}^{d} \|\breve{\pi}_{p_k}^{\perp}\mathcal{A}\|_F^2, \tag{5.1}$$

wherein $\breve{\pi}_k^{\perp} := 1 - \breve{\pi}_k$.

Proof. Assume w.l.o.g. that $\mathbf{p} = [1, 2, \ldots, d]$. We introduce a telescoping sum:

$$\begin{aligned}\mathcal{A} - \breve{\mathcal{A}} &= (\mathcal{A} - \breve{\pi}_1\mathcal{A}) + (\breve{\pi}_1\mathcal{A} - \breve{\pi}_2\breve{\pi}_1\mathcal{A}) + \cdots + (\breve{\pi}_{d-1}\cdots\breve{\pi}_1\mathcal{A} - \breve{\pi}_d\cdots\breve{\pi}_1\mathcal{A}) \\ &= \breve{\pi}_1^{\perp}\mathcal{A} + \breve{\pi}_2^{\perp}\breve{\pi}_1\mathcal{A} + \breve{\pi}_3^{\perp}\breve{\pi}_1\breve{\pi}_2\mathcal{A} + \cdots + \breve{\pi}_d^{\perp}\breve{\pi}_1\cdots\breve{\pi}_{d-1}\mathcal{A}. \end{aligned} \tag{5.2}$$

Consider any two distinct terms in the above. Let, for $i < j$,

$$\begin{aligned}\breve{\mathcal{B}}_{(i)} &:= (I - \breve{U}_i\breve{U}_i^T)\mathcal{A}_{(i)}(\breve{U}_1\breve{U}_1^T \otimes \cdots \otimes \breve{U}_{i-1}\breve{U}_{i-1}^T \otimes I \otimes \cdots \otimes I), \\ \breve{\mathcal{C}}_{(i)} &:= \breve{U}_i\breve{U}_i^T\mathcal{A}_{(i)}(\breve{U}_1\breve{U}_1^T \otimes \cdots \otimes \breve{U}_{i-1}\breve{U}_{i-1}^T \otimes \breve{U}_{i+1}\breve{U}_{i+1}^T \otimes \cdots \otimes \breve{U}_{j-1}\breve{U}_{j-1}^T \\ &\qquad \otimes\ I - \breve{U}_j\breve{U}_j^T \otimes I \otimes \cdots \otimes I);\end{aligned}$$

then we notice that $\breve{\mathcal{C}}_{(i)}^T\breve{\mathcal{B}}_{(i)} = 0$, because $\breve{U}_i\breve{U}_i^T$ is a projector. From (2.2),

$$\operatorname{trace}\left(\breve{\mathcal{C}}_{(i)}^T\breve{\mathcal{B}}_{(i)}\right) = \langle \breve{\pi}_i^{\perp}\breve{\pi}_1\cdots\breve{\pi}_{i-1}\mathcal{A},\ \breve{\pi}_j^{\perp}\breve{\pi}_1\cdots\breve{\pi}_i\cdots\breve{\pi}_{j-1}\mathcal{A}\rangle_F = 0.$$

This entails that $\breve{\pi}_i^{\perp}\breve{\pi}_1\cdots\breve{\pi}_{i-1}\mathcal{A} \perp \breve{\pi}_j^{\perp}\breve{\pi}_1\cdots\breve{\pi}_i\cdots\breve{\pi}_{j-1}\mathcal{A}$; i.e., these projected tensors are orthogonal with respect to the Frobenius norm. As the above holds for any $i < j$ and orthogonality is reflexive, all the terms in (5.2) are orthogonal with respect to one another, in the Frobenius norm. That completes the first part of the proof. The second part follows readily from the observation that an orthogonal projection onto a subspace can only decrease the Frobenius norm, due to (3.2). $\square$

In Figure 5.1 we visualize the above theorem for a third-order tensor and for the permutation $\mathbf{p} = [3, 2, 1]$. The cube is partitioned into octants. The shaded area in Figure 5.1(a) corresponds to $\pi_3^{\perp}\mathcal{A}$, in Figure 5.1(b) it corresponds to $\pi_2^{\perp}\pi_3\mathcal{A}$, and in Figure 5.1(c) to $\pi_1^{\perp}\pi_3\pi_2\mathcal{A}$. Other permutations result in different octants to be summed, but the resulting error is clearly the same.

The following error bounds of the T-HOSVD are an immediate corollary of Theorem 5.1. The upper bound was already stated in [10, Property 10]. Here, a new elegant proof based on the previous theorem is presented.[1]

[1] The proof in [10] unfortunately contains an error, as its second equality does not hold.

Corollary 5.2 (T-HOSVD error bounds). *Let $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, and let $\bar{\mathcal{A}}$ be the rank-$(r_1, \ldots, r_d)$ T-HOSVD of $\mathcal{A}$. Let the SVD of $\mathcal{A}_{(k)}$ be as in* (4.3)*. The error of the T-HOSVD approximation $\bar{\mathcal{A}}$ to $\mathcal{A}$ is then bounded by*

$$\max_k \|\tilde{\Sigma}_k\|_F^2 \leq \|\mathcal{A} - \bar{\mathcal{A}}\|_F^2 \leq \sum_{k=1}^{d} \|\tilde{\Sigma}_k\|_F^2. \tag{5.3}$$

Proof. The lower bound is proved by noticing that Theorem 5.1 holds for any permutation $\mathbf{p}$ of $\{1, 2, \ldots d\}$, the independence of the error on the permutation $\mathbf{p}$, and the positivity of the terms. The upper bound follows readily from (5.1) and the definition of the T-HOSVD factor matrices in (4.3). □

The T-HOSVD can be interpreted as an algorithm that minimizes the upper bound in Theorem 5.1, providing a strong rationale for the T-HOSVD approximation. Indeed, minimizing the upper bound yields

$$\min_{\pi_1,\ldots,\pi_d} \|\mathcal{A} - \pi_1 \cdots \pi_d \mathcal{A}\|_F^2 \leq \min_{\pi_1,\ldots,\pi_d} \sum_{k=1}^{d} \|\pi_k^\perp \mathcal{A}\|_F^2 = \sum_{k=1}^{d} \min_{\pi_k} \|\pi_k^\perp \mathcal{A}\|_F^2, \tag{5.4}$$

where the last equality follows from noticing that every term is minimized over a different projector, and that the T-HOSVD projectors are determined independently from one another. The solution to this minimization problem is given by choosing π_k to project onto the dominant subspace of the mode-k vector space, as in the T-HOSVD. Coincidentally, this also minimizes the lower bound in Corollary 5.2.

6. Sequentially truncated HOSVD.

In this section, we propose an alternative truncation strategy for the HOSVD. Contrary to the T-HOSVD, the order in which the modes are processed is relevant and leads to different approximations. The order in which modes are processed is denoted by a sequence $\mathbf{p}$. Throughout this section, we present our results only for the processing order $\mathbf{p} = [1, 2, \ldots, d]$, as this significantly simplifies the notation. It should be stressed, however, that many of the results depend on the permutation $\mathbf{p}$. For instance, the approximation error of our algorithm depends on the order in which the modes are processed.

6.1. Definition.

Optimization problem (1.1) can be expressed as

$$\begin{aligned}
&\min_{\pi_1,\ldots,\pi_d} \|\mathcal{A} - \pi_1\pi_2 \cdots \pi_d \mathcal{A}\|_F^2 \\
&\qquad = \min_{\pi_1,\ldots,\pi_d} \left(\|\pi_1^\perp \mathcal{A}\|_F^2 + \|\pi_2^\perp \pi_1 \mathcal{A}\|_F^2 + \cdots + \|\pi_d^\perp \pi_1 \cdots \pi_{d-1} \mathcal{A}\|_F^2 \right) \\
&\qquad = \min_{\pi_1} \left[\|\pi_1^\perp \mathcal{A}\|_F^2 + \min_{\pi_2} \left[\|\pi_2^\perp \pi_1 \mathcal{A}\|_F^2 + \min_{\pi_3} \left[\cdots + \min_{\pi_d} \|\pi_d^\perp \pi_1 \cdots \pi_{d-1} \mathcal{A}\|_F^2 \right] \right] \right],
\end{aligned}$$

by applying Theorem 5.1. Consider the minimization over π_1. It is not unreasonable to assume that the final error depends more strongly on the term $\|\pi_1^\perp \mathcal{A}\|_F^2$ than on the other terms. The subsequent terms will be minimized over other projectors, thereby diminishing the importance of a single projector. Therefore, $\hat{\pi}_1 := \arg\min_{\pi_1} \|\pi_1^\perp \mathcal{A}\|_F^2$ might be a good approximation to the optimal projector. By repeating the above argument for the projector π_2, and then π_3, and so on, we arrive at

$$\begin{aligned}
\min_{\pi_1,\ldots,\pi_d} \|\mathcal{A} - \pi_1\pi_2 \cdots \pi_d \mathcal{A}\|_F^2 &\leq \|\hat{\pi}_1^\perp \mathcal{A}\|_F^2 + \|\hat{\pi}_2^\perp \hat{\pi}_1 \mathcal{A}\|_F^2 + \cdots + \|\hat{\pi}_d^\perp \hat{\pi}_1 \cdots \hat{\pi}_{d-1} \mathcal{A}\|_F^2 \\
&= \|\mathcal{A} - \hat{\pi}_1 \cdots \hat{\pi}_d \mathcal{A}\|_F^2. \qquad (6.1)
\end{aligned}$$

Herein, the hat-projectors are defined recursively by

$$\hat{\pi}_k := \arg\min_{\pi_k} \|\pi_k^{\perp}\hat{\pi}_1\cdots\hat{\pi}_{k-1}\mathcal{A}\|_F^2 = \arg\max_{U_k\in\mathbb{R}^{n_k\times r_k}} \|U_kU_k^T\left[\hat{\pi}_1\cdots\hat{\pi}_{k-1}\mathcal{A}\right]_{(k)}\|_F^2. \tag{6.2}$$

Every processing order $\mathbf{p}$ of the modes yields a different optimization problem, and solution, of the above form. The truncation strategy that we propose consists of computing the solution to optimization problem (6.2). The solution can be obtained, for some $1 \leq k \leq d$, from a truncated SVD of the mode-k unfolding of $\hat{\pi}_1\cdots\hat{\pi}_{k-1}\mathcal{A}$. It is a tensor of the same size as $\mathcal{A}$, namely $n_1 \times n_2 \times \cdots \times n_d$. However,

$$\begin{aligned}\hat{U}_k &= \arg\max_{U_k\in\mathbb{R}^{n_k\times r_k}} \|U_kU_k^T\mathcal{A}_{(k)}(\hat{U}_1\hat{U}_1^T\otimes\hat{U}_2\hat{U}_2^T\otimes\cdots\otimes\hat{U}_{k-1}\hat{U}_{k-1}^T\otimes I\otimes\cdots\otimes I)^T\|_F^2\\ &= \arg\max_{U_k\in\mathbb{R}^{n_k\times r_k}} \|U_kU_k^T\mathcal{A}_{(k)}(\hat{U}_1\otimes\hat{U}_2\otimes\cdots\otimes\hat{U}_{k-1}\otimes I\otimes\cdots\otimes I)\|_F^2.\end{aligned}$$

Consequently, it is possible to compute the optimal projector by means of a truncated SVD of $(\hat{U}_1^T,\ldots,\hat{U}_{k-1}^T, I,\ldots,I)\cdot\mathcal{A}$, which is a tensor of size $r_1\times r_2\times\cdots\times r_{k-1}\times n_k\times\cdots\times n_d$. Whenever the tensor is strongly truncated, r_i is much smaller than n_i, thereby significantly improving the computational performance. The above is summarized in Algorithm 1, which computes the solution of optimization problem (6.2).[2]

Algorithm 1. Computing the sequentially truncated HOSVD.

```
input  : Tensor $\mathcal{A}$, truncation rank $(r_1, r_2, \ldots, r_d)$, and processing order $\mathbf{p}$.
output: Truncated core tensor $\hat{\mathcal{S}}$ and factor matrices $\{\hat{U}_k\}_k$.
$\hat{\mathcal{S}} \leftarrow \mathcal{A}$
for $k \leftarrow p_1, p_2, \ldots, p_d$ do
    % Compute the compact singular value decomposition of $\hat{\mathcal{S}}_{(k)}$
    $\hat{\mathcal{S}}_{(k)} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$, with $U_1 \in \mathbb{R}^{n_k\times r_k}$
    $\hat{U}_k \leftarrow U_1$
    $\hat{\mathcal{S}}_{(k)} \leftarrow \Sigma_1 V_1^T$
end
```

Note that, in Algorithm 1, we compute the compact[3] SVD, not the full SVD. The compact SVD is then truncated to the appropriate rank. Clearly, it is possible to replace this with an iterative algorithm that computes the desired singular triplets. In practice, the latter approach can be more efficient if the multilinear rank with which to approximate the tensor is very small.

Definition 6.1 (ST-HOSVD). *A rank-*$(r_1,\ldots,r_d)$ sequentially truncated higher-order singular value decomposition (ST-HOSVD) *of a tensor* $\mathcal{A}\in\mathbb{R}^{n_1\times\cdots\times n_d}$*, corresponding to the processing order* $\mathbf{p} = [1,2,\ldots,d]$*, is an approximation of the form*

$$\mathcal{A} \approx (\hat{U}_1,\hat{U}_2,\ldots,\hat{U}_d)\cdot\hat{\mathcal{S}} =: \hat{\mathcal{A}}_{\mathbf{p}} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_d},$$

whose truncated core tensor *is defined as*

$$(\hat{U}_1^T,\hat{U}_2^T,\ldots,\hat{U}_d^T)\cdot\mathcal{A} =: \hat{\mathcal{S}} \in \mathbb{R}^{r_1\times r_2\times\cdots\times r_d},$$

[2] A MATLAB implementation using the Tensor Toolbox [4] can be found in [58].

[3] The compact SVD of $A\in\mathbb{R}^{n\times m}$ is a decomposition such that $A = USV^T$ and $U\in\mathbb{R}^{n\times r}$, $S\in\mathbb{R}^{r\times r}$, and $V\in\mathbb{R}^{m\times r}$, with r the rank of A.

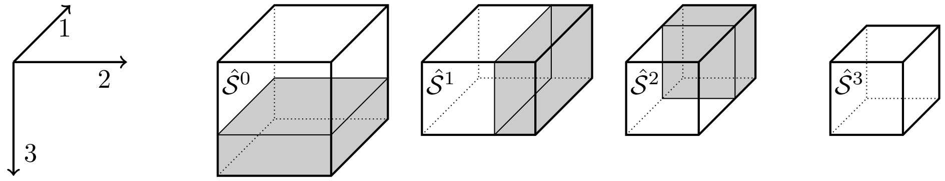


FIG. 6.1. *A graphical depiction of the sequential truncation of a third-order tensor, corresponding to the processing order* $[3, 2, 1]$ *of the modes.*

and where every factor matrix $\hat{U}_k \in \mathbb{R}^{n_k \times r_k}$ *has orthonormal columns. In terms of orthogonal multilinear projectors, one writes*

$$\hat{\mathcal{A}}_{\mathbf{p}} := \hat{\pi}_1 \hat{\pi}_2 \cdots \hat{\pi}_d \mathcal{A} := (\hat{U}_1 \hat{U}_1^T, \hat{U}_2 \hat{U}_2^T, \ldots, \hat{U}_d \hat{U}_d^T) \cdot \mathcal{A}.$$

*The k*th partially truncated core tensor *is defined as*

$$(6.3) \qquad (\hat{U}_1^T, \hat{U}_2^T, \ldots, \hat{U}_k^T, I, \ldots, I) \cdot \mathcal{A} =: \hat{\mathcal{S}}^k \in \mathbb{R}^{r_1 \times \cdots \times r_k \times n_{k+1} \times \cdots \times n_d},$$

with $\hat{\mathcal{S}}^0 := \mathcal{A}$ *and* $\hat{\mathcal{S}}^d = \hat{\mathcal{S}}$. *The* rank-$(r_1, \ldots, r_k, n_{k+1}, \ldots, n_d)$ partial approximation to $\mathcal{A}$ *is defined as*

$$(\hat{U}_1, \hat{U}_2, \ldots, \hat{U}_k, I, \ldots, I) \cdot \hat{\mathcal{S}}^k =: \hat{\mathcal{A}}^k \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d},$$

with $\hat{\mathcal{A}}^0 := \mathcal{A}$ *and* $\hat{\mathcal{A}}^d = \hat{\mathcal{A}}$.

The factor matrix $\hat{U}_k$, $1 \le k \le d$, *is the matrix of the* r_k *dominant left singular vectors of the mode-k vector space of* $\hat{\mathcal{S}}^{k-1}$. *It is obtained from the rank* r_k *truncated singular value decomposition of the* $(k-1)$*th partially truncated core tensor, as follows:*

$$(6.4) \qquad \hat{\mathcal{S}}^{k-1}_{(k)} = U_k \Sigma_k V_k^T = \begin{bmatrix} \hat{U}_k & \tilde{U}_k \end{bmatrix} \begin{bmatrix} \hat{\Sigma}_k & \\ & \tilde{\Sigma}_k \end{bmatrix} \begin{bmatrix} \hat{V}_k^T \\ \tilde{V}_k^T \end{bmatrix},$$

wherein $\hat{\Sigma}_k = \operatorname{diag}(\hat{\sigma}_{k,1}, \hat{\sigma}_{k,2}, \ldots, \hat{\sigma}_{k,r_k})$ *and* $\tilde{\Sigma}_k = \operatorname{diag}(\tilde{\sigma}_{k,r_k+1}, \tilde{\sigma}_{k,r_k+2}, \ldots, \tilde{\sigma}_{k,n_k})$.

The ST-HOSVD computes *a sequence* of approximations, $\hat{\mathcal{A}}^0, \hat{\mathcal{A}}^1, \ldots, \hat{\mathcal{A}}^d$, such that the multilinear rank of $\hat{\mathcal{A}}^k$ equals, in the first k modes, the desired dimension of the corresponding vector space. We term our approach "sequential" in that the mode-k projector depends on the previously computed projectors. In the remainder, we denote the ST-HOSVD projector onto mode k by $\hat{\pi}_k \mathcal{A} := (I, \ldots, I, \hat{U}_k \hat{U}_k^T, I, \ldots, I) \cdot \mathcal{A}$.

In Figure 6.1, we illustrate the operation of Algorithm 1. It represents the truncation of a third-order tensor $\mathcal{A} = \hat{\mathcal{S}}^0$ to the ST-HOSVD core tensor $\hat{\mathcal{S}} = \hat{\mathcal{S}}^3$, whereby the modes are processed in the order $\mathbf{p} = [3, 2, 1]$. First, the truncated SVD of the mode-3 vector space is computed. By projecting onto the span of the matrix of left singular vectors $\hat{U}_3$, the "energy" in the tensors is reordered. The Frobenius norm of the area shaded in gray is $\|\tilde{\Sigma}_3\|_F^2$, whereas the white area has a Frobenius norm equal to $\|\hat{\Sigma}_3\|_F^2$. By projecting onto the dominant subspace of the mode-3 vector space, we retain only the nonshaded area of $\hat{\mathcal{S}}^0$, resulting in the approximation $\hat{\mathcal{S}}^1$. In the next step, mode 2 is processed. The SVD is computed, and by projecting onto the space spanned by the left singular vectors, the energy is reordered. To obtain the next approximation, the shaded area of $\hat{\mathcal{S}}^1$ is set to zero. The procedure proceeds analogously in the last step. In the end, the ST-HOSVD core tensor is obtained.

By comparing (6.1) and (6.2) with (5.4), we arrive at an interesting relationship between the minimization problem solved by T-HOSVD and ST-HOSVD:

$$\min_{\pi_1,\ldots,\pi_d} \|\mathcal{A} - \pi_1\pi_2\cdots\pi_d\mathcal{A}\|_F^2 \leq \sum_{i=1}^{d} \min_{\pi_i} \|\pi_i^\perp \hat{\pi}_1 \cdots \hat{\pi}_{i-1}\mathcal{A}\|_F^2 \leq \sum_{i=1}^{d} \min_{\pi_i} \|\pi_i^\perp \mathcal{A}\|_F^2.$$

The last inequality holds because multilinear orthogonal projections only decrease the Frobenius norm. The optimization problem that is actually solved by the T-HOSVD can thus be considered as a simplification of optimization problem (6.2). This provides a strong rationale for the ST-HOSVD approximation. Furthermore, without truncating in every mode, the ST-HOSVD and HOSVD result in the same decomposition.

Theorem 6.2 (an alternative HOSVD algorithm). *A rank-$(n_1, n_2, \ldots, n_d)$ ST-HOSVD of $\mathcal{A} \in \mathbb{R}^{n_1\times n_2\times\cdots\times n_d}$ coincides with the HOSVD in Theorem* 4.1.

Proof. Consider any $1 \leq k \leq d$. From (6.3), (6.4), and from the observation that, in (6.4), $\hat{U}_k = U_k$ in the absence of truncation, we derive that

$$\mathcal{S}_{(k)}^{k-1} = I\mathcal{A}_{(k)}(U_1 \otimes \cdots \otimes U_{k-1} \otimes I \otimes \cdots \otimes I) = U_k\Sigma_k V_k^T,$$

and, because the Kronecker product preserves orthogonality [57],

$$\mathcal{A}_{(k)} = U_k\Sigma_k V_k^T (U_1 \otimes \cdots \otimes U_{k-1} \otimes I \otimes \cdots \otimes I)^T.$$

We note that the above is *the* singular value decomposition of $\mathcal{A}_{(k)}$. The left singular vectors, and thus the entire decomposition, must coincide with the HOSVD. Theorem 2 in [10] can then be applied to complete the proof. □

Consequently, the ST-HOSVD inherits the properties of the HOSVD in the absence of truncation. In particular, the ST-HOSVD of an order-2 tensor reduces to the matrix SVD, even in presence of truncation, which is not hard to prove.

6.2. Operation count.

One of the main advantages of the ST-HOSVD over the T-HOSVD algorithm is that it requires fewer floating point operations to compute the approximation. We restrict our estimates to cubic tensors. The generalization to arbitrary ranks and mode lengths is straightforward.

Property 6.3. *Let $\mathcal{A}$ be an order-d cubic tensor of size $n \times n \times \cdots \times n$. Let $\mathcal{A}$ be truncated to rank $(r, r, \ldots, r)$ by the ST-HOSVD (respectively, T-HOSVD). Assume an $O(m^2n)$ algorithm to compute the SVD of an $m \times n$ matrix, $m \leq n$. Then, the ST-HOSVD (respectively, T-HOSVD) requires*

$$O\left(\sum_{k=1}^{d} r^{k-1}n^{d-k+2} + \sum_{k=1}^{d} r^k n^{d-k}\right) \qquad \textit{and} \qquad O\left(dn^{d+1} + \sum_{k=1}^{d} r^k n^{d-k+1}\right)$$

operations to compute the approximation.

Proof. In the case of the ST-HOSVD, the SVD of an $n \times r^{k-1}n^{d-k}$ matrix should be computed in every mode, leading to the first sum. The next partially truncated core tensor can be obtained simply by scaling the right singular vectors with the corresponding singular values. This explains the last sum in the operation count.

The first term in the T-HOSVD estimate is due to the SVD, in every mode, of an $n \times n^{d-1}$ matrix to compute the factor matrices. The sum is required to compute the core tensor, by means of d matrix multiplications. □

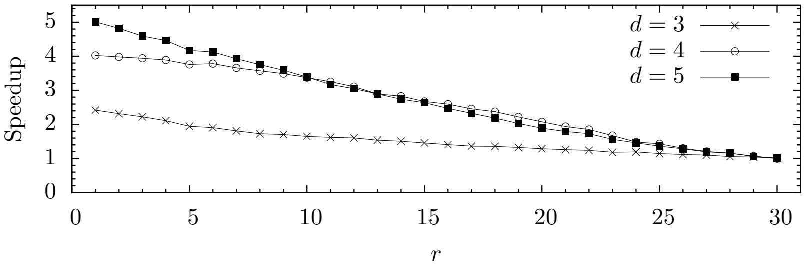


Fig. 6.2. *Speedup of ST-HOSVD over T-HOSVD for an order-d tensor of size* $30 \times 30 \times \cdots \times 30$ *which is truncated to rank* $(r, r, \ldots, r)$.

Computing the T-HOSVD can be much more expensive than an ST-HOSVD if $r \ll n$. For instance, if $r = O(1)$, then T-HOSVD requires $O(dn^{d+1})$ operations, versus $O(n^{d+1})$ for the ST-HOSVD. The speedup of ST-HOSVD over T-HOSVD can then be close to the order, d, of the tensor. This is illustrated in Figure 6.2. Herein, we truncate an order-d cubic $30 \times 30 \times \cdots \times 30$ tensor to multilinear rank $(r, r, \ldots, r)$ for different values of the order $3 \leq d \leq 5$ and rank $1 \leq r \leq 30$. We computed the T-HOSVD and ST-HOSVD approximation of this tensor and repeated this five times. Of these five runs, the minimum execution time was selected to determine the speedup of ST-HOSVD over T-HOSVD.

If the tensor is of size $n_1 \times n_2 \times \cdots \times n_d$ and is approximated by rank $(r_1, r_2, \ldots, r_d)$, then the processing order $\mathbf{p}$ is very relevant and may lead to large differences in the total number of operations. In fact, the speedup of ST-HOSVD over T-HOSVD may then exceed the order of the tensor, as we illustrate in section 8. A heuristic for the selection of the processing order is suggested in section 6.4. However, the ST-HOSVD algorithm always requires fewer operations than the T-HOSVD, regardless of the processing order and the requested truncation rank.

6.3. Approximation error. Several properties concerning the approximation error of the ST-HOSVD are investigated. First, we stress that it depends on the processing order of the modes, $\mathbf{p}$.

Example. Consider, for instance, the third-order tensor $\mathcal{A} \in \mathbb{R}^{3\times3\times3}$:

$$\mathcal{A}_{:,:,1} = \begin{bmatrix} 2 & 4 & 7 \\ 5 & 6 & 3 \\ 9 & 3 & 5 \end{bmatrix}, \qquad \mathcal{A}_{:,:,2} = \begin{bmatrix} 7 & 5 & 3 \\ 9 & 2 & 8 \\ 9 & 2 & 3 \end{bmatrix}, \qquad \mathcal{A}_{:,:,3} = \begin{bmatrix} 8 & 4 & 6 \\ 3 & 2 & 5 \\ 9 & 3 & 4 \end{bmatrix}.$$

If we approximate this tensor by a rank-$(2, 2, 2)$ ST-HOSVD decomposition, the following errors are obtained for the different permutations of $\{1, 2, 3\}$:

$$\|\mathcal{A} - \hat{\mathcal{A}}_{[1,2,3]}\|_F = 8.1912, \quad \|\mathcal{A} - \hat{\mathcal{A}}_{[1,3,2]}\|_F = 8.1932, \quad \|\mathcal{A} - \hat{\mathcal{A}}_{[2,1,3]}\|_F = 7.4799,$$
$$\|\mathcal{A} - \hat{\mathcal{A}}_{[2,3,1]}\|_F = 7.4497, \quad \|\mathcal{A} - \hat{\mathcal{A}}_{[3,1,2]}\|_F = 7.5001, \quad \|\mathcal{A} - \hat{\mathcal{A}}_{[1,2,3]}\|_F = 7.4835.$$

The error of the rank-$(2, 2, 2)$ T-HOSVD approximation is 8.8188, which is worse than every ST-HOSVD approximation.

The following theorem demonstrates that the error can be expressed exactly in terms of the singular values that are obtained from an execution of Algorithm 1.

Theorem 6.4 (error of the ST-HOSVD). *Let* $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, *and let* $\hat{\mathcal{A}}$ *be the* *rank-*$(r_1, \ldots, r_d)$ *ST-HOSVD of* $\mathcal{A}$ *as in Definition* 6.1*. Then*

$$\|\mathcal{A} - \hat{\mathcal{A}}\|_F^2 = \sum_{k=1}^{d} \sum_{i_k=r_k+1}^{n_k} \tilde{\sigma}_{k,i_k}^2 = \sum_{k=1}^{d} \left(\|\hat{\mathcal{A}}^{k-1}\|_F^2 - \|\hat{\mathcal{A}}^{k}\|_F^2 \right); \tag{6.5}$$

i.e., the squared error is the sum of the squares of the discarded singular values.

Proof. We derive an explicit formula for the error of two successive approximations $\hat{\mathcal{A}}^{k-1}$ and $\hat{\mathcal{A}}^{k}$, as they are defined in Definition 6.1, with $1 \leq k \leq d$. That is,

$$\begin{aligned} \epsilon_k^2 &:= \|\hat{\mathcal{A}}^{k-1} - \hat{\mathcal{A}}^{k}\|_F^2 \\ &= \|(\hat{U}_1, \ldots, \hat{U}_{k-1}, I, \ldots, I) \cdot \hat{\mathcal{S}}^{k-1} - (\hat{U}_1, \ldots, \hat{U}_k, I, \ldots, I) \cdot \hat{\mathcal{S}}^{k}\|_F^2 \\ &= \left\| \left(\hat{\mathcal{S}}_{(k)}^{k-1} - \hat{U}_k \hat{\mathcal{S}}_{(k)}^{k} \right) (\hat{U}_1 \otimes \cdots \otimes \hat{U}_{k-1} \otimes I \otimes \cdots \otimes I)^T \right\|_F^2 \\ &= \left\| \hat{\mathcal{S}}_{(k)}^{k-1} - \hat{U}_k \hat{\mathcal{S}}_{(k)}^{k} \right\|_F^2. \end{aligned} \tag{6.6}$$

From the definition of the partially truncated core tensor in (6.3), we note that

$$\hat{U}_k^T \hat{\mathcal{S}}_{(k)}^{k-1} = \left[(\hat{U}_1^T, \ldots, \hat{U}_k^T, I, \ldots, I) \cdot \mathcal{A} \right]_{(k)} = \hat{\mathcal{S}}_{(k)}^{k}.$$

However, from (6.4) it is also clear that

$$\hat{U}_k \hat{\mathcal{S}}_{(k)}^{k} = \hat{U}_k \hat{U}_k^T \hat{\mathcal{S}}_{(k)}^{k-1} = \hat{U}_k \hat{\Sigma}_k \hat{V}_k^T.$$

Substituting the above into (6.6) and using (6.4) again, we obtain

$$\epsilon_k^2 = \left\| \tilde{U}_k \tilde{\Sigma}_k \tilde{V}_k^T \right\|_F^2 = \|\tilde{\Sigma}_k\|_F^2 = \|\hat{\mathcal{A}}^{k-1}\|_F^2 - \|\hat{\Sigma}_k\|_F^2 = \|\hat{\mathcal{A}}^{k-1}\|_F^2 - \|\hat{\mathcal{A}}^{k}\|_F^2. \tag{6.7}$$

Finally, $\hat{\mathcal{A}}^{k-1} - \hat{\mathcal{A}}^{k} = \hat{\pi}_k^{\perp} \hat{\pi}_1 \cdots \hat{\pi}_{k-1} \mathcal{A}$, so that Theorem 5.1 concludes the proof. □

This theorem is useful in establishing a truncation strategy based on a numerical threshold, rather than some predefined approximation rank. Typically, in dimensionality reduction, we are interested in the tensor of the lowest multilinear rank that attains a certain error bound. It is therefore of practical importance to truncate the HOSVD such that the actual error matches as closely as possible with the desired error. For the ST-HOSVD we can rely on the above exact expression of the error, whereas for the T-HOSVD only an upper bound is known. Consequently, a straightforward truncation of the T-HOSVD can result in an actual error that is much[4] smaller than the threshold, implying unnecessary computations and storage costs.

In the T-HOSVD algorithm, a numerical truncation strategy can be based on the upper bound in Corollary 5.2. The T-HOSVD error will be smaller than ϵ if the SVD in mode k is truncated to yield an error smaller than ϵ_k, provided that the sum of the squares of these ϵ_k adds up to ϵ^2. We set $\epsilon_k^2 = \epsilon^2/d$ in our experiments. We pursue a different strategy with the ST-HOSVD. In mode k, there can be a discrepancy between the desired error ϵ_k and the actual error ϵ_k', which is the sum of the discarded singular values in that mode. This discrepancy can be taken into account, as follows. In mode k, we truncate the SVD to yield an error smaller than $((\epsilon^2 - \sum_{j=1}^{k-1} \epsilon_j'^2)/(d-k+1))^{1/2}$. In this manner, the final approximation error is still bounded by ϵ.

[4] The difference between the actual error and the threshold increases with the order of the tensor.

It is not required that we compute a compact SVD in Algorithm 1 in order to enable the above numerical thresholding technique. Noting $\|\hat{\mathcal{A}}^k\|_F^2 = \|\hat{\Sigma}_k\|_F^2$ in (6.7) allows us to efficiently update the error expansion in (6.5) during the execution of Algorithm 1, provided that the norm of $\mathcal{A}$ is computed beforehand.[5] We can then compute $\epsilon_k^2 = \|\hat{\mathcal{A}}^{k-1}\|_F^2 - \|\hat{\Sigma}_k\|_F^2$, which involves only quantities obtained from the truncated SVD.

Numerical example. Due to Theorem 6.4, the ST-HOSVD can compress the data more than the T-HOSVD while satisfying some relative error bound. Our purpose is to compress the $784 \times 5421 \times 10$ tensor, described in section 8.2, as much as possible, while obtaining a relative error no larger than $7 \cdot 10^{-2}$. The ST-HOSVD, with $\mathbf{p} = [3, 1, 2]$, resulted in a rank-$(214, 810, 10)$ approximation with a relative error of $6.9977 \cdot 10^{-2}$. The T-HOSVD produced a rank-$(261, 1456, 10)$ approximation whose relative error is $4.9841 \cdot 10^{-2}$. The T-HOSVD yields an error that is clearly better than requested. However, this comes at the cost of increased approximation rank and storage demands. The ST-HOSVD stores 15.34% of the original data, whereas the T-HOSVD stores 28.00%, nearly twice as much. Furthermore, the T-HOSVD algorithm is much slower. It completed in 50 minutes, whereas ST-HOSVD required only 3 minutes and 14 seconds. This represents a speedup factor of 15.4.

To the best of our knowledge, there is no closed formula for the error of the T-HOSVD in terms of the singular values that are computed by its algorithm. Therefore, no technique is available to truncate the T-HOSVD to yield an error close to a target error *using only the information provided by the singular values in every mode* and without first computing a more accurate HOSVD with a higher-than-required multilinear rank. The above example illustrates that the ST-HOSVD sometimes produces approximations whose error is much closer to the target error than that of the T-HOSVD.

Theorem 6.4 provides an expression for the error of an ST-HOSVD in terms of the singular values computed by its algorithm. Furthermore, ST-HOSVD also satisfies the upper bound on the error of the T-HOSVD, in Corollary 5.2, regardless of the processing order $\mathbf{p}$. That is, when truncating the T-HOSVD and ST-HOSVD to a given multilinear rank, both approximation errors are bounded by the same quantity.

THEOREM 6.5 (error bound of the ST-HOSVD). *Let $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, and let $\hat{\mathcal{A}}$ be the rank-$(r_1, \ldots, r_d)$ ST-HOSVD of $\mathcal{A}$, as defined in Definition* 6.1*. Let the SVD of $\mathcal{A}_{(k)}$ be given as in* (4.3)*. Then*

$$\min_k \|\tilde{\Sigma}_k\|_F^2 \leq \|\mathcal{A} - \hat{\mathcal{A}}\|_F^2 \leq \sum_{k=1}^{d} \|\tilde{\Sigma}_k\|_F^2$$

are bounds on the error of the ST-HOSVD.

Proof. The upper bound follows from

$$\|\hat{\pi}_k^{\perp}\hat{\pi}_1 \cdots \hat{\pi}_{k-1}\mathcal{A}\|_F^2 \leq \|\bar{\pi}_k^{\perp}\hat{\pi}_1 \cdots \hat{\pi}_{k-1}\mathcal{A}\|_F^2 \leq \|\bar{\pi}_k^{\perp}\mathcal{A}\|_F^2 = \|\tilde{\Sigma}_k\|_F^2$$

for all $1 \leq k \leq d$. Herein, $\bar{\pi}_k$ is the T-HOSVD projector in mode k. The first inequality in the above formula is due to the fact that the $\hat{\pi}_k$ projector is optimal, as it is derived from a truncated SVD. The second inequality is due to (3.2).

The lower bound follows from combining Theorem 5.1 and the fact that the ST-HOSVD error depends on the order in which it is computed. □

[5] Computing this norm does not affect the asymptotic time complexity.

6.4. A heuristic for the processing order. ST-HOSVD requires an additional parameter, the processing order $\mathbf{p}$, when compared to the T-HOSVD. The processing order affects the approximation error and the number of operations to compute the approximation. Selecting a good processing order is thus of importance. Problems of this type, which are combinatorial in nature, occur in other tensor decompositions as well, such as in the selection of the tree in hierarchical Tucker [18] and the processing order in the tensor-train decomposition [42]. Unfortunately, this problem has not yet been resolved in the context of the above decompositions.

Here, we propose a heuristic to choose the processing order if *no other information is known* about the tensor besides its size. If more information is available, such as dependencies between the modes or the multilinear rank of the approximation, other heuristics may be advisable. We suggest choosing an ordering that attempts to minimize the number of operations required to compute the dominant subspaces. A simple greedy algorithm[6] is proposed: select the mode that minimizes the operation count to compute the first SVD, then select the mode that minimizes the cost of the next SVD, and so on. If the compact SVD is employed to compute the dominant subspace, the greedy minimization is accomplished simply by processing the modes in order of increasing mode length. We thus propose

$$\mathbf{p} = [p_1, p_2, \ldots, p_d] \quad \text{such that } \; n_{p_1} \le n_{p_2} \le \cdots \le n_{p_d}.$$

The heuristic was applied to the experiments presented in section 8. In several numerical experiments we conducted, it was confirmed that this heuristic leads to an efficient construction of the ST-HOSVD when compared to other processing orders. We also observed, experimentally, that this heuristic often leads to a good approximation error. We believe that this is related to the observation that compression in the short modes, which are processed first, can affect the rank of the longer modes. From Definition 6.1, it is clear that with this heuristic the rank of mode p_i is

$$\operatorname{rank}\left(\mathcal{A}_{(p_i)}\right) = \min\left\{ n_{p_i}, \prod_{j<i} r_{p_j} \prod_{j>i} n_{p_j} \right\} \le \min\left\{ n_{p_i}, \prod_{j \ne i} n_{p_j} \right\}.$$

Consequently, if the compression ranks in the first few modes are such that the inequality becomes strict in the above formulation, it entails that the rank of mode p_i has actually *decreased* due to truncations in the previous modes. This could force more "energy" into the fewer terms that remain, leading, possibly, to a smaller truncation error. On the other hand, if this mode were processed first, the energy would be spread out over more terms. This could result in a larger truncation error. Choosing a good processing order that minimizes the approximation error is still an open question.

7. Error with respect to the T-HOSVD. Recall that the optimization problem solved by the T-HOSVD can be considered an approximation to the problem that ST-HOSVD solves. Both approximate the solution of the actual problem (1.1) and satisfy the same upper bound on their approximation error. Given these observations, we wonder whether the ST-HOSVD approximation is always better than the T-HOSVD.

[6] This does not always lead to the globally minimum number of operations for computing the ST-HOSVD, but generally it is quite good.

HYPOTHESIS 7.1. *Let $\mathcal{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, $\hat{\mathcal{A}}_{\mathbf{p}*} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ be the rank-$(r_1, \ldots, r_d)$ ST-HOSVD of $\mathcal{A}$ with optimal permutation order $\mathbf{p}^*$, and let $\bar{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ be the rank-$(r_1, \ldots, r_d)$ T-HOSVD of $\mathcal{A}$. Then*

$$\|\mathcal{A} - \hat{\mathcal{A}}_{\mathbf{p}^*}\|_F^2 \overset{?}{\leq} \|\mathcal{A} - \bar{\mathcal{A}}\|_F^2.$$

The above hypothesis is *false*, in general, as the next counterexample shows.

7.1. A counterexample. Consider the following fourth-order tensor:

$$\mathcal{A}_{:,:,1,1} = \begin{bmatrix} 0.5 & -1.7 \\ -1.3 & -0.6 \end{bmatrix}, \qquad \mathcal{A}_{:,:,2,1} = \begin{bmatrix} -2.4 & -0.1 \\ -0.7 & 1.4 \end{bmatrix},$$
$$\mathcal{A}_{:,:,1,2} = \begin{bmatrix} 0.1 & 0.1 \\ 2.2 & -0.8 \end{bmatrix}, \qquad \mathcal{A}_{:,:,2,2} = \begin{bmatrix} -0.3 & -2.5 \\ 0.0 & 0.3 \end{bmatrix}.$$

The rank-$(1, 1, 1, 1)$ T-HOSVD approximation is given by

$$\bar{\mathcal{A}} = \left(\begin{bmatrix} -0.97325 \\ -0.22975 \end{bmatrix}, \begin{bmatrix} -0.78940 \\ 0.61388 \end{bmatrix}, \begin{bmatrix} -0.31546 \\ 0.94894 \end{bmatrix}, \begin{bmatrix} -0.88167 \\ 0.47186 \end{bmatrix} \right) \cdot \begin{bmatrix} 2.57934 \end{bmatrix}.$$

The best ST-HOSVD approximation corresponds to the order $\mathbf{p}^* = [1, 3, 2, 4]$:

$$\hat{\mathcal{A}}_{\mathbf{p}^*} = \left(\begin{bmatrix} -0.97325 \\ -0.22975 \end{bmatrix}, \begin{bmatrix} -0.97310 \\ 0.23037 \end{bmatrix}, \begin{bmatrix} -0.09956 \\ 0.99503 \end{bmatrix}, \begin{bmatrix} -0.99692 \\ 0.07841 \end{bmatrix} \right) \cdot \begin{bmatrix} 2.53595 \end{bmatrix}.$$

The approximation errors are given respectively by

$$\|\bar{\mathcal{A}} - \mathcal{A}\|_F^2 = 18.68700 \quad \text{and} \quad \|\hat{\mathcal{A}}_{\mathbf{p}^*} - \mathcal{A}\|_F^2 = 18.90896,$$

which demonstrates that the T-HOSVD is better than the ST-HOSVD approximation. This counterexample was found by randomly sampling the fourth-order tensors of size $2 \times 2 \times 2 \times 2$, whose entries were drawn from a gamma distribution with mean 1, and truncating them to rank $(1, 1, 1, 1)$. Our experiments yielded seven counterexamples in ten million samples. The presented counterexample is the one that resulted in the largest difference in error, but even then the difference is only 0.59%.

While counterexamples to Hypothesis 7.1 may exist, our Monte Carlo experiments indicate that they are *extremely thinly spread* for third-order tensors whose entries are drawn from a normal, gamma, or uniform distribution. For instance, our experiments did not reveal counterexamples within 10^7 samples for $3 \times 3 \times 3$ tensors whose entries are drawn from a standard normal distribution and were truncated to a (uniformly) randomly chosen rank. In general, we noted that the probability of encountering a counterexample decreases as the multilinear rank to truncate to decreases.

7.2. A sufficient condition. While Hypothesis 7.1 does not hold in general, sufficient conditions can be derived under which it does hold [58], again providing a strong rationale for the sequential truncation strategy. We present one such condition here. It states that a third-order tensor which is truncated to rank 1 in at least one mode will be approximated better by the ST-HOSVD.

THEOREM 7.2. *Let $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Let $\hat{\mathcal{A}}_{\mathbf{p}}$ be the rank-$(1, r_1, r_2)$ ST-HOSVD of $\mathcal{A}$ corresponding to the processing order $\mathbf{p}$, and let $\bar{\mathcal{A}}$ be the T-HOSVD of $\mathcal{A}$ of the same multilinear rank. Then, Hypothesis* 7.1 *holds.*

Proof. Consider $\mathbf{p} = [1, 2, 3]$. Let $\hat{\pi}_1$ be the ST-HOSVD projector along the first mode and $\bar{\pi}_1$ the T-HOSVD projector. Clearly, $\hat{\pi}_1 = \bar{\pi}_1$, and $A = \hat{\pi}_1\mathcal{A} = \bar{\pi}_1\mathcal{A}$ is a matrix. From the Eckart–Young theorem, we know that the truncated SVD of A yields the best low-rank approximation [14, 17]. The ST-HOSVD computes precisely that. It will select the r_1 left singular vectors, which define the projector $\hat{\pi}_2$, project onto them, and then compute the r_2 left singular vectors of the projected matrix, which comprise the projector $\hat{\pi}_3$. However, these r_2 left singular vectors correspond to the r_2 dominant right singular vectors of A. Thus, the combined projector $\hat{\pi}_2\hat{\pi}_3 A$ results in the best rank-$\min\{r_1, r_2\}$ approximation of A. The T-HOSVD, on the other hand, does not compute the projectors in this manner. Hence, neither $\bar{\pi}_2$ and $\bar{\pi}_3$, nor their combined projector, are optimal in general. That entails

$$\|\hat{\pi}_1^\perp\mathcal{A}\|_F^2 + \|\hat{\pi}_1\mathcal{A} - \hat{\pi}_2\hat{\pi}_3\hat{\pi}_1\mathcal{A}\|_F^2 \le \|\hat{\pi}_1^\perp\mathcal{A}\|_F^2 + \|\hat{\pi}_1\mathcal{A} - \bar{\pi}_2\bar{\pi}_3\bar{\pi}_1\mathcal{A}\|_F^2.$$

The first term can be brought into the norm on both sides of the inequality. Theorem 5.1 completes the proof. □

The above theorem applies to any third-order tensor that is truncated to rank 1 in at least one mode. The order of the components in the rank $(1, r_1, r_2)$ that we assumed in Theorem 7.2 does not limit the generality. That is because the modes can be renumbered such that the theorem applies. Note in particular that the rank-$(1, 1, 1)$ ST-HOSVD approximation is better than the T-HOSVD approximation. It might thus yield a more interesting starting point for iterative algorithms that approximate the optimal rank-1 approximation to a tensor; see, e.g., [7, 11, 63].

The proof of the above theorem *cannot* be extended to higher orders, as the counterexample in section 7.1 already demonstrated.

8. Numerical experiments. In this section, we compare and analyze the performance of ST-HOSVD and T-HOSVD. Some additional Monte Carlo experiments with third-order tensors are presented in [58]. The main conclusions from these experiments [58] are: (1) the difference in approximation error between HOOI, T-HOSVD, and ST-HOSVD is small; (2) the difference is largest when truncating the tensor to a low multilinear rank; (3) the ST-HOSVD approximation error is always better than T-HOSVD, regardless of the processing order of the modes; (4) ST-HOSVD ranks between T-HOSVD and HOOI with respect to the approximation error; and (5) ST-HOSVD is the fastest algorithm. However, in practice, the tensors are much more structured than the models studied in [58], and the conclusions from those experiments may or may not be accurate in the applications of interest to researchers. Therefore, we limit our attention to three real applications in this paper.

The first example serves to demonstrate that ST-HOSVD nearly invariably improves upon T-HOSVD, while reducing the computational cost, if both are truncated to the same rank. It also reveals that the processing order of ST-HOSVD can sometimes be cleverly chosen for the application at hand; see also [58, section 7.5]. The second example investigates ST-HOSVD and T-HOSVD when a numerical threshold is used to truncate the approximation. It illustrates that an ST-HOSVD approximation can be constructed much faster than the T-HOSVD. Concurrently, it shows that ST-HOSVD results in an approximation whose error is much closer to the desired tolerance than T-HOSVD, resulting in lower storage costs. In the third example, we illustrate the difficulties that can arise when using the less accurate eigenvalue-based implementation of `nvecs` (see next paragraph). It reveals a class of problems wherein ST-HOSVD is always expected to be significantly faster than T-HOSVD. We also present results of an iterative algorithm to compute the projectors.

All experiments were conducted on a laptop comprising an Intel Core2Duo P7350 processing unit (2.0GHz) and 4GB main memory. The algorithms were implemented[7] in MATLAB 7.9.0, using the Tensor Toolbox v.2.4 [4]. However, we modified the `nvecs` implementation, which computes the dominant subspace in mode k, to use an SVD-based algorithm,[8] rather than computing the eigenvalues of the Gram matrix $A_{(k)}A_{(k)}^T$. While the original implementation is faster, it is well known that computing the SVD is more accurate. We illustrate this problem in section 8.3.

8.1. Dimensionality reduction for images. We investigate the use of ST-HOSVD, T-HOSVD, and HOOI for the compression of a set of images to a fixed multilinear rank (resulting in a fixed memory consumption). Applications of such a compressed representation are described in [59, 60, 61], where similar data sets are used.

HOOI is an iterative alternating least-squares algorithm for estimating the optimal orthogonal Tucker model of a specified multilinear rank [11]. Initializing HOOI with the T-HOSVD often results in an approximation whose error is close to the approximation error of the global optimum [11]. In this section, we assume that the HOOI solution represents the best approximation that can be achieved, and we compare ST-HOSVD and T-HOSVD relative to this alleged optimal solution. We used the `tucker_als` implementation [3] from the Tensor Toolbox v.2.4 [4]. The HOOI iterations were halted if the approximation error did not improve by more than 10^{-7} between two successive iterations, with the maximum number of iterations set to 50. That was sufficient to ensure convergence of HOOI.

The data set[9] consists of the Olivetti Research Laboratory face database [48]. This data set contains 10 different images of each of the 40 subjects. The 92×112 grayscale images of a subject differ in the lighting, facial expressions and facial details. We organized the images into a $10304 \times 40 \times 10$ tensor. The first mode corresponds to both pixel dimensions, which is referred to as a *texel* mode in the computer graphics literature. That is, every 92×112 image was vectorized into a vector of length 10304. The second mode corresponds to the subjects and the final to the different expressions.

We compressed the above data set to a fixed multilinear rank, and compared the three methods. In the texel mode we compressed to rank 1, 11, ..., 401. The subject mode was compressed to rank 1, 2, ..., 40, and the expression mode to 6 through 9. The approximation error for every combination of those mode ranks was computed.

We selected $\mathbf{p} = [3, 2, 1]$ as the processing order of the ST-HOSVD. This corresponds to the heuristic discussed in section 6.4.

In Figure 8.1, we present the results for truncating the expression mode to rank six.[10] We visualized the relative distance of the ST-HOSVD and T-HOSVD to the solution computed by HOOI. Clearly, the ST-HOSVD approximation is much closer to the (local) optimum, in terms of approximation error, than that of the T-HOSVD. In fact, the error of the ST-HOSVD was at most 1.012% worse than HOOI, whereas the T-HOSVD's maximum error was significantly higher, at 6.340%. On average[11] the ST-HOSVD error was 0.0985% higher than that of HOOI, and the error of T-HOSVD

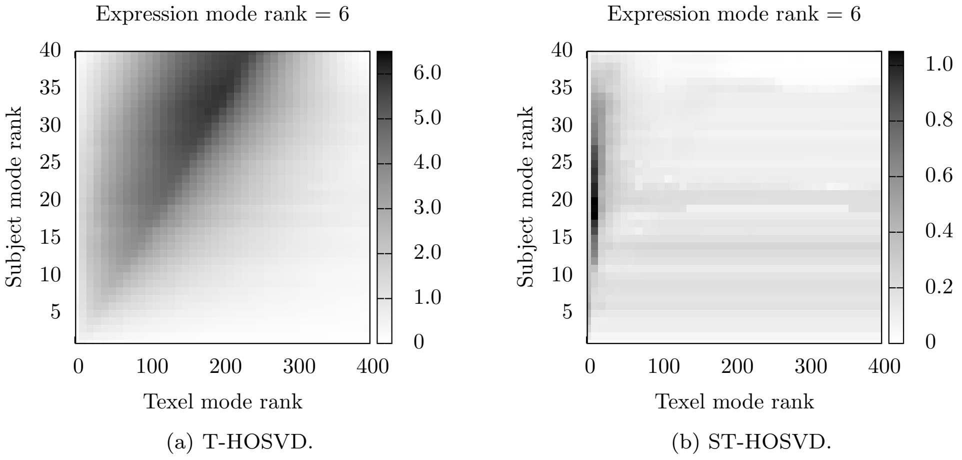
[7] The code can be found at http://people.cs.kuleuven.be/∼nick.vannieuwenhoven/.

[8] By using the compact SVD everywhere, we can be sure the timings are indicative of the actual performance, as the aforementioned SVD in MATLAB calls the corresponding LAPACK [1] routine.

[9] It can be found at http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[10] The results for truncating the expression mode to other ranks are very similar.

[11] The average was taken over the relative approximation errors for every multilinear rank we tested. Similar results are obtained when comparing the relative difference of the total approximation error over all multilinear ranks between the three methods.



(a) T-HOSVD.

(b) ST-HOSVD.

FIG. 8.1. *The relative approximation error* $(err_{HOSVD} - err_{HOOI})/err_{HOOI}$ *(expressed in %) in function of the multilinear approximation rank, with respect to the local optimum found by the HOOI algorithm, for the T-HOSVD* (a) *and ST-HOSVD* (b) *algorithms. A darker shade represents a larger (relative) error of the approximation. Note the different scales.*

was 2.115% higher than that of HOOI. This implies that, on average, T-HOSVD's distance to the optimum was 21.5 times the distance of ST-HOSVD to the optimum.

In Figure 8.1(a), there appears to be a sharp boundary line between large and small approximation errors relative to HOOI. Since the subject and expression modes are very short compared to the texel mode, the latter's rank cannot be higher than the product of the ranks of the aforementioned modes. Indeed, if the subject and expression modes are truncated to rank r_2 and r_3, respectively, the unfolding in the texel mode results in a $10304 \times r_2 r_3$ matrix, whose rank is bounded by $r_2 r_3$. The compression in the subject and expression mode thus strongly affects and limits the rank in the texel mode. The remaining energy after these two projections is spread out over only $r_2 r_3$ singular vectors, instead of over the 400 singular vectors one obtains by simply computing the dominant subspace of the texel mode (without first projecting in the other modes). This potentially concentrates more energy into fewer vectors, which could result in a smaller truncation error in that mode. It is precisely this information that we try to exploit by our choice of processing order for the ST-HOSVD. By processing the texel mode last, we ensure that its rank is already maximally reduced. The T-HOSVD, on the other hand, truncates all modes independently and consequently does not detect that, due to the truncations in other modes, the rank in the texel mode is, in fact, $r_2 r_3 < 400$. T-HOSVD is unable to exploit this information to construct a more accurate approximation. It is interesting to note that if the texel mode is not processed last, the sharp boundary is also present in the ST-HOSVD, and the approximation error is worse than the presented processing order.

Figure 8.1(b) features horizontal lines where the approximation error is constant. Because of the processing order, the rank in the texel mode is $r_2 r_3$, as explained in the previous paragraph. Consequently, "truncating" the texel mode to mode $r_1 > r_2 r_3$ results in the same approximation as truncating it to rank $r_2 r_3$, hence resulting in those horizontal lines. Any variations of the error on a horizontal line is due to the

HOOI algorithm, which finds different local optima.

Finally, we report the total execution time to compute all low-rank approximations. The ST-HOSVD required 91 minutes to compute the 6560 approximations, T-HOSVD computed for 207 minutes, and HOOI took 1841 minutes (an average of 20 iterations to reach convergence). The ST-HOSVD thus attains a speedup of 20.23 over HOOI and 2.27 over T-HOSVD. The ST-HOSVD solution may thus be more favorable in some applications, as it is, on average, within 0.1% of the optimum while cutting the execution time by a factor of 20 with respect to HOOI.

8.2. Handwritten digit classification. We revisit the classification of handwritten digits using the HOSVD, which was investigated by Savas and Eldén in [50]. The problem they consider consists of automatically classifying a set of grayscale images, representing handwritten digits, into ten classes, in order to predict the digit (or label) that is represented by the image. Algorithms for such a problem construct a (low-parameter) model from a set of training images for which the actual label is known. That model is employed to classify a disjoint set of test images. In [50, Algorithm 2], a HOSVD-based algorithm is presented. We demonstrate that using the ST-HOSVD instead of the T-HOSVD in this algorithm can significantly reduce the execution time of the training phase.

We compare the performance of the T-HOSVD and ST-HOSVD algorithms on the MNIST database,[12] which contains 60,000 training images and 10,000 test images. The 28×28 images are 8-bit grayscale. The training images are unequally distributed over the ten classes. Therefore, we restricted the number of training images in every class to 5421. The training images are grouped into a third-order tensor $\mathcal{A}$ of size $786 \times 5421 \times 10$. The first mode is the texel mode. The second mode corresponds to the training images. The third mode corresponds to the different classes. The vector $\mathcal{A}_{:,5,8}$ thus corresponds to the fifth image representing an eight.

To construct the model from the training images, Savas and Eldén [50] use the T-HOSVD to compress the data to a specified multilinear rank. The digit mode is not compressed. This results in an approximation $\mathcal{A} \approx (U, V, I) \cdot \mathcal{S}$. Thereafter, one basis matrix B_μ, constituting the first κ left singular vectors of $\mathcal{S}_{:,:,\mu}$, is computed for every class $1 \le \mu \le 10$, as in [50]. To classify a test image D, they vectorize the image, $d := \operatorname{vec}(D)$. The coordinates of the orthogonal projection of d onto the basis U, which corresponds to the texel mode, are determined: $d_p = U^T d$. This low-parameter approximation of d is projected orthogonally onto the space spanned by the basis matrix B_μ. Its residual is $r(d_p, \mu) := \|d_p - B_\mu B_\mu^T d_p\|_F$. The image d is then classified as $\arg\min_\mu r(d_p, \mu)$.

In this paper, we repeat the experiment of Savas and Eldén [50] and compare the relative performance of T-HOSVD and ST-HOSVD in compressing the data. We choose to truncate the T-HOSVD and ST-HOSVD to yield a relative approximation error of no more than 15%, using the technique described below Theorem 6.4, rather than selecting a multilinear rank beforehand. For the sake of comparison, an ST-HOSVD model of the same rank as the T-HOSVD is also presented (rST-HOSVD). In our experiments, we set $\kappa = 15$, which is a good choice judging from the data in [50].

The processing order of the ST-HOSVD was $\mathbf{p} = [1, 2]$, as the digit mode is skipped. This corresponds to the heuristic in section 6.4, ignoring the digit mode. ST-HOSVD thus first compresses the images (texel mode) simultaneously, and then

[12] The database can be obtained from http://yann.lecun.com/exdb/mnist/.

Table 8.1
A comparison between T-HOSVD and ST-HOSVD in classifying handwritten digits by an algorithm due to Savas and Eldén [50, Algorithm 2].

	T-HOSVD	ST-HOSVD	rST-HOSVD
Relative model error	$9.895 \cdot 10^{-2}$	$14.997 \cdot 10^{-2}$	$9.678 \cdot 10^{-2}$
Model rank	$(94, 511, 10)$	$(65, 142, 10)$	$(94, 511, 10)$
Training time (s)	2966.0	59.3	68.7
Classification time (s)	14.7	11.8	14.8
Classification error (%)	4.940	4.960	4.940
Storage (# of values)	87984	60840	87984

constructs a compressed representation over all training images. It is well known, in the context of classification, that classifiers constructed from the main features of the training images [31] actually perform *better* than classifiers constructed from the raw image data [50]. ST-HOSVD, with this processing order, can be interpreted as providing such a feature-extraction step, originating from the projection onto the dominant features—as determined by the SVD—in the first mode. In this case, knowledge of the application domain also *suggests* the use of the processing order $\mathbf{p} = [1, 2]$.

The results of our experiments are summarized in Table 8.1. First, we note that the ST-HOSVD model results in a relative error that is much closer to the target error than that of the T-HOSVD, while not significantly affecting the classification error. Because of this, the multilinear rank of the ST-HOSVD model is lower, which has several additional benefits in this application. ST-HOSVD classifies the test images 20% faster than T-HOSVD. Furthermore, the memory requirements are down by 31% from the T-HOSVD. ST-HOSVD stores a 786×65 U matrix and ten 65×15 matrices $\{B_\mu\}_{\mu=1}^{10}$, whereas T-HOSVD requires a 786×94 U matrix and ten 94×15 matrices.

The major advantage of the ST-HOSVD over T-HOSVD concerns its processing time. T-HOSVD computes the SVD of $\mathcal{A}_{(1)}$, a 786×54210 matrix, and $\mathcal{A}_{(2)}$, a 5421×7860 matrix. ST-HOSVD also requires the SVD of $\mathcal{A}_{(1)}$, but due to the projection step, it computes only the SVD of $\mathcal{A}_{(2)}(U \otimes I)$, which is a 5421×650 matrix. Consequently, the ST-HOSVD algorithm achieves a speedup factor of nearly 50 over the T-HOSVD in constructing the low-parameter third-order tensor model.

Comparing the rST-HOSVD and T-HOSVD, which have the same rank, we note that the relative approximation error of rST-HOSVD is better than that of the T-HOSVD. The training time is significantly shorter, while the classification error and time are equal.

8.3. Compression of simulation results. Lorente, Vega, and Velazquez [35] recently employed the HOSVD to compress an aerodynamics database consisting of numerical results of a multiparameter computational fluid dynamics simulation of the flow around an airfoil. We apply this idea to the solution of the much simpler 2D heat equation using a finite difference discretization. We seek the solution $u(x, y, t)$ of

$$\partial u/\partial t = \partial^2 u/\partial x^2 + \partial^2 u/\partial y^2$$

on the unit square $[0, 1]^2$, with boundary condition $0.25 - |0.5 - x| \cdot |0.5 - y|$. The initial heat distribution $u(x, y, 0)$ is also given by the last equation. We simulate this PDE up to $t = 0.25$. The PDE was discretized with a uniform mesh with cell size $(\Delta s, \Delta s, \Delta t)$ and solved using the explicit Euler method. The time step should be $0.25\Delta s^2$, in order to make the numerical scheme stable. We set $\Delta s = 10^{-2}$ and $\Delta t = 0.25 \cdot 10^{-4}$, resulting in a tensor of size $101 \times 101 \times 10001$.

Table 8.2

A comparison of three implementations of T-HOSVD and ST-HOSVD in compressing the results from a numerical simulation of the 2D heat equation.

	Type	Rank	Error	Time (s)	Compression
T-HOSVD	SVD	$(22, 22, 20)$	$8.510754 \cdot 10^{-5}$	9942.51	476.41
	EIG	$(22, 22, 20)$	$39.411250 \cdot 10^{-5}$	2371.64	476.41
	EIGS	$(22, 22, 20)$	$38.938427 \cdot 10^{-5}$	110.79	476.41
ST-HOSVD	SVD	$(22, 21, 19)$	$9.586941 \cdot 10^{-5}$	74.73	502.22
	EIG	$(22, 21, 19)$	$34.478420 \cdot 10^{-5}$	7.47	502.22
	EIGS	$(22, 21, 19)$	$34.480534 \cdot 10^{-5}$	5.42	502.22

The solution tensor was compressed to an absolute error of $\Delta s^2 = 10^{-4}$, the discretization accuracy, using the ST-HOSVD and T-HOSVD. We compare three implementations of the dominant subspace estimation algorithm: the SVD-based algorithm as used in this paper (`svd(A,'econ')`), the algorithm which computes the full eigendecomposition of the Gram matrix $A_{(k)}A_{(k)}^T$ or $A_{(k)}^T A_{(k)}$ depending on whichever is smaller (`eig(A)`), and an iterative algorithm which computes only the required dominant eigenvectors of the Gram matrix (`eigs(A,r)`). The latter is the default algorithm in the Tensor Toolbox v.2.4. In Table 8.2, we refer to these algorithms as "SVD," "EIG," and "EIGS," respectively. ST-HOSVD's processing order $\mathbf{p} = [1, 2, 3]$.

The results of our experiments are summarized in Table 8.2. The column "Type" indicates the implementation of the subspace estimation, and "Compression" shows the ratio between the number of floating point values used to store the solution tensor $(101^2 \cdot 10001)$ and the number used to store the multilinear approximation.

We note the very high compression factors in Table 8.2. The smallest possible rank of approximations that attain the discretization accuracy was determined experimentally to be rank $(21, 21, 19)$. Both SVD-based implementations of the numerically thresholded ST-HOSVD and T-HOSVD find an approximation whose rank is very close to this optimum. Its rank is small compared to the size of the tensor. That is because the values in the tensor represent an approximation to the C_0-continuous analytical solution of the PDE. Such function-related tensors can be compressed greatly [20, 27].

The table indicates that the approximation errors of the EIG and EIGS-based algorithms are worse than that of the SVD-based algorithm for a given multilinear rank. The error is at least 3.5 times higher than the error of the SVD-based implementations. Clearly, the discretization accuracy is not satisfied by a rank-$(22, 22, 20)$ tensor. In order to attain the discretization accuracy, the multilinear rank of the approximation has to be increased to $(86, 86, 50)$, which negatively affects the storage costs and compression time. The SVD-based implementations of ST-HOSVD and T-HOSVD compressed the data more than the EIG and EIGS-based implementations by a factor of 4.36 and 4.14, respectively, while attaining the requested error bound.

Again note that the ST-HOSVD yields a smaller rank and better compression factor than the T-HOSVD at a fixed error bound. At a fixed rank, on the other hand, the error of ST-HOSVD is better than that of T-HOSVD, regardless of the algorithm that implements the computation of the dominant subspaces.

The EIG-based implementations are indeed faster than the SVD-based implementations. The EIGS-based implementations are even faster. Clearly, the ST-HOSVD is *both* more accurate and faster than the T-HOSVD, regardless of the algorithm that implements the computation of the dominant subspace. The SVD, EIG, and EIGS implementations of ST-HOSVD attain a speedup of, respectively, 133.05, 317.49, and

20.44 over the corresponding implementation of the T-HOSVD.

The higher execution time of the T-HOSVD cannot be circumvented in this problem, wherein, for reasons of numerical stability of the discretization scheme, at least one unfolding results in a matrix that is approximately square. ST-HOSVD avoids the problem by first processing the short modes, as suggested by our heuristic. Because, in these applications, the modes are greatly compressible, the unfolding in the last mode(s) will result in a rectangular matrix with more rows than columns, rather than an approximately square matrix. Thus, the cost of computing the dominant subspace, regardless of implementation, is greatly reduced, resulting in significant speedups over the T-HOSVD.

9. Conclusions.

An error expression for a truncated orthogonal Tucker decomposition was presented. Based on this expression, we proposed an improved truncation strategy for the higher-order singular value decomposition. A truncation strategy based on a numerical threshold was presented that allows more accurate control over the final approximation error, using only the singular values computed during the construction of the ST-HOSVD model.

Numerical experiments indicate that the ST-HOSVD may be a suitable alternative to the T-HOSVD, as it can significantly reduce the number of floating point operations used to construct the model. In many cases, it also reduces the approximation error. In one application, ST-HOSVD resulted, on average, in an approximation whose error was only 0.1% higher than the (local) optimum computed by HOOI, while the execution time was cut by a factor of 20. In other applications, speedups of 50 and 133 in execution time were obtained over the T-HOSVD.

Despite its benefits, the ST-HOSVD also introduces a number of difficulties. For instance, the sequential approach destroys most of the structure in the original tensor. Therefore, straightforwardly applying the sequential truncation to sparse or otherwise structured tensors may be inadvisable. The ST-HOSVD is also serial in nature, whereas the T-HOSVD is parallel. In the latter, the SVDs in the different modes can be computed independently. Future research should investigate the degree to which these issues can be overcome. A final concern of the ST-HOSVD algorithm is related to the selection of the processing order. Although the heuristic we propose appears to be utile in the applications considered, we recognize that the knowledge on this topic is very incomplete and warrants further research.

Acknowledgments. H. Speleers, L. Sorber, and M. Van Barel are thanked for interesting and fundamental remarks. They are also thanked for their stimulating enthusiasm. L. Sorber is thanked for assistance with the images of the tensor decompositions and the code from the MultiLinear toolbox. L. De Lathauwer is kindly thanked for feedback on an early version of this manuscript. We also thank the anonymous reviewers for their helpful comments.

REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide, Third Edition*, SIAM, Philadelphia, PA, 1999.

[2] C. Andersson and R. Bro, *Improving the speed of multi-way algorithms: Part* I. *Tucker*3, Chemometrics and Intelligent Laboratory Systems, 42 (1998), pp. 93–103.

[3] B. Bader and T. Kolda, *Algorithm* 862*: MATLAB tensor classes for fast algorithm prototyping*, ACM Trans. Math. Software, 32 (2006), pp. 635–653.

[4] B. Bader and T. Kolda, *MATLAB Tensor Toolbox Version* 2.4, Mar. 2010, current version available online at http://www.sandia.gov/∼tgkdda/TensorToolbox/.

[5] F. Bell, D. Lambrecht, and M. Head-Gordon, *Higher order singular value decomposition in quantum chemistry*, Molec. Phys., 108 (2010), pp. 2759–2773.

[6] R. Bro and C. Andersson, *Improving the speed of multiway algorithms: Part* II*: Compression*, Chemometrics and Intelligent Laboratory Systems, 42 (1998), pp. 105–113.

[7] J. Chen and Y. Saad, *On the tensor SVD and the optimal low rank orthogonal approximation of tensors*, SIAM J. Matrix Anal. Appl., 30 (2009), pp. 1709–1734.

[8] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain, *Symmetric tensors and symmetric tensor rank*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1254–1279.

[9] R. Costantini, L. Sbaiz, and S. Süsstrunk, *Higher order SVD analysis for dynamic texture synthesis*, IEEE Trans. Image Process., 17 (2008), pp. 42–52.

[10] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278.

[11] L. De Lathauwer, B. De Moor, and J. Vandewalle, *On the best rank-*1 *and rank-*$(R_1, R_2, \ldots, R_n)$ *approximation of higher-order tensors*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1324–1342.

[12] L. De Lathauwer and J. Vandewalle, *Dimensionality reduction in higher-order signal processing and rank-*$(r_1, r_2, \ldots, r_n)$ *reduction in multilinear algebra*, Linear Algebra Appl., 391 (2004), pp. 31–55.

[13] V. de Silva and L.-H. Lim, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1084–1127.

[14] J. W. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, PA, 1997.

[15] L. Eldén and B. Savas, *A Newton–Grassmann method for computing the best multilinear rank-*(r_1, r_2, r_3) *approximation of a tensor*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 248–271.

[16] F. Estienne, N. Matthijs, D. Massart, P. Ricoux, and D. Leibovici, *Multi-way modelling of high-dimensionality electroencephalographic data*, Chemometrics and Intelligent Laboratory Systems, 58 (2001), pp. 59–72.

[17] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[18] L. Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 2029–2054.

[19] M. Haardt, F. Roemer, and G. Del Galdo, *Higher-order SVD-based subspace estimation to improve the parameter estimation accuracy in multidimensional harmonic retrieval problems*, IEEE Trans. Signal Process., 56 (2008), pp. 3198–3213.

[20] W. Hackbusch and B. Khoromskij, *Tensor-product approximation to operators and functions in high dimensions*, J. Complexity, 23 (2007), pp. 697–714.

[21] W. Hackbusch and S. Kühn, *A new scheme for the tensor representation*, J. Fourier Anal. Appl., 15 (2009), pp. 706–722.

[22] F. Hitchcock, *Multiple invariants and generalized rank of a p-way matrix or tensor*, J. Math. Phys., 7 (1927), pp. 39–79.

[23] M. Ishteva, P.-A. Absil, S. Van Huffel, and L. De Lathauwer, *Tucker compression and local optima*, Chemometrics and Intelligent Laboratory Systems, 106 (2010), pp. 57–64.

[24] M. Ishteva, P.-A. Absil, S. Van Huffel, and L. De Lathauwer, *Best low multilinear rank approximation of higher-order tensors, based on the Riemannian trust-region scheme*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 115–135.

[25] M. Ishteva, L. De Lathauwer, P.-A. Absil, and S. Van Huffel, *Dimensionality reduction for higher-order tensors: Algorithms and applications*, Int. J. Pure Appl. Math., 42 (2008), pp. 337–343.

[26] M. Ishteva, L. De Lathauwer, P.-A. Absil, and S. Van Huffel, *Differential-geometric Newton method for the best rank-*(r_1, r_2, r_3) *approximation of tensors*, Numer. Algorithms, 51 (2009), pp. 179–194.

[27] B. Khoromskij, *Structured rank-*$(r_1, \ldots, r_d)$ *decomposition of function-related tensors in* r^d, Comput. Methods Appl. Math., 6 (2006), pp. 194–220.

[28] E. Kofidis and P. A. Regalia, *On the best rank-*1 *approximation of higher-order supersymmetric tensors*, SIAM J. Matrix Anal. Appl., 23 (2002), pp. 863–884.

[29] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Rev., 51 (2009), pp. 455–500.

[30] P. Kroonenberg and J. de Leeuw, *Principal component analysis of three-mode data by means of alternating least squares algorithms*, Psychometrika, 45 (1980), pp. 69–97.

[31] C. Liu, K. Nakashima, H. Sako, and H. Fujisawa, *Handwritten digit recognition: Benchmarking of state-of-the-art techniques*, Pattern Recognition, 36 (2003), pp. 2271–2285.

[32] C. Liu, J. Zhou, and K. He, *Image compression based on truncated HOSVD*, in Proceedings of the International Conference on Information Engineering and Computer Science, IEEE Press, Piscataway, NJ, 2009, pp. 1–4.

[33] N. Liu, B. Zhang, J. Yan, Z. Chen, W. Liu, F. Bai, and L. Chien, *Text representation: From vector to tensor*, Proceedings of the IEEE International Conference on Data Mining, J. Han, B. W. Wah, V. Raghavan, X. Wu, and R. Rastogi, eds., IEEE Computer Society, Los Alamitos, CA, 2005, pp. 725–728.

[34] X. Liu, L. De Lathauwer, F. Janssens, and B. De Moor, *Hybrid clustering of multiple information sources via HOSVD*, in Advances in Neural Networks - ISNN 2010, L. Zhang, B.-L. Lu, and J. Kwok, eds., Lecture Notes in Comput. Sci. 6064, Springer, Berlin, Heidelberg, 2010, pp. 337–345.

[35] L. Lorente, J. Vega, and A. Velazquez, *Compression of aerodynamic databases using high-order singular value decomposition*, Aerospace Sci. Tech., 14 (2010), pp. 168–177.

[36] M. Markaki and Y. Stylianou, *Discrimination of speech from nonspeech in broadcast news based on modulation frequency features*, Speech Communication, 53 (2011), pp. 726–735.

[37] N. Mesgarani, M. Slaney, and S. Shamma, *Discrimination of speech from nonspeech based on multiscale spectro-temporal modulations*, IEEE Trans. Audio Speech, Language Process., 14 (2006), pp. 920–930.

[38] D. Muti and S. Bourennane, *Survey on tensor signal algebraic filtering*, Signal Process., 87 (2007), pp. 237–249.

[39] J. Nagy and M. Kilmer, *Kronecker product approximation for preconditioning in three-dimensional imaging applications*, IEEE Trans. Image Process., 15 (2006), pp. 604–613.

[40] L. Omberg, G. Golub, and O. Alter, *A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies*, Proc. Natl. Acad. Sci. USA, 104 (2007), pp. 18371–18376.

[41] L. Omberg, J. Meyerson, K. Kobayashi, L. Drury, J. Diffley, and O. Alter, *Global effects of DNA replication and DNA replication origin activity on eukaryotic gene expression*, Molec. Systems Biol., 5 (2009), 312.

[42] I. V. Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comput., 33 (2011), pp. 2295–2317.

[43] I. V. Oseledets and E. E. Tyrtyshnikov, *Breaking the curse of dimensionality, or how to use SVD in many dimensions*, SIAM J. Sci. Comput., 31 (2009), pp. 3744–3759.

[44] I. Oseledets and E. Tyrtyshnikov, *Recursive decomposition of multidimensional tensors*, Dokl. Math., 80 (2009), pp. 460–462.

[45] J. M. Papy, L. De Lathauwer, and S. Van Huffel, *Exponential data fitting using multilinear algebra: The single-channel and multi-channel case*, Numer. Linear Algebra Appl., 12 (2005), pp. 809–826.

[46] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, Manchester, UK, 1992.

[47] Y. Saad, *Iterative Methods for Sparse Linear Systems, Second Edition,* SIAM, Philadelphia, PA, 2003.

[48] F. Samaria and A. Harter, *Parameterisation of a stochastic model for human face identification*, in Proceedings of the Second IEEE Workshop on Applications of Computer Vision, IEEE Press, Piscataway, NJ, pp. 138–142.

[49] B. Savas, *Analyses and Tests of Handwritten Digit Recognition Algorithms*, examensarbete (Master's thesis), Department of Mathematics, Tekniska Högskolan i Linköping, Linköping, Sweden, 2003.

[50] B. Savas and L. Eldén, *Handwritten digit classification using higher order singular value decomposition*, Pattern Recognition, 40 (2007), pp. 993–1003.

[51] B. Savas and L.-H. Lim, *Quasi-Newton methods on Grassmannians and multilinear approximations of tensors*, SIAM J. Sci. Comput., 32 (2010), pp. 3352–3393.

[52] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen, *CubeSVD: A novel approach to personalized web search*, in Proceedings of the 14th international conference on World Wide Web (WWW '05), ACM, New York, 2005, pp. 382–390.

[53] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos, *Tag recommendations based on tensor dimensionality reduction*, in Proceedings of the 2008 ACM Conference on Recommender Systems, New York, 2008, ACM, New York, 2008, pp. 43–50.

[54] L. Tucker, *Implications of factor analysis of three-way matrices for measurement of change*, in Problems in Measuring Change, C. Harris, ed., University of Wisconsin Press, Madison, WI, 1963, pp. 122–137.

[55] L. Tucker, *The extension of factor analysis to three-dimensional matrices*, in Contributions to Mathematical Psychology, H. Gulliksen and N. Frederiksen, eds., Holt, Rinehardt, & Winston, New York, 1964, pp. 110–127.

[56] L. Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 31 (1966), pp. 279–311.

[57] C. Van Loan, *The ubiquitous Kronecker product*, J. Comput. Appl. Math., 123 (2000), pp. 85–100.

[58] N. Vannieuwenhoven, R. Vandebril, and K. Meerbergen, *On the Truncated Multilinear Singular Value Decomposition*, Technical Report TW589, Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium, 2011.

[59] M. Vasilescu and D. Terzopoulos, *Multilinear image analysis for facial recognition*, in Proceedings of the 16th International Conference on Pattern Recognition, IEEE Computer Society, Los Alamitos, CA, 2002, Vol. 2, pp. 511–514.

[60] M. Vasilescu and D. Terzopoulos, *Multilinear subspace analysis of image ensembles*, IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition, 2 (2003), pp. 93–99.

[61] M. Vasilescu and D. Terzopoulos, *TensorTextures: Multilinear image-based rendering*, ACM Trans. Graph., 23 (2004), pp. 336–342.

[62] H. Wang and N. Ahuja, *Facial expression decomposition*, in Proceedings of the Ninth IEEE International Conference on Computer Vision., 2003, Vol. 2, pp. 958–965.

[63] T. Zhang and G. H. Golub, *Rank-one approximation to high order tensors*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 534–550.