Tensor Fields for Multilinear Image Representation and Statistical Learning Models Applications

Tiene Andre Filisbino and Gilson Antonio Giraldi
Department of Computer Science
National Laboratory for Scientific Computing (LNCC)
Petrópolis, Brazil
{tiene,gilson}@lncc.br

Carlos Eduardo Thomaz
Department of Electrical Engineering
FEI
São Bernardo do Campo, Brazil
cet@fei.edu.br

***Abstract*—Nowadays, higher order tensors have been applied to model multidimensional image data for subsequent tensor decomposition, dimensionality reduction and classification tasks. In this paper, we survey recent results with the goal of highlighting the power of tensor methods as a general technique for data representation, their advantage if compared with vector counterparts and some research challenges. Hence, we firstly review the geometric theory behind tensor fields and their algebraic representation. Afterwards, subspace learning, dimensionality reduction, discriminant analysis and reconstruction problems are considered following the traditional viewpoint for tensor fields in image processing, based on generalized matrices. We show several experimental results to point out the effectiveness of multilinear algorithms for dimensionality reduction combined with discriminant techniques for selecting tensor components for face image analysis, considering gender classification as well as reconstruction problems. Then, we return to the geometric approach for tensors and discuss opened issues in this area related to manifold learning and tensor fields, incorporation of prior information and high performance computational requirements. Finally, we offer conclusions and final remarks.**

***Keywords*-Tensor Fields; Dimensionality Reduction; Tensor Subspace Learning; Ranking Tensor Components; Reconstruction; MPCA; Face Image Analysis.**

I. Introduction

Many areas such as pattern recognition, computer vision, signal processing and medical image analysis, require the managing of huge data sets with a large number of features or dimensions. Behind such *big data* we have medical imaging, scientific data produced by numerical simulations and acquisition systems as well as complex systems from Nature, Internet, economy/finance, among other areas in Society [1]. The analysis of such data is ubiquitous, involving multilevel, multi-scale and statistical viewpoints which raised an intriguing question: can we envisage a conceptual construction that plays for data analysis the same role that geometry and statistical mechanics play for physics?

Firstly of all, we agree that geometry and topology are the natural tools to handle large, high-dimensional databases because we can encode in a geometric concept (like a differentiable manifold) notions related to similarity, vectors and tensors while the topology allows to synthesize knowledge about connectivity to understand how data is organized on different levels [2], [3]. In this scenario, manifold learning methods and tensor fields in manifolds offer new ways of mining data spaces for information retrieval. The main assumption in manifold learning is that the data points, or samples, lye on a low-dimensional manifold $\mathcal{M}$, embedded in some high-dimensional space [4]. Therefore, dimensionality reduction should be used for discarding redundancy and reduce the computational cost of further operations. Manifold learning methods look for a compact data representation by recovering the data geometry and topology [5]. The manifold structure offers resources to build vector and tensor fields which can be applied to encode data features. Also, data points must appear according to some probability density function. In this way, geometry, topology and statistical learning could be considered as the building blocks of the *conceptual construction* mentioned in the above question [6], [7].

In this paper we focus on geometric and statistical learning elements related to tensor fields for data analysis. Therefore, we start with differentiable manifolds as the support for tensor fields [8]. Then, we cast in this geometric viewpoint the traditional algebraic framework for tensor methods used in multidimensional image databases analysis, based on generalized matrices. We review advanced aspects in tensor methods for data representation combined with statistical learning for extracting meaningful information from high-dimensional image databases.

Tensor methods offer an unified framework because vectors and matrices are first and second order tensors, respectively. Besides, colored images can be represented as third order tensors and so on. Then, multilinear methods can be applied for dimensionality reduction and analysis. Tensor representation for images was proposed in [9] by using a singular value decomposition method. There are supervised and unsupervised techniques in this field. The former is composed by methods based on scatter ratio maximization (discriminant analysis with tensor representation (DATER), uncorrelated multilinear discriminant analysis (UMLDA)), and scatter difference maximization (general tensor discriminant analysis (GTDA) and tensor rank-one discriminant analysis (TR1DA)). The unsupervised class (do not take class labels into account) includes the least square error minimization (concurrent subspace analysis (CSA), incremental tensor analysis (ITA), tensor rank-one decomposition (TROD)) and variance maximization (multilinear

principal component analysis (MPCA) and its variants) as well as multilinear independent components analysis (MICA) [10], [11], [12]. On the other hand, most of the mentioned techniques depends on iterative methods, which are sensitive to initialization, and involve NP-hard problems, which have motivated specific works as the ones reported in [13], [14].

The applications of multilinear data representation include face image analysis [15], face recognition under multiple viewpoints and varying lighting conditions, digital number recognition, video content representation and retrieval, face transfer that maps video recorded performances of one individual to facial animations of another one, gait recognition [11], geoscience and remote sensing data mining, visualization and computer graphics techniques based on tensor decomposition (see [16] and references therein).

This paper reviews the main topics involved in the application of tensor methods for image analysis: multilinear representation for high-dimensional data; dimensionality reduction; discriminant analysis; classification; reconstruction, that is, visualize the information captured by the discriminant tensor components. The presentation follows the references [11], [17], [15], [14], [18] to survey results that emphasize the power of tensor techniques for data representation, their advantage against linear counterparts and research challenges. We discuss the solution for the problem of subspace learning in tensor spaces in the context of MPCA. Next, pattern recognition tasks associated to tensor discriminant analysis and classification are presented. The problem of ranking tensor components to identify tensor subspaces for separating sample groups is focused on the context of statistical learning techniques, based on the covariance structure of the database, the Fisher criterion and discriminant weights computed through separating hyperplanes [15].

In the experimental results we focus on tensor components for face image analysis considering gender classification as well as reconstruction problems [17], [15]. We perform subspace learning through the MPCA. Then, the mentioned ranking techniques are used to select discriminant MPCA subspaces. We investigate the efficiency of the selected tensor components for gender classification and reconstruction problems using the FEI face image database[1]. We shall highlight the fact that the pipeline used to perform the gender experiment is not limited to image analysis. We can say that it is 'general' in the sense that, given a multidimensional database that can be represented by higher order tensors, it can be used to select discriminant subspaces for subsequent analysis.

Moreover, we add a section to discuss challenges and perspectives in the area. Our presentation is motivated by aspects related to data modeling and computational requirements. So, we return to the geometric viewpoint and show the link between manifold learning and tensor fields in order to sketch the application of such framework for data analysis. In addition, we discuss the problem of explicitly incorporating prior information in multilinear frameworks to allow an automatic selective treatment of the variables that compose the patterns of interest. However, further developments in tensor-based data mining will led to huge memory requirements and computational complexity issues which are also discussed in the perspective section.

This paper is organized as follows. Section II gives the algebraic and geometric background. Next, the section III reviews the tensor framework for dimensionality reduction. In section IV, the MPCA technique is presented. The approaches for ranking tensor components are described in section V. The section VI shows the experimental results. In Section VII, we discuss important points that have emerged from this work. Section VIII presents our concerns about open issues in tensor techniques for data analysis. Finally, we end with the conclusions in section IX. An extended version of the whole material is presented in [16].

II. Tensor Fields in Differentiable Manifolds

In this paper, the bold uppercase symbols represent tensor objects, such as $\mathbf{X}, \mathbf{Y}, \mathbf{T}$; the normal uppercase symbols represent matrices, data sets and subspaces (P, U, D, S, etc.); the bold italic lowercase symbols represent vectors, such as $\mathbf{x}$, $\mathbf{y}$; and the normal Greek lowercase symbols represent scalar numbers (λ, α, etc.).

The concept of tensor can be introduced through an elegant geometric framework, based in the notion of tensor product of vector spaces [19]. This approach is simple, straightforward, and it makes explicit the fact that the generalized matrices are just local representations of more general objects; named tensor fields.

A. Tensor Product Spaces

The simplest way to define the tensor product of two vector spaces V_1 and V_2, with dimensions $dim\,(V_1) = n$ and $dim\,(V_2) = m$, is by creating a new vector space analogously to multiplication of integers. For instance if $\{\mathbf{e}_1^1, \mathbf{e}_1^2, \mathbf{e}_1^3, ..., \mathbf{e}_1^n\}$ and $\{\mathbf{e}_2^1, \mathbf{e}_2^2, \mathbf{e}_2^3, ..., \mathbf{e}_2^m\}$ are basis in V_1 and V_2, respectively, then, the tensor product between these spaces, denoted by $V_1 \otimes V_2$, is a vector space that has the following properties:

1) Dimension:
$$dim\,(V_1 \otimes V_2) = n.m, \tag{1}$$

2) Basis:
$$V_1 \otimes V_2 = span\left\{\mathbf{e}_1^i \otimes \mathbf{e}_2^j; 1 \leq i \leq n, 1 \leq j \leq m\right\}, \tag{2}$$

3) Tensor product of vectors (Multilinearity): Given $\mathbf{v} = \sum_{i=1}^{n} v_i \mathbf{e}_1^i$ and $\mathbf{u} = \sum_{j=1}^{m} u_j \mathbf{e}_2^j$ we define:
$$\mathbf{v} \otimes \mathbf{u} = \sum_{i=1}^{n}\sum_{j=1}^{m} v_i u_j \mathbf{e}_1^i \otimes \mathbf{e}_2^j. \tag{3}$$

Generically, given vector spaces $V_1, V_2, \ldots, V_n$, such that $dim\,(V_i) = m_i$, and $\{\mathbf{e}_i^1, \mathbf{e}_i^2, ..., \mathbf{e}_i^{m_i}\}$ is a basis for V_i , we can define:
$$V_1 \otimes V_2 \otimes \ldots \otimes V_n$$

[1]This database can be downloaded from http://www.fei.edu.br/∼cet/facedatabase.html

$$= span\left\{\mathbf{e}_1^{i_1} \otimes \mathbf{e}_2^{i_2} \otimes \ldots \otimes \mathbf{e}_n^{i_n};\ \mathbf{e}_k^{i_k} \in V_k\right\}, \quad (4)$$

and the properties above can be generalized in a straightforward way.

B. Differentiable Manifolds and Tensors

A differentiable manifold of dimension m is a set $\mathcal{M}$ and a family of injections $\{\varphi_\alpha\}_{\alpha\in I}$, $\varphi_\alpha : U_\alpha \subset \mathbb{R}^m \to \mathcal{M}$ where U_α is an open set of $\mathbb{R}^m$ and I an index set, such that [8]:

1) $\cup_{\alpha\in I}\varphi_\alpha(U_\alpha) = \mathcal{M}$.

2) For every $\alpha, \beta \in I$, with $\varphi_\alpha(U_\alpha) \cap \varphi_\beta(U_\beta) = W \neq \emptyset$, the sets $\varphi_\alpha^{-1}(W)$ and $\varphi_\beta^{-1}(W)$ are open sets in $\mathbb{R}^m$ and the chart transition $\varphi_\beta^{-1} \circ \varphi_\alpha : \varphi_\alpha^{-1}(W) \to \varphi_\beta^{-1}(W)$ is differentiable.

3) The family $\{(U_\alpha, \varphi_\alpha)\}$ is maximal respect to properties (1) and (2).

The properties (1) and (2) allow to define a natural topology over $\mathcal{M}$: a set $A \subset \mathcal{M}$ is an open set of $\mathcal{M}$ if $\varphi_\alpha^{-1}(A \cap \varphi_\alpha(U_\alpha))$ is an open set of $\mathbb{R}^m$, $\forall\alpha \in I$.

We shall notice that, if $p \in \varphi_\alpha(U_\alpha)$ then $\varphi_\alpha^{-1}(p) = (x_1(p), ..., x_m(p)) \in \mathbb{R}^m$. So, $\varphi_\alpha(U_\alpha)$ is called a coordinate neighborhood and the pair $(U_\alpha, \varphi_\alpha)$ a system of local coordinates for $\mathcal{M}$ in p. If $\varphi_\beta^{-1} \circ \varphi_\alpha \in C^k$, with $k \geq 1$, $\forall\alpha, \beta \in I$, then we say that $\mathcal{M}$ is a $\mathbf{C}^k$ differentiable manifold, or simply $\mathbf{C}^k$ manifold. If $k = \infty$, $\mathcal{M}$ is called a smooth manifold.

Let $\mathcal{M}$ be a C^k manifold of dimension m with local coordinates $x : U \subset \mathbb{R}^m \to \mathcal{M}$, at a point p. A tangent vector $\mathbf{v}$ to $\mathcal{M}$ at p can be expressed in the local coordinates x as:

$$\mathbf{v} = \sum_{i=1}^{m}\left(v_i \frac{\partial}{\partial x_i}\right), \quad (5)$$

where the vectors:

$$B = \left\{\frac{\partial}{\partial x_1}, ..., \frac{\partial}{\partial x_m}\right\}, \quad (6)$$

are defined by the local coordinates.

The set of all tangent vectors to $\mathcal{M}$ at p is called the tangent space to $\mathcal{M}$ at p, and it is denoted by $T_p(\mathcal{M})$. The vectors in the set (6) determine an natural basis for $T_p(\mathcal{M})$. A vector field $\mathbf{v}$ over a manifold $\mathcal{M}$ is a function that associates to each point $p \in \mathcal{M}$ a vector $\mathbf{v}(p) \in T_p(\mathcal{M})$. Therefore, in the local coordinates x:

$$\mathbf{v}(p) = \sum_{i=1}^{m}\left(v_i(p) \frac{\partial}{\partial x_i}(p)\right), \quad (7)$$

where now we explicit the fact that expressions (5) and (6) are computed in each point $p \in \mathcal{M}$.

The notion of tensor field is formulated as a generalization of the vector field formulation using the concept of tensor product of section II-A. So, given the subspaces $T_p^i(\mathcal{M}) \subset T_p(\mathcal{M})$, with $dim\left(T_p^i(\mathcal{M})\right) = m_i$, $i = 1, 2, \cdots, n$, the tensor product of these spaces, denoted by $T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$, is a vector space defined by expression (4) with $V_i = T_p^i(\mathcal{M})$ and individual basis $\left\{\mathbf{e}_k^{i_k}(p), i_k = 1, 2, \cdots, m_k\right\} \subset T_p^k(\mathcal{M})$; that means, a natural basis for the vector space $T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$ is the set:

$$\left\{\mathbf{e}_1^{i_1}(p) \otimes \mathbf{e}_2^{i_2}(p) \otimes \cdots \otimes \mathbf{e}_n^{i_n}(p), \quad \mathbf{e}_k^{i_k}(p) \in T_p^k(\mathcal{M})\right\}. \quad (8)$$

In this context, a tensor $\mathbf{X}$ of order n in $p \in \mathcal{M}$ is defined as an element $\mathbf{X}(p) \in T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$; that is, an abstract algebraic entity that can be expressed as [19]:

$$\mathbf{X}(p) = \sum_{i_1, i_2, \cdots, i_n} \mathbf{X}_{i_1, i_2, \cdots, i_n}(p)\, \mathbf{e}_1^{i_1}(p) \otimes \mathbf{e}_2^{i_2}(p) \otimes \cdots \otimes \mathbf{e}_n^{i_n}(p). \quad (9)$$

Analogously to the vector case, a tensor field of order n over a manifold $\mathcal{M}$ is a function that associates to each point $p \in \mathcal{M}$ a tensor $\mathbf{X}(p) \in T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$. If the manifold $\mathcal{M}$ is an Euclidean space of dimension m, then $T_p(\mathcal{M})$ can be identified with the $\mathbb{R}^m$, and, consequently, $T_p^k(\mathcal{M})$ is identified with $\mathbb{R}^{m_k}$. So, in this case, we can discard the dependence to $p \in \mathcal{M}$ in expression (9) and write:

$$\mathbf{X} = \sum_{i_1, i_2, \cdots, i_n} \mathbf{X}_{i_1, i_2, \cdots, i_n} \mathbf{e}_1^{i_1} \otimes \mathbf{e}_2^{i_2} \otimes \cdots \otimes \mathbf{e}_n^{i_n}. \quad (10)$$

Therefore, we can say that the tensor $\mathbf{X}$ in expression (10) is just a generalized matrix $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$, in the sense that it can be represented by the n dimensional array $\mathbf{X}_{i_1, i_2, \cdots, i_n}$ everywhere in the Euclidean space. This is the context in which tensors have been used in computer vision, image processing and visualization techniques.

Each element of the basis given by expression (8), is called rank-1 tensor in the multilinear subspace learning literature because it equals to the tensor product of n vectors $\mathbf{e}_1^{i_1}(p) \otimes \mathbf{e}_2^{i_2}(p) \otimes \cdots \otimes \mathbf{e}_n^{i_n}(p)$ [13]. Expression (9), or its simplified version (10), gives the decomposition of the tensor $\mathbf{X}$ in the components of the tensor basis. In the sequel, we survey topics related to subspace learning and pattern recognition applied to multidimensional image data represented through tensors given by expression (10). Afterwards, in section VIII, we return to the general theory defined by equation (9) and discuss its perspectives for data analysis.

III. Dimensionality Reduction in Tensor Spaces

Now, let us consider the new basis:

$$\widetilde{B} = \left\{\widetilde{\mathbf{e}}_1^{i_1} \otimes \widetilde{\mathbf{e}}_2^{i_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{i_n}, \quad \widetilde{\mathbf{e}}_k^{i_k} \in \mathbb{R}^{m_k}\right\}, \quad (11)$$

as well as basis change matrices $R^k \in \mathbb{R}^{m_k \times m_k}$, defined by:

$$\mathbf{e}_k^{i_k} = \sum_{j_k=1}^{m_k} R_{i_k j_k}^k \widetilde{\mathbf{e}}_k^{j_k}, \quad (12)$$

where $k = 1, 2, \cdots, n$ and $i_k = 1, 2, \cdots, m_k$.

In the tensor product framework, to get the new representation of the tensor $\mathbf{X}$ in the basis $\widetilde{B}$ it is just a matter of inserting expression (12) in the tensor product representation

given by equation (10).Then, using the multilinearity of the tensor product (expression (3)) we get:

$$\mathbf{X} = \sum_{j_1,j_2,\cdots,j_n} \widetilde{\mathbf{X}}_{j_1,j_2,\cdots,j_n} \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \quad (13)$$

with:

$$\widetilde{\mathbf{X}}_{j_1,j_2,\cdots,j_n} = \sum_{i_1,i_2,\cdots,i_n} \mathbf{X}_{i_1,i_2,\cdots,i_n} R^1_{i_1 j_1} R^2_{i_2 j_2} \ldots . R^n_{i_n j_n}.$$

These expressions give the decomposition, or representation, of the tensor $\mathbf{X}$ in the basis $\widetilde{B}$. Now, let us perform dimensionality reduction by truncating the basis change matrices to get projection matrices $U^k \in \mathbb{R}^{m_k \times m\prime_k}$, as follows:

$$U^k_{i_k j_k} = R^k_{i_k j_k}, \quad i_k = 1, 2, \cdots, m_k; \quad j_k = 1, 2, \cdots, m\prime_k, \quad (14)$$

with $k = 1, 2, \cdots, n$, $m\prime_k \leq m_k$. Then, using again the multilinearity of the tensor product, we obtain a new tensor:

$$\mathbf{Y} = \sum_{j_1,j_2,\cdots,j_n} \mathbf{Y}_{j_1,j_2,\cdots,j_n} \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \quad (15)$$

where:

$$\mathbf{Y}_{j_1,j_2,\cdots,j_n} = \sum_{i_1,i_2,\cdots,i_n} \mathbf{X}_{i_1,i_2,\cdots,i_n} U^1_{i_1 j_1} U^2_{i_2 j_2} \ldots U^n_{i_n j_n}. \quad (16)$$

The operations in expression (16) can be represented through the usual mode-k product, applied when considering a tensor $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ as a generalized matrix [20], as follows.

Definition 1. *The mode-k product of tensor* $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ *with the matrix* $A \in \mathbb{R}^{m\prime_k \times m_k}$ *is given by:*

$$(\mathbf{X} \times_k A)_{i_1,\ldots,i_{k-1},i,i_{k+1},\ldots,i_n}$$
$$= \sum_{j=1}^{m_k} \mathbf{X}_{i_1,\cdots,i_{k-1},j,i_{k+1},\cdots i_n} A_{i,j}, \quad i = 1, 2, ..., m\prime_k. \quad (17)$$

So, we can show by induction that the expression (16) can be computed as:

$$\mathbf{Y}_{j_1,j_2,\cdots,j_n} = \left(\mathbf{X} \times_1 U^{1^T} \times_2 U^{2^T} ... \times_n U^{n^T} \right)_{j_1,j_2,\cdots,j_n}, \quad (18)$$

or, in a compact form:

$$\mathbf{Y} = \mathbf{X} \times_1 U^{1^T} \times_2 U^{2^T} ... \times_n U^{n^T}. \quad (19)$$

Expression (19) has the advantage of been more simple and compact than the algebraic counterpart given by equations (15)-(16). That is way it is used everywhere in the tensor literature for image analysis and visualization applications [10], [21]. Besides, the generalized array viewpoint simplifies the definition of other tensor operations also important in the context of multidimensional data manipulation by exploring the isomorphism between $\mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ and $\mathbb{R}^{m_1 \cdot m_2 \cdots m_n}$. For instance, the notions of internal product and norm in $\mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ are induced, in a natural manner, from the $\mathbb{R}^{m_1 \cdot m_2 \cdots m_n}$ space as follows.

Definition 2. *The internal product between two tensors* $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ *and* $\mathbf{Y} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ *is defined by:*

$$\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{i_1=1,\ldots,i_n=1}^{m_1,\ldots,m_n} \mathbf{X}_{i_1,..,i_n} \mathbf{Y}_{i_1,..,i_n} \quad (20)$$

Definition 3. *The Frobenius norm of a tensor is given by the expression* $\| \mathbf{X} \| = \sqrt{\langle \mathbf{X}, \mathbf{X} \rangle}$*, and the distance between tensors* $\mathbf{X}$ *and* $\mathbf{Y}$ *is computed by:*

$$D(\mathbf{X}, \mathbf{Y}) = \| \mathbf{X} - \mathbf{Y} \| . \quad (21)$$

Besides, operations that transform a multidimensional array into a two dimensional one (matrix) are also useful.

Definition 4. *The mode-k flattening of an n-th order tensor* $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$ *into a matrix* $X_{(k)} \in \mathbb{R}^{m_k \times \Pi_{i \neq k} m_i}$*, denoted by* $X_{(k)} \Longleftarrow_k \mathbf{X}$*, is defined by expression:*

$$X_{(k)_{i_k,j}} = \mathbf{X}_{i_1,\ldots,i_n}, \quad j = 1 + \sum_{l=1,l\neq k}^{n} (i_l - 1) \Pi_{0=l+1,0\neq k}^{n} m_0. \quad (22)$$

Finally, we shall notice that the tensor $\mathbf{Y}$ in expression (19) is the lower dimensional representation, or projection, of the tensor $\mathbf{X}$ in the reduced space $\mathbb{R}^{m\prime_1 \times m\prime_2 \times \ldots \times m\prime_n}$. Therefore, we must perform the reconstruction that generates a tensor $\mathbf{X}_i^R \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n}$, calculated by [20]:

$$\mathbf{X}_i^R = \mathbf{X}_i \times_1 U^1 U^{1^T} ... \times_n U^n U^{n^T}, \quad (23)$$

which, in visual databases, allows to verify how good a low dimensional representation will look like. We must define optimality criteria to seek for suitable matrices U^1, U^2,...,U^n. The next section revise this point using statistical learning methods.

IV. Subspace Learning in Tensor Spaces

Now, let us consider a database D with N elements that can be represented through $n-th$ order tensors:

$$D = \left\{ \mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \ldots \times m_n},\ i = 1, 2, ..., N \right\}. \quad (24)$$

We can fit the problem of dimensionality reduction in the context of subspace learning approaches by calculating n projection matrices that maximize the total scatter; that means, they perform variance maximization by solving the problem [11]:

$$(U^j|_{j=1}^n) = \arg \max_{U^j|_{j=1}^n} \frac{1}{N} \sum_{i=1}^{N} ||\mathbf{Y}_i - \overline{\mathbf{Y}}||^2, \quad (25)$$

where $\mathbf{Y}_i$ is given by expression (19) and $\overline{\mathbf{Y}}$ is the mean tensor computed by:

$$\overline{\mathbf{Y}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{Y}_i. \quad (26)$$

The algorithm to solve problem (25) is steered by the following result.

Theorem 1. *Let $U^k \in \mathbb{R}^{m_k \times m\prime_k}$ $i = 1, 2, \ldots, n$, be the solution to (25). Then, given the projection matrices $U^1, ..., U^{k-1}, U^{k+1}, ..., U^n$, the matrix U^k consists of the $m\prime_k$ principal eigenvectors of the matrix:*

$$\Phi^{(k)} = \sum_{i=1}^{N}(X_{i(k)} - \overline{X}_{(k)})U_{\Phi^{(k)}}.U^T_{\Phi^{(k)}}.(X_{i(k)} - \overline{X}_{(k)})^T, \quad (27)$$

where $X_{i(k)}$ and $\overline{X}_{(k)}$ are the mode-k flattening of sample tensor X_i and of the global mean $\overline{X}$, respectively, and:

$$U_{\Phi^{(k)}} = U^{k+1} \otimes U^{k+2} \otimes ... \otimes U^n \otimes U^1 \otimes U^2 \otimes ... \otimes U^{k-1}. \quad (28)$$

Proof: see [11], [22].

The MPCA is formulated based on the variance maximization criterion. Hence, starting from the Theorem 1, the iterative procedure given by the Algorithm 1 can be implemented to solve problem (25). In this procedure, called MPCA Algorithm in [11], the projection matrices are computed one by one with all the others fixed. Once the matrices $U_t^1, ..., U_t^{k-1}, U_t^{k+1}, ..., U_t^n$ are fixed we can seek for the optimum U_t^k by solving the optimization problem given by equation (25) respect to the matrix U_t^k. Then, the objective function in expression (25), can not decrease after each interaction.

The reduced dimensions $m_k^{'}$, $k = 1, 2, \ldots, n$ must be specified in advance or determined by some heuristic. In [11] it is proposed to compute these values in order to satisfy the criterion:

$$\frac{\sum_{i_k=1}^{m'_k}\lambda_{i_k(k)}}{\sum_{i_k=1}^{m_k}\lambda_{i_k(k)}} > \Omega \quad (29)$$

where Ω is a threshold to be specified by the user and $\lambda_{i_k(k)}$ is the $i_k th$ eigenvalue of $\Phi^{(k)*}$.

The total scatter Υ (Algorithm 1, lines 5 and 9) is related to the total scatter tensor defined by [11]:

$$\mathbf{\Psi}_{j_1,j_2,\cdots,j_n} = \sum_{i=1}^{N}\frac{\left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right)^2}{N}, \quad (30)$$

which offers also a straightforward way to rank tensor components as we will see in section VI.

V. Ranking Tensor Components

The problem of ranking components is very known in the context of PCA. In this case, it was observed that, since PCA explains the covariance structure of all the data its most expressive components, that is, the first principal components

Algorithm 1: MPCA Algorithm.

Input: Samples $\{\mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times ... \times m_n},\ i = 1, ..., N\}$; dimensions $m_k^{'}$; $k = 1, \cdots, n$.

1 **Preprocessing:** Center the input samples as $\{\widetilde{\mathbf{X}}_i = \mathbf{X}_i - \overline{\mathbf{X}}, i = 1, ..., N\}$, where $\overline{\mathbf{X}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_i$ is the sample mean.

2 Initialization: Calculate the eigen-decomposition of $\Phi^{(k)*} = \sum_{i=1}^{N}\widetilde{\mathbf{X}}_{i(k)} \cdot \widetilde{\mathbf{X}}_{i(k)}^T$, and set U_0^k to consist of the eigenvectors corresponding to the most significant $m_k^{'}$ eigenvalues, for $k = 1, ..., n$.

3 **Local optimization:**

4 Calculate $\widetilde{\mathbf{Y}}_i = \widetilde{\mathbf{X}}_i \times_1 U_0^{1^T} ... \times_n U_0^{n^T}, i = 1, ..., N$;

5 Calculate $\Upsilon_0 = \sum_{i=1}^{N}||\widetilde{\mathbf{Y}}_i||_F^2$;

6 **for** $t = 1, ...$ *to* T_{max} **do**

7 **for** $k = 1, ...$ *to* n **do**

8 Set the matrix U_t^k to consist of the $m_k^{'}$ leading eigenvectors of $\Phi^{(k)}$, defined in expression (27);

9 Calculate $\widetilde{\mathbf{Y}}_i, i = 1, ..., N$ and Υ_t;

10 **if** $|\Upsilon_t - \Upsilon_{t-1}| < \eta$ **then**

11 break;

Output: Projection matrices $U^k = U_t^k, k = 1, ..., n$.

with the largest eigenvalues, do not necessarily represent important discriminant directions to separate sample groups [23]. The Figure 1 is a simple example that pictures this fact. Both Figures 1.(a) and 1.(b) represent the same data set. Figure 1.(a) just shows the PCA directions ($\overline{x}$ and $\overline{y}$) and the distribution of the samples over the space. However, in Figure 1.(b) we distinguish two patterns: plus $(+)$ and minus $(-)$. We observe that the principal PCA direction $\overline{x}$ can not discriminate samples of the considered groups.

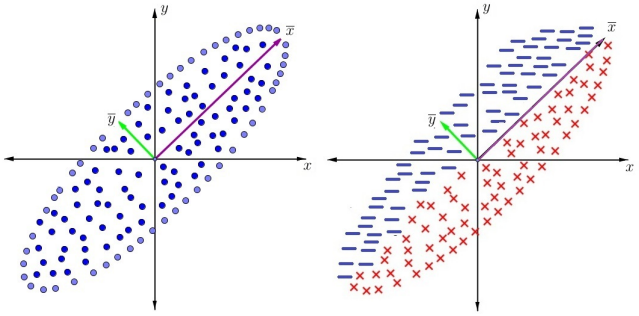


Fig. 1. (a) Data distribution and PCA directions. (b) The same population but distinguishing patterns plus $(+)$ and minus $(-)$.

This observation motivates the application and development of other techniques, like linear discriminant analysis (LDA), or its regularized version named MLDA, and discriminant principal components analysis (DPCA), to identify the most important linear directions for separating sample groups rather than PCA [24], [25]. Behind the LDA is the Fisher criterion which have inspired ranking methods for tensor components

[11]. The DPCA has been extended for tensor databases generating the TDPCA [17], [15]. The extension of the maximum variance criterion for ranking MPCA tensor components is ambiguous because there is no a closed-form solution for the corresponding subspace learning problems unless in particular cases [17]. Therefore, we shall consider some sound methodology to estimate the covariance structure of tensor subspaces [17]. In the following subsections we revise these approaches.

A. Spectral Structure of MPCA Subspaces

The first point is how to estimate the variance explained by each tensor component? In the MPCA algorithm each subspace:

$$\left\{\widetilde{\mathbf{e}}_k^{j_k}, \quad j_k = 1, 2, \cdots, m\prime_k\right\}, \quad k = 1, 2, \cdots, n,$$

is obtained by taking the first $m\prime_k$ leading eigenvectors of a covariance-like matrix $\Phi^{(k)}$. So, let:

$$\left\{\lambda_{j_k}^k, \quad j_k = 1, 2, \cdots, m\prime_k\right\}, \quad k = 1, 2, \cdots, n,$$

the associated eigenvalues. The data distribution in each subspace can be represented by the vector:

$$\mathbf{v}_k = \frac{1}{\beta}\sum_{j_k=1}^{m\prime_k} \lambda_{j_k}^k \widetilde{\mathbf{e}}_k^{j_k}, \quad k = 1, 2, \cdots, n,$$

where β is a normalization factor that takes into account the tensor space dimension and the number N of samples: $\beta = N \cdot \Pi_{k=1}^n m_k$. Therefore, following [17], the variance explained by each element of basis $\widetilde{B}$ in expression (11) can be estimated by calculating:

$$\mathbf{v}_1 \otimes \mathbf{v}_2 \otimes \cdots \otimes \mathbf{v}_n$$
$$= (\frac{1}{\beta})^n \sum_{j_1, j_2, \cdots, j_n} \lambda_{j_1}^1 \lambda_{j_2}^2 \cdots \lambda_{j_n}^n \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}. \quad (31)$$

and, consequently, we can rank the MPCA tensor components by sorting (in decreasing order) the set:

$$E = \left\{\lambda_{j_1, j_2, \cdots, j_n} = (\frac{1}{\beta})^n \lambda_{j_1}^1 \lambda_{j_2}^2 \cdots \lambda_{j_n}^n,\ j_k = 1, 2, \cdots, m\prime_k\right\}, \quad (32)$$

to obtain the principal MPCA tensor components, likewise in the PCA methodology. The sorted sequence $\{\lambda_{j_1, j_2, \cdots, j_n}\}$ can be re-indexed using just one index $\{\lambda_i;\ i = 1, 2, ..., \prod_{k=1}^n m\prime_k\}$ that tells the number of the principal MPCA component in the sorted sequence.

B. Tensor Discriminant Principal Components

The problem of using a linear classifier to rank tensor components, named tensor discriminant principal components analysis (TDPCA) in [17], consists of the following steps. Firstly, we perform dimensionality reduction using the MPCA subspaces, by computing the low dimensional representation $\mathbf{Y}_i$ of each tensor $\mathbf{X}_i$ through expression (19). The goal of this step is to discard redundancies of the original representation. Then, supposing a two-class problem, a linear classifier is estimated using the projected training samples $\mathbf{Y}_i \in \mathbb{R}^{m\prime_1 \times m\prime_2 \times \ldots \times m\prime_n}$ and the corresponding labels. The separating hyperplane is defined through a discriminant tensor $\mathbf{W} \in \mathbb{R}^{m\prime_1 \times m\prime_2 \times \ldots \times m\prime_n}$ while the discriminant features given by:

$$\tilde{y}_i = \langle \mathbf{Y}_i, \mathbf{W} \rangle, \quad (33)$$

for $i = 1, 2, \cdots, N$, are used for classification.

We can investigate the components $\mathbf{W}_{i_1, .., i_n}$ of the discriminant tensor $\mathbf{W}$ to determine the discriminant contribution for each feature. So, following the same idea proposed in [23] for PCA subspaces, these components are weights in expression (33) that determine the discriminant contribution of each feature $\mathbf{Y}_{i; i_1, .., i_n}$; that means, weights that are estimated to be approximately 0 have negligible contribution on the discriminant scores $\tilde{y}_i$ described in equation (33), indicating that the corresponding features are not significant to separate the sample groups. In contrast, largest weights (in absolute values) indicate that the corresponding features contribute more to the discriminant score and consequently are important to characterize the differences between the groups.

Therefore, differently from the criterion of section V-A, we are selecting among the MPCA components the "directions" that are efficient for discriminating the sample groups rather than representing all the samples. Such a set of components ordered in decreasing order of the discriminant weights is called the tensor discriminant principal components [17], [15]. The TDPCA with discriminant tensor given by the SVM (MLDA) linear classifier is named TDPCA-SVM (TDPCA-MLDA) in the following sections [15].

C. Fisher Criterion for Tensor Component Analysis

The key idea of Fisher criterion is to separate samples of distinct groups by maximizing their between-class separability while minimizing their within-class variability [24]. It is implemented in [26] through the discriminant tensor criterion given by:

$$(U^1, U^2, \ldots, U^n)$$
$$= \arg\max_{U^j|_{j=1}^n} \frac{\sum_{c=1}^C N_c \cdot || \left(\mathbf{X}_c - \overline{\mathbf{X}}\right) \times_1 U^1 ... \times_n U^n||^2}{\sum_{i=1}^N || \left(\mathbf{X}_i - \overline{\mathbf{X}}_{c_i}\right) \times_1 U^1 ... \times_n U^n||^2}, \quad (34)$$

where C is the number of classes, N_c is the number of elements of class c, $\mathbf{X}_c$ is the average tensor of the samples belonging to class c, $\overline{\mathbf{X}}$ is the average tensor of all the samples and $\overline{\mathbf{X}}_{c_i}$ is the average of the class corresponding to the ith tensor.

Following the expressions (15)-(16), and by considering Frobenius norm (Definition 3) we can interchange the summations to get:

$$(U^1, U^2, \ldots, U^n)$$
$$= \arg\max_{U^j|_{j=1}^n} \frac{\sum_J \left(\sum_{c=1}^C N_c \cdot \left(\mathbf{Y}_{c;J} - \overline{\mathbf{Y}}_J\right)^2\right)}{\sum_J \left(\sum_{i=1}^N \left(\mathbf{Y}_{i;J} - \overline{\mathbf{Y}}_{c_i;J}\right)^2\right)}, \quad (35)$$

where $J = (j_1, j_2, \cdots, j_n)$. Expression (35) motivates the discriminability criterion presented in [11] without a formal

(a) (b)

Fig. 2. The frontal neutral profile of an individual in the FEI database. (b) Frontal profile of the mean tensor for the FEI database (gray scale).

justification. Specifically, from expression (35) we can postulate that the larger is the value of $\Gamma_{j_1,j_2,\cdots,j_n}$ computed by:

$$\Gamma_{j_1,j_2,\cdots,j_n} = \frac{\sum_{c=1}^{C} N_c \cdot \left(\mathbf{Y}_{c;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{j_1,j_2,\cdots,j_n}\right)^2}{\sum_{i=1}^{N} \left(\mathbf{Y}_{i;j_1,j_2,\cdots,j_n} - \overline{\mathbf{Y}}_{c_i;j_1,j_2,\cdots,j_n}\right)^2}, \quad (36)$$

then more discriminant is the tensor component $\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}$ for samples classification.

VI. Application: FEI Database Analysis

In this section we perform gender experiments using the face image database maintained by the Department of Electrical Engineering of FEI, São Bernardo do Campo (SP), Brazil [2]. There are 14 images for each of 200 individuals, a total of 2800 images. All images are colorful and taken against a white homogenous background in an upright frontal position with 11 neutral profile rotation of up to about 180 degrees, one facial expression (smile) and two poses with variations in illumination. Scale might vary about 10% and the original size of each image is 640×480 pixels. All faces are mainly represented by students and staff at FEI, between 19 and 40 years old with distinct appearance, hairstyle, and adorns. The number of male and female subjects are exactly the same and equal to 100 [23]. Figure 2.(a) shows the frontal profile of a sample that has been used in the experiments (more examples in [16]).

For memory requirements, we convert each pose to gray scale before computations. Therefore, each sample is represented by a tensor $\mathbf{X} \in \mathbb{R}^{640} \otimes \mathbb{R}^{480} \otimes \mathbb{R}^{11}$ and the frontal pose of the mean tensor is pictured in Figure 2.(b). The images are well scaled and aligned. These features make the FEI database very attractive for testing dimensionality reduction in tensor spaces.

Details about the implementation of the mentioned tensor methods are given in the section VI-A. We have carried out experiments to understand the main differences between the tensor principal components selected by the target methods (section VI-B). Then, in sections VI-C and VI-D, we have investigated the effectiveness of the ranking approaches for recognition and reconstruction tasks, respectively. In this paper, we focus on MPCA subspaces. Comparisons with CSA results can be found in [15].

[2]This database can be downloaded from http://www.fei.edu.br/∼cet/facedatabase.html

A. Implementation Details

We consider the following tensor subspaces generated by the MPCA procedure described in the Algorithm 1: (a) Subspace S_1^{mpca}, obtained with $m\prime_1 = 479$ for $m\prime_2 = 639$ and $m\prime_3 = 11$; (b) Tensor subspace S_2^{mpca} obtained by setting the dimensionality $m\prime_1 = 33$, $m\prime_2 = 42$ and $m\prime_3 = 9$, computed by expression (29) with $\Omega = 0,97$, according to [11].

These components are sorted following the rules of section V. We have observed that the spectral variances computed in expression (32) are not null. Therefore, we retain all the tensor components performing tensor subspaces with dimension $3366891 \approx 3,36 \cdot 10^6$ for S_1^{mpca} and dimension $33 \cdot 42 \cdot 9 = 12474$ for the subspace S_2^{mpca}. Following equation (12), each obtained set of projection matrices generates a new basis $\widetilde{B}$ like the one in expression (11) and a tensor $\mathbf{X}$ can be represented in $\widetilde{B}$ by a tensor $\mathbf{Y}$ computed by expression (15). Besides, each ranking technique induces a natural order in the set $I = \{(j_1, j_2, \cdots, j_n), \quad j_k = 1, 2, \cdots, m\prime_k\}$, that means, $(j_1, j_2, \cdots, j_n)$ preceed $\left(\widehat{j}_1, \widehat{j}_2, \cdots, \widehat{j}_n\right)$ if $\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}$ is ranked higher than $\widetilde{\mathbf{e}}_1^{\widehat{j}_1} \otimes \widetilde{\mathbf{e}}_2^{\widehat{j}_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{\widehat{j}_n}$ according to the considered ranking method.

Let, $\hat{I} \subset I$ and the subspace $\widehat{B} = \left\{\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \quad (j_1, j_2, \ldots, j_n) \in \hat{I}\right\} \subset \widetilde{B}$. The projection of a tensor $\mathbf{X}$ in $\widehat{B}$ is a tensor $\widehat{\mathbf{Y}}$ that can be simply computed by:

$$\widehat{\mathbf{Y}} = \sum_{j_1,j_2,\ldots,j_n} \mathbf{T}_{j_1,j_2,\cdots,j_n} \times \mathbf{Y}_{j_1,j_2,\cdots,j_n} \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}, \quad (37)$$

where, the characteristic tensor $\mathbf{T}$ is defined by:

$$\mathbf{T}_{j_1,j_2,\cdots,j_n} = \begin{cases} 1, & if\ \widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n} \in \widehat{B}, \\ 0, & otherwise. \end{cases} \quad (38)$$

If we denote $\widehat{\mathbf{Y}}_{j_1,j_2,\cdots,j_n} = \mathbf{T}_{j_1,j_2,\cdots,j_n} \times \mathbf{Y}_{j_1,j_2,\cdots,j_n}$ then the corresponding reconstruction is obtained by computing:

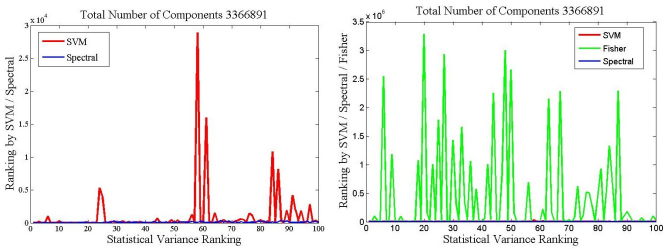
$$\mathbf{X}^R = \widehat{\mathbf{Y}} \times_1 U_1 \times_2 U_2 \times_3 \ldots \times_n U_n. \quad (39)$$

Another important element for the discussions of the subsequent sections is the application of the scatter tensor defined by expression (30) to rank tensor components of $\widetilde{B}$ by sorting (in decreasing order) the set:

$$V = \{\Psi_{j_1,j_2,\cdots,j_n}, \quad j_k = 1, 2, \cdots, m\prime_k\}. \quad (40)$$

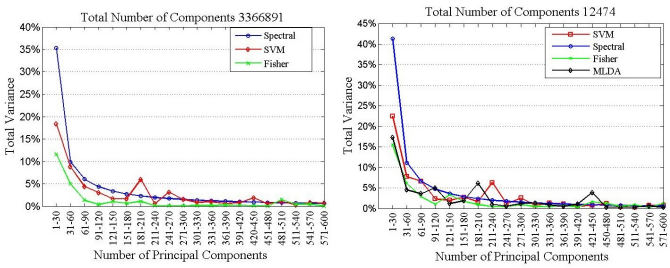
Therefore, $\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}$ is ranked higher than $\widetilde{\mathbf{e}}_1^{\widehat{j}_1} \otimes \widetilde{\mathbf{e}}_2^{\widehat{j}_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{\widehat{j}_n}$ if $\Psi_{j_1,j_2,\cdots,j_n} > \Psi_{\widehat{j}_1,\widehat{j}_2,\cdots,\widehat{j}_n}$. We shall observe that $\Psi_{j_1,j_2,\cdots,j_n}$ is the statistical variance associated to the component $\widetilde{\mathbf{e}}_1^{j_1} \otimes \widetilde{\mathbf{e}}_2^{j_2} \otimes \cdots \otimes \widetilde{\mathbf{e}}_n^{j_n}$. Therefore, when using this criterion we say that we are ranking tensor components by their statistical variances.

In this work, the tensor operations are implemented using the functions of the tensor toolbox available at [27] for MatLab. Examples of tensor operations as well as other library functions can be also found in [27].



(a) (b)

Fig. 3. Subspace S_1^{mpca}: (a) Top 100 statistical variance tensor principal components (horizontal axis), ranked by SVM hyperplane and spectral variance, using the FEI database. (b) The same information of figure (a) but including the ranking by Fisher criterion.



(a) (b)

Fig. 4. (a) Amount of total variance explained by the 600 S_1^{mpca} most expressive tensor components selected by spectral variance, TDPCA-SVM, and Fisher criteria. (b) Amount of total variance using the 600 most expressive tensor components of S_2^{mpca}, including also the TDPCA-MLDA components.

B. Ranking and Understanding Tensor Components

Now, we determine the discriminant contribution of each tensor component in the S_1^{mpca} by investigating its ranking respect to each approach presented on section V. Due to memory restrictions we could not include the TDPCA-MLDA criterion in this case. In the horizontal axis of Figures 3.(a)-(b), we read the sorting by statistical variance, as described above (expression (40)). Then, we consider each criterion of section V and compute the new rank of each component. For instance, the $25th$ tensor component respect to the statistical variance was ranked as the $16th$ for the spectral, the $1782898th$ and $3873th$ when using the Fisher and TDPCA-SVM criteria, respectively. This figure shows that the ranking obtained by the spectral variance is the closer one to the statistical variance method. A similar behaviour was observed for the S_2^{mpca} subspace. This observation experimentally confirms that expression (31) can be used to estimate the variance explained by each tensor component. Therefore, we only consider the spectral variance from here to the end of this section.

We observe also from Figure 3.(b) that the Fisher criterion is the one that most disagree with the statistical variance. This is also observed for the TDPCA-SVM ranking, although less intensely. For instance, the $15th$ principal statistical variance component were ranked as $13th$ and $10294th$ by the SVM and Fisher criteria and the $21th$ principal statistical variance component was ranked as the $15th$, $164114th$ by the SVM and Fisher criteria, respectively. As pointed out in [23], this behaviour is related with the fact that, the first principal components with the largest variances are not necessarily the most discriminant ones. On the other hand the $59th$ respect to the statistical variance was ranked as the first one respect to the Fisher and $4th$ respect to the TDPCA-SVM. Since principal components with lower variances describe particular information related to few samples, these results indicate the ability of TDPCA and Fisher for zooming into the details of group differences.

The total variance explained by the 600 most expressive tensor components for the gender experiment is illustrated in Figure 4 when using the considered subspaces. This figure shows that as the dimension of the spectral most expressive subspace increases, there is an exponential decrease in the amount of total variance explained by the first spectral tensor principal components, as already observed in [17]. However, the corresponding variances explained by the tensor principal components selected by the other criteria do not follow the same behavior. Specifically, we observe in Figure 4.(a) some oscillations along the TDPCA-SVM and Fisher spectrum. For instance, the amount of the total variance explained by the $181-210$ TDPCA-SVM components is a local maximum for the corresponding distribution. The Fisher spectrum also presents some oscillations in the amount of the total variance explained between the $91th$ and $240th$ principal components and shows a local maximum for $481-510$ tensor components. Oscillations also appear in the TDPCA-MLDA plot in Figure 4.(b).

In order to quantify the discriminant power of the principal components, we present in Figure 5 the amount of total discriminant information, in descending order, explained by each one of the first 400 tensor principal components selected by the TDPCA and Fisher approaches. The proportion of total discriminant information t described by the k^{th} tensor principal component can be calculated as follows:

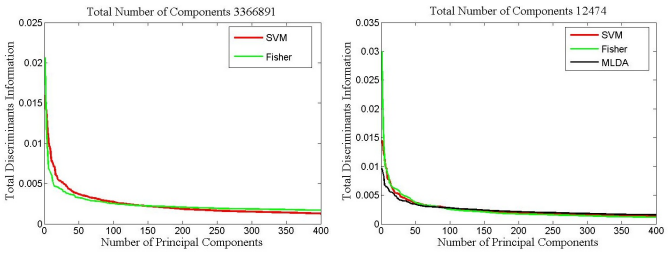
$$t_k = \frac{|\sigma_k|}{\sum_{j=1}^{m} |\sigma_j|}, k = 1, 2, \cdots, m, \tag{41}$$

where m is the subspace dimension, and $[\sigma_1, \sigma_2, ..., \sigma_m]$ are the weights computed by the MLDA/SVM separating hyperplane (section V-B) as well as the Fisher criterion (expression(36)).

Figure 5 shows that as the dimension of the TDPCA-SVM, TDPCA-MLDA and Fisher subspace increases there is an exponential decrease in the amount of total discriminant information described by the corresponding principal components.

C. Recognition Rates in Gender Experiments

In this section, we have compared the effectiveness of the tensor principal components ranked according to the spectral variance, Fisher criterion and TDPCA techniques (section V) on recognition tasks. The 10-fold cross validation method is used to evaluate the classification performance of the tensor subspaces. In these experiments we have assumed equal prior probabilities and misclassification costs for both groups. We



(a) (b)

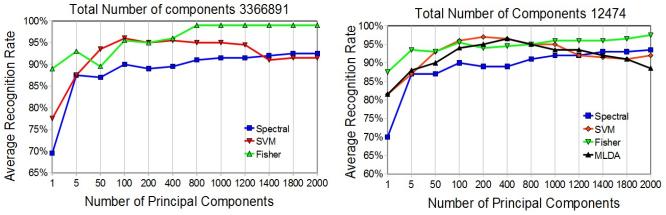
Fig. 5. (a) Amount of total discriminant information calculated in equation (41) and explained by TDPCA-SVM and Fisher tensor principal components for gender experiment for S_1^{mpca} subspace. (b) Analogous, but now using S_2^{mpca} subspace and including the TDPCA-MLDA.

consider the tensor subspaces S_1^{mpca} and S_2^{mpca}, as defined in section VI-A, and the ranking techniques to sort the corresponding basis. Generically, let $\widetilde{B}_{mpca}$ the obtained sorted basis. Then, we take the sequence of subspaces $\widehat{B}_j \subset \widetilde{B}_{mpca}$, where $\widehat{B}_1$ contains only the first tensor component of $\widetilde{B}_{mpca}$; $\widehat{B}_2$ contains the first and second tensor principal components of $\widetilde{B}_{mpca}$ and so on. Given a test observation $\mathbf{X}_t \in D$, it is projected in the subspace $\widetilde{B}_{mpca}$ through expression (15), generating a tensor $\mathbf{Y}_t$. We perform the same operation for each class mean $\overline{\mathbf{X}}_i$ generating the tensor $\overline{\mathbf{Y}}_i$. Next, we compute the Mahalanobis distance from $\mathbf{Y}_t$ to $\overline{\mathbf{Y}}_i$ to assign that observation to either the male or female groups. That is, we have assigned $\mathbf{Y}_t$ to class i that minimizes:

$$d_i^k(\mathbf{Y}_t) = \sum_{j=1}^{k} \frac{1}{\lambda_j} (\mathbf{Y}_{t;j} - \overline{\mathbf{Y}}_{i;j})^2 \qquad (42)$$

where λ_j is the corresponding spectral variance, k is the number of tensor principal components retained and $\mathbf{Y}_{t;j}$ is the projection of tensor $\mathbf{Y}_t$ in the jth tensor component (the same for $\overline{\mathbf{Y}}_{i;j}$). In the recognition experiments, we have considered different number of tensor principal components ($k = 1, 5, 50, 100, 200, 400, 800, 1000, 1200, 1400, 1800, 2000$) to calculate the recognition rates of the corresponding methods of selecting principal components.

Figure 6 shows the average recognition rates of the 10-fold cross validation of the gender experiments using the FEI database with the considered subspaces (be careful about the non-uniform scale of the horizontal axis). Figure 6.(a) shows that for the smallest numbers of components considered ($1 \leq k \leq 5$) the Fisher criterion achieves higher recognition rates than the other ones for S_1^{mpca}. However, when taking $30 < k < 100$ the TDPCA-SVM approach performs better than Fisher and spectral methods. For $k \in [100, 400]$ the Fisher criterion and TDPCA-SVM perform close to each other and they outperform all the others selected tensor subspaces. The recognition rates in all the experiments reported in Figure 6.(a) fall into the range $[0.7, 0.99]$ with the maximum achieved by the Fisher method for $k \geq 800$. In the case of S_2^{mpca} subspace, we can observe in Figure 6.(b) that the Fisher outperforms the other ones for $1 \leq k < 50$. Then, the TDPCA-SVM recognition rates equal the Fisher accuracy for $50 < k < 100$, while for $100 < k < 400$ the TDPCA-SVM is the best one. In the range $400 < k < 800$ both TDPCA-SVM and TDPCA-MLDA achieve the same accuracy and they outperform the counterparts in this interval. Finally, for $800 < k \leq 2000$ the Fisher criterion becomes the best among all the other methods again.



(a) (b)

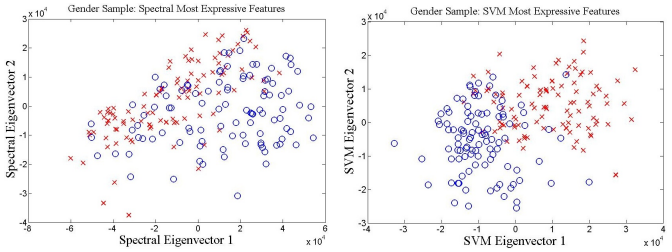
Fig. 6. (a)-(b) Average recognition rates for S_1^{mpca} (S_2^{mpca}, for (b)) components selected by the largest spectral variances, TDPCA-SVM, TDPCA-MLDA, and Fisher criteria.

When comparing the plots of Figure 6.(a)-(b) we observe that the Fisher criterion gets better recognition rates for the smallest subspace dimensions ($k \leq 5$). Then, it is clearly outperformed for the TDPCA in some closed interval $[k_1, k_2]$ and, finally, the Fisher criterion achieves the highest recognition rates, or is equivalent to TDPCA, for $k_2 \leq k \leq 2000$. The values for k_1 and k_2 are given on Table I.

Figure	k_1	k_2
6.(a)	32	100
6.(b)	100	800

TABLE I
TABLE FOR THE INTERSECTION POINTS k_1 AND k_2.

The Figures 6.(a)-(b) show that the spectral is, in general, outperformed by the other methods for low values of k. Then, its recognition rates increases and it achieves the highest performance with $k = 2000$ as we can observe in Figures 6.(a)-(b). We must be careful about the fact that the spectral is the only unsupervised method considered in this paper. This fact impacts the recognition rates because there is not incorporation of prior information in the spectral process for ranking tensor components. Besides, since the spectral method works with the covariance structure of all the data its most expressive components, that is, the first principal components with the largest (estimated) variances, are not necessarily the most discriminant ones. This fact can be visualized in Figure 7 which shows the tensor samples from the FEI database projected on the first two principal components selected by spectral for S_2^{mpca} and selected by the TDPCA-SVM for the same subspace. A visual inspection of this figure shows that spectral (Figures 7.(a)) clearly fails to recover the important features for separating male from female samples if compared with the TDPCA-SVM result pictured on Figure 7.(b). This observation agrees with the recognition rates reported for TDPCA-SVM if compared with the recognition rates for spectral observed on Figure 6.(b), for $k < 5$.



(a) (b)

Fig. 7. (a) Two-dimensional most expressive subspace identified by spectral for the gender experiment (male "+" and female "o") for S_2^{mpca}. (b) Two-dimensional most expressive subspace selected by TDPCA-SVM and the projected samples, for S_2^{mpca}.

Also, when $k > 1000$ the Fisher technique achieves recognition rates close to 100% for all reported experiments. For $k < 800$ the Fisher, LDA and SVM criteria performs better or equivalent than the spectral ones for both MPCA subspaces, being close to each other for $100 \leq k \leq 400$ in Figure 6.(a) and for $100 \leq k \leq 800$ in Figure 6.(b). This observation agrees with the fact that the total discriminant information pictured by Figure 5 does not indicate significant differences for these techniques in the range $50 \leq k \leq 400$.

The Figures 6.(a)-(b) show that for both the MPCA subspaces the Fisher criterion outperforms the other ones for $1 \leq k \leq 5$, for classification tasks. If considered the spectral criterion, such result agrees with the fact that the first principal components with the largest variances, do not necessarily represent important discriminant "directions" to separate sample groups.

D. Tensor Components and Reconstruction

Firstly, we shall study the reconstruction error, which is quantified through the root mean squared error (RMSE), computed as follows:

$$RMSE\left(\widehat{B}\right) = \sqrt{\frac{\sum_{i=1}^{N} ||\mathbf{X}_i^R - \mathbf{X}_i||^2}{N}}, \qquad (43)$$

where $\widehat{B}$ is the subspace for projection.

The Figure 8 shows the RMSE of the reconstruction process for the subspace S_2^{mpca} with tensor components sorted according to the ranking techniques of section V, generating the basis $\widetilde{B}_{mpca_2}$. To build the plots in Figure 8 we follow the same procedure explained in the beginning of section VI-C and take a sequence of principal subspaces $\widehat{B}_j \subset \widetilde{B}_{mpca_2}$. Next, each tensor $\mathbf{X}_i \in D$ is projected in the subspace $\widetilde{B}_{mpca_2}$ through expression (15), generating a tensor $\mathbf{Y}_i$ which is projected on each basis $\widehat{B}_j$ following equation (37). Each obtained tensor $\widehat{\mathbf{Y}}_i$ is reconstructed, generating the tensor named $\mathbf{X}_i^R$ above, following equation (39). Finally, we compute the sequence $RMSE\left(\widehat{B}_j\right)$ and plot the points $\left(j, RMSE\left(\widehat{B}_j\right)\right)$, $j = 1, 5, \ldots$ to obtain each line in Figure 8. We use a non-uniform scale in the horizontal axis in order to clarify the visualization.

We observe from Figure 8 that the RMSE for spectral is lower than the RMSE for the other methods. So, while in the classification tasks the spectral is, in general, outperformed by the other methods, in the reconstruction the spectral becomes the most efficient technique. On the other hand, we observe from Figure 8 that the RMSE for Fisher is higher or equivalent than the RMSE for the other ranking techniques. Therefore, the superiority of the Fisher criterion for classification is not observed for reconstruction tasks.

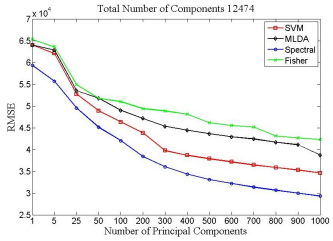
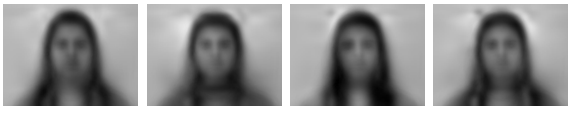


Fig. 8. RMSE for subspaces $\widehat{B}_j \subset S_2^{mpca}$ selected by TDPCA-SVM (red line), TDPCA-MLDA (black line), Spectral (blue line) and Fisher (green line).

In order to visualize these facts we picture in Figure 9 the frontal pose of the reconstruction of a sample of the FEI database using the first 100 tensor principal components selected by each one of the considered ranking techniques. The original image is shown on Figure 2.(a) and the Figures 9.(a)-(d) show the obtained reconstruction when using $\widehat{B}_{100} \subset S_2^{mpca}$. However, we must be careful when comparing the reconstruction results. In fact, we can visually check the fact that Figures 9.(a),(b),(d) are more related with the mean image (Figure 2.(b)) then with the test sample shown in Figure 2.(a) while Figure 9.(c) is less deviated to the mean. Taking this fact into account, a visual inspection shows that the reconstruction based on spectral tensor principal components gives a better reconstruction, as indicated by Figure 8.



(a) (b) (c) (d)

Fig. 9. Reconstruction using the first 100 tensor principal components of S_2^{mpca} selected by: (a) Fisher. (b) TDPCA-MLDA. (c) Spectral. (d) TDPCA-SVM.

VII. Discussion

The Figures 6.(a)-(b) show that for both the MPCA subspaces the Fisher criterion outperforms the other ones in the range $1 \leq k \leq 5$, for classification tasks. If considered the spectral criterion, such result agrees with the fact that the first principal components with the largest variances, do not necessarily represent important discriminant "directions" to separate sample groups.

In [28] it is emphasised that high-dimensional spaces but small sample size data leads to side-effects, like overfitting, which can significantly impact in the generalisation performance of SVM. Such observation could explain the decreasing in the recognition rates observed for the TDPCA-SVM when

increasing the subspace dimension, observed in Figures 6.(a)-(b).

The spectral technique could be used to estimate the variance or discriminability of each tensor component for any dimensionality reduction method that computes the covariance structure for each component space $\mathbb{R}^{m_k}$. For instance, the three variants of MPCA named RMPCA, NMPCA and UMPCA as well as the ITA, which is an incremental version of CSA, could be augmented with the covariance structure computed by the spectral method.

In this paper we focus on the SVM and MLDA methods to demonstrate the TDPCA technique. Obviously, any other separating hyperplane could be used to compute the weights for ranking tensor components. Besides, the Fisher criterion could be applied to discriminate important features in any tensor subspace for supervised classification tasks.

It is important to observe that Fisher and TDPCA techniques need a training set once they are supervised methods. However, there are applications for which the patterns that characterizes different groups are not very clear even for an expertise (medical imaging analysis, for instance). In these cases, the spectral technique could be applied to compute the covariance structure for mining the data set in order to look for the patterns that most vary in the samples.

The comparison between the recognition rates of TDPCA-MLDA and TDPCA-SVM shows that the former is equivalent or outperforms the latter for $1 \leq k \leq 5$, but TDPCA-SVM outperforms the TDPCA-MLDA in the range $5 < k < 400$, as observed in Figures 6.(b). For larger values of k the performance plots of both techniques oscillate and it is not possible to decide the better one for gender tasks. Both MLDA and linear SVM seek to find a decision boundary that separates data into different classes as well as possible. The MLDA depends on all of the data, even points far away from the separating hyperplane and consequently is expected to be less robust to gross outliers [24]. The description of the SVM solution, on the other hand, does not make any assumption on the distribution of the data, focusing on the observations that lie close to the opposite class, that is, on the observations that most count for classification. As a consequence, SVM is more robust to outliers, zooming into the subtleties of group differences [24]. We expect that these differences between MLDA and SVM will influence the recognition rates of TDPCA-MLDA and TDPCA-SVM when there is some class overlap in the original high-dimensional space due to subtle differences between the sample groups; for example, in facial expression experiments.

It is worthwhile to highlight the advantages of multilinear methods against linear (non-tensor) ones for pattern recognition applications. In fact, the non-tensor counterpart of the MPCA is the PCA technique [24]. To apply the PCA algorithm we need to vectorize the samples to get vectors $\mathbf{v} \in \mathbb{R}^{m_1 \cdot m_2 \ldots m_n}$. The large number of features involved is cumbersome for linear techniques due to the computational cost and memory requirements. In fact, for PCA we must diagonalize a covariance matrix $C \in \mathbb{R}^{m_1 \cdot m_2 \ldots m_n} \times \mathbb{R}^{m_1 \cdot m_2 \ldots m_n}$ while for MPCA we get covariance matrices $C^k \in \mathbb{R}^{m_k} \times \mathbb{R}^{m_k}$. On the other hand, the recognition tasks that we consider in section VI-C have small training sets if compared to the number of features involved ('small sample size problem': $N \ll m_1 \cdot m_2 \ldots m_n$). As a consequence, the computational complexity for linear methods can be reduced due to the fact that, in this case, the number of independent components generated by the PCA is $(N-1)$ or less [24]. However, the number of available projection directions in multilinear methods can be much larger than the sample number, as we can notice from expression (14) because $m\prime_1 \cdot m\prime_2 \cdot \ldots \cdot m\prime_n \gg N$, in general (due to the flattening operation each matrix $\Phi^{(k)}$ in the line 8 of the Algorithm 1 is computed using a number of $N \cdot \amalg_{i \neq k} m_i$ samples which implies that the rank of $\Phi^{(k)}$ is not so far from m_k in the applications). This property reduces the susceptibility of tensor methods to the small sample size problem which directly increases the performance of multilinear techniques above PCA based algorithms, as already reported in [29].

VIII. Perspectives

In the sections III-VII we have not mentioned the support manifold for the considered tensors. However, if we return to tensor field definition in expression (9), the first point to tackle is the computation of the differential structure behind the manifold $\mathcal{M}$. Such task can be performed by manifold learning techniques based on local methods, that attempt to preserve the structure of the data by seeking to map nearby data points into nearby points in the low-dimensional representation. Then, the global manifold information is recovered by minimizing the overall reconstruction error. Traditional manifold learning techniques like Locally Linear Embedding (LLE) and Local Tangent Space Alignment (LTSA) and Hessian Eigenmaps, as well as the more recent Local Riemannian Manifold Learning (LRML), belong to this category of nonlinear dimensionality reduction methods (see [30] and references therein). The obtained manifold structure can offer the environment to define tensor fields through expression (9), and the application of such machinery for data analysis composes the first topic in this section.

Next, the development of a weighted procedure for multilinear subspace analysis is discussed in order to incorporate high level semantics in the form of labeled data or spatial maps computed in cognitive experiments. However, application of tensor techniques for data analysis is computational involved which opens new perspectives in parallel and distributed memory computing.

A. Manifold Learning and Tensor Fields

In order to implement manifold learning solution, we shall take the samples of a database $\mathcal{D} = \{p_1, \ldots, p_N\} \subset \mathbb{R}^D$ and performs the following steps: (a) Recover the data topology using some similarity measure; (b) Determination of the manifold dimension m; (c) Construction of a neighborhood system $\{U_\alpha\}_{\alpha \in I}$, where I is an index set and $U_\alpha \subset \mathbb{R}^m$; (d) Manifold learning by computing the local coordinate for each

neighborhood. The output of these steps is a family of local coordinate systems $\{(U_\alpha, \varphi_\alpha)\}_{\alpha \in I}$, for $\mathcal{M}$, according to the definition of differentiable manifold given in section II-B.

The local coordinate systems allow to compute also $\widetilde{\mathcal{D}} = \{z_1, z_2, \cdots, z_N\} \subset \mathbb{R}^m$, the lower dimensional representation of the data samples, as well as tangent vectors:

$$\mathbf{v}(p_j) = \sum_{i=1}^{n} \left(v_i(p_j) \frac{\partial \varphi_\alpha}{\partial x_i}(z_j) \right), \tag{44}$$

where $z_j = \varphi_\alpha^{-1}(p_j)$. Moreover, we can define the tangent space $T_p(\mathcal{M})$ and, consequently, a tensor field computed by expression (9), with $\mathbf{e}_k^{i_k}(p) \in T_p^k(\mathcal{M})$.

To simplify notation, we can return to the generalized matrix notion and represent a tensor $\mathbf{X}(p) \in T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$ as:

$$[\mathbf{X}(p)]_{i_1, i_2, \cdots, i_n} = \mathbf{X}_{i_1, i_2, \cdots, i_n}(p). \tag{45}$$

where we have omitted the reference to the basis B for simplicity. In the above expression, we use the symbol $[\cdot]$ to indicate that we are considering the matrix representation of the tensor field. Therefore, we shall read the above expression as the element $i_1, i_2, \cdots, i_n$ of the matrix representation of the tensor field $\mathbf{X}$ computed at the point $p \in \mathcal{M}$.

In this scenario, the mode-k product, given in Definition 1, is a local version of the tensor product between tensors $\mathbf{X}(p) \in T_p^1(\mathcal{M}) \otimes T_p^2(\mathcal{M}) \otimes \cdots \otimes T_p^n(\mathcal{M})$ and $\mathbf{A}(p) \in T_p'^k(\mathcal{M}) \otimes T_p^k(\mathcal{M})$, defined by:

$$[(\mathbf{X} \otimes \mathbf{A})(p)]_{i_1, \ldots, i_{k-1}, i_k, i, j_1, i_{k+1}, \ldots, i_n} = \mathbf{X}_{i_1, i_2, \cdots, i_n}(p) \cdot \mathbf{A}_{i, j_1}(p),$$

followed by a contraction in the i_k and j_1 indices, that means:

$$[(\mathbf{X} \times_k \mathbf{A})(p)]_{i_1, \ldots, i_{k-1}, i, i_{k+1}, \ldots, i_n} = \sum_{j=1}^{m_k} \mathbf{X}_{i_1, \cdots, i_{k-1}, j, i_{k+1}, \cdots i_n}(p) \mathbf{A}_{i, j}(p),$$

where $i_s = 1, 2, \ldots, m_s$, and $i = 1, 2, ..., m\prime_k$. Besides, the projection matrices $U^k \in \mathbb{R}^{m_k \times m\prime_k}$ are replaced to projection tensors $\mathbf{U}^k(p) \in T_p^k(\mathcal{M}) \otimes T_p'^k(\mathcal{M})$ and the tensor $\mathbf{Y}$ in expression (19) is given by:

$$\mathbf{Y}(p) = \left(\mathbf{X} \times_1 \mathbf{U}^{1^T} \times_2 \mathbf{U}^{2^T} ... \times_n \mathbf{U}^{n^T} \right)(p), \tag{46}$$

where:

$$[\mathbf{Y}(p)]_{j_1, j_2, \cdots, j_n} = \left[\left(\mathbf{X} \times_1 \mathbf{U}^{1^T} \times_2 \mathbf{U}^{2^T} ... \times_n \mathbf{U}^{n^T} \right)(p) \right]_{j_1, j_2, \cdots, j_n}, \tag{47}$$

with $\mathbf{Y}(p) \in T_p'^1(\mathcal{M}) \otimes T_p'^2(\mathcal{M}) \otimes \cdots \otimes T_p'^n(\mathcal{M})$, and $dim(T_p'^k(\mathcal{M})) = m\prime_k$, $k = 1, 2, \cdots, n$.

The application of the above concepts for data analysis depends on the following issues: (a) Manifold learning to build the local coordinate systems $\{(U_\alpha, \varphi_\alpha)\}_{\alpha \in I}$, for $\mathcal{M}$; (b) Discrete tensor field computation $\mathbf{X}(p_i)$, $i = 1, 2, \cdots, N$; (c) Local subspace learning technique to perform dimensionality reduction to compute the discrete tensor field $\mathbf{Y}(p_i)$, $i = 1, 2, \cdots, N$, given by expressions (46)-(47).

We believe that the tensor field concept together with the manifold structure offer a powerful framework for data modeling and analysis. However, fundamental issues are involved in the steps (a)-(c) above, related to data space topology/geometry as well as computational complexity of the necessary algorithms [31].

B. Application of Spatial Weighting Maps for Tensor Spaces

Despite of the well-known attractive properties of MPCA, its approach does not incorporate prior information in order to steer subspace computation. Important features may be discarded if dimensionality reduction is performed without prior information, which reduces the accuracy of subsequent data analysis [32]. Such problem has been addressed in the context of PCA which can be extended to a weighted version by introducing a spatial weighting value for each pixel i, $1 \leq i \leq n$, in the image. The spatial weighting vector:

$$\mathbf{w} = [w_1, w_2, \ldots, w_n]^T \tag{48}$$

is such that $w_j \geq 0$ and $\sum_{j=1}^{n} w_j = 1$. So, when N samples are observed, the weighted sample correlation matrix R^* can be described by

$$R^* = \left\{ r_{jk}^* \right\} = \left\{ \frac{\sum_{i=1}^{N} \sqrt{w_j}(x_{ij} - \bar{x}_j)\sqrt{w_k}(x_{ik} - \bar{x}_k)}{\sqrt{\sum_{i=1}^{N}(x_{ij} - \bar{x}_j)^2}\sqrt{\sum_{i=1}^{N}(x_{ik} - \bar{x}_k)^2}} \right\}. \tag{49}$$

for $j = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, n$. The sample correlation r_{jk}^* between the j^{th} and k^{th} variables is equal to w_j when $j = k$, $r_{jk}^* = r_{kj}^*$ for all j and k, and the weighted correlation matrix R^* is a nxn symmetric matrix.

Let the weighted correlation matrix R^* have respectively P^* and Λ^* eigenvector and eigenvalue matrices, that is,

$$P^{*T} R^* P^* = \Lambda^*. \tag{50}$$

The set of m ($m \leq n$) eigenvectors of R^*, that is, $P^* = [\mathbf{p}_1^*, \mathbf{p}_2^*, \ldots, \mathbf{p}_m^*]$, which corresponds to the m largest eigenvalues, defines a new orthonormal coordinate system for the training set matrix X and is called here as the spatially weighted principal components.

The remaining question now is: how to define spatial weights w_j that incorporate the prior knowledge extracted from the labeled data and can be systematically computed through the supervised information available? In [32] we address this task through spatial weights obtained as the output of a linear learning process, like SVM or LDA, for separating tasks in binary classification problems. Such approach can be extended to the tensor case by using the discriminant tensor $\mathbf{W}$, like performed in section V-B. Then, we consider the absolute values $|\mathbf{W}_{i_1, .., i_n}|$ to compute positive factors $\widetilde{\mathbf{W}}_{j_1, j_2, \cdots, j_n}$ to weight tensor features. Other statistical techniques based

on image descriptors or even cognitive experiments shall be considered to define spatial weighting maps for tensor subspaces computation. Further research must be undertaken to explore these possibilities for feature extraction and pattern recognition, specially in the context of face and medical image analysis where linear counterparts have demonstrated the capabilities of such weighting maps [32].

C. High Performance Requirements

From a theoretical viewpoint it is proved that many naturally occurring problems for tensors of order $n = 3$ are NP-hard [14]; that is, solutions to the hardest problems in NP can be found by answering questions about tensors in $\mathbb{R}^{m_1 \times m_2 \times m_3}$. For instance, it is demonstrated that the graph 3-colorability is polynomially reducible to the tensor eigenvalue problem over $\mathbb{R}$ (given $\mathbf{X} \in \mathbb{R}^{m \times m \times m}$ find a nonzero vector $x \in \mathbb{R}^m$ and $\lambda \in \mathbb{R}$ such that $\sum_{i,j=1}^{m} \mathbf{X}_{ijk} x_i x_j = \lambda x_k$, $k = 1, 2. \ldots, m$). Thus, deciding tensor eigenvalue over $\mathbb{R}$ is NP-hard. Finding the best rank-1 tensor decomposition is also an NP-hard problem [33]. Therefore, while very useful in practice, tensor methods present important computational challenges in the case of big tensors.

Hence, high performance algorithms have been proposed in order to address the computation requirements behind tensor operations. For instance, in [18] it is proposed a distributed memory parallel algorithm and implementation for computing the Tucker decomposition [10] of general dense tensors. The algorithm casts local computations in terms of specific routines to exploit optimized, architecture-specific components with the aim of reducing inter processor communication during data distributions and corresponding parallel computations.

In general, the tensor subspace learning and decomposition depend on intermediate products whose size can be much larger than the final result of the computation, referred to as the intermediate data explosion problem (computation of matrix $\Phi^{(k)}$ for MPCA). Also, the entire tensor needs to be accessed in each iteration, requiring large amounts of memory and incurring large data transport costs. Besides, in some operations (like mode-k flattening in Definition 4), the tensor data is accessed in different orders inside each iteration, which makes efficient block caching of the tensor data for fast memory access difficult if the tensor data is not replicated multiple times in memory.

In the specific cases of the techniques discussed in this paper, the asymptotic analysis is very useful to give the notion of the amount of required computation. Since MPCA iteratively compute the solution we can perform the computational complexity through the analysis of one iteration [11]. For simplicity, it is assumed that $m_1 = m_2 = \ldots m_n = m$. The most demanding steps in terms of floating-point operations are the formation of the matrix $\Phi^{(k)}$, the eigen-decomposition, and the computation of the multilinear projection $\widetilde{\mathbf{Y}}_i$. The corresponding computations complexities are given, respectively, by: $O\left(N \cdot n \cdot m^{(n+1)}\right)$, $O\left(m^3\right)$ and $O\left(n \cdot m^{(n+1)}\right)$. Therefore, if $m \geq 3$ we get a computational complexity of $O\left(N \cdot n \cdot m^{(n+1)}\right)$ for MPCA. So, MPCA faces a computational bottleneck because the computational requirements increase exponentially with the dimension n. From the view point of memory requirements it should be noticed that all the computation can be incrementally performed by reading $\mathbf{X}_i$ sequentially from the disk. Therefore, the memory requirements is limited by the number of elements of each tensor $\mathbf{X}_i$, given by $O\left(\prod_{i=1}^{n} m_i\right)$.

The spectral technique needs the computation of $O\left(\prod_{i=1}^{n} m\prime_i\right)$ variances and each variance needs n floating-point operation to be calculated which renders a computational complexity of $O\left(\prod_{i=1}^{n} m\prime_i\right)$ (see expression (32)). The computational complexity of the TDPCA-MLDA and TDPCA-SVM are given by the computational complexity of MLDA and SVM in the reduced space $\mathbb{R}^{m\prime_1 \times m\prime_2 \times \ldots \times m\prime_n}$, which are given, respectively, by [34], [35]: $O(min(N, \prod_{i=1}^{n} m\prime_i) \cdot \prod_{i=1}^{n} m\prime_i))$ and $O(max(N \cdot \prod_{i=1}^{n} m\prime_i) \cdot N^2)$. The Fisher criterion depends on the computation of $O\left(\prod_{i=1}^{n} m\prime_i\right)$ numbers and each one needs $O\left(N\right)$ floating-point operation to be calculated which renders a computational complexity of $O\left(N \cdot \prod_{i=1}^{n} m\prime_i\right)$. If we consider small sample size problems ($N \ll \prod_{i=1}^{n} m\prime_i$) then we obtain that the computational complexity of spectral is the lowest one ($O\left(\prod_{i=1}^{n} m\prime_i\right)$), followed by TDPCA-MLDA and Fisher ($N \cdot O\left(\prod_{i=1}^{n} m\prime_i\right)$). The computational complexity of TDPCA-SVM is the highest one, given by ($N^2 \cdot O\left(\prod_{i=1}^{n} m\prime_i\right)$). If $m = \min\left\{m\prime_i; \quad i = 1, 2, \cdot, \cdot, n\right\}$, then all the above results led to a computational complexity that depends on m^n and, like MPCA, increase exponentially with the tensor order n.

Despite of the solution in high performance computing already found in the literature [36], we still need the development of software architectures exploring new possibilities in hardware and theoretical developments to address the mentioned bottlenecks in order to make big tensor decomposition practical as well as scalable.

IX. Conclusions

In this paper, we review subspace learning, dimensionality reduction, classification and reconstruction problems in tensor spaces. We discuss these problems and show their particular solution in the context of MPCA. In the specific area of pattern recognition, we review advanced methodologies for tensor discriminant analysis. The problem of ranking tensor components to identify tensor subspaces for separating sample groups is focused on the context of statistical learning techniques.

We discuss the mentioned problem using the traditional algebraic formulation of tensors. However, we consider the geometric approach for tensor fields to discuss opened issues related to manifold learning and tensors. High-dimensional modeling is becoming challenging across the data-based modeling community because of advances in sensor and storage technologies that allow the generation of huge amounts of data related to complex phenomena in society and Nature. Tensor techniques offer a structure for data representation and analysis, as already demonstrated in the literature. However, we must pay attention in some aspects when considering

tensor-structured data sets. Tensor-based research is not just matrix-based research with additional subscripts. Tensors are geometric objects and the connection between their geometric, statistics and algebraic theories must be understood in order to fully explore tensor-based computation for data analysis. The incorporation of prior knowledge to steer the data mining tasks and the help of parallel computation were also discussed as fundamental research directions in this area.

REFERENCES

[1] A. E. Hassanien, A. Taher Azar, V. Snasael, J. Kacprzyk, and J. H. Abawajy, Eds., *Big Data in Complex Systems*, ser. Studies in Big Data. Springer, 2015, vol. 9. I

[2] M. Rasetti and E. Merelli, "The topological field theory of data: a program towards a novel strategy for data mining through data language," *Journal of Physics: Conference Series*, vol. 626, no. 1, p. 012005, 2015. I

[3] A. Kuleshov and A. Bernstein, *Machine Learning and Data Mining in Pattern Recognition: 10th Int. Conf., MLDM 2014, St. Petersburg, Russia, July 21-24, 2014. Proceedings*. Springer International Publishing, 2014, ch. Manifold Learning in Data Mining Tasks, pp. 119–133. I

[4] J. Zhang, H. Huang, and J. Wang, "Manifold learning for visualizing and analyzing high-dimensional data," *IEEE Intelligent Systems*, vol. 25, no. 4, pp. 54–61, 2010. I

[5] J. A. Lee and M. Verleysen, *Nonlinear Dimensionality Reduction*, 1st ed. Springer Publishing Company, Incorporated, 2007. I

[6] K. Sun and S. Marchand-Maillet, "An information geometry of statistical manifold learning," in *Proc. of the 31th Int. Conf. on Machine Learn., ICML 2014, Beijing, China, 21-26 June 2014*, 2014, pp. 1–9. I

[7] G. A. Giraldi, E. C. Kitani, E. Del-Moral-Hernandez, and C. E. Thomaz, "Discriminant component analysis and self-organized manifold mapping for exploring and understanding image face spaces," in *Survey paper from SIBGRAPI Tutorial*, Aug 2011, pp. 25–38. I

[8] B. Dubrovin, A. Fomenko, and S. Novikov, *Modern geometry: Methods and Applications*. New York: Springer-Verlag, 1992. I, II-B

[9] M. A. O. Vasilescu and D. Terzopoulos, *Computer Vision — ECCV 2002, Part I*. Springer, Berlin Heidelberg, 2002, ch. Multilinear Analysis of Image Ensembles: TensorFaces, pp. 447–460. I

[10] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognition*, vol. 44, no. 7, pp. 1540–1551, 2011. I, III, VIII-C

[11] H. Lu, K. Plataniotis, and A. Venetsanopoulos, "Mpca: Multilinear principal component analysis of tensor objects," *Neural Networks, IEEE Trans. on*, vol. 19, no. 1, pp. 18–39, Jan 2008. I, IV, IV, 11, 11, V, V-C, VI-A, VIII-C

[12] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear independent components analysis," in *CVPR (1)*. IEEE Computer Society, 2005, pp. 547–553. I

[13] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "On the best rank-1 and rank-(r1,r2,. . .,rn) approximation of higher-order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1324–1342, Mar. 2000. I, II-B

[14] C. J. Hillar and L.-H. Lim, "Most tensor problems are np-hard," *J. ACM*, vol. 60, no. 6, pp. 45:1–45:39, Nov. 2013. I, VIII-C

[15] T. A. Filisbino, G. A. Giraldi, and C. E. Thomaz, "Comparing ranking methods for tensor components in multilinear and concurrent subspace analysis with applications in face images," *Int. J. Image Graphics*, vol. 15, no. 1, 2015. I, V, V-B, VI

[16] T. Filisbino, G. Giraldi, and C. Thomaz, "Tensor algebra for high-dimensional image database representation: Statistical learning and geometric models," National Laboratory for Scientific Computing, Tech. Rep., 2016. [Online]. Available: https://www.dropbox.com/s/kbcgdfww7yjxgzf/Survey-Paper-Tutorial-Extended-Sib-20-07-2016.pdf?dl=0 I, VI

[17] ——, "Defining and sorting tensor components for face image analysis," National Laboratory for Scientific Computing, Tech. Rep., 2013. [Online]. Available: http://www.lncc.br/pdf_consultar.php?idt_arquivo=6370&mostrar=1&teste=1. I, V, V-A, V-B, V-B, VI-B

[18] W. Austin, G. Ballard, and T. G. Kolda, "Parallel tensor compression for large-scale scientific data," *CoRR*, vol. abs/1510.06689, 2015. I, VIII-C

[19] S. Liu and Q. Ruan, "Orthogonal tensor neighborhood preserving embedding for facial expression recognition," *Pattern Recognition*, vol. 44, no. 7, pp. 1497 – 1513, 2011. II, II-B

[20] D. Xu, L. Zheng, S. Lin, H.-J. Zhang, and T. S. Huang, "Reconstruction and recognition of tensor-based objects with concurrent subspaces analisys," *IEEE Trans. on Circuits and Systems for video technology*, vol. 1051/8215, 2008. III, III

[21] R. Pajarola, S. K. Suter, and R. Ruiters, "Tensor approximation in visualization and computer graphics," in *Eurographics 2013 - Tutorials*, no. t6. Girona, Spain: Eurographics Association, May 2013. III

[22] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl*, vol. 21, pp. 1253–1278, 2000. IV

[23] C. E. Thomaz and G. A. Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," *Image Vision Comput.*, vol. 28, no. 6, pp. 902–913, June 2010. V, V-B, VI, VI-B

[24] T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning," *Springer*, 2001. V, V-C, VII

[25] M. Zhu, "Discriminant analysis with common principal components," *Biometrika*, vol. 93(4), pp. 1018–1024, 2006. V

[26] S. Yan, D. Xu, Q. Yang, L. Zhang, X. Tang, and H.-J. Zhang, "Discriminant analysis with tensor representation," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, June 2005, pp. 526–532 vol. 1. V-C

[27] B. W. Bader, T. G. Kolda *et al.*, "Matlab tensor toolbox version 2.6," Available online, February 2015. [Online]. Available: http://www.sandia.gov/~tgkolda/TensorToolbox/ VI-A

[28] S. Klement, A. Madany Mamlouk, and T. Martinetz, "Reliability of Cross-Validation for SVMs in High-Dimensional, Low Sample Size Scenarios," in *ICANN '08: Proceedings of the 18th Int. Conf. on Artificial Neural Networks, Part I*. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 41–50. VII

[29] D. Xu, S. Yan, L. Zhang, S. Lin, H.-J. Zhang, and T. Huang, "Reconstruction and recognition of tensor-based objects with concurrent subspaces analysis," *Circuits and Systems for Video Technology, IEEE Trans. on*, vol. 18, no. 1, pp. 36–47, Jan 2008. VII

[30] G. F. M. Jr., G. A. Giraldi, C. E. Thomaz, and D. Millan, "Composition of local normal coordinates and polyhedral geometry in riemannian manifold learning," *IJNCR*, vol. 5, no. 2, pp. 37–68, 2015. VIII

[31] A. Ozdemir, M. A. Iwen, and S. Aviyente, "Locally linear low-rank tensor approximation," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Dec 2015, pp. 839–843. VIII-A

[32] C. E. Thomaz, E. L. Hall, G. A. Giraldi, P. G. Morris, R. Bowtell, and M. J. Brookes, "A priori-driven multivariate statistical approach to reduce dimensionality of meg signals," *Electronics Letters*, vol. 49, no. 18, pp. 1123–1124, August 2013. VIII-B, VIII-B

[33] N. Ravindran, N. D. Sidiropoulos, S. Smith, and G. Karypis, "Memory-efficient parallel computation of tensor and matrix products for big tensor decomposition," *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, 2014. VIII-C

[34] D. Cai, "Spectral regression: A regression framework for efficient regularized subspace learning," Ph.D. dissertation, University of Illinois at Urbana-Champaign, http://hdl.handle.net/2142/1170, 2009. VIII-C

[35] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, *Feature Extraction: Fundations and Applications*. Springer, Berlin, Heidelberg, New York, 2006. VIII-C

[36] A. P. Liavas and N. D. Sidiropoulos, "Parallel algorithms for constrained tensor factorization via alternating direction method of multipliers," *IEEE Trans. on Signal Processing*, vol. 63, no. 20, pp. 5450–5463, Oct 2015. VIII-C