Published in the *Proc. of the European Conf. on Computer Vision (ECCV '02)*, Copenhagen, Denmark, May, 2002, 447–460.

Multilinear Analysis of Image Ensembles: TensorFaces

M. Alex O. Vasilescu and Demetri Terzopoulos

Courant Institute, New York University, USA
Department of Computer Science, University of Toronto, Canada

Abstract. Natural images are the composite consequence of multiple factors related to scene structure, illumination, and imaging. Multilinear algebra, the algebra of higher-order tensors, offers a potent mathematical framework for analyzing the multifactor structure of image ensembles and for addressing the difficult problem of disentangling the constituent factors or modes. Our multilinear modeling technique employs a tensor extension of the conventional matrix singular value decomposition (SVD), known as the N-mode SVD. As a concrete example, we consider the multilinear analysis of ensembles of facial images that combine several modes, including different facial geometries (people), expressions, head poses, and lighting conditions. Our resulting "TensorFaces" representation has several advantages over conventional eigenfaces. More generally, multilinear analysis shows promise as a unifying framework for a variety of computer vision problems.

1 Introduction

Natural images are formed by the interaction of multiple factors related to scene structure, illumination, and imaging. Human perception remains robust despite significant variation of these factors. For example, people possess a remarkable ability to recognize faces when confronted by a broad variety of facial geometries, expressions, head poses, and lighting conditions, and this ability is vital to human social interaction. Developing a similarly robust computational model of face recognition remains a difficult open problem whose solution would have substantial impact on biometrics for identification, surveillance, human-computer interaction, and other applications.

Linear algebra, i.e., the algebra of matrices, has traditionally been of great value in the context of image analysis and representation. The Fourier transform, the Karhonen-Loeve transform, and other linear techniques have been veritable workhorses. In particular, principal component analysis (PCA) has been a popular technique in facial image recognition, as has its refinement, independent component analysis (ICA) [2]. By their very nature, however, these offspring of linear algebra address single-factor variations in image formation. Thus, the conventional "eigenfaces" facial image recognition technique [13, 17] works best when person identity is the only factor that is permitted to vary. If other factors, such as lighting, viewpoint, and expression, are also permitted to modify facial images, eigenfaces face difficulty.

In this paper, we employ a more sophisticated mathematical approach in the analysis and representation of images that can account explicitly for each of the multiple factors

inherent to image formation. Our approach is that of multilinear algebra—the algebra of higher-order tensors. The natural generalization of matrices (i.e., linear operators defined over a vector space), tensors define multilinear operators over a *set* of vector spaces. Subsuming conventional linear analysis as a special case, tensor analysis offers a unifying mathematical framework suitable for addressing a variety of computer vision problems. Tensor analysis makes the assumption that images formed as a result of some multifactor confluence are amenable to linear analysis as each factor or mode is allowed to vary in turn, while the remaining factors or modes are held constant.[1]

We focus in this paper on the higher-order generalization of PCA and the singular value decomposition (SVD) of matrices for computing principal components. Unlike the matrix case for which the existence and uniqueness of the SVD is assured, the situation for higher-order tensors is not as simple. Unfortunately, there does not exist a true "tensor SVD" that offers all the nice properties of the matrix SVD [6]. There are multiple ways to decompose tensors orthogonally. However, one multilinear extension of the matrix SVD to tensors is most natural. We demonstrate the application of this N-*mode SVD* to the representation of collections of facial images, where multiple modes are permitted to vary. The resulting representation separates the different modes underlying the formation of facial images, hence it is promising for use in a robust facial recognition algorithm.

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 covers the foundations of tensor algebra that are relevant to our approach. Section 4 formulates the tensor decomposition algorithm which is central to our multilinear analysis. Section 5 applies our multilinear analysis algorithm to the analysis of facial images. Section 6 concludes the paper and proposes future research topics.

2 Related Work

Prior research has approached the problem of facial representation for recognition by taking advantage of the functionality and simplicity of matrix algebra. The well-known family of PCA-based algorithms, such as eigenfaces [13, 17] and Fisherfaces [1] compute the PCA by performing an SVD on a $XY \times P$ *data matrix* of "vectorized" $X \times Y$ pixel images of P people. These linear models are suitable in the case where the identity of the subject is the only variable accounted for in image formation. Various researchers have attempted to deal with the shortcomings of PCA-based facial image representation in less constrained (multi-factor) situations, for example, by employing better classifiers [11].

Bilinear models have attracted attention because of their richer representational power. The *2-mode analysis* technique for analyzing (statistical) data matrices of scalar entries is described by Magnus and Neudecker [8]. 2-mode analysis was extended to vector entries by Marimont and Wandel [9] in the context of characterizing color surface and illuminant spectra. Freeman and Tenenbaum [4, 14] applied this extension in three different perceptual domains, including face recognition.

[1] Also of interest is the fact that, from a probabilistic point of view, multilinear algebra is to higher-order statistics what linear algebra is to second-order statistics [3].

As was pointed out by Shashua and Levin [12], the natural representation of a collection of images is a three-dimensional array, or 3rd-order tensor, rather than a simple matrix of vectorized images. They develop compression algorithms for collections of images, such as video images, that take advantage of spatial (horizontal/vertical) and temporal redundancies, leading to higher compression rates compared to applying conventional PCA on vectorized image data matrices.

In addressing the motion analysis/synthesis problem, Vasilescu [19, 18] structured motion capture data in tensor form and developed an algorithm for extracting "human motion signatures" from the movements of multiple subjects each performing several different actions. The algorithm she described performed 3-mode analysis (with a dyadic decomposition) and she identified the more general motion analysis problem involving more than two factors (people, actions, cadences, ...) as one of N*-mode analysis* on higher-order tensors. N-mode analysis of observational data was first proposed by Tucker [16], who pioneered 3-mode analysis, and subsequently developed by Kapteyn *et al.* [5, 8] and others, notably [3].

The N-mode SVD facial image representation technique that we develop in this paper subsumes the previous methods reviewed above. In particular, when presented with matrices of vectorized images that are amenable to simple, linear analysis, our method reduces to SVD, hence PCA; i.e., the eigenfaces of Sirovich and Kirby or Turk and Pentland. When the collection of images is more appropriately amenable to bilinear analysis, our technique reduces to the "style/content" analysis of Freeman and Tenenbaum. More importantly, however, our technique is capable of handling images that are the consequence of any number of multilinear factors of the sort described in the introduction.

3 Relevant Tensor Algebra

We now introduce the notation and basic definitions of multilinear algebra. Scalars are denoted by lower case letters ($a, b, \ldots$), vectors by bold lower case letters ($\mathbf{a}, \mathbf{b} \ldots$), matrices by bold upper-case letters ($\mathbf{A}, \mathbf{B} \ldots$), and higher-order tensors by calligraphic upper-case letters ($\mathcal{A}, B \ldots$).

A *tensor*, also known as n-way array or multidimensional matrix or n-mode matrix, is a higher order generalization of a vector (first order tensor) and a matrix (second order tensor). Tensors are multilinear mappings over a set of vector spaces. The *order* of tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is N. An element of $\mathcal{A}$ is denoted as $\mathcal{A}_{i_1 \ldots i_n \ldots i_N}$ or $a_{i_1 \ldots i_n \ldots i_N}$ or where $1 \leq i_n \leq I_n$.

An $N^{th}-$order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ has *rank-1* when it is expressible as the outer product of N vectors: $\mathcal{A} = \mathbf{u}_1 \circ \mathbf{u}_2 \circ \ldots \circ \mathbf{u}_N$. The tensor element is expressed as $a_{ij \ldots m} = u_{1i} u_{2j} \ldots u_{Nm}$, where u_{1i} is the i^{th} component of $\mathbf{u}_1$, etc. The *rank* of a N^{th} order tensor $\mathcal{A}$, denoted R=rank($\mathcal{A}$), is the minimal number of rank-1 tensors that yield $\mathcal{A}$ in a linear combination:

$$\mathcal{A} = \sum_{r=1}^{R} \sigma_r \mathbf{u}_1^{(r)} \circ \mathbf{u}_2^{(r)} \circ \ldots \circ \mathbf{u}_N^{(r)}. \tag{1}$$

A singular value decomposition (SVD) can be expressed as a *rank decomposition* as is shown in the following simple example:

$$\mathbf{M} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \sigma_{11} & 0 \\ 0 & \sigma_{22} \end{bmatrix} \begin{bmatrix} f & g \\ h & i \end{bmatrix} = \sigma_{11} \begin{bmatrix} a \\ c \end{bmatrix} \circ \begin{bmatrix} f \\ g \end{bmatrix} + \sigma_{22} \begin{bmatrix} b \\ d \end{bmatrix} \circ \begin{bmatrix} h \\ i \end{bmatrix} \tag{2}$$

$$= \mathbf{U}_1 \mathbf{\Sigma} \mathbf{U}_2^T \tag{3}$$

$$= \begin{bmatrix} \mathbf{u}_1^{(1)} & \mathbf{u}_1^{(2)} \end{bmatrix} \begin{bmatrix} \sigma_{11} & 0 \\ 0 & \sigma_{22} \end{bmatrix} \begin{bmatrix} \mathbf{u}_2^{(1)} & \mathbf{u}_2^{(2)} \end{bmatrix}^T \tag{4}$$

$$= \sum_{i=1}^{R=2} \sum_{j=1}^{R=2} \sigma_{ij} \mathbf{u}_1^{(i)} \circ \mathbf{u}_2^{(j)} \tag{5}$$

Note that a singular value decomposition is a *combinatorial orthogonal rank decomposition* (5), but that the reverse is not true; in general, rank decomposition is not necessarily singular value decomposition. For further discussion on the differences between matrix SVD, rank decomposition and orthogonal rank decomposition for higher order tensors see [6].

Next, we generalize the definition of column and row rank of matrices. In tensor terminology, column vectors are referred to as mode-1 vectors and row vectors as mode-2 vectors. The mode-n vectors of an N^{th} order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ are the I_n-dimensional vectors obtained from $\mathcal{A}$ by varying index i_n while keeping the other indices fixed. The mode-n vectors are the column vectors of matrix $\mathbf{A}_{(n)} \in \mathbb{R}^{I_n \times (I_1 I_2 \ldots I_{n-1} I_{n+1} \ldots I_N)}$ that results from *flattening* the tensor $\mathcal{A}$, as shown in Fig. 1. The *n-rank* of $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$, denoted R_n, is defined as the dimension of the vector space generated by the mode-n vectors:

$$R_n = rank_n(\mathcal{A}) = rank(\mathbf{A}_{(n)}). \tag{6}$$

A generalization of the product of two matrices is the product of a tensor and a matrix. The *mode-n product* of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_n \times \ldots \times I_N}$ by a matrix $\mathbf{M} \in \mathbb{R}^{J_n \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{M}$, is a tensor $\mathcal{B} \in \mathbb{R}^{I_1 \times \ldots \times I_{n-1} \times J_n \times I_{n+1} \times \ldots \times I_N}$ whose entries are computed by

$$(\mathcal{A} \times_n \mathbf{M})_{i_1 \ldots i_{n-1} j_n i_{n+1} \ldots i_N} = \sum_{i_n} a_{i_1 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} m_{j_n i_n}. \tag{7}$$

The mode-n product can be expressed in tensor notation as follows:

$$\mathcal{B} = \mathcal{A} \times_n \mathbf{M}, \tag{8}$$

or, in terms of flattened matrices,

$$\mathbf{B}_{(n)} = \mathbf{M} \mathbf{A}_{(n)}. \tag{9}$$

The mode-n product of a tensor and a matrix is a special case of the inner product in multilinear algebra and tensor analysis. In the literature, it is often denoted using Einstein summation notation. For our purposes, however, the mode-n product symbol is more suggestive of multiplication and expresses better the analogy between matrix and tensor SVD [16] (see Section 4). The mode-n product has the following properties:

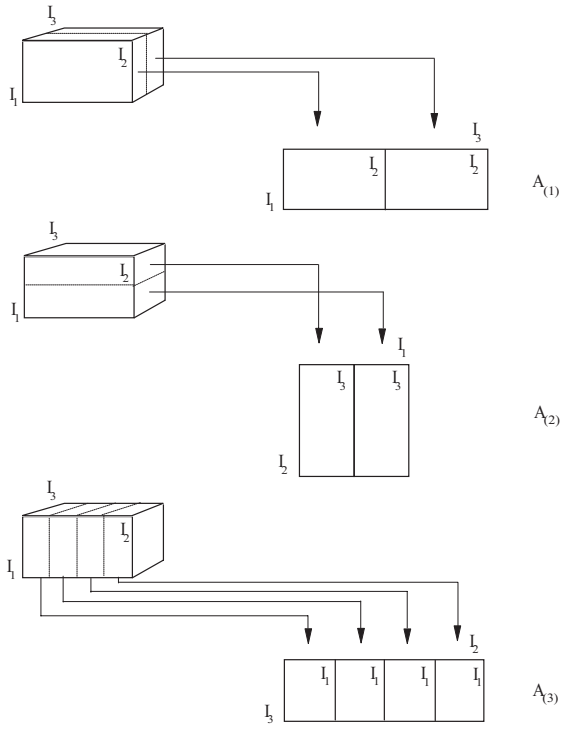

Fig. 1. Flattening a (3rd-order) tensor. The tensor can be flattened in 3 ways to obtain matrices comprising its mode-1, mode-2, and mode-3 vectors.

1. Given a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times \ldots I_n \times \ldots I_m \cdots}$ and two matrices, $\mathbf{U} \in \mathbb{R}^{J_m \times I_m}$ and $\mathbf{V} \in \mathbb{R}^{J_n \times I_n}$ the following property holds true:

$$\mathcal{A} \times_m \mathbf{U} \times_n \mathbf{V} = (\mathcal{A} \times_m \mathbf{U}) \times_n \mathbf{V} \tag{10}$$
$$= (\mathcal{A} \times_n \mathbf{V}) \times_m \mathbf{U} \tag{11}$$
$$= \mathcal{A} \times_n \mathbf{V} \times_m \mathbf{U} \tag{12}$$

2. Given a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times \ldots \times I_n \times \ldots \times I_N}$ and two matrices, $\mathbf{U} \in \mathbb{R}^{J_n \times I_n}$ and $\mathbf{V} \in \mathbb{R}^{K_n \times J_n}$ the following property holds true:

$$(\mathcal{A} \times_n \mathbf{U}) \times_n \mathbf{V} = \mathcal{A} \times_n (\mathbf{V}\mathbf{U}) \tag{13}$$

4 Tensor Decomposition

A matrix $\mathbf{D} \in \mathbb{R}^{I_1 \times I_2}$ is a two-mode mathematical object that has two associated vector spaces, a row space and a column space. SVD orthogonalizes these two spaces and decomposes the matrix as $\mathbf{D} = \mathbf{U}_1 \mathbf{\Sigma} \mathbf{U}_2^T$, the product of an orthogonal column-space represented by the left matrix $\mathbf{U}_1 \in \mathbb{R}^{I_1 \times J_1}$, a diagonal singular value matrix

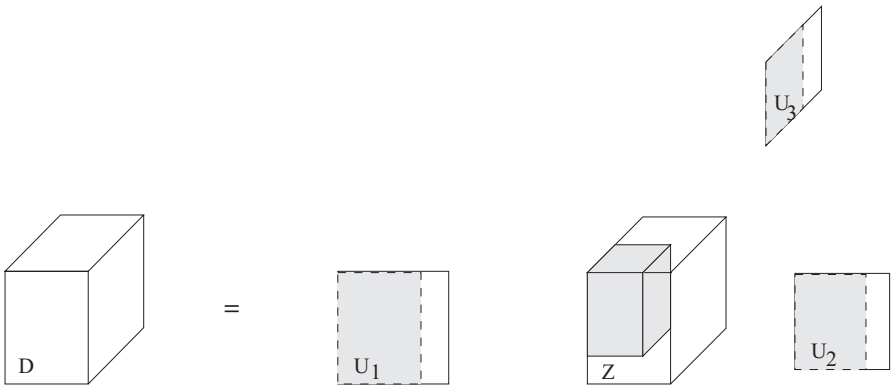

Fig. 2. An N-mode SVD orthogonalizes the N vector spaces associated with an order-N tensor (the case $N = 3$ is illustrated).

$\mathbf{\Sigma} \in \mathbb{R}^{J_1 \times J_2}$, and an orthogonal row space represented by the right matrix $\mathbf{U}_2 \in \mathbb{R}^{I_2 \times J_2}$. In terms of the mode-$n$ products defined in the previous section, this matrix product can be rewritten as $\mathbf{D} = \mathbf{\Sigma} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2$.

By extension, an order $N > 2$ tensor or n-way array $\mathcal{D}$ is an N-dimensional matrix comprising N spaces. "N-mode SVD" is a an extension of SVD that orthogonalizes these N spaces and expresses the tensor as the mode-n product (7) of N-orthogonal spaces

$$\mathcal{D} = \mathcal{Z} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \ldots \times_n \mathbf{U}_n \ldots \times_N \mathbf{U}_N, \tag{14}$$

as illustrated in Fig. 2 for the case $N = 3$. Tensor $\mathcal{Z}$, known as the *core tensor*, is analogous to the diagonal singular value matrix in conventional matrix SVD. It is important to realize, however, that the core tensor does not have a diagonal structure; rather, $\mathcal{Z}$ is in general a full tensor [6]. The core tensor governs the interaction between the *mode matrices* $\mathbf{U}_n$, for $n = 1, \ldots, N$. Mode matrix $\mathbf{U}_n$ contains the orthonormal vectors spanning the column space of the matrix $\mathbf{D}_{(n)}$ that results from the mode-n flattening of $\mathcal{D}$, as was illustrated in Fig. 1. [2]

[2] Note that the N-mode SVD can be expressed as an expansion of mutually orthogonal rank-1 tensors (analogous to equation (5)), as follows:

$$\mathcal{D} = \sum_{i_1=1}^{R_1} \ldots \sum_{i_n=1}^{R_n} \ldots \sum_{i_N=1}^{R_N} z_{i_1 \ldots i_N} \mathbf{U}_1^{(i_1)} \circ \ldots \circ \mathbf{U}_n^{(i_n)} \circ \ldots \mathbf{U}_N^{(i_N)},$$

where $\mathbf{U}_n^{(i_n)}$ is the i_n column vector of the matrix $\mathbf{U}_n$. In future work, we shall address the problem of finding the best rank-$(R_1, R_2, \ldots, R_N)$ tensor. This is not to be confused with the classical "rank-R problem" [7].

4.1 The N-Mode SVD Algorithm

In accordance with the above theory, our N-mode SVD algorithm for decomposing $\mathcal{D}$ is as follows:

1. For $n = 1, \ldots, N$, compute matrix $\mathbf{U}_n$ in (14) by computing the SVD of the flattened matrix $\mathbf{D}_{(n)}$ and setting $\mathbf{U}_n$ to be the left matrix of the SVD.[3]
2. Solve for the core tensor as follows

$$\mathcal{Z} = \mathcal{D} \times_1 \mathbf{U}_1^T \times_2 \mathbf{U}_2^T \ldots \times_n \mathbf{U}_n^T \ldots \times_N \mathbf{U}_N^T. \tag{15}$$

5 TensorFaces: Multilinear Analysis of Facial Images

As we stated earlier, image formation depends on scene geometry, viewpoint, and illumination conditions. Multilinear algebra offers a natural approach to the analysis of the multifactor structure of image ensembles and to addressing the difficult problem of disentangling the constituent factors or modes.

In a concrete application of our multilinear image analysis technique, we employ the Weizmann face database of 28 male subjects photographed in 15 different poses under 4 illuminations performing 3 different expressions. We used a portion of this database, employing images in 5 poses, 3 illuminations, and 3 expressions.[4] Using a global rigid optical flow algorithm, we roughly aligned the original 512×352 pixel images relative to one reference image. The images were then decimated by a factor of 3 and cropped as shown in Fig. 3, yielding a total of 7943 pixels per image within the elliptical cropping window. Our facial image data tensor $\mathcal{D}$ is a $28 \times 5 \times 3 \times 3 \times 7943$ tensor. The number of modes is $N = 5$.

We apply multilinear analysis to the facial image data using the N-mode decomposition algorithm described in Section 4. The 5-mode decomposition of $\mathcal{D}$ is

$$\mathcal{D} = \mathcal{Z} \times_1 \mathbf{U}_{\text{people}} \times_2 \mathbf{U}_{\text{views}} \times_3 \mathbf{U}_{\text{illums}} \times_4 \mathbf{U}_{\text{expres}} \times_5 \mathbf{U}_{\text{pixels}}, \tag{16}$$

where the $28 \times 5 \times 3 \times 3 \times 7943$ core tensor $\mathcal{Z}$ governs the interaction between the factors represented in the 5 mode matrices: The 28×28 mode matrix $\mathbf{U}_{\text{people}}$ spans the space of people parameters, the 5×5 mode matrix $\mathbf{U}_{\text{views}}$ spans the space of viewpoint parameters, the 3×3 mode matrix $\mathbf{U}_{\text{illums}}$ spans the space of illumination parameters and the 3×3 mode matrix $\mathbf{U}_{\text{expres}}$ spans the space of expression parameters. The 7943×7943 mode matrix $\mathbf{U}_{\text{pixels}}$ orthonormally spans the space of images.

[3] When $\mathbf{D}_{(n)}$ is a non-square matrix, the computation of $\mathbf{U}_n$ in the singular value decomposition $\mathbf{D}_{(n)} = \mathbf{U}_n \mathbf{\Sigma} \mathbf{V}_n^T$ can be performed efficiently, depending on which dimension of $\mathbf{D}_{(n)}$ is smaller, by decomposing either $\mathbf{D}_{(n)} \mathbf{D}_{(n)}^T = \mathbf{U}_n \mathbf{\Sigma}^2 \mathbf{U}_n^T$ and then computing $\mathbf{V}_n^T = \mathbf{\Sigma}^+ \mathbf{U}_n^T \mathbf{D}_{(n)}$ or by decomposing $\mathbf{D}_{(n)}^T \mathbf{D}_{(n)} = \mathbf{V}_n \mathbf{\Sigma}^2 \mathbf{V}_n^T$ and then computing $\mathbf{U}_n = \mathbf{D}_{(n)} \mathbf{V}_n \mathbf{\Sigma}^+$.

[4] A computer-controlled robot arm positioned the camera to $\pm 34^\circ$, $\pm 17^\circ$, and 0°, the frontal view in the horizontal plane. The face was illuminated by turning on and off three light sources fixed at the same height as the face and positioned to the left, center, and right of the face. For additional details, see [10].

(a)

(b)

Fig. 3. The facial image database (28 subjects $\times$ 45 images per subject). (a) The 28 subjects shown in expression 2 (smile), viewpoint 3 (frontal), and illumination 2 (frontal). (b) The full image set for subject 1. Left to right, the three panels show images captured in illuminations 1, 2, and 3. Within each panel, images of expressions 1, 2, and 3 are shown horizontally while images from viewpoints 1, 2, 3, 4, and 5 are shown vertically. The image of subject 1 in (a) is the image situated at the center of (b).

Our multilinear analysis, which we call *TensorFaces*, subsumes linear, PCA analysis or conventional eigenfaces. Each column of $\mathbf{U}_{\text{pixels}}$ is an "eigenimage". These eigenimages are identical to conventional eigenfaces [13, 17], since the former were computed by performing an SVD on the mode-5 flattened data tensor $\mathcal{D}$ which yields the matrix $\mathbf{D}_{(\text{pixels})}$ whose columns are the vectorized images. To further show mathematically that PCA is a special case of our multilinear analysis, we write the latter in terms of matrix notation. A matrix representation of the N-mode SVD can be obtained by unfolding $\mathcal{D}$ and $\mathcal{Z}$ as follows:

$$\mathbf{D}_{(n)} = \mathbf{U}_n \mathbf{Z}_{(n)} (\mathbf{U}_{n-1} \otimes \ldots \otimes \mathbf{U}_1 \otimes \mathbf{U}_N \otimes \ldots \otimes \mathbf{U}_{n+2} \otimes \mathbf{U}_{n+1})^T, \tag{17}$$

where $\otimes$ denotes the matrix Kronecker product. Using (17) we can express the decomposition of $\mathcal{D}$ as

$$\underbrace{\mathbf{D}_{(\text{pixels})}}_{\text{image data}} = \underbrace{\mathbf{U}_{\text{pixels}}}_{\text{basis vectors}} \underbrace{\mathbf{Z}_{(\text{pixels})} (\mathbf{U}_{\text{expres}} \otimes \mathbf{U}_{\text{illums}} \otimes \mathbf{U}_{\text{views}} \otimes \mathbf{U}_{\text{people}})^T}_{\text{coefficients}}. \tag{18}$$

The above matrix product can be interpreted as a standard linear decomposition of the image ensemble, where the mode matrix $\mathbf{U}_{\text{pixels}}$ is the PCA matrix of basis vectors and the associated matrix of coefficients is obtained as the product of the flattened core tensor times the Kronecker product of the people, viewpoints, illuminations, and expressions mode matrices. Thus, as we stated above, our multilinear analysis subsumes linear, PCA analysis.

The advantage of multilinear analysis is that the core tensor $\mathcal{Z}$ can transform the eigenimages present in the matrix $\mathbf{U}_{\text{pixels}}$ into *eigenmodes*, which represent the principal axes of variation across the various modes (people, viewpoints, illuminations, expressions) and represents how the various factors interact with each other to create an image. This is accomplished by simply forming the product $\mathcal{Z} \times_5 \mathbf{U}_{\text{pixels}}$. By contrast, PCA basis vectors or eigenimages represent only the principal axes of variation across images. To demonstrate, Fig. 4 illustrates in part the results of the multilinear analysis of the facial image tensor $\mathcal{D}$. Fig. 4(a) shows the first 10 PCA eigenimages contained in $\mathbf{U}_{\text{pixels}}$. Fig. 4(b) illustrates some of the eigenmodes in the product $\mathcal{Z} \times_5 \mathbf{U}_{\text{pixels}}$. A few of the lower-order eigenmodes are shown in the three arrays. The labels at the top of each array indicate the names of the horizontal and vertical modes depicted by the array. Note that the basis vector at the top left of each panel is the average over all people, viewpoints, illuminations, and expressions, and that the first column of eigenmodes (people mode) is shared by the three arrays.

PCA is well suited to parsimonious representation, since it orders the basis vectors according to their significance. The standard PCA compression scheme is to truncate the higher order eigenvectors associated with this representation. Our multilinear analysis enables an analogous compression scheme, but it offers much greater control. It allows the strategic truncation of higher-order eigenmodes depending on the task at hand and the modalities that should be represented most faithfully.

Multilinear analysis subsumes mixtures of probabilistic PCA or view-based models [15, 11] when one uses a different choice of basis functions. Starting with the eigenmodes $\mathcal{Z} \times_5 \mathbf{U}_{\text{pixels}}$, we multiply the viewpoint parameter matrix $\mathbf{U}_{\text{views}}$ to form the product $\mathcal{Z} \times_2 \mathbf{U}_{\text{views}} \times_5 \mathbf{U}_{\text{pixels}}$, which yields the principal axes of variation of the image

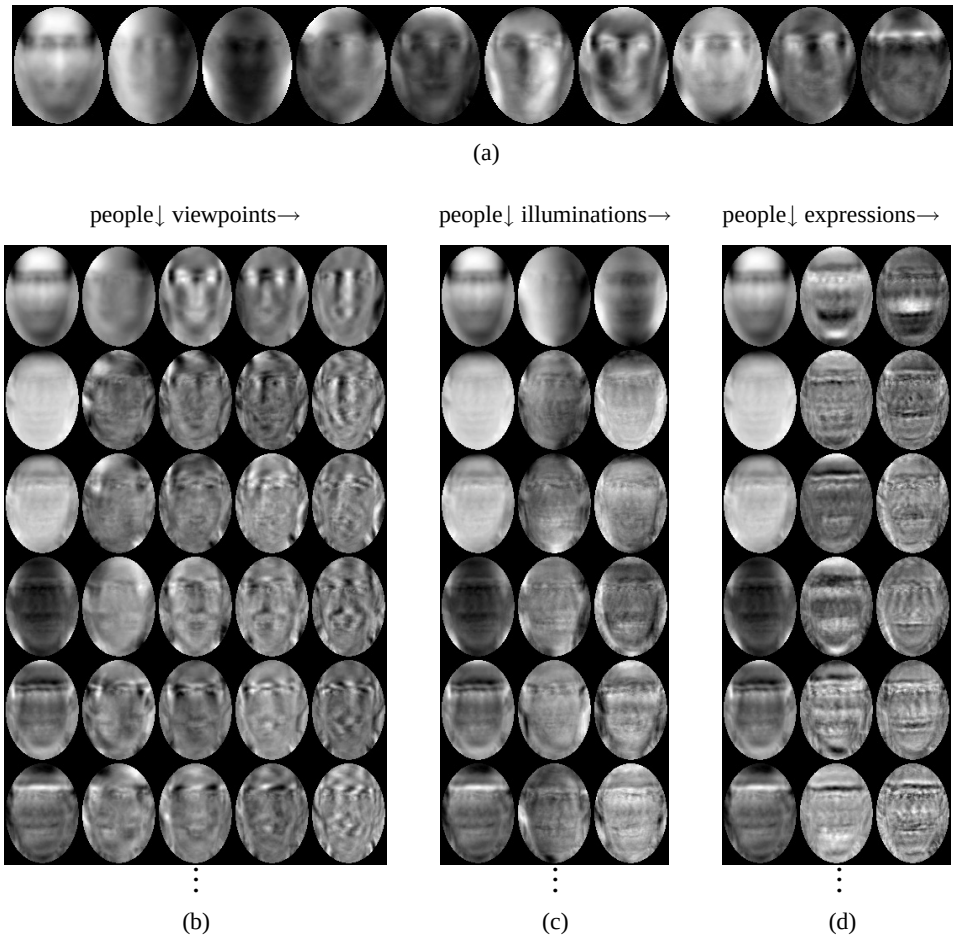

Fig. 4. Some of the basis vectors resulting from the multilinear analysis of the facial image data tensor $\mathcal{D}$. (a) The first 10 PCA eigenvectors (eigenfaces), which are contained in the mode matrix $\mathbf{U}_{\text{pixels}}$, and are the principal axes of variation across all images. (b,c,d) A partial visualization of the product $\mathcal{Z} \times_5 \mathbf{U}_{\text{pixels}}$, in which the core tensor $\mathcal{Z}$ transforms the eigenvectors $\mathbf{U}_{\text{pixels}}$ to yield a 5-mode, $28 \times 5 \times 3 \times 3 \times 7943$ tensor of eigenmodes which capture the variability across modes (rather than images). Some of the first few eigenmodes are shown in the three arrays. The labels at the top of each array indicate the names of the horizontal and vertical modes depicted in that array. Note that the basis vector at the top left of each panel is the average over all people, viewpoints, illuminations, and expressions (the first column of eigenmodes (people mode) is shared by the three arrays).

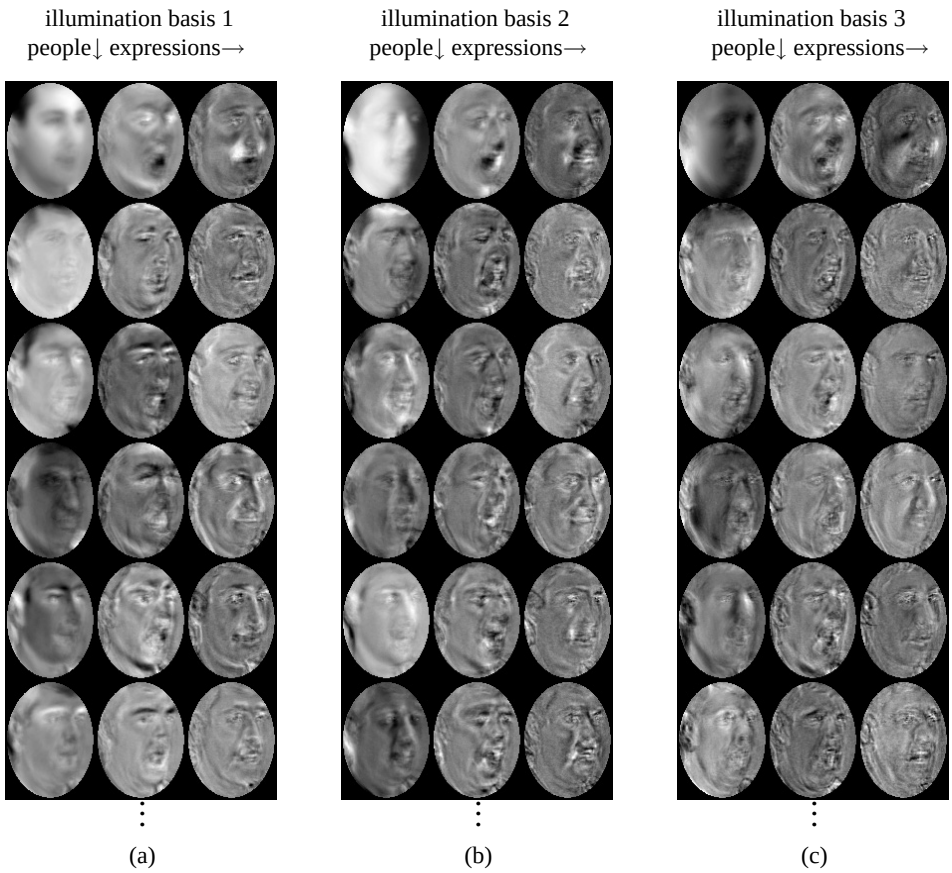

Fig. 5. Some of the eigenvectors in the $28 \times 3 \times 3 \times 7943$ tensor $\mathcal{Z} \times_2 \mathbf{U}_{\text{views}} \times_5 \mathbf{U}_{\text{pixels}}$ for viewpoint 1. These eigenmodes are viewpoint specific.

ensemble across the people mode, illumination mode, and expression mode for each of the 5 viewpoints. Fig. 5 shows the eigenvectors that span all the images in viewpoint 1. In essence, the multilinear analysis provides for each viewpoint the principal axes of a multidimensional Gaussian.

Similarly, we can define a person specific set of eigenvectors that span all the images. Fig. 6(a–c) illustrates the effect of multiplying the eigenvectors of Fig. 4(b–d) by $\mathbf{U}_{\text{people}}$ to obtain the $5 \times 3 \times 3 \times 7943$ tensor of eigenvectors $\mathcal{Z} \times_1 \mathbf{U}_{\text{people}} \times_5 \mathbf{U}_{\text{pixels}}$. These new eigenvectors are now person-specific. The figure shows all of the eigenvectors for slice 1 of the tensor, associated with subject 1 in Fig. 3(a). The eigenvectors shown capture the variations across the distribution of images of this particular subject over all viewpoints, expressions, and illuminations. Fig. 6(d–e) shows portions of slices 2 and 3 through the tensor (the upper 3×3 portions of arrays analogous to that in (a) of the figure are shown), showing some of the eigenvectors specific to subject 2 and to subject 3, respectively.

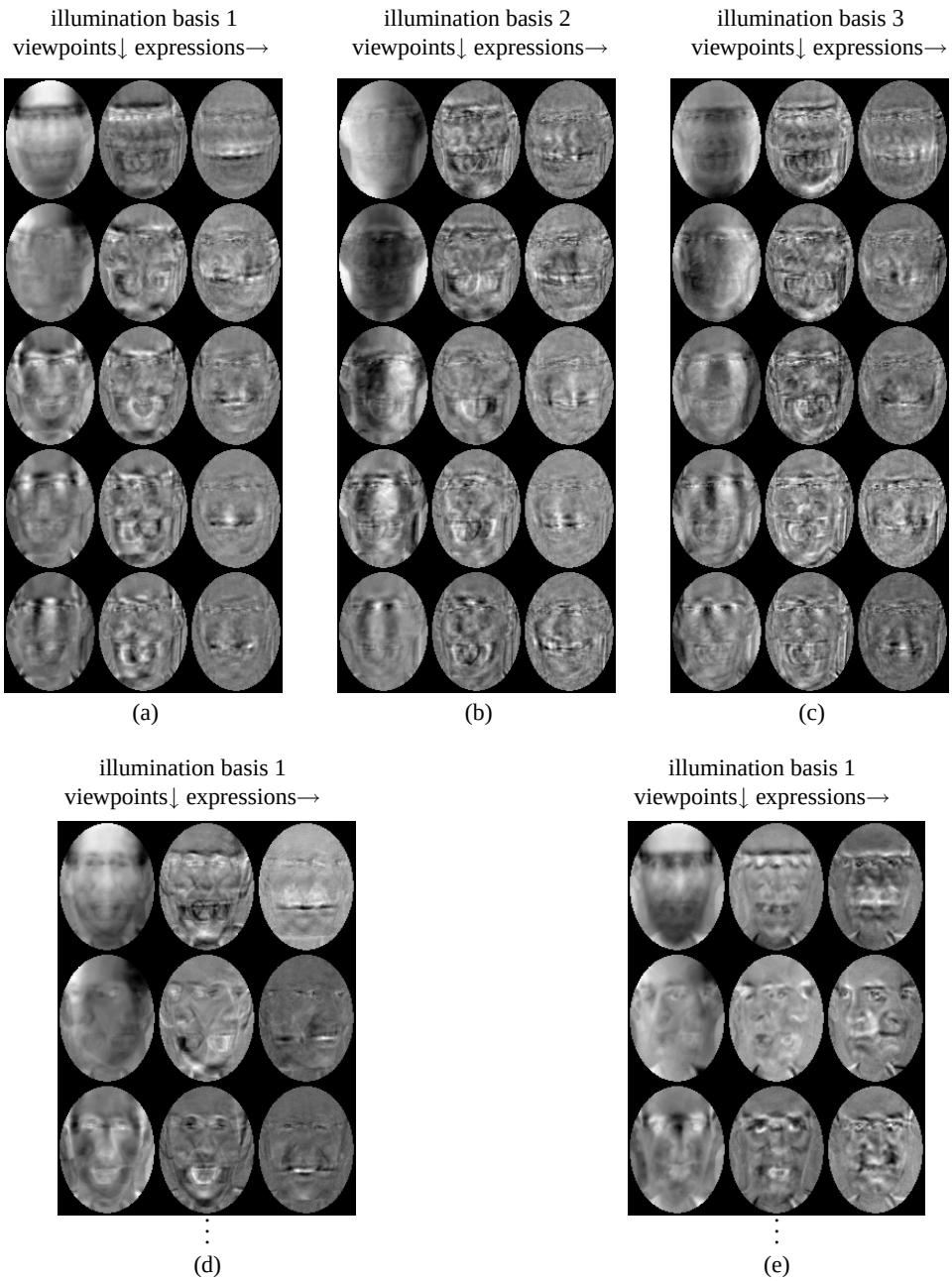

Fig. 6. (a,b,c) All the eigenvectors in the $5 \times 3 \times 3 \times 7943$ tensor $\mathcal{Z} \times_1 \mathbf{U}_{\text{people}} \times_5 \mathbf{U}_{\text{pixels}}$ for subject 1. This is the top slice (subject 1 in Fig. 3(a)) of the tensor depicted in Fig. 4(b–d) but multiplied by $\mathbf{U}_{\text{people}}$, which makes the eigenvectors person-specific. (d) Person specific eigenvectors for subject 2 and (e) for subject 3; the upper 3×3 portions of arrays analogous to that in (a) are shown.

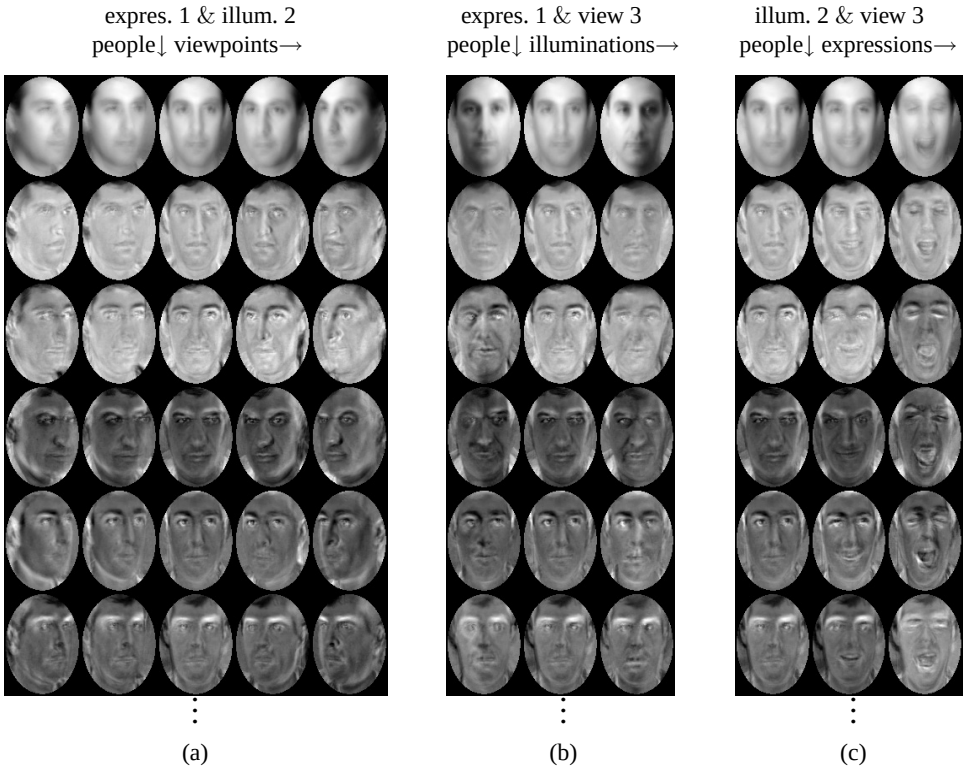

Fig. 7. This $28 \times 5 \times 3 \times 3 \times 7943$ tensor $\mathcal{Z} \times_2 \mathbf{U}_{\text{views}} \times_3 \mathbf{U}_{\text{illums}} \times_4 \mathbf{U}_{\text{expres}} \times_5 \mathbf{U}_{\text{pixels}}$ defines 45 different basis for each combination of viewpoints, illumination and expressions. These basis have 28 eigenvectors which span the people space. The topmost row across the three panels depicts the average person, while the eigenvectors in the remaining rows capture the variability across people in the various viewpoint, illumination, and expression combinations. (a) The first column is the basis spanning the people space in viewpoint 1, illumination 2 and expression 1, the second column is the basis spanning the people space in viewpoint 2, illumination 2 and expression 1, etc. (b) The first column is the basis spanning the people space in viewpoint 1, illumination 1 and expression 1, the second column is the basis spanning the people space in viewpoint 1, illumination 2 and expression 1, etc. (c) The first column is the basis spanning the people space in viewpoint 3, illumination 2 and expression 1, the second column is the basis spanning the people space in viewpoint 3, illumination 2 and expression 2, etc.

An important advantage of multilinear analysis is that it maps all images of a person, regardless of viewpoint, illumination and expression, to the same coefficient vector, given the appropriate choice of basis, thereby achieving zero intra-class scatter. Thus, multilinear analysis creates well separated people classes by maximizing the ratio of inter-class scatter to intra-class scatter [1]. By comparison, PCA will represent each different image of a person with a different vector of coefficients.

In our facial image database there are 45 images per person that vary with viewpoint, illumination, and expression. PCA represents each person as a set of 45 vector-valued coefficients, one for each image in which the person appears. The length of each PCA coefficient vector is $28 \times 5 \times 3 \times 3 = 1215$. By contrast, multilinear analysis enables us to represent each person with a single vector coefficient of dimension 28 relative to the bases comprising the tensor $\mathcal{Z} \times_2 \mathbf{U}_{\text{views}} \times_3 \mathbf{U}_{\text{illums}} \times_4 \mathbf{U}_{\text{expres}} \times_5 \mathbf{U}_{\text{pixels}}$, some of which are shown in Fig. 7. Each column in the figure is a basis and it is composed of 28 eigenvectors. In any column, the first eigenvector depicts the average person and the remaining eigenvectors capture the variability across people, for the particular combination of viewpoint, illumination, and expression associated with that column. The eigenvectors in any particular row play the same role in each column. This is the reason why images of the same person taken under different viewpoint, illumination, and expression conditions are projected to the same coefficient vector by the bases associated with these conditions.

6 Conclusion

We have identified the analysis of an ensemble of images resulting from the confluence of multiple factors related to scene structure, illumination, and viewpoint as a problem in multilinear algebra. Within this mathematical framework, the image ensemble is represented as a higher-dimensional tensor. This image data tensor must be decomposed in order to separate and parsimoniously represent the constituent factors. To this end, we prescribe the "N-mode SVD" algorithm, a multilinear extension of the conventional matrix singular value decomposition (SVD).

Although we have demonstrated the power of N-mode SVD using ensembles of facial images, which yielded TensorFaces, our tensor decomposition approach shows promise as a unifying mathematical framework for a variety of computer vision problems. In particular, it subsumes as special cases the simple linear (1-factor) analysis associated with conventional SVD and principal components analysis (PCA), as well as the incrementally more general bilinear (2-factor) analysis that has recently been investigated in the context of computer vision [4, 14]. Our completely general multilinear approach accommodates any number of factors by taking advantage of the mathematical machinery of tensors.

Not only do tensor decompositions play an important role in the factor analysis of multidimensional datasets, as described in this paper, but they also appear in conjunction with higher order statistics (higher order moments and cumulants) that are employed in independent component analysis (ICA). Hence, we can potentially apply tensor decomposition to ICA.

In future work, we will develop algorithms that exploit our multilinear analysis framework in a range of applications, including image compression, resynthesis, and recognition.

References

1. P.N. Belhumeur, J. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *Proceedings of the European Conference on Computer Vision*, pages 45–58, 1996.
2. R. Chellappa, C.L. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83(5):705–740, May 1995.
3. L. de Lathauwer. *Signal Processing Based on Multilinear Algebra*. PhD thesis, Katholieke Univ. Leuven, Belgium, 1997.
4. W. Freeman and J. Tenenbaum. Learing bilinear models for two-factor problems in vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 554–560, 1997.
5. A. Kapteyn, H. Neudecker, and T. Wansbeek. An approach to n-mode component analysis. *Psychometrika*, 51(2):269–275, June 1986.
6. T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.
7. J. B. Kruskal. Rank, decomposition, and uniqueness for 3-way and n-way array. In R. Coppi and S. Bolasco, editors, *Multiway Data Analysis*, pages 7–18, Amsterdam, 1989. North Holland.
8. J. R. Magnus and H. Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. John Wiley & Sons, New York, New York, 1988.
9. D.H. Marimont and B.A. Wandell. Linear models of surface and illuminance spectra. *J. Optical Society of America, A.*, 9:1905–1913, 1992.
10. Y. Moses, S. Edelman, and S. Ullman. Generalization to novel images in upright and inverted faces. *Perception*, 25:443–461, 1996.
11. A. Pentland and B. Moghaddam. View-based and modular eigenspaces for face recognition. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1994.
12. A. Shashua and A. Levin. Linear image coding for regression and classification using the tensor-rank principle. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page in press, Hawai, 2001.
13. L. Sirovich and M. Kirby. Low dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A.*, 4:519–524, 1987.
14. J.B. Tenenbaum and W.T. Freeman. Separating style and content. In M. Moser, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, pages 662–668. MIT Press, 1997.
15. M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analysers. *Neural Computation*, 11(2):443–482, 1999.
16. L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.
17. M. A. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
18. M. A. O. Vasilescu. An algorithm for extracting human motion signatures. In *IEEE Conference on Computer Vision and Pattern Recognition*, Hawai, 2001. in press.
19. M. A. O. Vasilescu. Human motion signatures for character animation. In *ACM SIGGRAPH 2001 Conf. Abstracts and Applications*, page 200, Los Angeles, August 2001.