Pattern Recognition 42 (2009) 2850 – 2862

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A unified tensor framework for face recognition

Santu Rana*, Wanquan Liu, Mihai Lazarescu, Svetha Venkatesh

Department of Computing, Curtin University of Technology, GPO Box U1987, Perth, WA 6845, Australia

ARTICLE INFO

Article history:
Received 29 August 2008
Received in revised form 26 February 2009
Accepted 7 March 2009

Keywords:
Face recognition
Tensor
Multilinear algebra

ABSTRACT

In this paper we propose a new optimization framework that unites some of the existing tensor based methods for face recognition on a common mathematical basis. Tensor based approaches rely on the ability to decompose an image into its constituent factors (i.e. person, lighting, viewpoint, etc.) and then utilizing these factor spaces for recognition. We first develop a multilinear optimization problem relating an image to its constituent factors and then develop our framework by formulating a set of strategies that can be followed to solve this optimization problem. The novelty of our research is that the proposed framework offers an effective methodology for explicit non-empirical comparison of the different tensor methods as well as providing a way to determine the applicability of these methods in respect to different recognition scenarios. Importantly, the framework allows the comparative analysis on the basis of quality of solutions offered by these methods. Our theoretical contribution has been validated by extensive experimental results using four benchmark datasets which we present along with a detailed discussion.

1. Introduction

In recent years, tensor based approaches have been widely studied as a viable solution to the challenging problem of face recognition across different lightings, viewpoints, expression, etc. Tensor modeling of face images [17] assumes that the face images are multilinear function of different factors of variation, i.e. person, lighting, viewpoint, pixel, etc. Whilst it is understood in appearance based face recognition research that face images actually span a non-linear manifold in the pixel space, factor based multilinear assumption seems to provide a good approximation with encouraging recognition results reported in [16,14,13]. These tensor based approaches generally perform better than the linear methods such as, the eigenface method [15] and are less computationally expensive than the non-linear methods. All the algorithms in [16,14,13] are based on the same multilinear PCA decomposition, however, they differ widely in their developments. Till now the only means of comparison between these methods is through testing them on various databases, which has the associated problem of dealing with inconsistent order of performance over different datasets. In absence of any theoretical basis of comparison it remains difficult to assess their performance over general scenarios without actually doing any experiments.

Our contribution is to propose a new tensor based face recognition framework that unites all these algorithms in [16,14,13] in one common mathematical basis, and in effect develop the required structure for explicit, non-empirical comparison. Tensor analysis through any form of multilinear decomposition is actually factor analysis with different assumptions on the relations of the factors. For face images, tensor analysis provides an ability to express a face image in terms of its constituent factors (i.e. person, lighting, viewpoint, etc.), while the relation between these factors are governed by the choice of specific multilinear decompositions (e.g. multilinear PCA/multilinear ICA, etc.) used. Therefore, the basic principle of any tensor based approach can be best understood in terms of the way it utilizes different factors. All existing methods lack this perspective wherein the methods are developed mostly through mechanical operations including tensor unfolding, matrix reorganization, etc., and thus provide no fundamental understanding. The description through *eigenmodes* and its variants are, in our opinion, secondary in nature, which neither describes the methods in a clear and concise manner nor provides a basis for comparison based on the quality of solutions. Essentially, we build our framework based on factor analysis principles and by re-formulating the above methods in our framework, we obtain clearer and more concise interpretation of their objectives.

We start by developing an optimization problem relating a face image to its constituent factors. This stems directly from the way face images are modeled through a tensor and its decomposition. We also note that from factor analysis principles, one factor is invariant to the changes of other factors, which implies that all the images of

* Corresponding author.
E-mail addresses: santu.rana@cs.curtin.edu.au (S. Rana), wanquan@cs.curtin.edu.au (W. Liu), m.lazarescu@cs.curtin.edu.au (M. Lazarescu), svetha@cs.curtin.edu.au (S. Venkatesh).

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.03.018

a person have the same value for its person-factor irrespective of the lighting or viewpoint conditions. Similarly, this notion carries forward to all the other factors. This independence property of factors is useful in the sense that if we can obtain the person-factor value or person-space representation of an image, we can readily identify it by matching it to the stored person-space representations of the known persons. We can solve this optimization problem directly using Alternating Least Squares (ALS) method to compute all the factors of that image. However, in practice ALS is extremely slow in convergence and most of the times fails to converge if the factor spaces are high-dimensional or highly correlated, as noted also in [11].

The shortcomings associated with the direct approach have motivated us to investigate alternative ways to solve this problem. In order to do so, we have developed a general framework, that offers two key advantages. Firstly, it allows the different assumptions about factors to be explicitly articulated. Secondly, it enables the development of different strategies for optimization that exploit specific factor spaces and their relationships. Within this parameters we reformulate the three existing approaches, i.e. MPCA-LV [16], MPCA-JS [14], MPCA-PS [13] along with the direct approach using ALS, which we name as MPCA-ML. The most important contribution of this paradigm is that it helps us to analyze their failure modes for different recognition scenarios. From the analysis, we realize that MPCA-ML is not a preferable choice because of the drawback of ALS when factor spaces are high-dimensional in the cases of large databases and the indefinite cost of testing associated with it. Similarly, the solution to MPCA-LV is heavily dependent on the assumption that the training dataset includes all possible test lighting/viewpoint conditions and is brittle when this condition is not true. MPCA-JS overcomes all the previous problems and offers the ideal solution for the optimization problem. However, the solution obtained is in terms of a descriptor from which different factors are impossible to separate, as opposed to the previous methods where person-space representations are used for discrimination. The discriminating power of this descriptor could be worse than using unique person factors, especially when the number of factors is high leading to an inconsistent behavior over different datasets. For MPCA-PS the overall solution is suboptimal, yet with the strategy adopted, MPCA-PS overcomes the limitation of MPCA-LV. Furthermore, it also improves upon MPCA-JS by keeping the person factor separate and using it indirectly for recognition. By being dependent on the person factor for discrimination it is able to provide low complexity testing and at the same time offers a consistent performance regardless of the number of factors as opposed to the case of MPCA-JS. The only point of failure may occur when dealing with similar looking persons whose person-factor values may be closer, thus resulting in higher uncertainty during recognition. However, as this failure mode is rare, we can expect that MPCA-PS will provide a consistent performance over a large number of datasets. MPCA-ML will fail for larger databases while MPCA-LV will perform poorly when test conditions are unseen. MPCA-JS generally provides a good performance, however, as the number of factors increases, the performance will decrease. It is imperative to note that we only gained this understanding by using our proposed framework.

Extensive experimentation is conducted over publicly available benchmark datasets to thoroughly compare each of the approaches and to validate and augment our understanding about them. The outline of the paper is as follows: Section 2 includes a review of few preliminary definitions and results related to tensor and tensor modeling of face images, while Section 3 includes a brief review of the existing tensor approaches for face recognition. In Section 4 we elaborate our framework and propose four different face recognition methods on this framework, Section 5 contains experimental results and detailed analysis of the results based on the proposed framework, and finally, Section 6 concludes the work.

2. Mathematical background

In this section, we first review a number of key definitions specific to tensors, then briefly describe multilinear PCA, and then propose our unified tensor modeling for face recognition.

2.1. Definitions related to tensor

2.1.1. Tensor as a higher order matrix

Mathematically, a tensor is an object which extends the notion of scalar, vector and matrix. Whilst a vector is an element of a vector space, a tensor is an element of a tensor space, which is defined as a tensor product of vector spaces.[1] The tensor product of two vectors results in a matrix and tensor product of N vectors results in a N-dimensional matrix. As a consequence, any data matrix of any dimensionality can be assumed to be a tensor with an implicit tensor space. Following this, we use the terms 'N-dim matrix' and 'N'th order tensor' interchangeably to refer to the same object. Also we note that, by kth 'mode' of a tensor we refer to the kth constituent vector space of the 'tensor space' the tensor belongs to.

2.1.2. Matricization of tensor: tensor unfolding at kth mode

Matricization of a tensor refers to the process of creating a matrix from a tensor of order > 2. This operation is needed to create more complex tensor operations using the usual matrix operations as the building blocks and by doing so it becomes easy to propagate the properties associated with matrix operations to the tensor case. It is often referred to as unfolding operation and defined specific to a mode. Following [11] the formal definition is given below:

Definition 1. Let A be a tensor of order N with $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. Then $A_{(k)}$ is denoted as tensor unfolding of A at kth mode, and defined as such that $A_{(k)} \in \mathbb{R}^{I_k \times (I_{k+1} \ldots I_N I_1 \ldots I_{k-1})}$ and it contains the element $a_{i_1 .. i_N}$ at i_kth row and at $[(i_{k+1}-1)I_{k+2} \ldots I_N I_1 \ldots I_{k-1} + (i_{k+2}-1)I_{k+3} \ldots I_N I_1 \ldots I_{k-1} + \cdots + (i_N - 1)I_1 \ldots I_{k-1} + (i_1 - 1)I_2 \ldots I_{k-1} + (i_2 - 1)I_3 \ldots I_{k-1} + \cdots + i_{k-1}]$th column.

Intuitively, the operation indicates slicing the tensor along a particular direction depending on the mode of unfolding, and then putting the slices side by side in a matrix. The pictorial description is given in Fig. 1 for a third order tensor. Correspondingly, folding of an unfolded tensor refers to the reverse operation of unfolding.

2.1.3. Matrix times tensor: mode-k multiplication

The mode-k multiplication of a tensor $A \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ by a matrix $U \in \mathbb{R}^{J_k \times I_k}$ is denoted by $B = A \times_k U$. $B \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_{k-1} \times J_k \times I_{k+1} \cdots \times I_N}$ and the entries of B are defined by

$$B_{i_1 \ldots i_{k-1} j_k i_{k+1} \ldots i_N} = \sum_{i_k=1}^{I_k} A_{i_1 \ldots i_{k-1} i_k i_{k+1} \ldots i_N} . U_{j_k, i_k} \tag{1}$$

Alternatively, it can also be shown that,

$$(A \times_k U)_{(k)} = U \cdot A_{(k)} \tag{2}$$

2.2. Multilinear PCA

PCA of an ensemble of images is performed by computing SVD of the image data matrix, $D \in \mathbb{R}^{I_1 \times I_2}$, whose columns contain *zero-mean* vectored images of size I_2. SVD orthogonalizes the two

[1] Let A, B are the vector spaces over the field $\mathbb{R}$. $A \otimes B$ is the tensor product of the vector spaces A and B and is constituted by all formal linear combinations of terms of the form $a \otimes b$: $\sum \alpha a \otimes b$, where $\alpha \in \mathbb{R}, a \in A$ and $b \in B$.

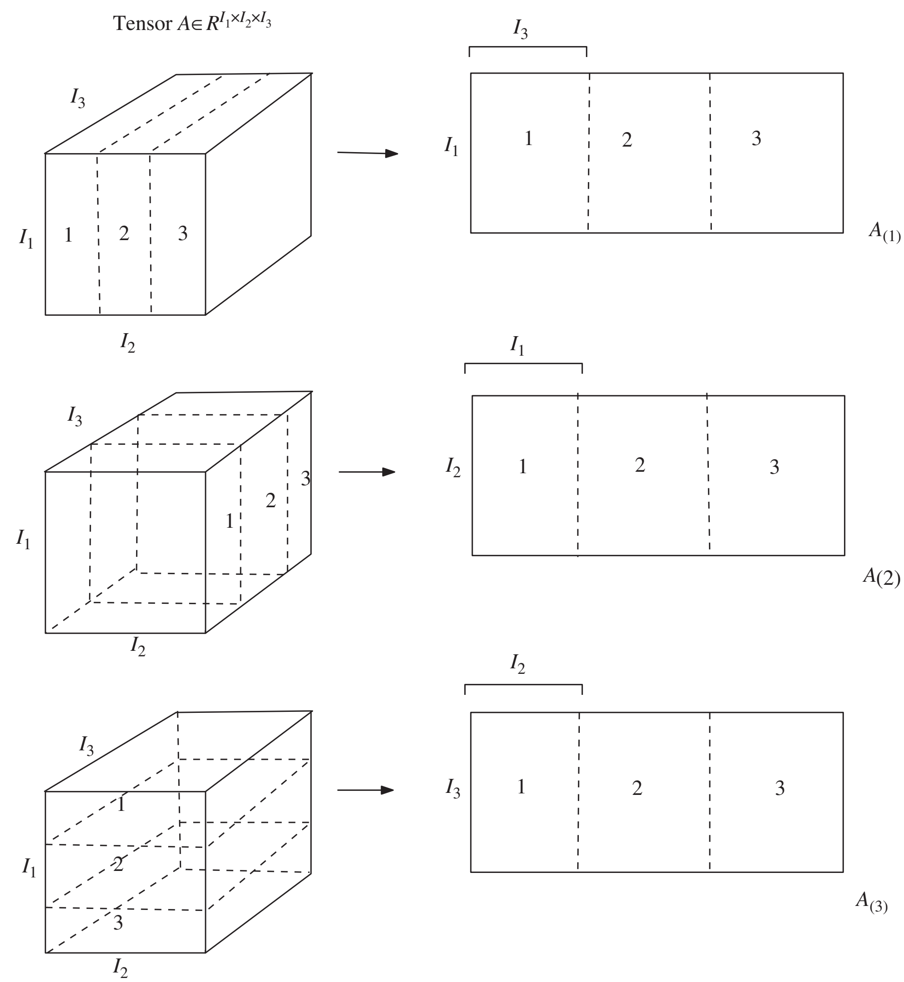

Fig. 1. Unfolding of a third order tensor $A \in \mathbb{R}^{3\times3\times3}$ in three different modes.

associated vector spaces of the two mode matrix D and decomposes the matrix as

$$D = U\Sigma V^T \tag{3}$$

where Σ is a diagonal matrix containing ordered *eigenvalues* and U, V are two orthonormal matrices, which are simply defined as *left-singular matrix* and *right-singular matrix*, respectively. Following the least squares formulation for PCA, it can be shown that, U contains ordered *principal direction of variation* (principal components) of the dataset in its columns. Similarly, N-mode SVD, a generalization of the SVD for higher order matrices [3], orthogonalizes "N" associated vector spaces of an N-way matrix (or Nth order tensor). If $\mathscr{D}$ is an nth order tensor and $\mathscr{D} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_n}$, the application of n-mode SVD orthogonalizes "n" associated vector spaces of $\mathscr{D}$ and decomposes the tensor as

$$\mathscr{D} = S\times_1 U^1 \times_2 \cdots \times_k U^k \cdots \times_n U^n \tag{4}$$

where $U^k, \forall k \in \{1, 2, \ldots, n\}$, is an orthonormal matrix and contains the ordered principal components for the kth mode. S is called the *core tensor*. For higher order cases ($n > 2$), S is not guaranteed to be diagonal, though there is an ordering in the *subtensors* of S and the *subtensors* are mutually orthogonal.[2] The decomposition algorithm is as follows:

(1) For $k = 1, \ldots, n$, compute matrix U^k by computing SVD on the *mode-k* flattening of the tensor $\mathscr{D}$ and set the left singular matrix as U^k.
(2) Compute core tensor S as

$$S = \mathscr{D}\times_1 U^{1^T} \times_2 \cdots \times_k U^{k^T} \ldots \times_n U^{n^T} \tag{5}$$

2.3. Tensor model for face recognition

Let us assume that our database contains images of persons with variations in lighting and viewpoint only. The tensor representation of the database is given by

$$T(i_p, i_l, i_v) = I_{P_{i_p}, L_{i_l}, V_{i_v}} \tag{6}$$

where $I_{P_{i_p}, L_{i_l}, V_{i_v}}$ is the image vector of i_pth person at i_lth lighting and i_vth viewpoint. T is a tensor of order 4 and,

$$T \in \mathbb{R}^{N_p\times N_l\times N_v\times N_x}$$

[2] A subtensor $S_{i_n=\alpha}$ for $S \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_n}$ is obtained by fixing the nth index to α. Orthogonality implies $\langle S_{i_n=\alpha}, S_{i_n=\beta}\rangle = 0$ for $\alpha \neq \beta$ and ordering implies $\|S_{i_1}\| \geqslant \|S_{i_2}\| \geqslant \cdots \geqslant \|S_{i_n}\|$.

where N_p is the number of persons, N_l & N_v represent the number of lighting and viewpoint instances, respectively, and N_x is the size of the image vector. Following the arrangement of images in T, the first mode will be referred to as the *person* mode, second as *lighting* mode, third as *viewpoint* mode and the fourth as *pixel* mode.

Multilinear PCA of tensor T yields four orthogonal subspaces, wherein each subspace corresponds to one mode of variation. This is represented as follows:

$$T = S \times_1 U^P \times_2 U^L \times_3 U^V \times_4 U^X \tag{7}$$

S is called the core tensor and the columns of U^P, U^L, U^V and U^X define the person, lighting, viewpoint and the pixel subspaces, respectively. The columns in U^X represent traditional *eigenfaces* and the columns of U^P, U^L and U^V, represent the $N'_p(N'_p \leqslant N_p)$, $N'_l(N'_l \leqslant N_l)$ and $N'_v(N'_v \leqslant N_v)$ dominant eigenvectors (or the principle axes of variation) of the person, lighting and viewpoint subspaces, respectively. We refer to these axes of variation as *eigenperson*, *eigenlighting* and *eigenviewpoint*, respectively. Also, it can be noted that each row of U^P corresponds to a particular person. We refer i_pth row of U^P as the *person-space* representation of the person i_p. Interestingly, all the training images of i_pth person, irrespective to any variation, are represented by a single point in the person-space. Similarly, i_lth row of U^L refers to the *lighting-space* representation of the i_lth lighting and i_vth row of U^V refers to the *viewpoint-space* representation of the i_vth viewpoint. The core tensor, $S \in \mathbb{R}^{N'_p \times N'_l \times N'_v \times N'_x}$, controls the mutual interaction between the person, lighting, viewpoint and pixel subspaces.

3. Related work

Linear models such as the eigenface [15] is suitable for face image analysis when the identity of persons is the only factor that is allowed to vary. However, when there are variation in multiple modes such as face images of different persons at different lightings and viewpoints then linear systems can no longer provide a suitable framework for analysis. As an alternative, multilinear modeling has been proposed in [16], where face images are assumed to be the result of multilinear interaction between different factors such as person, lighting, viewpoint, etc. Multilinear analysis provides concepts of eigenperson, eigenlighting, eigenviewpoint along with the traditional notion of eigenface as a special case. The power of such modeling lies in that it enables the construction of effective representations, depending on the variations observed in each subspace and the importance given to the associated factor. Face recognition in this framework was first studied in [16], and it showed a huge improvement in recognition performance over linear methods on a limited set of experiments. Recognition was performed by estimating person-space representation of a test image and then comparing it with the stored person-space representations. Similar approaches have been used in [18] for expression invariant face recognition, in [6] for simultaneous super-resolution and recognition and in [7] for human gait recognition. However, it has been pointed out in [14] that these approaches are not suitable for recognition in unseen modes as they do not utilize eigendecomposition information in factor spaces other than the person-space. Through intuitive justification that the core tensor is more suitable for face recognition in unseen modes as it contains information on variation for all the factors, they have proposed a new approach for recognition. In this approach, multilinear eigenmodes, which are the results of interaction between eigenpersons, eigenlightings and eigenviewpoints, are used to define a joint space. This space is used for face image representation and subsequent recognition through a comparison with the stored joint space coefficients of the training images. This approach supersedes previous algorithms both on performance and efficiency aspects over an extended set of experiments. However, this algorithm becomes prohibitively costly for large databases as recognition requires comparison with all the training images, leading to a test complexity of $O(N_T)$, where N_T is the total number of images. To reduce the testing complexity, [13] suggested a new approach that utilizes person-specific eigenmodes to define a set of projection bases, with each projection basis corresponding to a person. The test image is projected on the each basis, and the basis retaining the most information after projection determines the identity of the testing image. Reconstruction error is used to quantify the loss of information. This is an effort to compartmentalize the multilineareigenmodes space of [14] corresponding to persons. This results in a testing complexity of only $O(N_p)$, where N_p is the number of persons in the training database. As $N_p \ll N_T$, it provides a significant speedup over [14], while performing at par in recognition.

4. Proposed recognition framework

From (7), if $u_p(\in \mathbb{R}^{N'_p}), u_l(\in \mathbb{R}^{N'_l})$ and $u_v(\in \mathbb{R}^{N'_v})$ are person-space, lighting-space and viewpoint-space projections, respectively, for a test image I_T, we have,

$$I_T = S \times_1 u_p \times_2 u_l \times_3 u_v \times_4 U^X \tag{8}$$

In order to represent a testing image with the tensor framework, we need to compute u_p, u_l and u_v for I_T. For this purpose, we can formulate a multilinear least squares problem as,

$$\min_{u_p, u_l, u_v} \|I_T - S \times_1 u_p \times_2 u_l \times_3 u_v \times_4 U^X\|_2 \tag{9}$$

Next we will show four different ways to solve this optimization problem for recognition and demonstrate that some of them are just alternative derivations of the existing algorithms.

4.1. Approach 1: MPCA-ML

We have mentioned earlier that each person in the training database has a unique representation in the person-space. Therefore, the most direct way to perform recognition is to compare the person-space projection of the test image I_T with the person-space representation of all the training persons. We can solve the multilinear optimization problem (9) directly using alternating least squares method [9]. In this method u_p, u_l and u_v are initialized to some random values and then Eq. (9) is alternatively optimized over a single variable while keeping others fixed. After a sufficient number of iterations, we hope to achieve the convergence, indicated by minute difference in the consecutive estimates of the values of u_p, u_l and u_v. The method can be terminated when all the differences are below certain threshold or a certain number of iterations have reached. The recognition algorithm is elaborated in Algorithm 1.

The major advantage of this solution is that it is easy to incorporate any prior knowledge of lighting or pose of the test image by initializing or fixing u_l and u_v to certain values. Also, we obtain all the factor space projection of the test image, enabling us to estimate lighting or pose of the face in the same framework. However, as discussed in the Introduction, ALS method suffers from poor convergence rate and may fail to converge in case of high-dimensionality of factor spaces. This results in an extremely expensive method coupled with poor performance for large databases.

Algorithm 1. Testing algorithm for the MPCA-ML.

Test image: I_T

(1) ALS method: initialize $u_p \in \mathbb{R}^{N'_p}, u_l \in \mathbb{R}^{N'_l}$ and $u_v \in \mathbb{R}^{N'_v}$ to some random values.
- Step 1: Compute u_p^{new} as,
  $u_p^{new} = I_T \times ((S\times_2 u_l \times_3 u_v \times_4 U^X)_{(person)})^+$
- Step 2: Compute u_l^{new} as,
  $u_l^{new} = I_T \times ((S\times_1 u_p^{new} \times_3 u_v \times_4 U^X)_{(lighting)})^+$
- Step 3: Compute u_v^{new} as,
  $u_v^{new} = I_T \times ((S\times_1 u_p^{new} \times_2 u_l^{new} \times_4 U^X)_{(viewpoint)})^+$
- Step 4: Check convergence criterion and set $u_p = u_p^{new}$, $u_l = u_l^{new}$ and $u_v = u_v^{new}$.
- Step 5: Go to Step 1, if convergence is not achieved.

(2) If $\{c_{i_p}\}_{i_p=1}^{N_p}$ are the rows of U^P and if i_{p_m} minimizes
$\min_{i_p} \|u_p - c_{i_p}\|$
then I_T belongs to the person i_{p_m}

4.2. Approach 2: MPCA-LV

As seen in the previous algorithm, even though a direct multilinear solution using ALS offers a simple approach to solution, the computation requirement can be high even for small databases. Also, we estimate both the lighting-space and viewpoint-space projection of the test image, though only the person-space projection is sufficient for recognition. Considering this, the computational cost can be reduced if we avoid estimating u_l, u_v and instead solve (9) over a fixed set of $\{(u_l, u_v)\}$. Let (u_l^*, u_v^*) be a member from the set $\{(u_l, u_v)\}$, then Eq. (9) can be reformulated as

$$\min_{u_p} \|I_T - S\times_1 u_p \times_2 u_l^* \times_3 u_v^* \times_4 U^X\|_2 \tag{10}$$

Following the definition of mode-k multiplication to a tensor, this can equivalently be written as

$$\min_{u_p} \|I_T - u_p \times (S\times_2 u_l^* \times_3 u_v^* \times_4 U^X)_{(person)}\|_2 \tag{11}$$

Fortunately, this is a linear equation and the optimal u_p for (u_l^*, u_v^*) can be computed as:

$$u_p = I_T \times (S\times_2 u_l^* \times_3 u_v^* \times_4 U^X)^+_{(person)} \tag{12}$$

where the superscript $^+$ implies *Moore–Penrose pseudoinverse*. For each possible pairs of $\{u_l, u_v\}$ we obtain an optimal solution for u_p, resulting in a set of $\{u_p\}$. To create the set $\{u_l, u_v\}$, we assume that our training database is well represented, implying that the lighting conditions and pose of the testing images are very close to the training lighting conditions and pose. Therefore, we fix u_l to the set of training lightings and u_v to the set of training viewpoints. Let $u_l^{k_1} = k_1$th row of U^L and $u_v^{k_2} = k_2$th row of U^V, then the set is created as, $\{(u_l^{k_1}, u_v^{k_2})$ for $k_1 = 1, \ldots, N_l k_2 = 1, \ldots, N_v\}$. Clearly the cardinality of the set is $N_l N_v$. Using (12), we generate a set of person-space representation for I_T, over the set $\{(u_l, u_v)\}$ as

$$\begin{aligned} u_p^{k_1k_2} &= I_T \times (S\times_2 u_l^{k_1} \times_3 u_v^{k_2} \times_4 U^X)^+_{(person)} \quad \forall k_1, k_2 \\ &= I_T \times A_{k_1k_2} \end{aligned} \tag{13}$$

The set of person-space representations, $\{u_p^{k_1k_2}$ for $k_1 = 1, \ldots, N_l k_2 = 1, \ldots, N_v\}$, is then compared pairwise to the person-space representation of the training persons and the best matching person is found. The recognition algorithm is elaborated in Algorithm 2.

Algorithm 2. Testing algorithm for MPCA-LV.

Input: Test image $= I_T$
Precalculate: $A_{k_1k_2} = (S\times_2 u_l^{k_1} \times_3 u_v^{k_2} \times_4 U^X)^+_{(person)}$

(1) For $k_1 = 1, \ldots, N_l$ and $k_2 = 1, \ldots, N_v$ compute,
$u_p^{k_1k_2} = I_T \times A_{k_1k_2}$

(2) If $\{c_{i_p}\}_{i_p=1}^{N_p}$ are the rows of U^P and if i_{p_m} minimizes
$\min_{i_p,k_1,k_2} \|u_p^{k_1k_2} - c_{i_p}\|$
then I_T belongs to the person i_{p_m}

This algorithm is an attempt to solve the optimization problem in a deterministic time, though at the cost of obtaining suboptimal solution. The suboptimal solution is near-optimal when our assumption that the test images are not at wide variation compared to the training images is valid. However, if the variation is large, the estimation of u_p would be quite inaccurate and the recognition performance based on u_p would be poor. On the other hand, we still retain the advantage of incorporating prior knowledge in terms of manipulating u_l and u_v individually, either by fixing them to a certain set of values or excluding a certain subset of values from the existing set. It is also obvious that this approach is similar to the idea of [16]. However, by re-deriving it on our framework we are able to gain more insights on how this algorithm is expected to behave in different scenarios. Specifically, we show that the unability of this approach to handle faces at unseen modes as noted in [14,13] is simply because it is not designed to handle such conditions; thus providing a much simpler explanation than those offered in [14,13].

4.3. Approach 3: MPCA-JS

In this approach we exploit the structure of the core tensor to cast (9) as a linear least squares problem. From (9), we obtain person-space, lighting-space and viewpoint-space descriptions for a test image. The core tensor contains the multilinear relationship between the person-space, lighting-space and the viewpoint-space. The person-space, lighting-space and viewpoint-space projection vectors interacts with the core tensor in the mode-specific linear way to approximate the image. Through manipulating the structure inside the core tensor, we will shift the multilinearity from the core tensor side to the description side. An image will now be described by a multilinear function of person-space, lighting-space and viewpoint-space projection vectors and will be approximated by a linear multiplication of the description vector with a matrix derived from the core tensor. As a result, instead of multilinear optimization we obtain a linear equation to compute the description vector for a test image, thereby saving computation cost and solving it optimally. Following this idea, we provide the necessary derivation to develop a recognition algorithm.

Let us start by defining,

$$\mathscr{A} = S\times_4 U^X \tag{14}$$

With analogy to the definition of T, we can state that,

$$\mathscr{A}(i_p, i_l, i_v) = I_{P^e_{i_p} L^e_{i_l} V^e_{i_v}} \tag{15}$$

where $I_{P^e_{i_p} L^e_{i_l} V^e_{i_v}}$ is the image of i_pth eigenperson at i_lth eigenlighting and i_vth eigenviewpoint. Following the usual definition of u_p, u_l and u_v for the test image I_T we can write that,

$$I_T = S\times_1 u_p \times_2 u_l \times_3 u_v \times_4 U^X \tag{16}$$

And, following the definition of $\mathscr{A}$ from (14) we can rewrite the above as

$$I_T = \mathscr{A}\times_1 u_p \times_2 u_l \times_3 u_v \tag{17}$$

Next we provide the formulation to shift the multilinearity from the core tensor side to the description vector side. Applying the rule of mode multiplication, (17) can be written as

$$
\begin{aligned}
I_T &= \sum_{i_v=1}^{N'_v} u_v(i_v).\sum_{i_l=1}^{N'_l} u_l(i_l).\sum_{i_p=1}^{N'_p} u_p(i_p).I_{P^e_{i_p}L^e_{i_l}V^e_{i_v}} \\
&= \sum_{i_v=1}^{N'_v} u_v(i_v).\sum_{i_l=1}^{N'_l} u_l(i_l).u_p.\begin{bmatrix} I_{P^e_1L^e_{i_l}V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_{i_l}V^e_{i_v}} \end{bmatrix} \\
&= \sum_{i_v=1}^{N'_v} u_v(i_v).\left\{ u_l(1).u_p\begin{bmatrix} I_{P^e_1L^e_1V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_1V^e_{i_v}} \end{bmatrix} + \cdots + u_l(N'_l).u_p\begin{bmatrix} I_{P^e_1L^e_{N'_l}V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_{N'_l}V^e_{i_v}} \end{bmatrix} \right\} \\
&= \sum_{i_v=1}^{N'_v} u_v(i_v).u_l.\begin{bmatrix} u_p\begin{bmatrix} I_{P^e_1L^e_1V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_1V^e_{i_v}} \end{bmatrix} & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & u_p\begin{bmatrix} I_{P^e_1L^e_{N'_l}V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_{N'_l}V^e_{i_v}} \end{bmatrix} \end{bmatrix} \\
&= \sum_{i_v=1}^{N'_v} u_v(i_v).u_l.\underset{(N'_l \times N'_pN'_l)}{\begin{bmatrix} u_p & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & u_p \end{bmatrix}}.\begin{bmatrix} \begin{bmatrix} I_{P^e_1L^e_1V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_1V^e_{i_v}} \end{bmatrix} \\ \cdots \\ \cdots \\ \begin{bmatrix} I_{P^e_1L^e_{N'_l}V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_{N'_l}V^e_{i_v}} \end{bmatrix} \end{bmatrix}
\end{aligned}
\tag{18}
$$

Let us denote

$$
\begin{bmatrix} u_p & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & u_p \end{bmatrix}.\begin{bmatrix} \begin{bmatrix} I_{P^e_1L^e_1V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_1V^e_{i_v}} \end{bmatrix} \\ \cdots \\ \cdots \\ \begin{bmatrix} I_{P^e_1L^e_{N'_l}V^e_{i_v}} \\ \cdots \\ I_{P^e_{N'_p}L^e_{N'_l}V^e_{i_v}} \end{bmatrix} \end{bmatrix} = Y_{i_v},
$$

then we can rewrite (18) as

$$
\begin{aligned}
I_T &= \sum_{i_v=1}^{N'_v} u_v(i_v).u_l.Y_{i_v} \\
&= u_v.\begin{bmatrix} u_l.Y_{i_v=1} & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & u_l.Y_{i_v=N'_v} \end{bmatrix} = u_v.\underset{(N'_v \times N'_lN'_v)}{\begin{bmatrix} u_l & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & u_l \end{bmatrix}}.\begin{bmatrix} Y_{i_v=1} \\ .. \\ .. \\ Y_{i_v=N'_v} \end{bmatrix}
\end{aligned}
\tag{19}
$$

Substituting Y_{i_v} with its definition and after few more steps we obtain

$$
I_T = u_v.\underset{(N'_v \times N'_lN'_v)}{\begin{bmatrix} u_l & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & u_l \end{bmatrix}}.\underset{(N'_vN'_l \times N'_vN'_lN'_p)}{\begin{bmatrix} u_p & \ldots & \ldots & 0 \\ \ldots & u_p & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & u_p \end{bmatrix}}.\underset{(N'_vN'_lN'_p \times N_x)}{\begin{bmatrix} I_{P^e_1L^e_1V^e_1} \\ \cdots \\ I_{P^e_{N'_p}L^e_1V^e_1} \\ I_{P^e_1L^e_2V^e_1} \\ \cdots \\ I_{P^e_{N'_p}L^e_{N'_l}V^e_1} \\ I_{P^e_1L^e_1V^e_2} \\ \cdots \\ \cdots \\ I_{P^e_{N'_p}L^e_{N'_l}V^e_{N'_v}} \end{bmatrix}}
$$

$$
\Rightarrow I_T = f(u_p, u_l, u_v).\tilde{\mathscr{A}} \tag{20}
$$

$f : \mathbb{R}^{N'_p} \times \mathbb{R}^{N'_l} \times \mathbb{R}^{N'_v} \to \mathbb{R}^{N'_pN'_lN'_v}$ is a multilinear function over u_p, u_l, u_v. $\tilde{\mathscr{A}}$ contains images of all the eigenpersons at all the eigenlightings and eigenviewpoints, and therefore, a derivative from the core tensor. Let $d_T = f(u_p, u_l, u_v)$ be the description vector for I_T, then we can reformulate (9) as

$$
\min_{d_T} \|I_T - d_T.\tilde{\mathscr{A}}\|_2 \tag{21}
$$

Fortunately, this is a linear problem and the least squares solution for d_T is computed as

$$
d_T = I_T \times \tilde{\mathscr{A}}^{+} \tag{22}
$$

It is to be noted that, the optimal solution for d_T implies optimal solution for u_p, u_l and u_v as well, in contrast to the previous algorithm MPCA-LV, where we had a suboptimal solution to the optimization problem. However, on the flip side, given d_T, we cannot compute u_p uniquely, therefore, cannot use the usual recognition approach of comparing u_p in the person-space as followed by both MPCA-ML and MPCA-LV. We need to compute description vectors for all the training images and store them for recognition. Recognition follows an eigenface like algorithm, where d_T is compared to the stored description vectors and the best matching training image is found. Next, we will show that the description vectors of training images can be directly calculated from the matrices U^P, U^L and U^V in an efficient way.

Definition 2. Let us define C_P as

$$
C_P = \begin{bmatrix} C_P^{(1)} \\ C_P^{(2)} \\ \cdots \\ \cdots \\ C_P^{(N_p)} \end{bmatrix} \tag{23}
$$

where $C_P^{(k)}$ is defined as

$$
C_P^{(k)} = \begin{bmatrix} u_p^{(k)} & 0 & \ldots & 0 \\ 0 & u_p^{(k)} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & u_p^{(k)} \end{bmatrix} \tag{24}
$$

where $u_p^{(k)}$(=kth row of U^P) is repeated diagonally for $N'_lN'_v$ times.

Similarly we can define C_L as

$$C_L = \begin{bmatrix} C_L^{(1)} \\ C_L^{(2)} \\ \dots \\ \dots \\ C_L^{(N_l)} \end{bmatrix} \tag{25}$$

where $C_L^{(k)}$ is defined as

$$C_L^{(k)} = \begin{bmatrix} u_l^{(k)} & 0 & \dots & 0 \\ 0 & u_l^{(k)} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & u_l^{(k)} \end{bmatrix} \tag{26}$$

where $u_l^{(k)}$(=kth row of U^L) is repeated diagonally for $N'_v N_p$ times.

And C_V is

$$C_V = \begin{bmatrix} C_V^{(1)} \\ C_V^{(2)} \\ \dots \\ \dots \\ C_V^{(N_v)} \end{bmatrix} \tag{27}$$

where $C_V^{(k)}$ is defined as,

$$C_V^{(k)} = \begin{bmatrix} u_v^{(k)} & 0 & \dots & 0 \\ 0 & u_v^{(k)} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & u_v^{(k)} \end{bmatrix} \tag{28}$$

where $u_v^{(k)}$(=kth row of U^V) is repeated diagonally for $N_p N_l$ times.

Based on above definitions, the following theorem provides the desired relation between the image description vector and C_P, C_L and C_V, which were in turn calculated from the matrices U^P, U^L and U^V.

Theorem 1. *Let $\mathscr{M} = C_V \times C_L \times C_P$. If m_k is the kth row of the matrix $\mathscr{M}$ then,*

$$m_k = I_{P_{i_p} L_{i_l} V_{i_v}} \times \tilde{\mathscr{A}}^+ \tag{29}$$

where $i_p = ((k-1) \bmod N_p + 1)$, $i_l = ((\lceil k/N_p \rceil - 1) \bmod N_l + 1)$ and $i_v = (\lceil k/(N_p \times N_l) \rceil)$.

Proof. Refer to Appendix A.

It follows from the above theorem that matrix $\mathscr{M} = C_V \times C_L \times C_P$ contains the description vectors, $f(u_p, u_l, u_v)$ of all the training images. Each row m_k of the matrix $\mathscr{M}$ refers to the description vector of a training image, whose person, lighting and viewpoint indices are provided by the above theorem. The algorithm for testing is given in Algorithm 3.

Algorithm 3. Testing algorithm for MPCA-JS.

Input: Test image $= I_T$

Precalculate: $P = \tilde{\mathscr{A}}^+$

(1) Given the test image I_T, find the corresponding description vector d_T as:
$d_T = I_T \times P$

(2) Use a Nearest Neighbor classifier to find the best matching description vector m_b, i.e. that minimizes,
$\min_k \|d_T - m_k\|$ for $k = 1, \dots, (N_p \times N_l \times N_v)$
where m_k is the kth row of the matrix $\mathscr{M}$. The distance measure we use is the *cosine distance*. For two vectors **a** and **b** the *cosine distance* between them is defined as,
$cosine\_dist(\mathbf{a}, \mathbf{b}) = \dfrac{\langle \mathbf{a}, \mathbf{b} \rangle}{\|\mathbf{a}\| \|\mathbf{b}\|}$

(3) The person identity, p_T for the test image is the person identity of the best matching vector m_b (bth row of $\mathscr{M}$) and it is given by,
$p_T = ((b-1) \text{mod} N_p + 1)$

Clearly, this algorithm provides a deterministic approach to solve (9) without losing the optimality in solution, and hence, superior to both the previous algorithms. Another major advantage is that it follows the conventional method of recognition, i.e. projection (generation of description vector) followed by classification. Therefore, sophisticated classifiers can be used to improve the performance of the testing algorithm. However, a naive implementation using NN classifier can be enormously expensive, as the search complexity is $O(N_p N_l N_v N'_p N'_l N'_v)$. In comparison to it, MPCA-ML has a search complexity of only $O(N_p N'_p)$ and MPCA-LV has a search complexity of only $O(N_l N_v N_p N'_p)$. Another point to be noted that, it actually uses a modified descriptor for recognition, as opposed to the person factor used in the previous method. Also, it is obvious from the algorithm that this approach is similar to the ideas presented in [14]. However, through this alternative derivation we are able describe the methods in a cleaner way by explicitly stating the quality of solution it offers and then linking the performance to it. However, this approach is distinct than the previous approaches in the sense that it uses a multilinear combination of all factors as the descriptor instead of the usual person-space representation. The power of discrimination of this descriptor may reduce considerably as more and more factors are added with the person-factor. Therefore, this method will provide good performance when number of factors is low, and may fail when the number becomes higher. This particular insight cannot be obtained from the description given in the initial paper [14].

4.4. Approach 4: MPCA-PS

In all the previous approaches, whilst solving (9) we let u_p to be a free variable and solved the optimization problem. For recognition, we either directly used u_p for recognition as in MPCA-ML and MPCA-LV, or used a description that is a function of u_p, as in MPCA-JS. However, as we recognize persons that are present in our training database, we expect u_p to be the same as that of person-space representation of the person the test image belongs to. Following this idea, we can simplify the optimization problem of (9), by finding its minima over the set of person-space representations of the training persons. As each individual row of U^P refers to a specific person, we make a candidate set of u_p such that $\{u_p^k = k\text{th row of } U^P$, for $k = 1, 2, \dots, N_p\}$. For each u_p^k we generate a multilinear optimization subproblem that is multilinear in u_l, u_v. We follow similar techniques as in MPCA-JS to solve these multilinear subproblems and finally we compute the overall minima out of those subproblems. The specific u_p^k, which provides the overall minima, reveals the identity of the test image as that of person k. Precisely, we modify (9) as

$$\min_{k,u_l,u_v} \|I_T - S\times_1 u_p^k \times_2 u_l \times_3 u_v \times_4 U^X\|_2 \tag{30}$$

Next, we develop a mechanism to solve this optimization problem. Let us start by defining,

$$\mathscr{B}_k = S\times_1 u_p^k \times_4 U^X \tag{31}$$

Clearly, $\mathscr{B} \in \mathbb{R}^{1\times N_l'\times N_v'\times N_x}$. With analogy to the definition of T, we can state that,

$$\mathscr{B}_k(1, i_l, i_v) = I_{P_k L_{i_l}^e V_{i_v}^e} \tag{32}$$

where $I_{P_k L_{i_l}^e V_{i_v}^e}$ is the image of kth person at i_lth eigenlighting and i_vth eigenviewpoint. If the test image, I_T belongs to the Kth person and, u_l, u_v are its projection in lighting and viewpoint-spaces, respectively, then,

$$\begin{aligned} I_T &= S\times_1 u_p^K \times_2 u_l \times_3 u_v \times_4 U^X \\ &= \mathscr{B}_K \times_2 u_l \times_3 u_v \end{aligned} \tag{33}$$

Following the rule of mode multiplication, (33) can be written as

$$\begin{aligned} I_T &= \sum_{i_v=1}^{N_v'} u_v(i_v) \cdot \sum_{i_l=1}^{N_l'} u_l(i_l) \cdot I_{P_K L_{i_l}^e V_{i_v}^e} \\ &= \sum_{i_v=1}^{N_v'} u_v(i_v).u_l. \begin{bmatrix} I_{P_K L_1^e V_{i_v}^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_{i_v}^e} \end{bmatrix} \\ &= u_v(1).u_l \begin{bmatrix} I_{P_K L_1^e V_1^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_1^e} \end{bmatrix} + \cdots + u_v(N_v').u_l \begin{bmatrix} I_{P_K L_1^e V_{N_v'}^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_{N_v'}^e} \end{bmatrix} \\ &= u_v. \begin{bmatrix} u_l \begin{bmatrix} I_{P_K L_1^e V_1^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_1^e} \end{bmatrix} & \cdots & 0 \\ 0 & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & u_l \begin{bmatrix} I_{P_K L_1^e V_{N_v'}^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_{N_v'}^e} \end{bmatrix} \end{bmatrix} \end{aligned} \tag{34}$$

Continuing the derivation we obtain,

$$\begin{aligned} I_T &= u_v \underset{N_v' \times N_l' N_v'}{\begin{bmatrix} u_l & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & u_l \end{bmatrix}} . \begin{bmatrix} \begin{bmatrix} I_{P_K L_1^e V_1^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_1^e} \end{bmatrix} \\ \cdots \\ \cdots \\ \begin{bmatrix} I_{P_K L_1^e V_{N_v'}^e} \\ \cdots \\ I_{P_K L_{N_l'}^e V_{N_v'}^e} \end{bmatrix} \end{bmatrix} \\ \Rightarrow I_T &= g(u_v, u_l).\tilde{\mathscr{B}}_K \end{aligned} \tag{35}$$

where $g : \mathbb{R}^{N_l'} \times \mathbb{R}^{N_v'} \to \mathbb{R}^{N_l' N_v'}$ is a multilinear function over u_l, u_v. $\tilde{\mathscr{B}}_k$ contains images of kth person at all the eigenlightings and eigenviewpoints. Let, $c_T = g(u_v, u_l)$ is the description vector of lighting and pose for I_T then we can reformulate (30) corresponding to the person K as

$$\min_{c_T} \|I_T - c_T \times \tilde{\mathscr{B}}_K\|_2 \tag{36}$$

This is a linear optimization problem such and the least squares solution for c_T, specific to $k = K$ is computed as

$$c_T^K = I_T \times \tilde{\mathscr{B}}_K^+ \tag{37}$$

In order to find the optimal k, we can write (30) as

$$\min_k \|I_T - c_T^k \times \mathscr{B}_k\|_2 \tag{38}$$

The algorithm for testing is given in Algorithm 4.

Algorithm 4. Testing algorithm for MPCA-PS.

Testing: I_T is the test image.

(1) For every $k = 1, \ldots, N_p$ compute
  - $c_T^k = I_T \times \tilde{\mathscr{B}}_k^+$
  - $e^k = \|I_T - c_T^k \times \tilde{\mathscr{B}}_k\|$

  The distance measure we use is Euclidean distance.

(2) If e^{k_m} is the smallest of $\{e_k\}_{k=1}^{N_p}$ then I_T belongs to the person k_m.

In this approach we are enforcing suboptimality to the solution by optimizing (9) over a set of u_p. However, for a fixed u_p we solve for optimal values of u_l and u_v. Though both MPCA-LV and this approach generate suboptimal solutions, the fundamental difference between them is that, in MPCA-LV we fix u_l, u_v to the training lighting and viewpoint cases and optimally solve for u_p, whereas in this case we fix u_p to the training persons and optimally solve for u_l, u_v. Fixing u_l, u_v implies that we expect lighting condition and viewpoint of a test image to be almost similar to any of the training lighting and viewpoint cases. However, fixing u_p implies that we expect the test image to have the same person-space representation as the person the test image belongs to. Clearly, the later assumption is more logical and the suboptimal solution for this case, should almost be near-optimal, implying we should expect similar recognition performance as that of MPCA-JS. Moreover, as this algorithm has search complexity of only $O(N_p N_x)$, it is much more efficient than MPCA-JS. On the down side, due to the way recognition is performed, we cannot further reduce the time. In comparison to it, MPCA-JS can be made faster by using sophisticated classifiers. It is also obvious that this approach is similar to the ideas presented in [13]. However, with this derivation we are able to express it in a clearer fashion by explicitly stating the quality of solution it offers. Firstly, it solves the optimization problem almost-optimally, thereby overcoming the drawback of MPCA-LV. Secondly, it utilizes person-space indirectly for recognition thereby overcoming the limitation of MPCA-JS. This explanation is much neat than the *eigenmode*-based explanation as offered in the initial paper [13].

4.5. Theoretical analysis of four approaches

As we see in Table 1, we were able to describe all the four methods in a coherent way in terms of assumption they make and strategies they adopt to solve the multilinear optimization problem (9). Critically, we are also able to outline their failure modes based on the analysis of solution mechanisms they follow and get an understanding of their expected behaviors and consequent deployability in different face recognition scenarios. Firstly, the direct approach

Table 1
Summary of all the four approaches developed in this paper for face recognition using tensor methods.

Method	Assumptions	Solution strategy (A)	Quality of solution	Recognition method (B)	Recognition cost= cost of (A)+cost of (B)
MPCA-ML	None	Brute force—using alternating least squares (ALS)	Optimal if global optima is reached (however, ALS fails to converge if factor spaces are high-dimensional or factors are highly correlated)	Compare u_p with stored person-space projection of known persons	Indefinite $+ O(N_p N'_p)$
MPCA-LV	Lighting and viewpoint for the test image is almost similar to training lightings and viewpoints	Fix (u_l, u_v) to training lighting–viewpoint combination and simplify (9) to obtain $N_l N_v$ no. of linear optimization problems (11). Solve all the linear optimization problems to generate a candidate set of u_p	Sub-optimal (extremely poor if test lighting/viewpoint conditions are quite different from that of the training set)	Compare the candidate set of u_p with stored person-space projection of known persons in a pair-wise manner	$N_l N_v$ projections+ $O(N_l N_v N_p N'_p)$
MPCA-JS	None	Manipulate the structure of core tensor to cast the multilinear optimization problem of (9) as a linear optimization problem in (21). The solution is obtained as a descriptor, that is a multilinear function of all the factors	Optimal (however, the factors are unseparable and the multilinear descriptor may not have the same discriminative power as that of u_p)	Compare the description vector (of length $N'_v N'_l N'_p$) with stored description vectors of all training images	Only one projection+ $O(N_v N_l N_p\ N'_v N'_l N'_p)$
MPCA-PS	Person-space projection of the test image should be close to the person-space representation of the training person the test image belongs to	Fix u_p to the person-space representation of training persons and divide (9) into N_p no. of multilinear optimization subproblems, each specific to one person (30). Manipulate the structure of core tensor to cast each of them as a linear optimization problem (36)	Sub-optimal (however, highly optimal if testing involves only known persons and it indirectly uses u_p to discriminate)	Among the N_p optimization problems, find the one which generate the least error in approximating the test image	N_p projections $+ O(N_p N_x)$

MPCA-ML is not preferable for two reasons: (1) the failure of the optimization routine when the factor spaces are high-dimensional as in the cases of large databases and (2) indefinite cost of testing. MPCA-LV, though a linear time algorithm, is heavily dependent on the assumption that the training dataset contains images that cover all possible test lighting/viewpoint conditions and therefore is brittle when the conditions are not met. MPCA-JS overcomes all the previous problems and offers the best solution for the optimization problem. However, the solution obtained is in terms of a descriptor from which different factors are impossible to separate. The discriminating power of this descriptor is hard to predict and could be worse than using unique person factors, particularly when the number of factors goes higher because of the reduced influence of the unique person factor in the descriptor. For MPCA-PS the overall solution is suboptimal as we do not compute person-factor value, however, for each fixed value of the person-factor (respective to each known persons) we solve for all other factors optimally. Thereby, the solution is nearly optimal if the test images are only from known persons. With this strategy, MPCA-PS can on one hand overcome the shortcoming of the MPCA-LV approach while on the other hand it improves upon MPCA-JS by keeping the person factor separate and using it indirectly for recognition. By being dependent on person factor for discrimination it is able to provide low complexity testing while at the same time providing immunity from any performance variation due to varying number of factors as was in the case of MPCA-JS. Hence, we can expect that MPCA-PS will provide consistent performance over any kind of dataset. MPCA-ML will fail for larger databases. MPCA-LV will have a poor performance when test conditions are unseen while MPCA-JS will typically provide good performance with the condition that an increase in the number of factors will negatively impact the overall performance. In the next section we will validate these theoretical insights by experimentation on different databases.

5. Experiments, analysis and evaluation

In this section, we provide experimental results of the different tensor based approaches we derived, on a number of publicly available benchmark datasets. Specifically, we used PEAL [4], YaleB frontal [5], Extended YaleB [12] and Weizmann face database for experiments. The Extended YaleB database contains images of 38 persons at 64 different lightings and at nine different viewing directions. The YaleB frontal is a subset of the Extended YaleB database, which contains images of 38 persons at 64 different lighting conditions, at the frontal viewpoint only. PEAL is a face-database of Chinese nationals at different poses, lightings and expressions [4]. From this database we created PEAL lighting variation dataset by choosing images of 150 persons at different lighting conditions. The lighting conditions have three different modes: light focussing on the middle of the faces (M), light focussing from up (U) and light focussing from down (D). Each mode has light focussing from five different directions ($0^\circ, \pm 45^\circ, \pm 90^\circ$), resulting in 15 different lighting conditions, though images at all 15 lighting conditions are not available for all persons. Primarily the light used is of incandescent type, however, occasionally fluorescent lamps were used, resulting in a maximum of five extra lighting conditions. In the dataset the minimum number of image a person has is six whilst the maximum number is 20. However, only three images per person are available at exactly the same lighting conditions, i.e. at M—0°, U—0° and D—0°. For the Weizmann face database, we have access to face images of 28 persons at five different viewpoints ($0^\circ, \pm 17^\circ, \pm 34^\circ$), three different lightings and two different expressions. Prior to the experiments, all the images were cropped and their eye-points were manually aligned. Then all the image vectors were normalized to unity. For HOSVD and other tensor operations, we used the tensor toolbox developed by Bader and Kolda in MATLAB™ [1].

Table 2
Recognition performance on YaleB lighting variation dataset.

Recognition method	Avg. accuracy (%) (std. deviation)			
	5 train	10 train	15 train	20 train
PCA	53.71 (9.6)	69.20 (5.3)	77.69 (4.8)	88.29 (3.9)
MPCA-ML	71.67 (11.5)	85.92 (4.3)	91.37 (2.3)	93.36 (2.0)
MPCA-LV	43.22 (14.3)	56.41 (9.6)	63.30 (9.3)	72.66 (8.6)
MPCA-JS	76.16 (10.0)	87.52 (5.8)	92.78 (2.5)	95.02 (1.7)
MPCA-PS	**77.33** (11.1)	**90.40** (4.1)	**94.38** (1.5)	**95.52** (1.5)
PCA + LDA	72.53 (11.7)	85.07 (6.1)	90.63 (3.6)	93.44 (3.0)
PCA + LPP	72.98 (10.8)	86.97 (4.3)	91.84 (2.4)	94.38 (2.0)

Table 3
Testing time for YaleB lighting variation dataset.

Recognition method	Avg. time (s)			
	5 train	10 train	15 train	20 train
PCA	0.11	0.21	0.34	0.45
MPCA-ML	14.75	18.83	25.76	32.70
MPCA-LV	0.83	2.86	4.37	5.85
MPCA-JS	0.27	0.85	1.41	2.20
MPCA-PS	0.10	**0.15**	**0.19**	**0.24**
PCA + LDA	**0.09**	0.16	0.25	0.31
PCA + LPP	0.10	0.21	0.32	0.43

Test time is for 100 test images.

For the YaleB frontal dataset, four sets of experiments were performed with randomly chosen 5, 10, 15 and 20 lighting conditions for training and the rest for testing. For experiments on the Extended YaleB database, 16 representative lighting conditions at five representative viewpoints were used for training and the rest for testing. For experiments on Weizmann face database 3 representative viewpoints (at 0° and ±34°), at two randomly selected lighting conditions and at all the expressions were used for training and the rest for testing. For experiments on the PEAL lighting variation dataset, three lighting conditions at M—0°, U—0° and D—0° were used for training and the rest 1040 images for testing. Thus, in both the PEAL database and YaleB frontal database, the test images were at unseen lighting. For both Extended YaleB database and Weizmann database, the test images were either at unseen lighting or unseen viewpoints or both. For the experiments on YaleB frontal, each set of experiments was repeated 20 times and the average and the standard deviations of the results are reported. Experiments were performed with MPCA-ML, MPCA-LV [16], MPCA-JS [14], MPCA-PS [13] along with PCA [15] for baselining and LDA [2] and LPP [8] to benchmark results produced by the tensor based methods against the leading ones. All the timings reported here are for Matlab code running on a Intel Xeon 8×2.3 GHz server with 16 GB RAM without any extra effort for code optimization.

We have used *energy thresholding* to select the number of *eigenvectors* to be retained (i.e. for PCA and for Pixel mode of MPCA the energy threshold = 0.96, at all other modes of MPCA the threshold = 0.99). LDA and LPP was performed in the reduced dimension after performing PCA on the data with the same threshold to maintain uniformity. Face recognition using PCA, LDA and LPP used Euclidean distance measure for comparison, while all tensor based approaches used cosine distance measure. For MPCA-ML, the convergence was decided to have been achieved when norm of the difference of two consecutive estimates of u_p, u_l, u_v reached less than 0.000001. However, the maximum number of iteration was capped at 2000.

Tables 2 and 3 present the recognition performance and test time for experiments on YaleB dataset. Experimental results for Extended YaleB database, Weizmann database and PEAL lighting variation dataset have been tabulated in Tables 4, 5 and 6, respectively.

From the results we can list the following observations:

- In most of the experiments, MPCA-PS outperformed all other approaches including LDA and LPP. Particularly, in Weizmann dataset it showed excellent recognition accuracy. We attribute this to the nature of the dataset, where the appearance of persons are very much different from each other, resulting in a lesser overlap among *person-specific eigenmode* spaces. The opposite reasoning applies to the PEAL dataset, where persons share some similarities in the appearances, resulting in a higher overlap of the *person-specific eigenmode* spaces. Therefore, we observe that MPCA-PS performed somewhat worse than MPCA-JS in the PEAL dataset, though comparable to LDA and LPP. Nevertheless, efficiency wise MPCA-PS always outperformed other approaches by significant amount. Also in Table 2 note that the MPCA-PS have the least variation of results due to different partitioning of YaleB lighting variation dataset for higher training cases, which shows the robustness of the algorithm. In Extended YaleB dataset, which is the largest among all the datasets we have, MPCA-PS is almost twice faster than PCA and MPCA-LV, almost 350 times faster than MPCA-ML, and around 14 times faster than MPCA-JS.
- In most of the cases, MPCA-JS performed almost similar to MPCA-PS, except in the experiments on Weizmann dataset. There, it performed even worse than PCA. The reason, as pointed out correctly in [13], is the poor generalization of the NN classifier in the high-dimensional *multilinear eigenmode* space. Also it is to be noted that in the PEAL dataset it performed better than LDA and LPP as well.
- In all the experiments, MPCA-LV performed very poorly in terms of recognition accuracy, which is reflective of our analysis discussed before, that this method is not suitable for application for recognition under unseen lightings and viewpoints.
- Finally we observe that, MPCA-ML showed poor performance except in the case of YaleB lighting variation dataset, which concurred with our discussion of this method in the corresponding section. Better performance in YaleB is the result of small number of variables as well as their small dimensions. Compared to YaleB lighting variation dataset, YaleB extended dataset has three variables, Weizmann has four variables and although PEAL dataset has only two variables the person-space is considerably higher (129 compared to $\leqslant 38$ for YaleB), which results in poor performance of MPCA-ML in these datasets. We also note the high testing time required for this method. To further study the behavior of ALS we experimented on Extended YaleB, Weizmann and PEAL dataset, by initializing person-space variable u_p with the expected solution, i.e. the known person-space representation of the test person. The intent is to find the performance of MPCA-ML when theALS gets closer to the global minima. This accuracies are shown in brackets in the MPCA-ML columns of Tables 4–6. We observe the recognition for this case goes closer to the best recognition rate as obtained from MPCA-JS and MPCA-PS. Therefore, establishing our viewpoint that the poor performance of MPCA-ML is actually the result of the poor behavior of ALS and not the way the problem is formulated.

From the above discussion we can conclude that the approaches MPCA-LV and MPCA-ML with ALS as the optimization solver is not suitable for any practical deployments. A possible further research can be envisioned to discover a better optimization routine than ALS and investigate the performance of MPCA-ML in that setting. However, within our realm of study it appears that any possible practical tensor based face recognition application may be built around either MPCA-JS or MPCA-PS on a case by case basis. Whilst in absolute term it is not possible to predict superior recognition performance of one above the other, we can state that on an average case MPCA-PS will provide acceptable accuracy at the least test time, making it a better candidate than the other.

Table 4
Experimental results for Extended YaleB dataset with lighting + viewpoint variation.

	Recognition method						
	PCA	MPCA-ML	MPCA-LV	MPCA-JS	MPCA-PS	PCA + LDA	PCA + LPP
Accuracy (%)	67.48	75.95 (79.77)	49.89	84.20	**85.03**	**85.03**	84.30
Test time (s)	1.78	283.54	1.51	10.54	**0.79**	1.23	1.64

Test time is for 100 test images.

Table 5
Experimental results for Weizmann database with lighting+viewpoint variation.

	Recognition method						
	PCA	MPCA-ML	MPCA-LV	MPCA-JS	MPCA-PS	PCA + LDA	PCA + LPP
Accuracy (%)	86.71	84.52 (86.31)	69.64	77.18	**98.21**	97.82	91.67
Test time (s)	0.20	40.38	0.20	0.53	**0.12**	0.12	0.20

Test time is for 100 test images.

Table 6
Experimental results for PEAL lighting variation dataset.

	Recognition method						
	PCA	MPCA-ML	MPCA-LV	MPCA-JS	MPCA-PS	PCA + LDA	PCA + LPP
Accuracy (%)	74.79	74.04 (83.08)	75.10	**92.79**	87.79	89.89	88.94
Test time (s)	2.45	445.01	1.31	2.14	**1.23**	1.68	2.44

Test time is for 100 test images.

6. Conclusion

In this paper we unified a number of the existing tensor based approaches for face recognition on a single optimization framework. The major contribution of work has been the development of the framework which is effectively a structure that enables an explicit, non-empirical comparison of the tensor methods. This in turn has enabled us to gain a deep understanding of the applicability of these methods to different face recognition tasks. From our analysis, we have determined that the MPCS-PS overall is likely to produce the best performance for all types of datasets, despite the fact that it provides a sub-optimal solution. MPCA-JS provides an optimal solution and generally produces good results but is susceptible to an increase in the number of variations. The third approach evaluated, MPCA-ML, is impractical when dealing with large databases. Our observations for MPCA-ML suggest that the problem lies in the optimization process which may not converge to an acceptable solution when factor spaces are high-dimensional. The MPCA-LV approach will, in most cases, have the worst performance overall as it is not designed to handle unseen test conditions and therefore it is the least preferred solution.

We have conducted an extensive testing and evaluation of these methods with four benchmark datasets: YaleB lighting variation dataset, Extended YaleB dataset with lighting and viewpoint variation, Weizmann dataset and PEAL dataset. The results we have obtained confirm our theoretical analysis of the methods. Specifically, MPCA-LV performed poorly in all the experiments as recognition was always performed with at least one unseen mode (i.e. either lighting or viewpoint or both unseen). MPCA-ML gave good performance on YaleB lighting variation dataset as this is a small dataset with only one mode of variation (i.e. lighting), however, performance deteriorated considerably for other datasets which have either a large number of persons or a higher number of varying modes. In general, MPCA-JS showed good performance when number of varying modes is small, but fared badly on the Weizamann dataset as it has three varying modes (i.e. lighting, viewpoint and expression). Overall MPCA-PS provided good and consistent recognition accuracy. Its performance was often better than the leading face recognition methods, such as LDA and LPP. Moreover, it takes the least testing time among all the methods compared. Therefore, in our opinion, the MPCA-PS approach makes a competent candidate for use in real-world face recognition applications.

Future work may involve investigating more effective and efficient optimization routine for the MPCA-ML method and more rigorous validation of the MPCA-PS approach over larger and complex databases. Also, this method can be tried over face images taken with different types of sensor such as near-infrared sensors as discussed in [10]. We also envisage that any future development of tensor based approach for different types of face recognition problem will benefit from the framework presented in our paper and may eventually further research interest in this direction.

Acknowledgments

We thank the anonymous reviewers for their constructive comments, that helped us to improve this paper.

Appendix A. Proof of Theorem 1

Proof. Let us start with the definition of $\tilde{\mathscr{A}}$,

$$\tilde{\mathscr{A}} = \begin{bmatrix} I_{P_1^e L_1^e V_1^e} \\ \cdots \\ I_{P_{N_p'}^e L_1^e V_1^e} \\ I_{P_1^e L_2^e V_1^e} \\ \cdots \\ I_{P_{N_p'}^e L_{N_l'}^e V_1^e} \\ I_{P_1^e L_1^e V_2^e} \\ \cdots \\ \cdots \\ I_{P_{N_p'}^e L_{N_l'}^e V_{N_v'}^e} \end{bmatrix} \quad \text{(A.1)}$$

Let

$$D_1^{(k)} = C_P^{(k)} \times \tilde{\mathscr{A}} \tag{A.2}$$

then substituting the $C_P^{(k)}$ from (24) and $\tilde{\mathscr{A}}$ from (A.1),

$$D_1^{(k)} = \begin{bmatrix} u_p^k & 0 & \dots & 0 \\ 0 & u_p^k & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & u_p^k \end{bmatrix} . \begin{bmatrix} I_{P_1^e L_1^e V_1^e} \\ \dots \\ I_{P^e_{N'_p} L_1^e V_1^e} \\ I_{P_1^e L_2^e V_1^e} \\ \dots \\ I_{P^e_{N'_p} L^e_{N'_l} V_1^e} \\ I_{P_1^e L_1^e V_2^e} \\ \dots \\ \dots \\ I_{P^e_{N'_p} L^e_{N'_l} V^e_{N'_\nu}} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{j=1}^{N'_p} u_p^k(j).I_{P_j^e L_1^e V_1^e} \\ \sum_{j=1}^{N'_p} u_p^k(j).I_{P_j^e L_2^e V_1^e} \\ \dots\dots\dots\dots \\ \sum_{j=1}^{N'_p} u_p^k(j).I_{P_j^e L^e_{N'_l} V_1^e} \\ \sum_{j=1}^{N'_p} u_p^k(j).I_{P_j^e L_1^e V_2^e} \\ \dots\dots\dots\dots. \\ \dots\dots\dots\dots. \\ \sum_{j=1}^{N'_p} u_p^k(j).I_{P_j^e L^e_{N'_l} V^e_{N'_\nu}} \end{bmatrix} = \begin{bmatrix} I_{P_k L_1^e V_1^e} \\ I_{P_k L_2^e V_1^e} \\ \dots\dots\dots\dots \\ I_{P_k L^e_{N'_l} V_1^e} \\ I_{P_k L_1^e V_2^e} \\ \dots\dots\dots\dots. \\ \dots\dots\dots\dots. \\ I_{P_k L^e_{N'_l} V^e_{N'_\nu}} \end{bmatrix} \tag{A.3}$$

where $I_{P_k L^e_{k_1} V^e_{k_2}} = \sum_{j=1}^{N'_p} u_p^k(j).I_{P_j^e L^e_{k_1} V^e_{k_2}}$ is the image of kth actual person at k_1th eigenlighting and k_2th eigenviewpoint.

Let

$$D_1 = C_P \times \tilde{\mathscr{A}} \tag{A.4}$$

then substituting C_P from (23)

$$D_1 = [C_P^{(1)} C_P^{(2)} \dots C_P^{(N_p)}]^T \tilde{\mathscr{A}} = [D_1^{(1)} D_1^{(2)} \dots D_1^{(N_p)}]^T = \begin{bmatrix} I_{P_1 L_1^e V_1^e} \\ \dots \\ I_{P_1 L^e_{N'_l} V_1^e} \\ I_{P_1 L_1^e V_2^e} \\ \dots \\ I_{P_1 L^e_{N'_l} V^e_{N'_\nu}} \\ I_{P_2 L_1^e V_1^e} \\ \dots \\ \dots \\ I_{P_{N_p} L^e_{N'_l} V^e_{N'_\nu}} \end{bmatrix} \tag{A.5}$$

Following a similar derivation

$$D_2 = C_L \times D_1 = \begin{bmatrix} I_{P_1 L_1 V_1^e} \\ \dots \\ I_{P_1 L_1 V^e_{N'_\nu}} \\ I_{P_2 L_1 V_1^e} \\ \dots \\ I_{P_{N_p} L_1 V^e_{N'_\nu}} \\ I_{P_1 L_2 V_1^e} \\ \dots \\ \dots \\ I_{P_{N_p} L_{N_l} V^e_{N'_\nu}} \end{bmatrix} \tag{A.6}$$

and finally,

$$D_3 = C_V \times D_2 = \begin{bmatrix} I_{P_1 L_1 V_1} \\ \dots \\ I_{P_{N_p} L_1 V_1} \\ I_{P_1 L_2 V_1} \\ \dots \\ I_{P_{N_p} L_{N_l} V_1} \\ I_{P_1 L_1 V_2} \\ \dots \\ \dots \\ I_{P_{N_p} L_{N_l} V_{N_\nu}} \end{bmatrix} \tag{A.7}$$

The structure of D_3 is such that the image $I_{P_{i_p} L_{i_l} V_{i_\nu}}$ can be found at

$$(N_p \times N_l \times (i_\nu - 1) + N_p \times (i_l - 1) + i_p)\text{th row of } D_3 \tag{A.8}$$

Now,

$$\begin{aligned} D_3 &= C_V \times D_2 \\ &= C_V \times C_L \times D_1 \\ &= C_V \times C_L \times C_P \times \tilde{\mathscr{A}} \\ &= \mathscr{M} \times \tilde{\mathscr{A}} \end{aligned}$$

$$\Rightarrow \mathscr{M} = D_3 \times \tilde{\mathscr{A}}^+ \quad (^+\text{implies pseudoinverse}) \tag{A.9}$$

If m_k is the kth row of $\mathscr{M}$ and if

$$k = N_p \times N_l \times (i_\nu - 1) + N_p \times (i_l - 1) + i_p \tag{A.10}$$

then,

$$\begin{aligned} m_k &= k\text{th row of } D_3 \times \tilde{\mathscr{A}}^+ \\ &= I_{P_{i_p} L_{i_l} V_{i_\nu}} \times \tilde{\mathscr{A}}^+ \end{aligned} \tag{A.11}$$

Solving (A.10) for i_p, i_l and i_ν we obtain, $i_p = ((k-1) \bmod N_p + 1)$, $i_l = ((\lceil k/N_p \rceil - 1) \bmod N_l + 1)$ and $i_\nu = (\lceil k/(N_p \times N_l) \rceil)$. □

References

[1] B.W. Bader, T.G. Kolda, Tensor toolbox version 2.2, Copyright 2007, Sandia National Laboratories, Available at: ⟨http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/⟩.

[2] P. Belhumeur, J. Hespanha, D. Kriegman, Eigenfaces vs. fisherfaces: recognition using class specific linear projection, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7) (July 1997) 711–720.

[3] L. De Lathauwer, B.D. Moor, J. Vandewalle, A multilinear singular value decomposition, SIAM Journal on Matrix Analysis and Applications 21 (4) (2000) 1253–1278.
[4] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, D. Zhao, The cas-peal large-scale Chinese face database and baseline evaluations, IEEE Transactions on Systems, Man and Cybernetics, Part A 38 (1) (2008) 149–161.
[5] A. Georghiades, P. Belhumeur, D. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 643–660.
[6] K.J.S. Gong, Multi-modal tensor face for simultaneous super-resolution and recognition, in: 10th IEEE International Conference on Computer Vision, ICCV 2005, vol. 2, 17–21 October 2005, pp. 1683–1690.
[7] A. Haiping Lu, K.N. Plataniotis, Venetsanopoulos. Multilinear principal component analysis of tensor objects for recognition, in: 18th International Conference on Pattern Recognition, ICPR 2006, vol. 2, 2006, pp. 776–779.
[8] X. He, P. Niyogi, Locality preserving projections, Advances in Neural Information Processing Systems, 16, MIT Press, Cambridge, MA, 2003.
[9] P. Kroonenberg, J.D. Leeuw, Principal component analysis of three-mode data by means of alternating least squares algorithms, Psychometrika 45 (1980) 69–97.
[10] A. Kumar, T. Srikanth, Online personal identification in night using multiple face representations, in: 19th International Conference on Pattern Recognition, ICPR 2008, vols. 1–4, December 2008.
[11] L.D. Lathauwer, B.D. Moor, J. Vandewalle, On the best rank-1 and rank-$(r1, r2, \ldots, rn)$ approximation of higher-order tensors, SIAM Journal on Matrix Analysis and Applications 21 (4) (2000) 1324–1342.
[12] K.-C. Lee, J. Ho, D. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (5) (2005) 684–698.
[13] S. Rana, W. Liu, M. Lazarescu, S. Venkatesh, Efficient tensor based face, in: 19th International Conference on Pattern Recognition, ICPR 2008, December 2008, pp. 1–4.
[14] S. Rana, W. Liu, M. Lazarescu, S. Venkatesh, Recognising faces in unseen modes: a tensor based approach, in: 19th International Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8, June 2008.
[15] M. Turk, A. Pentland, Face recognition using eigenfaces, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1991, Proceedings CVPR '91, June 1991, pp. 3–6.
[16] M. Vasilescu, D. Terzopoulos, Multilinear image analysis for facial recognition, in: Proceedings of the 16th International Conference on Pattern Recognition, vol. 2, 2002, pp. 511–514.
[17] M.A.O. Vasilescu, D. Terzopoulos, Multilinear analysis of image ensembles: tensorfaces, in: ECCV '02: Proceedings of the 7th European Conference on Computer Vision—Part I, Springer, London, UK, 2002, pp. 447–460.
[18] H. Wang, N. Ahuja, Facial expression decomposition, in: Proceedings of the 9th IEEE International Conference on Computer Vision, 2003, vol. 2, 16–16 October 2003, pp. 958–965.

About the Author—SANTU RANA is currently a Ph.D. student at the School of Computing at Curtin University of Technology, Perth, Western Australia. He completed his Bachelor in Electrical Engineering from Jadavpur University, Kolkata, in the year 2003, followed by Masters in System Science & Automation from Indian Institute of Science in the year 2005. His research interests include vision, pattern recognition and machine learning.

About the Author—WANQUAN LIU received the B.Sc, degree in Applied Mathematics from Qufu Normal University, PR China, in 1985, the M.Sc. degree in Control Theory and Operation Research from Chinese Academy of Science in 1988 and the Ph.D. degree in Electrical Engineering from Shanghai Jiaotong University, in 1993. He once held the ARC Fellowship and JSPS Fellowship and attracted research funds from different resources. He is currently a Senior Lecturer in the Department of Computing at Curtin University of Technology. His research interests include face recognition, large scale pattern recognition, signal processing, machine learning and intelligent systems.

About the Author—MIHAI LAZARESCU is currently a Senior Lecturer at Curtin University of Technology and has over 7 years of research and development experience in pattern recognition and machine learning. His research interests are in stream data mining, machine learning, case-based reasoning and vision. He is a Senior Member of IMPCA (Institute for Multi-sensor Processing & Content Analysis), a member of the Institute of Electrical and Electronic Engineers, Australian Pattern Recognition Society as well as the Australian Computer Society. Dr. Lazarescu has been active reviewer for several journals and conferences and has published more than 45 peer reviewed papers.

About the Author—SVETHA VENKATESH is John Curtin Distinguished Professor at the School of Computing at Curtin University of Technology, Perth, Western Australia. Svetha has extensive experience in low-level vision, pattern recognition and multimedia content analysis. Her research is in the areas of large scale pattern recognition, image understanding, multimedia analysis and applications of computer vision to image and video indexing and retrieval. She is the author of about 320 research papers in these areas and is currently the Director of the Institute of Multi-sensor Processing and Content Analysis. She is also the Associate Editor of IEEE Transaction in Multimedia and ACM Transactions on Multimedia Computing, Communications and Applications (2008 onwards). She has been awarded the Fellowship of the International Association of Pattern Recognition and Fellow of Australian Academy of Technological Science and Engineering.