Recognising faces in unseen modes: a tensor based approach

Santu Rana, Wanquan Liu, Mihai Lazarescu and Svetha Venkatesh
Dept. of Computing, Curtin University of Technology
GPO Box U1987, Perth, WA 6845, Australia.
{santu.rana, wanquan, m.lazarescu, svetha}@cs.curtin.edu.au

Abstract

This paper addresses the limitation of current multilinear techniques (multilinear PCA, multilinear ICA) when applied to face recognition for handling faces in unseen illumination and viewpoints. We propose a new recognition method, exploiting the interaction of all the subspaces resulting from multilinear decomposition (for both multilinear PCA and ICA), to produce a new basis called multilinear-eigenmodes. This basis offers the flexibility to handle face images at unseen illumination or viewpoints. Experiments on benchmarked datasets yield superior performance in terms of both accuracy and computational cost.

1. Introduction

In biometric based security applications, facial images are an important modality for identification. In order to be robust, face recognition algorithms must correctly recognize a face irrespective of variations in facial expression, viewpoint or lighting condition. Popular face recognition algorithms such as the Eigenfaces [14] and Fisherface [5], are only suitable for situations when the identity of the person is the only factor being considered, encountering difficulty when there are variations in lighting, expression etc. This is because these linear models are not intrinsically equipped to deal with variations in more than one factor. Attempts have been made to overcome limitations imposed by linear models by introducing non-linearity in the classification stage, but this incurs higher computational costs [12].

Recently, multilinear models [17] have been proposed as an alternative to accommodate variations across multiple factors in a natural way. Facial images are organised as a data tensor, with different factors of variation modelled as different modes of the data tensor. Subsequent application of Higher Order SVD, a generalisation of SVD for higher order matrices, generates subspaces related to every factor of variation. The power of such modelling lies in that it enables us to construct effective representations, depending on the variations observed in each subspace and the importance given to the associated factor [17]. The effectiveness of such a representation results in better face recognition performance than the linear models, as reported by Vasilescu *et.al.* in [16]. However, in their approach, only the person-mode decomposition is used for recognition, whilst other mode decompositions are used optionally to reduce the dimensionality of associated vector-spaces (*e.g.* removing the dimensions with low variance). More precisely, if we want to identify persons when the facial images are only subjected to varying lighting and viewpoints, a set of *eigenmodes* are calculated for each combination of lighting and viewpoint. These *eigenmodes* are similar to *eigenfaces*, however, whilst *eigenfaces* capture variations over all the images, *eigenmodes* capture variations over images at particular combinations of lighting and viewpoint. These eigenmodes constitute the basis of each vector space, and thus there is a separate vector space for each combination of lighting and viewpoint. The notion of multilinearity implies that for training images, each person is defined by the same coefficient vector across all the bases. A test image is projected on every basis and a set of candidate coefficient vectors is generated. The set is then compared pair-wise to the set of stored person-specific combination vectors and the best match is found.

A similar approach has also been used in [18] for expression invariant face recognition, in [9] for simultaneous super-resolution and recognition and in [10] for gait recognition. A similar recognition approach has also been used for Multilinear ICA decomposition [15]. An analysis of these approaches reveals the following shortcomings:

1. Though multilinear decomposition is used, essentially they compute a set of coupled bases, based only on person-mode decomposition. This, we believe, is a severe under-utilisation of the multilinear decomposition, which provides a mechanism to unearth the hidden *multilinear relationship* between all factors of variations (*i.e.* person, lighting and viewpoint)

2. The recognition procedures in [16] [18] need to per-

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

form linear projections at each lighting-viewpoint combination, increasing computational cost with increase in the number of lighting-viewpoint combinations.

3. Testing images at novel lighting or viewpoints has not been investigated well, as the projection bases do not contain any information regarding variations in these factors.

Based on these observations and motivated to fully use the information of *multilinear relations* between the factors, we propose a recognition approach overcoming the shortcomings outlined above. In particular, we base our approach on the core tensor, which represents how the various factors interact with each other in a *multilinear way* to create an image. When the core tensor is multiplied by the pixel space eigenvectors (*i.e. eigenimages*), it transforms the *eigenimages* into *eigenmodes*. Contrary to the previous person-specific *eigenmodes*, these *multilinear eigenmodes* are the result of interaction between *eigen-persons*, *eigen-lightings* and *eigen-viewpoints* and used as the projection basis in the proposed method. Also the proposed basis contains information on the variations of all the factors, thereby rendering itself suitable to handle unseen variation in factors. A linear projection operator is then defined over this basis, giving a joint person-lighting-viewpoint description for a facial image. Further, we provide a mechanism to compute the description vectors for the training set in an efficient way directly from the decomposition result. We store the description vectors for all the training images along with the projection operator. Given a test image, we find the description vector by employing the projection operator and then identify the closest matching training image as the identity of the test image.

Extensive experimentation comparing our proposed method with the existing tensor based methods[16] shows that the proposed method has two distinct advantages. First, our proposed approach performs much better than the existing approaches in terms of correctly identifying persons in the novel scenarios proving that our method, based on the *multilinear relations* between all the factors is more flexible in dealing with unseen variations. Second, in most of the experiments our proposed approach took less time to test an image than the existing approach. This is because, we exploit the fact that in some situations, even if there is a huge number of actual lighting or viewpoint instances, few *eigenlightings* and *eigenviewpoints* can capture most of the variations and thus the method lends itself to an effective computation. We note that current tensor approaches concentrate in the decomposition of the tensor [13], whilst we focus in exploiting the decomposition in a novel way for recognition.

2. Background

In this section we review multilinear PCA and ICA, as well as existing approaches for facial recognition in multilinear frameworks.

2.1. Multilinear PCA

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

$$D = U \Sigma V^T \tag{2.1}$$

where Σ is a diagonal matrix containing ordered *eigenvalues* and U is a *orthonormal* matrix containing ordered *principal direction of variation* (principal components) in its columns. Similarly, N-mode SVD, a generalization of the SVD for higher order matrices [6], orthogonalizes "N" associated vector spaces of an N-way matrix (or N'th order tensor). If $\mathcal{D}$ is a n'th order tensor and $\mathcal{D} \in \mathcal{R}^{I_1 \times I_2 \times \ldots \times I_n}$, application of n-mode SVD orthogonalizes "n" associated vector spaces of $\mathcal{D}$ and decomposes the tensor as,

$$\mathcal{D} = S \times_1 U^1 \times_2 \ldots \times_k U^k \ldots \times_n U^n \tag{2.2}$$

where, $\times_k$ denotes *mode-k* product and the *orthonormal* matrix, U^k contains ordered principal components for the k'th 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 [1]. 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 $\mathcal{D}$ and set the left singular matrix as U^k.

2. Compute core tensor S as,

$$S = \mathcal{D} \times_1 U^{1^T} \times_2 \ldots \times_k U^{k^T} \ldots \times_n U^{n^T} \tag{2.3}$$

2.2. Multilinear ICA

ICA is a generalisation of PCA, in the sense that, while PCA only decorrelates the data, ICA seeks to make the data as independent as possible. ICA can be applied in two different ways [4]. In Architecture I, ICA is applied to D^T to generate a set of *independent components*. Essentially, ICA starts with the PCA decomposition of (2.1) and rotates

[1] A subtensor $S_{i_n=\alpha}$ for $S \in \mathcal{R}^{I_1 \times I_2 \times \ldots \times I_n}$ is obtained by fixing the n'th index to α. Orthogonality implies $\langle S_{i_n=\alpha}, S_{i_n=\beta} \rangle = 0$ for $\alpha \neq \beta$ and ordering implies $||S_{i_1}|| \geq ||S_{i_2}|| \geq \ldots \geq ||S_{i_n}||$

the principal components such that they become statistically *independent* [3]. The rotation is performed as,

$$\begin{aligned} D^T &= V\Sigma U^T \\ &= (V\Sigma W^{-1})(WU^T) \\ &= K^T C^T \end{aligned} \tag{2.4}$$

where W is an invertible transformation computed by the ICA algorithm and $C = UW^T$ contains the set of independent components in its columns. The rotation matrix W can be computed by first setting an objective function that holds the notion of the specific form of statistical *independence* we are seeking and then by optimizing the objective function [1]. Dimensionality reduction is usually performed in the PCA stage. In Architecture II, ICA is applied to D to generate a set of *independent coefficients* for the data. ICA is applied after PCA, to rotate the principal components in such a way that the coefficients become statistically *independent* (the component vectors need not be). Starting from (2.1) the rotation is performed as,

$$\begin{aligned} D &= U\Sigma V^T \\ &= (UW^{-1})(W\Sigma V^T) \\ &= CK \end{aligned} \tag{2.5}$$

where $C = UW^{-1}$ is the basis matrix. Again, W is an invertible matrix that is computed by the ICA algorithm. Similar to the multilinear PCA, multilinear ICA is computed by performing ICA and calculating the basis matrix at each mode [15]. Decomposition by multilinear ICA is represented as,

$$\mathcal{D} = Z \times_1 C^1 \times_2 ... \times_k C^k ... \times_n C^n \tag{2.6}$$

where, Z is called the *core tensor* and C^k is the basis matrix for k'th mode. The algorithm for multilinear ICA decomposition is as follows,

1. For $k = 1, ..., n$, compute matrix U^k and W^k by computing ICA on the *mode-k* flattening of the tensor $\mathcal{D}$ and set,
   (a) $C^k = U^k W^{k^T}$ for multilinear Architecture I
   (b) $C^k = U^k W^{k^{-1}}$ for multilinear Architecture II
2. Compute core tensor S as,

$$S = \mathcal{D} \times_1 C^{1^+} \times_2 ... \times_k C^{k^+} ... \times_n C^{n^+} \tag{2.7}$$

where, C^{k^+} implies *pseudoinverse* of the matrix C^k.

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,

$$T(i_p, i_l, i_v) = I_{P_{i_p}, L_{i_l}, V_{i_v}} \tag{2.8}$$

where, $I_{P_{i_p}, L_{i_l}, V_{i_v}}$ is the image vector of i_p'th person at i_l'th lighting and i_v'th viewpoint. T is a tensor of order 4 and,

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

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. This tensor can be decomposed using either Multilinear PCA, or Multilinear ICA. Multilinear PCA yields four orthogonal subspaces, wherein each subspace corresponds to one mode of variation. This is represented as follows:

$$T = S \times_1 U^P \times_2 U^L \times_3 U^V \times_4 U^X \tag{2.9}$$

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 $N'_p (N'_p \leq N_p)$, $N'_l (N'_l \leq N_l)$ and $N'_v (N'_v \leq N_v)$ dominant eigenvectors (or the principle axes of variation) of the person, lighting and viewpoint subspaces respectively. We refer to these axes of variation as *eigen-person*, *eigen-lighting* and *eigen-viewpoint* respectively. The core tensor, $S \in \mathcal{R}^{N'_p \times N'_l \times N'_v \times N'_x}$, controls the mutual interaction between the person, lighting, viewpoint and pixel subspaces. Multilinear ICA analysis of T provides us a set of four independent components, related to each mode of variation as follows,

$$T = Z \times_1 C^P \times_2 C^L \times_3 C^V \times_4 C^X \tag{2.10}$$

where Z is the core tensor and the columns of C^P, C^L, C^V and C^X define independent components related to the person, lighting, viewpoint and pixels respectively.

2.3.1 Existing recognition approach

In the Multilinear ICA(MICA) framework[15], recognition uses the statistical independence property of ICA to simultaneously recognise person, lighting and viewpoint. However, if we are only interested in identifying person, the recognition processes for MICA and Multilinear PCA(MPCA)[16] are the same. Hence, we elaborate on the recognition approach using Multilinear PCA decomposition, keeping in mind the differences for Multilinear ICA that the core matrix S, and the subspace matrices should be replaced by the core matrix Z, and corresponding ICA basis matrices.

We start with the Multilinear decomposition of the tensor T as defined in (2.9), from which we define , $\mathcal{B}$ as,

$$\mathcal{B} = S \times_2 U^L \times_3 U^V \times_4 U^X \tag{2.11}$$

Therefore, $B \in \mathcal{R}^{N'_p \times N_l \times N_v \times N_x}$. If $\mathcal{B}_{person}$ denotes the unfolding of the tensor $\mathcal{B}$ in the person mode, then

$$\mathcal{B}_{person} = \begin{bmatrix} I_{P^e_1 L_1 V_1} & I_{P^e_1 L_2 V_1} & .. & I_{P^e_1 L_{N_l} V_{N_v}} \\ I_{P^e_2 L_1 V_1} & I_{P^e_2 L_2 V_1} & .. & .. \\ .. & .. & .. & .. \\ .. & .. & .. & .. \\ I_{P^e_{N'_p} L_1 V_1} & .. & .. & I_{P^e_{N'_p} L_{N_l} V_{N_v}} \end{bmatrix} \quad (2.12)$$

where, $I_{P^e_{i_p} L_{i_l} V_{i_v}}$ is the image of i_p'th *eigenperson* for the i_l'th lighting and i_v'th viewpoint. This can be rewritten as:

$$\mathcal{B}_{person} = \begin{bmatrix} B_{(1,1)} & B_{(2,1)} & .... & B_{(N_l,N_v)} \end{bmatrix} \quad (2.13)$$

where,

$$B_{(i_l,i_v)} = \begin{bmatrix} I_{P^e_1 L_{i_l} V_{i_v}} \\ I_{P^e_2 L_{i_l} V_{i_v}} \\ .... \\ .... \\ I_{P^e_{N'_p} L_{i_l} V_{i_v}} \end{bmatrix} \quad (2.14)$$

$B_{(i_l,i_v)}$ is used as the projection basis for the (i_l, i_v)'th combination. This results in $N_l \times N_v$ number of distinct projection bases. It is also evident from (2.11) and (2.9) that,

$$\begin{aligned} T &= \mathcal{B} \times_1 U^P \\ or,\quad T_{(i_l,i_v)} &= U^P \times \mathcal{B}_{(i_l,i_v)} \\ or,\quad I_{P_k, L_{i_l}, V_{i_v}} &= c_k \times \mathcal{B}_{(i_l,i_v)} \end{aligned} \quad (2.15)$$

where, c_k is the k'th row of the matrix U^P and it is specific to the k'th person. The recognition algorithm is based on (2.15). A test image is projected on the basis of $B_{(i_l,i_v)}$, for all i_l and i_v to generate a set of candidate coefficient vectors $\{c_{i_l,i_v}\}$. The best matching c_{p_m} (*i.e.* that minimizes $||c_p - c_{i_l,i_v}||$, for all i_l, i_v and p) identifies the test image as that of the person p_m.

3. Proposed Recognition Approach

Motivated by our desire to exploit the *multilinear relations* amongst the factors, we present a new recognition approach using the information in the core tensor. We derive the recognition scheme for the Multilinear PCA decomposition. For recognition using Multilinear ICA, the core matrix, S and the subspace matrices should be replaced by the core matrix, Z and corresponding component matrices. Starting from (2.9) we define $\mathcal{A}$ as,

$$\mathcal{A} = S \times_4 U^X \quad (3.1)$$

Let $\mathcal{A}_{person}$ denote the unfolding of the tensor $\mathcal{A}$ in the person mode, then

$$\mathcal{A}_{person} = \begin{bmatrix} I_{P^e_1 L^e_1 V^e_1} & I_{P^e_1 L^e_2 V^e_1} & .. & I_{P^e_1 L^e_{N'_l} V^e_{N'_v}} \\ I_{P^e_2 L^e_1 V^e_1} & I_{P^e_2 L^e_2 V^e_1} & .. & .. \\ .. & .. & .. & .. \\ .. & .. & .. & .. \\ I_{P^e_{N'_p} L^e_1 V^e_1} & .. & .. & I_{P^e_{N'_p} L^e_{N'_l} V^e_{N'_v}} \end{bmatrix} \quad (3.2)$$

where, $I_{P^e_{i_p} L^e_{i_l} V^e_{i_v}}$ is the image of i_p'th *eigenperson* for i_l'th *eigen-lighting and* i_v'th *eigen-viewpoint*. Further, $\mathcal{A}_{person}$ can be rewritten as,

$$\mathcal{A}_{person} = \begin{bmatrix} A_{(1,1)} & A_{(2,1)} & .... & A_{(N'_l,N'_v)} \end{bmatrix} \quad (3.3)$$

where,

$$A_{(i_l,i_v)} = \begin{bmatrix} I_{P^e_1 L^e_{i_l} V^e_{i_v}} \\ I_{P^e_2 L^e_{i_l} V^e_{i_v}} \\ .... \\ .... \\ I_{P^e_{N'_p} L^e_{i_l} V^e_{i_v}} \end{bmatrix} \quad (3.4)$$

We define a multilinear eigen-space as,

$$\tilde{\mathcal{A}} = \begin{bmatrix} A_{(1,1)} \\ A_{(2,1)} \\ .... \\ .... \\ A_{(N'_l,N'_v)} \end{bmatrix} \quad (3.5)$$

$\tilde{\mathcal{A}}$ constitutes a vector space spanned by the rows of $\tilde{\mathcal{A}}$. The $N'_p \times N'_l \times N'_v$ rows combine elements of *eigen-person*, *eigen-lighting* and *eigen-viewpoint* and form the basis of this vector space. We call this basis *multilinear eigenmodes* and use it to derive a description for a facial image. The proposed basis has two distinct advantages. First, compared to the conventional tensor based method, the proposed basis involves *eigenmodes* across person as well as all the factors of variation, thereby truly exploiting the *multilinear relations* obtained from the multilinear decompositions. Second, like PCA and unlike the conventional tensor based method, the proposed method defines a unified basis for projection. However, our proposed basis is physically more interpretable than that of the PCA, in case of multi-factor variation in the dataset.

The projection matrix, P for the basis $\tilde{\mathcal{A}}$ is,

$$P = \tilde{\mathcal{A}}^+ \quad (3.6)$$

where, $\tilde{\mathcal{A}}^+$ is the *Moore-Penrose pseudoinverse* of $\tilde{\mathcal{A}}$. Next we will show that the coefficient vector of training images for the projection matrix in (3.6) can be directly calculated from the matrices U^P, U^L and U^V in an efficient way.

Definition 1. *Let us define $\mathcal{UP}$ as,*

$$\mathcal{UP} = \begin{bmatrix} U^P & 0 & 0 & 0 \\ 0 & U^P & 0 & 0 \\ .... & .... & .... & .... \\ .... & .... & .... & .... \\ 0 & 0 & 0 & U^P \end{bmatrix} \tag{3.7}$$

here, U^P is repeated diagonally for $N_l \times N_v$ times.

Let us define $\mathcal{UL}$ as,

$$\mathcal{UL} = \begin{bmatrix} UL & 0 & 0 & 0 \\ 0 & UL & 0 & 0 \\ .... & .... & .... & .... \\ .... & .... & .... & .... \\ 0 & 0 & 0 & UL \end{bmatrix} \tag{3.8}$$

here, UL is repeated diagonally for N_v times and it's defined as,

$$UL = \begin{bmatrix} UL(1,1) \\ UL(2,1) \\ .... \\ UL(N'_p, 2) \\ UL(1,2) \\ .... \\ .... \\ UL(N'_p, N_l) \end{bmatrix} \tag{3.9}$$

where $UL(i,j)$ is a row vector of size $N'_p \times N'_l$ defined as,

$$UL(i,j)_{(i+N'_p\times(k-1))} = \begin{cases} U^L(j,k) & \textit{for } k = 1,...,N'_l \\ 0 & \textit{otherwise} \end{cases} \tag{3.10}$$

Let us define $\mathcal{UV}$ as,

$$\mathcal{UV} = \begin{bmatrix} UV(1,1) \\ UV(2,1) \\ .... \\ UV(N'_p \times N'_l, 1) \\ UV(1,2) \\ .... \\ .... \\ UV(N'_p \times N'_l, N_v) \end{bmatrix} \tag{3.11}$$

where $UV(i,j)$ is a row vector of size $N'_p \times N'_l \times N'_v$ defined as,

$$UV(i,j)_{(i+N'_p\times N'_l(k-1))} = \begin{cases} U^V(j,k) & \textit{for } k = 1,..,N'_v \\ 0 & \textit{otherwise} \end{cases} \tag{3.12}$$

Theorem 1. *Let $\mathcal{M} = \mathcal{UP} \times \mathcal{UL} \times \mathcal{UV}$. If m_k is the k'th row of the matrix $\mathcal{M}$ then,*

$$m_k = I_{P_{i_p} L_{i_l} V_{i_v}} \times P \tag{3.13}$$

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

Proof. See Appendix A for the proof. □

It follows from the above theorem that matrix $\mathcal{M} = \mathcal{UP} \times \mathcal{UL} \times \mathcal{UV}$ contains the coefficients of projection of all the training images for the projection matrix P and thus provides an efficient way to compute the coefficient set. Further it also provides insights on how the decomposition, projection matrix and the coefficients are related to each other. Each row m_k of the matrix $\mathcal{M}$ refers to a training image, whose person, lighting and viewpoint indices are provided by the above theorem. For testing we need to store the projection matrix, P and the coefficient matrix, $\mathcal{M}$. The algorithm for testing is given in the Algorithm 1.

Algorithm 1 Testing algorithm for the Proposed method

1. Given the test image I_T, find the corresponding description vector m_T as,

$$m_T = I_T \times P$$

2. Use a Nearest Neighbour classifier to find the best matching description vector m_b *i.e.* that minimizes,

$$\min_k ||m_T - m_k|| \text{ for } k = 1, ..., (N_p \times N_l \times N_v)$$

   where, m_k is the k'th row of the matrix $\mathcal{M}$. The distance measure we use is the *cosine distance*. For two vectors **a** and **b** the *cosine distance* between them is defined as,

$$cosine\_dist(\mathbf{a}, \mathbf{b}) = \frac{\langle \mathbf{a}, \mathbf{b} \rangle}{||\mathbf{a}||\,||\mathbf{b}||}$$

3. The person identity, p_T for the test image is the person identity of the best matching vector m_b and it is given by,

$$p_T = ((m_b - 1) \bmod N_p + 1)$$

As our method uses joint person-factor space description for recognition, we will refer to it as MPCA-JS or MICA-JS, depending on the specific multilinear decomposition used.

4. Experiments, Analysis and Evaluation

We used the Extended YaleB and PEAL databases in our experiments. The Extended YaleB database contains images of 38 persons at 64 different illumination conditions and at 9 different viewpoints for each illumination condition [8] [11]. The PEAL database contains images of Chinese nationals at different pose, expression and illumination[7]. For our experiments we have only chosen frontal images of 20 persons at 20 different illuminations. 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 Kolda *et. al.* in MATLAB™ [2]. FastICA© package was used for ICA and Multilinear ICA computation. For the PEAL dataset, four set of experiments were performed with 5, 7, 9 & 11 lighting conditions as training while the rest were used for testing. For experiments on the Extended YaleB database 30 lighting conditions at 5 viewpoints are used for training and the rest for testing. Thus in the PEAL database, the test images were at unseen lightings and in the Extended YaleB database, the test images were at unseen lighting and viewpoints. Each set of experiments was repeated 10 times on 10 random partitions of the database and average of the results are reported. The performance of our proposed recognition procedure is compared with the PCA[14], ICA[4], conventional MPCA[16] and conventional MICA[15] methods of face recognition.

Traditionally, when PCA is used for recognition, the last few eigenvectors are removed to improve performance. Here we used *energy thresholding* to retain top-*k* eigenvectors such that,

$$\min_k \frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{n} \lambda_j} \geq thresh \qquad (4.1)$$

where, λ_j is the eigenvalue corresponding to the j'th eigenvector and $\lambda_{j-1} \geq \lambda_j$, n is the total number of eigenvectors and $thresh$ is the user specified threshold. In the case of Multilinear PCA, the same *energy thresholding* was used to select the "k" in each mode. In our experiments we set 0.96 as the threshold for the PCA based method, 0.96 as the threshold for Multilinear PCA in pixel mode and 1.0 in other modes. In the case of ICA and Multilinear ICA, we maintained the same dimensionality as those of respective PCA and Multilinear PCA and basis matrices were computed using ICA-Architecture II. While PCA and ICA based recognition methods used the Euclidian distance for comparison, multilinear based methods used Cosine distance, which was used in previous research papers [16]. Table 1 and 2 show the results of experiments on PEAL and Extended YaleB databases respectively.

Recognition method	***Accuracy (%)***			
	5 train	**7 train**	**9 train**	**11 train**
PCA	69.47	72.96	75.59	74.44
ICA	74.63	77.92	80.19	79.39
MPCA	73.23	78.42	81.14	77.39
MPCA-JS	**86.20**	**90.58**	**93.50**	**93.67**
MICA	74.26	80.46	81.23	79.50
MICA-JS	**84.90**	**88.23**	**92.55**	**92.33**

Table 1. Experimental results on PEAL lighting variation dataset.

As the Table 1 shows, our proposed recognition approach improves the recognition performance significantly over the conventional multilinear based approaches. While the conventional MPCA is slightly better than PCA, our method, MPCA-JS provides much better performance. This is also the case with ICA, where conventional MICA outperforms ICA and our method, MICA-JS, outperforms conventional MICA. The best performance on this database is given by MPCA-JS.

Recognition method	PCA	MPCA	MPCA-JS
Accuracy (%)	72.45	38.70	**85.45**
Recognition method	ICA	MICA	MICA-JS
Accuracy (%)	73.15	38.85	**85.45**

Table 2. Experimental results on Extended YaleB dataset.

Table 2 provides a set of results that is worth noting. Specifically, MPCA and MICA actually fared significantly worse compared with PCA and ICA, which is surprising. However, the conventional multilinear methods only use *person*-mode decomposition for recognition and hence they are not able to truly realize the power of multilinear analysis. Our proposed method, which meaningfully uses decomposition in all the modes, shows a much improved performance when compared with the linear methods.

Recognition method	***Test time for 100 test cases (sec.)***			
	5 train	**7 train**	**9 train**	**11 train**
PCA	0.14	0.17	0.22	0.26
MPCA	0.92	1.50	2.11	2.65
MPCA-JS	**0.24**	**0.39**	**0.50**	**0.59**

Table 3. Test time for 100 test cases on PEAL lighting variation dataset.

Table 3 shows the time taken in testing by PCA, MPCA and MPCA-JS respectively on the PEAL lighting variation dataset. All the timings are for Matlab code running on a Pentium dual core 1.86GHz system, having 2GB RAM. It shows that MPCA-JS is more efficient for testing than the existing multilinear recognition method.

5. Conclusions and Future Work

In this paper we exploited the interaction of subspaces, resulting from multilinear analysis, to propose a novel face recognition method capable of handling unseen variations in images. Experimental results on both PEAL and Extended YaleB show the superiority of our proposed recognition method over the conventional multilinear recognition methods, whilst in some situations conventional multilinear methods performed worse than the linear methods, our proposed method consistently outperformed linear methods. Possible future works include understanding of the proposed *multilinear eigenmode space* and exploiting any structure therein to further improve the recognition performance.

References

[1] J. K. A. Hyvrinen and E. Oja. *Independent Component Analysis*. John Wiley & Sons., 2001.

[2] B. W. Bader and T. G. Kolda. Tensor toolbox version 2.2, Copyright 2007, Sandia National Laboratories, Available at *http://csmr.ca.sandia.gov/∼tgkolda/TensorToolbox/*.

[3] M. S. Bartlett. *Face Image Analysis by Unsupervised Learning*. Kluwer Academic Publishers, Norwell, MA, USA, 2001.

[4] M. J. Bartlett M.S. and S. T.J. Face recognition by independent component analysis. *Neural Networks, IEEE Transactions on*, 13(6):1450–1464, Nov 2002.

[5] P. N. Belhumeur, P. H. Jo and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(7):711–720, 1997.

[6] L. De Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[7] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao. The cas-peal large-scale chinese face database and baseline evaluations. *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, 38(1):149–161, Jan. 2008.

[8] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: illumination cone models for face recognition under variable lighting and pose. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(6):643–660, Jun 2001.

[9] K. J. S. Gong. Multi-modal tensor face for simultaneous super-resolution and recognition. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2:1683–1690 Vol. 2, 17-21 Oct. 2005.

[10] A. Haiping Lu; Plataniotis, K.N.; Venetsanopoulos. Multilinear principal component analysis of tensor objects for recognition. *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, 2:776–779, 0-0 0.

[11] K.-C. Lee, J. Ho, and D. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(5):684–698, May 2005.

[12] Q. Liu, R. Huang, H. Lu, and S. Ma. Kernel-based nonlinear discriminant analysis for face recognition. *J. Comput. Sci. Technol.*, 18(6):788–795, 2003.

[13] S. W. Park and M. Savvides. Estimating mixing factors simultaneously in multilinear tensor decomposition for robust face recognition and synthesis. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, page 49, Washington, DC, USA, 2006. IEEE Computer Society.

[14] M. Turk and A. Pentland. Face recognition using eigenfaces. *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 586–591, 3-6 Jun 1991.

[15] D. Vasilescu, M.A.O.; Terzopoulos. Multilinear independent components analysis. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 1:547–553 vol. 1, 20-25 June 2005.

[16] M. Vasilescu and D. Terzopoulos. Multilinear image analysis for facial recognition. *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, 2:511–514 vol.2, 2002.

[17] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I*, pages 447–460, London, UK, 2002. Springer-Verlag.

[18] H. Wang and N. Ahuja. Facial expression decomposition. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 958–965 vol.2, 16-16 Oct. 2003.

A. Proof of the Theorem 1

Proof. Let us refer to $\tilde{\mathcal{A}}$ in (3.5), which is,

$$\tilde{\mathcal{A}} = \begin{bmatrix} A_{(1,1)} \\ A_{(2,1)} \\ .... \\ .... \\ A_{(N'_l,N'_v)} \end{bmatrix} \tag{A.1}$$

From the definition of $A_{(i_l,i_v)}$ (3.4) we can observe that $\tilde{\mathcal{A}}$ contains images of interaction between N'_p *eigen-persons,* N'_l *eigen-lightings* and N'_v *eigen-viewpoints*. The images in $\tilde{\mathcal{A}}$ are organized in such a way that, the index for the *eigen-persons* varies the fastest followed by the index of the *eigen-lightings* and the index for the *eigen-viewpoints* varies slowest. This structure implies that the image of interaction between the i'_p'th *eigen-person*, i'_l'th *eigen-lighting* and i'_v'th *eigen-viewpoint*, $I_{P^e_{i'_p} L^e_{i'_l} V^e_{i'_v}}$, can be found at the

$$(N'_p \times N'_l \times (i'_v - 1) + N'_l \times (i'_l - 1) + i'_p)\text{'th row of } \tilde{\mathcal{A}}$$

We know that the image due to the interaction between the i'_p'th *eigen-person*, i'_l'th *eigen-lighting* and i_v'th actual viewpoint, $I_{P^e_{i'_p} L^e_{i'_l} V_{i_v}}$, can be calculated as,

$$\begin{aligned} I_{P^e_{i'_p} L^e_{i'_l} V_{i_v}} &= \sum_{i'_v=1}^{N'_v} U^V(i_v, i'_v) \times I_{P^e_{i'_p} L^e_{i'_l} V^e_{i'_v}} \\ &= \sum_{i'_v=1}^{N'_v} \{U^V(i_v, i'_v) \times (N'_p \times N'_l \times (i'_v - 1) \\ &\qquad +N'_l \times (i'_l - 1) + i'_p)\text{'th row of } \tilde{\mathcal{A}}\} \\ &= [i_v\text{'th row of } U^V] \times Y \times \tilde{\mathcal{A}} \end{aligned} \tag{A.2}$$

where, Y is a *selection* matrix of size $N'_v \times (N'_p \times N'_l \times N'_v)$ whose i'_v'th row is defined as,

$$Y_{i'_v,k} = \begin{cases} 1 & \text{if } k = \{(N'_p \times N'_l \times (i'_v - 1) + \\ & \qquad N'_l \times (i'_l - 1) + i'_p)\} \\ 0 & \text{otherwise} \end{cases} \tag{A.3}$$

Clearly, the first row of Y has entry 1 at the $(N'_l \times (i'_l - 1) + i'_p)$'th position and the rest are zero, the second row has entry 1 at $(N'_p \times N'_l + N'_l \times (i'_l - 1) + i'_p)$'th position and the rest are zero, the third row has an entry 1 at $(N'_p \times N'_l \times 2 + N'_l \times (i'_l - 1) + i'_p)$'th position and the rest are zero and so on. Hence if,

$$Z = [i_v\text{'th row of } U^V] \times Y \tag{A.4}$$

then,

$$Z_{(i+N'_p \times N'_l(k-1))} = \begin{cases} U^V(i_v, k) & \text{for } k = 1, .., N'_v \\ 0 & \text{otherwise} \end{cases} \tag{A.5}$$

where $i = N'_l \times (i'_l - 1) + i'_p$ and Z is same as the definition of $UV(i, i_v)$ in (3.12). Therefore, from (A.2) and (A.4),

$$\begin{aligned} I_{P^e_{i'_p} L^e_{i'_l} V_{i_v}} &= Z \times \tilde{\mathcal{A}} \\ &= UV(N'_p \times N'_l(k-1), i_v) \times \tilde{\mathcal{A}} \end{aligned} \tag{A.6}$$

Let us denote,

$$\mathcal{B} = \mathcal{UV} \times \tilde{\mathcal{A}} \tag{A.7}$$

where $\mathcal{UV}$ is as defined in (3.11). From (3.11) and (A.6) it is easy to see that $\mathcal{UV}$ transforms the images of $\tilde{\mathcal{A}}$ into the images, which are due to the interaction of N'_p *eigen-persons* and N'_l *eigen-lightings* at N_v actual viewpoints. The organization of the images in $\mathcal{B}$ is similar to $\tilde{\mathcal{A}}$. Now we can formulate a similar argument to prove that pre-multiplication of $\mathcal{B}$ by $\mathcal{UL}$ generates images of N'_p *eigen-persons* at N_l actual lightings and N_v actual viewpoints. Let $\mathcal{C} = \mathcal{UL} \times \mathcal{B}$ then consequently pre-multiplication of $\mathcal{C}$ by $\mathcal{UP}$ generates images of N_p actual persons at N_l actual lightings and N_v actual viewpoints. Let

$$\begin{aligned} \mathcal{D} &= \mathcal{UP} \times \mathcal{C} \\ &= \mathcal{UP} \times \mathcal{UL} \times \mathcal{B} \\ &= \mathcal{UP} \times \mathcal{UL} \times \mathcal{UV} \times \tilde{\mathcal{A}} \end{aligned} \tag{A.8}$$

The structure of $\mathcal{D}$ is again similar to $\tilde{\mathcal{A}}$ and the image of i_p'th person at i_l'th lighting and at i_v'th viewpoint can be found at,

$$(N_p \times N_l \times (i_v - 1) + N_l \times (i_l - 1) + i_p)\text{'th row of } \mathcal{D} \tag{A.9}$$

Let us define $\mathcal{M} = \mathcal{UP} \times \mathcal{UL} \times \mathcal{UV}$, then from (A.8) we obtain,

$$\begin{aligned} D &= \mathcal{M} \times \tilde{\mathcal{A}} \\ \Longrightarrow \quad M &= \mathcal{D} \times \tilde{\mathcal{A}}^{+} \\ &= \mathcal{D} \times P \end{aligned} \tag{A.10}$$

where $P = \tilde{\mathcal{A}}^{+}$. From (A.9) and from (A.10) we observe that $(N_p \times N_l \times (i_v - 1) + N_l \times (i_l - 1) + i_p)$'th row of $\mathcal{M}$ contains the coefficient of projection of the i_p'th person at i_l'th lighting and at i_v'th viewpoint ($I_{P_{i_p} L_{i_l} V_{i_v}}$), for the projection matrix P. That implies that if m_k is the k'th row of the matrix $\mathcal{M}$ then,

$$m_k = I_{P_{i_p} L_{i_l} V_{i_v}} \times P \tag{A.11}$$

where,

$$k = (N_p \times N_l \times (i_v - 1) + N_l \times (i_l - 1) + i_p) \tag{A.12}$$

Solving (A.12) for i_p, i_l and i_v we obtain, $i_p = ((k - 1) \bmod N_p + 1)$, $i_l = ((\lceil \frac{k}{N_p} \rceil - 1) \bmod N_l + 1)$ and $i_v = ((\lceil \frac{k}{N_p \times N_l} \rceil - 1) \bmod N_v + 1)$. □