A Convergent Solution to Tensor Subspace Learning

Huan Wang[1]
[1] IE, Chinese University of Hong Kong, Hong Kong
hwang5@ie.cuhk.edu.hk

Shuicheng Yan[2]**,Thomas Huang**[2]
[2] ECE, University of Illinois at Urbana Champaign, USA
{scyan,huang}@ifp.uiuc.edu

Xiaoou Tang [1,3]
[3] Microsoft Research Asia
Beijing, China
xitang@microsoft.com

Abstract

Recently, substantial efforts have been devoted to the subspace learning techniques based on tensor representation, such as 2DLDA [Ye *et al.*, 2004], DATER [Yan *et al.*, 2005] and Tensor Subspace Analysis (TSA) [He *et al.*, 2005]. In this context, a vital yet unsolved problem is that the computational convergency of these iterative algorithms is not guaranteed. In this work, we present a novel solution procedure for general tensor-based subspace learning, followed by a detailed convergency proof of the solution projection matrices and the objective function value. Extensive experiments on real-world databases verify the high convergence speed of the proposed procedure, as well as its superiority in classification capability over traditional solution procedures.

1 Introduction

Subspace learning algorithms [Brand, 2003] such as Principal Component Analysis (PCA) [Turk and Pentland, 1991] and Linear Discriminant Analysis (LDA) [Belhumeur *et al.*, 1997] traditionally express the input data as vectors and often in a high-dimensional feature space. In real applications, the extracted features are usually in the form of a multidimensional union, *i.e.* a tensor, and the vectorization process destroys this intrinsic structure of the original tensor form. Another drawback brought by the vectorization process is the *curse of dimensionality* which may greatly degrade the algorithmic learnability especially in the small sample size cases.

Recently substantial efforts have been devoted to the employment of tensor representation for improving algorithmic learnability [Vasilescu and Terzopoulos, 2003]. Among them, 2DLDA [Ye *et al.*, 2004] and DATER [Yan *et al.*, 2005] are tensorized from the popular vector-based LDA algorithm. Although the initial objectives of these algorithms are different, they all end up with solving a higher-order optimization problem, and commonly iterative procedures were used to search for the solution. A collective problem encountered by their solution procedures is that the iterative procedures are not guaranteed to converge, since in each iteration, the optimization problem is approximately simplified from the *Trace Ratio* form $\arg\max_{U^k} Tr({U^k}^T S_k^p U^k)/Tr({U^k}^T S^k U^k)$ [1] to the *Ratio Trace* form $\arg\max_{U^k} Tr[\ ({U^k}^T S^k U^k)^{-1}({U^k}^T S_k^p U^k)\ ]$ in order to obtain a closed-form solution for each iteration. Consequently, the derived projection matrices are unnecessary to converge, which greatly limits the application of these algorithms since it is unclear how to select the iteration number and the solution is not optimal even in the local sense.

In this work, by following the graph embedding formulation for general dimensionality reduction proposed by [Yan *et al.*, 2007], we present a new solution procedure for subspace learning based on tensor representation. In each iteration, instead of transforming the objective function into the ratio trace form, we transform the trace ratio optimization problem into a *trace difference* optimization problem $\max_{U^k} Tr[{U^k}^T(S_k^p - \lambda S^k)U^k]$ where λ is the objective function value computed from the solution $(U^k|_{k=1}^n)$ of the previous iteration. Then, each iteration is efficiently solved with the eigenvalue decomposition method [Fukunaga, 1991]. A detailed proof is presented to justify that λ, namely the value of the objective function, will increase monotonously, and also we prove that the projection matrix U^k will converge to a fixed point based on the *point-to-set map* theories [Hogan, 1973].

It is worthwhile to highlight some aspects of our solution procedure to general subspace learning based on tensor representation here:

1. The value of the objective function is guaranteed to monotonously increase; and the multiple projection matrices are proved to converge. These two properties ensure the algorithmic effectiveness and applicability.
2. Only eigenvalue decomposition method is applied for iterative optimization, which makes the algorithm extremely efficient; and the whole algorithm does not suffer from the singularity problem that is often encountered by the traditional generalized eigenvalue decomposition method used to solve the ratio trace optimization problem.
3. The consequent advantage brought by the sound theoretical foundation is the enhanced potential classification

[1] Matrices S_k^p and S^k are both positive semidefinite and more detailed definitions are described afterward.

capability of the derived low-dimensional representation from the subspace learning algorithms.

The rest of this paper is organized as follows. Section II reviews the general subspace learning based on tensor representation, and then we introduce our new solution procedure along with the theoretical convergency proof in section III. By taking the Marginal Fisher Analysis (MFA) algorithm proposed in [Yan *et al.*, 2007] as an example, we verify the convergency properties of the new proposed solution procedure and the classification capability of the derived low-dimensional representation is examined with a set of experiments on the real-world databases in Section IV.

2 Subspace Learning with Tensor Data

In this section, we present a general subspace learning framework by encoding data as tensors of arbitrary order, extended from the one proposed by [Yan *et al.*, 2007] and taking the data inputs as vectors. The concepts of tensor inner production, mode-k production with matrix, and mode-k unfolding are referred to the work of [Yan *et al.*, 2005].

2.1 Graph Embedding with Tensor Representation

Denote the sample set as $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N], \mathbf{X}_i \in R^{m_1 \times m_2 \times \ldots \times m_n}, i = 1, \ldots N$, with N as the total number of samples. Let $G = \{X, S\}$ be an undirected similarity graph, called an *intrinsic graph*, with vertex set $\mathbf{X}$ and similarity matrix $S \in R^{N \times N}$. The corresponding diagonal matrix D and the Laplacian matrix L of the graph G are defined as

$$L = D - S, \quad D_{ii} = \sum_{j \neq i} S_{ij} \ \forall \ i. \tag{1}$$

The task of graph embedding is to determine a low-dimensional representation of the vertex set $\mathbf{X}$ that preserves the similarities between pairs of data in the original high-dimensional feature space. Denote the low-dimensional embedding of the vertices as $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_N]$, where $\mathbf{Y}_i \in R^{m'_1 \times m'_2 \times \ldots \times m'_n}$ is the embedding for the vertex $\mathbf{X}_i$, with the assumption that $\mathbf{Y}_i$ is the mode-k production of $\mathbf{X}_i$ with a series of column orthogonal matrices $U^k \in R^{m_k \times m'_k}$,

$$\mathbf{Y}_i = \mathbf{X}_i \times_1 U^1 \times_2 U^2 \ldots \times_n U^n, \quad {U^k}^T U^k = I_{m'_k}, \tag{2}$$

where $I_{m'_k}$ is an m'_k-by-m'_k identity matrix. To maintain similarities among vertex pairs according to the graph preserving criterion [Yan *et al.*, 2007], we have

$$(U^k)^*|_{k=1}^n = \arg\min_{U^k|_{k=1}^n} \frac{\sum_{i \neq j} \parallel \mathbf{Y}_i - \mathbf{Y}_j \parallel^2 S_{ij}}{f(U^k|_{k=1}^n)} \tag{3}$$

$$= \arg\min_{U^k|_{k=1}^n} \frac{\sum_{i \neq j} \parallel (\mathbf{X}_i - \mathbf{X}_j) \times_k U^k|_{k=1}^n \parallel^2 S_{ij}}{f(U^k|_{k=1}^n)}, \tag{4}$$

where $f(U^k|_{k=1}^n)$ is a function that poses extra constraint for the graph similarity preserving criterion. Here $U^k|_{k=1}^n$ means the sequence U^1, U^2 to U^n and so for the other similar representations in the following parts of this work. Commonly, $f(U_k|_{k=1}^n)$ may have two kinds of definitions. One is for scale normalization, that is,

$$f(U_k|_{k=1}^n) = \sum_{i=1}^{N} \|\mathbf{X}_i \times_k U^k|_{k=1}^n\|^2 B_{ii}, \tag{5}$$

where B is a diagonal matrix with non-negative elements. The other is a more general constraint which relies on a new graph, referred to as *penalty graph* with similarity matrix S^p, and is defined as

$$f(U^k|_{k=1}^n) = \sum_{i \neq j} \|(\mathbf{X}_i - \mathbf{X}_j) \times_k U^k|_{k=1}^n\|^2 S^p_{ij}. \tag{6}$$

Without losing generality, we assume that the constraint function is defined with penalty matrix for simplicity; and for scale normalization constraint, we can easily have the similar deduction for our new solution procedure. Then, the general formulation of the tensor-based subspace learning is expressed as

$$\arg\max_{U^k|_{k=1}^n} \frac{\sum_{i \neq j} \parallel (\mathbf{X}_i - \mathbf{X}_j) \times_k U^k|_{k=1}^n \parallel^2 S^p_{ij}}{\sum_{i \neq j} \parallel (\mathbf{X}_i - \mathbf{X}_j) \times_k U^k|_{k=1}^n \parallel^2 S_{ij}}. \tag{7}$$

Recent studies [Shashua and Levin, 2001] [Ye, 2005] [Ye *et al.*, 2004] [Yan *et al.*, 2005] have shown that dimensionality reduction algorithms with data encoded as high-order tensors usually outperform those with data represented as vectors, especially when the number of training samples is small. Representing images as 2D matrices instead of vectors allows correlations between both rows and columns to be exploited for subspace learning.

Generally, no closed-form solution exists for (7). Previous works [Ye *et al.*, 2004] [Yan *et al.*, 2005] utilized iterative procedures to search for approximate solutions. First, the projection matrices $U^1, \ldots, U^n$ are initialized arbitrarily; then each projection matrix U^k is refined by fixing the other projection matrices $U^1, \ldots, U^{k-1}, U^{k+1}, \ldots, U^n$ and solving the optimization problem:

$$U^{k^*} = \arg\max_{U^k} \frac{\sum_{i \neq j} \|{U^k}^T Y^k_i - {U^k}^T Y^k_j\|^2 S^p_{ij}}{\sum_{i \neq j} \|{U^k}^T Y^k_i - {U^k}^T Y^k_j\|^2 S_{ij}} \tag{8}$$

$$= \arg\max_{U^k} \frac{Tr({U^k}^T S^p_k U^k)}{Tr({U^k}^T S^k U^k)} \tag{9}$$

where Y^k_i is the mode-k unfolding matrix of the tensor $\tilde{\mathbf{Y}}_i = \mathbf{X}_i \times_1 U^1 \ldots \times_{k-1} U^{k-1} \times_{k+1} U^{k+1} \ldots \times_n U^n$ and $S^k = \sum_{i \neq j} S_{ij}(Y^k_i - Y^k_j)(Y^k_i - Y^k_j)^T, S^p_k = \sum_{i \neq j} S^p_{ij}(Y^k_i - Y^k_j)(Y^k_i - Y^k_j)^T$.

The optimization problem in (9) is still intractable, and traditionally its solution is approximated by transforming the objective function in (9) into a more tractable approximate form, namely, Ratio Trace form,

$$U^{k^*} = \arg\max_{U^k} Tr((({U^k}^T S^k U^k)^{-1}({U^k}^T S^p_k U^k)) \tag{10}$$

which can be directly solved with the generalized eigenvalue decomposition method. However, this distortion of the objective function leads to the computational issues as detailed in the following subsection.

2.2 Computational Issues

As the objective function in each iteration is changed from the trace ratio form (9) to the ratio trace form (10), the deduced solution can satisfy neither of the two aspects: 1) the objective function value in (7) can monotonously increase; and 2) the solution $(U^1, U^2, \ldots, U^n)$ can converge to a fixed point. In this work, we present a convergent solution procedure to the optimization problem defined in (7).

3 Solution Procedure and Convergency Proof

In this section, we first introduce our new solution procedure to the tensor-based subspace learning problems, and then give the convergency proof to the two aspects mentioned above.

As described above, there does not exist closed-form solution for the optimization problem (7), and we solve the optimization problem also in an iterative manner. For each iteration, we refine one projection matrix by fixing the others and an efficient method is proposed for this refinement. Instead of solving a ratio trace optimization problem (10) for an approximate solution, we transform the trace ratio optimization problem (9) into a *trace difference* optimization problem defined as

$$U^{k^*} = \arg\max_{U^k} Tr({U^k}^T(S_k^p - \lambda S^k)U^k), \qquad (11)$$

where λ is the value of objective function (7) computed from the projection matrices of the previous iteration.

Though the iterative procedure may converge to a *local* optimum for the optimization problem (7), it can monotonously increase the objective function value as proved later, which directly leads to its superiority over the ratio trace based optimization procedure, since the step-wise solution of the latter is unnecessarily optimal for (9).

We iteratively refine the projection matrices, and the detailed solution procedure to solve the tensor-based general subspace learning problem is listed in Algorithm 1.

3.1 Analysis of Monotonous Increase Property

Rewrite the objective function of (7) as

$$G(U^k|_{k=1}^n) = \frac{\sum\limits_{i\neq j} \|(\mathbf{X}_i - \mathbf{X}_j) \times_k U^k|_{k=1}^n\|^2 S_{ij}^p}{\sum\limits_{i\neq j} \|(\mathbf{X}_i - \mathbf{X}_j) \times_k U^k|_{k=1}^n\|^2 S_{ij}}, \qquad (13)$$

and then we have the theory as below:

Theorem-1. By following the terms in Algorithm-1 and Eqn. (13), we have

$$G(U_t^1, \ldots, U_t^{k-1}, U_{t-1}^k, U_{t-1}^{k+1} \ldots, U_{t-1}^n) \leq G(U_t^1, \ldots, U_t^{k-1}, U_t^k, U_{t-1}^{k+1} \ldots, U_{t-1}^n). \qquad (14)$$

Proof. Denote $g(U) = Tr(U^T(S_k^p - \lambda S^k)U)$ where

$$\lambda = G(U_t^1, \ldots, U_t^{k-1}, U_{t-1}^k, U_{t-1}^{k+1} \ldots, U_{t-1}^n),$$

then we have

$$g(U_{t-1}^k) = 0.$$

Algorithm 1 . Procedure to Tensor-based Subspace Learning

1: **Initialization.** Initialize $U_0^1, U_0^2, \ldots, U_0^n$ as arbitrary column orthogonal matrices.
2: **Iterative optimization.**
For t=1, 2, ..., T_{max}, Do
For k=1, 2, ..., n, Do

1. Set $\lambda = \frac{\sum_{i\neq j} \|(\mathbf{X}_i - \mathbf{X}_j) \times_o {U_t^o}|_{o=1}^{k-1} \times_o {U_{t-1}^o}|_{o=k}^{n}\|^2 S_{ij}^p}{\sum_{i\neq j} \|(\mathbf{X}_i - \mathbf{X}_j) \times_o {U_t^o}|_{o=1}^{k-1} \times_o {U_{t-1}^o}|_{o=k}^{n}\|^2 S_{ij}}$.
2. Compute S^k and S_k^p as in (9) based on the projection matrices $U_t^1, \ldots, U_t^{k-1}$ and $U_{t-1}^{k+1}, \ldots, U_{t-1}^n$.
3. Conduct Eigenvalue Decomposition:
$$(S_k^p - \lambda S^k)v_j = \lambda_j v_j,\ j = 1, \ldots, m_k',$$
where v_j is the eigenvector corresponding to the j-th largest eigenvalue λ_j.
4. Reshape the projection directions for the sake of orthogonal transformation invariance:
   (a) Set $V = [v_1, v_2, \ldots, v_{m_k'}]$;
   (b) Let $S^v = VV^T(\sum_i X_i^k {X_i^k}^T)VV^T$, where X_i^k is the mode-k unfolding of the tensor $\mathbf{X}_i$;
   (c) Conduct Eigenvalue Decomposition as
   $$S^v u_i = \gamma_i\, u_i. \qquad (12)$$
5. Set the column vectors of matrix U_t^k as the leading eigenvectors, namely, $U_t^k = [u_1, u_2, \ldots, u_{m_k'}]$.

End
If $\|U_t^k - U_{t-1}^k\| < \sqrt{m_k m_k'}\,\varepsilon, k = 1, 2, \ldots, n$ (ε is set to 10^{-4} in this work), then break.

End
3: **Output the projection matrices** U^k=U_t^k, k=1, 2, ..., n.

Moreover, from $U^TU = I_{m_k'}$, it is easy to prove that

$$\sup g(U) = \sum_{j=1}^{m_k'} \lambda_j.$$

From Algorithm 1, we have $g(U_t^k) = \sum_{j=1}^{m_k'} \lambda_j$, and hence

$$g(U_t^k) \geq g(U_{t-1}^k) = 0.$$

Then, $Tr({U_t^k}^T(S_k^p - \lambda S^k)U_t^k) \geq 0$. As matrix S^k is positive semidefinite [2], we have

$$\frac{Tr({U_t^k}^T S_k^p U_t^k)}{Tr({U_t^k}^T S^k U_t^k)} \geq \lambda,$$

that is,

$$G(U_t^o|_{o=1}^{k-1}, U_{t-1}^o|_{o=k}^n) \leq G(U_t^o|_{o=1}^k, U_{t-1}^o|_{o=k+1}^n)$$

[2] Though S^k may have zero eigenvalues, $Tr({U_t^k}^T S^k U_t^k)$ will be positive when m_k' is larger than the number of the zero eigenvalues.

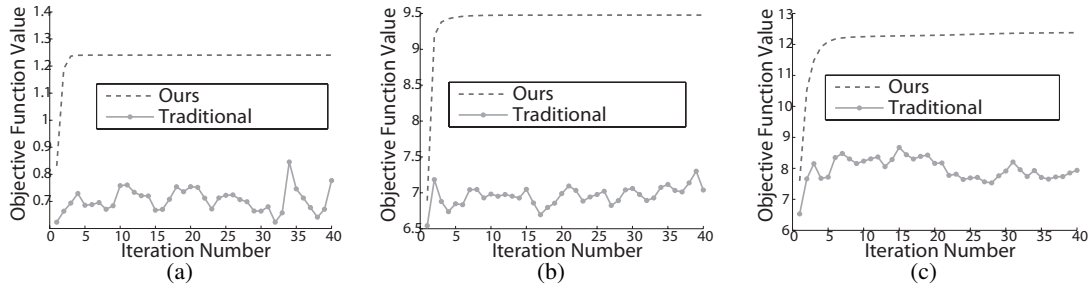


Figure 1: The value of the objective function (7) vs. iteration number. (a) USPS database (b) ORL database, and (c) CMU PIE database. Here the *traditional* method means the solution procedure based ratio trace optimization.

From theorem-1, we can conclude that the value of the objective function monotonously increases.

3.2 Proof of Convergency

To prove the convergency of the projection matrices $U^1, U^2, \ldots, U^n$, we need the concept of *point-to-set* map. The power set $\wp(\chi)$ of a set χ is the collection of all subsets of χ. A *point-to-set* map Ω is a function: $\chi \rightarrow \wp(\chi)$. In our solution procedure to tensor-based subspace learning, the map from $(U_{t-1}^k|_{k=1}^n)$ to $(U_t^k|_{k=1}^n)$ can be considered as a point-to-set map, since each U_t^k is invariant under any orthogonal transformation.

Strict Monotony. An algorithm is a point-to-set map Ω:$\chi \rightarrow \wp(\chi)$. Given an initial point x_0, an algorithm generates a sequence of points via the rule that $x_t \in \Omega(x_{t-1})$. Suppose $J : \chi \rightarrow R_+$ is continuous, non-negative function, an algorithm is called *strict monotony* if 1) $y \in \Omega(x)$ implies that $J(y) \geq J(x)$, and 2) $y \in \Omega(x)$ and $J(y) = J(x)$ imply that $y = x$.

Let set χ be the direct sum of the orthogonal matrix space $O^{m_k \times m'_k}$, that is, the data space $\chi = O^{m_1 \times m'_1} \bigoplus O^{m_2 \times m'_2} \bigoplus \ldots \bigoplus O^{m_n \times m'_n}$, then the Algorithm 1 produces a point-to-set algorithm with respect to $J(x) = G(U^k|_{k=1}^n)$, and it can be proved to be strictly monotonic as follows.

Theorem-2. The point-to-set map from Algorithm 1 is strictly monotonic.

Proof. From theorem-1, we have $G(U_{t-1}^k|_{k=1}^n) \leq G(U_t^k|_{k=1}^n)$, and hence the first condition for strict monotony is satisfied. For the second condition, we take U^1 as an example to prove that this condition is also satisfied. If $G(U_{t-1}^k|_{k=1}^n) = G(U_t^k|_{k=1}^n)$, then from the proof of theorem-1, we have $g(U_{t-1}^1) = g(U_t^1)$ with $\lambda = G(U_{t-1}^k|_{k=1}^n)$ and S^k, S_k^p computed from $(U_{t-1}^k|_{k=1}^n)$. From the proof of theorem-1, we can have that there only exists one orthogonal transformation[3] between U_{t-1}^1 and U_t^1. As shown in Algorithm 1, this kind of orthogonal transformation has been normalized by the reshaping step, hence we have U_{t-1}^1=U_t^1. Similarly, we can prove that $U_t^k = U_{t-1}^k$ for $k = 1, 2, \ldots, n$, hence the second condition is also satisfied and the Algorithm 1 is strictly monotonic.

Theorem-3 [Meyer, 1976]. Assume that the algorithm Ω is strictly monotonic with respect to J and it generates a sequence $\{x_t\}$ which lies in a compact set. If χ is normed, then $\|x_t - x_{t-1}\| \rightarrow 0$.

From theorem-3, we can have the conclusion that the obtained $(U_t^k|_{k=1}^n)$ will converge to a local optimum, since the χ is compact and with norm definition.

4 Experiments

In this section, we systematically examine the convergency properties of our proposed solution procedure to tensor-based subspace learning. We take the Marginal Fisher Analysis (MFA) as an instance of general subspace learning, since MFA has shown to be superior to many traditional subspace learning algorithms such as Linear Discriminant Analysis (LDA); more details on the MFA algorithm is referred to [Yan *et al.*, 2007]. Then, we evaluate the classification capability of the derived low-dimensional representation from our solution procedure compared with the traditional procedure proposed in [Ye *et al.*, 2004] and [Yan *et al.*, 2005]. For tensor-based algorithm, the image matrix, $2^r d$ tensor, is used as input, and the image matrix is transformed into the corresponding vector as the input of vector-based algorithms.

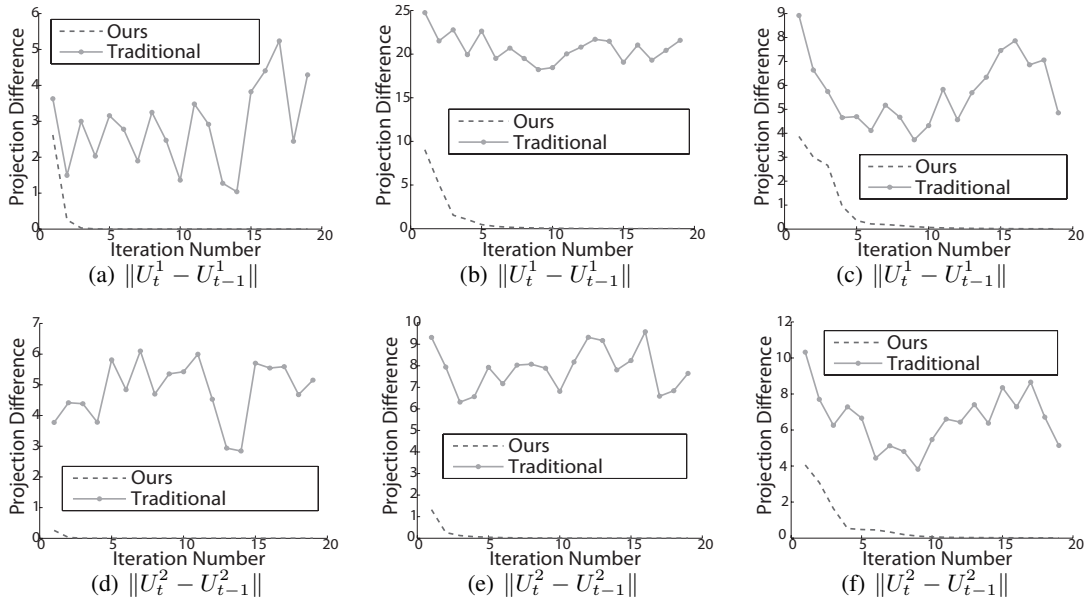
4.1 Data Sets

Three real-world data sets are used. One is the USPS handwritten dataset [4] of 16-by-16 images of handwritten digits with pixel values ranging between -1 and 1. The other two are the benchmark face databases, ORL and CMU PIE [5]. For the face databases, affine transform is performed on all the samples to fix the positions of the two eyes and the mouth center. The ORL database contains 400 images of 40 persons, where each image is normalized to the size of 56-by-46 pixels. The CMU PIE (Pose, Illumination, and Expression) database contains more than 40,000 facial images of 68 people. In our experiment, a subset of five near frontal poses

[3]This claim is based on the assumption that there do not exist duplicated eigenvalues in (11).

[4]Available at: http://www-stat-class.stanford.edu/ tibs/ElemStatLearn/data.html

[5]Available at http://www.face-rec.org/databases/.



(a) $\|U_t^1 - U_{t-1}^1\|$ (b) $\|U_t^1 - U_{t-1}^1\|$ (c) $\|U_t^1 - U_{t-1}^1\|$

(d) $\|U_t^2 - U_{t-1}^2\|$ (e) $\|U_t^2 - U_{t-1}^2\|$ (f) $\|U_t^2 - U_{t-1}^2\|$

Figure 2: The step difference of the projection matrices vs. iteration number. (a,d) USPS database, (b,e) ORL database, and (c,f) CMU PIE database.

(C27, C05, C29, C09 and C07) and illuminations indexed as 08 and 11 are used and normalized to the size of 32-by-32.

4.2 Monotony of Objective Function Value

In this subsection, we examine the monotony property of the objective function value from our solution procedure compared with the optimization procedure that step-wisely transforms the objective function into the ratio trace form. The USPS, ORL and PIE databases are used for this evaluation. The detailed results are shown in Figure 1. It is observed that the traditional ratio trace based procedure does not converge, while our new solution procedure guarantees the monotonous increase of the objective function value and commonly our new procedure will converge after about 4-10 iterations. Moreover, the final converged value of the objective function from our new procedure is much larger than the value of the objective function for any iteration of the ratio trace based procedure.

4.3 Convergency of the Projection Matrices

To evaluate the solution convergency property compared with the traditional ratio trace based optimization procedure, we calculate the difference norm of the projection matrices from two successive iterations and the detailed results are displayed in Figure 2. It demonstrates that the projection matrices converge after 4-10 iterations for our new solution procedure; while for the traditional procedure, heavy oscillations exist and the solution does not converge. As shown in Figure 3, the recognition rate is sensitive to the oscillations caused by the unconvergent projection matrices and the classification accuracy is degraded dramatically.

Table 1: Recognition error rates (%) on the ORL database.

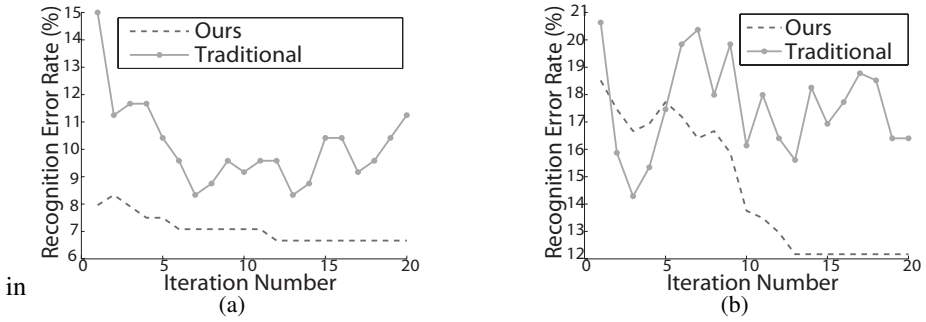
Method	G3P7	G4P6	G5P5
w/o DR.	28.57	24.17	21.5
LDA	17.86	17.08	11.00
MFA_RT	17.50	16.25	10.50
MFA_TR	13.93	10.00	6.50
TMFA_RT	12.14	11.67	5.00
TMFA_TR	**11.07**	**6.67**	**4.00**

Table 2: Recognition error rates (%) on the PIE database.

Method	G3P7	G4P6	G5P5
w/o DR.	49.89	31.75	30.16
LDA	18.82	19.84	18.10
MFA_RT	16.55	15.61	13.65
MFA_TR	14.97	13.49	9.52
TMFA_RT	14.74	14.29	**3.81**
TMFA_TR	**13.61**	**12.17**	9.52

4.4 Face Recognition

In this subsection, we conduct classification experiments on the benchmark face databases. The Tensor Marginal Fisher Analysis algorithm based on our new solution procedure (TMFA_TR) is compared with the traditional ratio trace based Tensor Marginal Fisher Analysis (TMFA_RT), LDA, Ratio Trace based MFA (MFA_RT) and Trace Ratio based MFA (MFA_TR), where MFA_TR means to conduct tensor-based MFA by assuming n=1. To speed up model training, PCA is conducted as a preprocess step for vector-based algorithms. The PCA dimension is set as N-N_c (N is the sample number



in

Figure 3: Recognition error rate (%) vs. iteration number. (a) ORL database(G4P6), and (b) CMU PIE database(G4P6).

and N_c is the class number), which is equivalent to the case for Fisherface algorithm [Belhumeur *et al.*, 1997]. The same graph configuration with nearest neighbor $k = 3$ for the intrinsic graph and $k_p = 40$ for the penalty graph is adopted for all the MFA based algorithms. Since the traditional tensor subspace learning algorithms do not converge, we terminate the process after 3 iterations.

For comparison, the classification result on the original gray-level features without dimensionality reduction is also reported as the baseline, denoted as 'w/o DR.' in the result tables. In all the experiments, the Nearest Neighbor method is used for final classification. All possible dimensions of the final low-dimensional representation are evaluated, and the best results are reported. For each database, we test various configurations of training and testing sets for the sake of statistical confidence, denoted as $'GxPy'$ for which x images of each subject are randomly selected for model training and the remaining y images of each subject are used for testing. The detailed results are listed in Table 1 and 2. From these results, we can have the following observations:

1. TMFA_TR mostly outperforms all the other methods concerned in this work, with only one exception for the case $G5P5$ on the CMU PIE database.

2. For vector-based algorithms, the trace ratio based formulation (MFA_TR) is consistently superior to the ratio trace based one (MFA_RT) for subspace learning.

3. Tensor representation has the potential to improve the classification performance for both trace ratio and ratio trace formulations of subspace learning.

5 Conclusions

In this paper, a novel iterative procedure was proposed to directly optimize the objective function of general subspace learning based on tensor representation. The convergence of the projection matrices and the monotony property of the objective function value were proven. To the best of our knowledge, it is the first work to give a convergent solution for general tensor-based subspace learning.

6 Acknowledgement

The work described in this paper was supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region and DTO Contract NBCHC060160 of USA.

References

[Belhumeur *et al.*, 1997] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE TPAMI*, pages 711–720, 1997.

[Brand, 2003] M. Brand. Continuous nonlinear dimensionality reduction by kernel eigenmaps. *International Join Conference on Artificial Intelligence (IJCAI)*, 2003.

[Fukunaga, 1991] K. Fukunaga. Introduction to statistical pattern recognition. *Academic Press, second edition*, 1991.

[He *et al.*, 2005] X. He, D.Cai, and P. Niyogi. Tensor subspace analysis. *Advances in Neural Information Processing Systems*, 2005.

[Hogan, 1973] W. Hogan. Point-to-set maps in mathematical programming. *SIAM Rev.*, 15(3):591–603, 1973.

[Meyer, 1976] R. Meyer. Sufficient conditions for the convergence of monotonic mathematical programming algorithms. *J. Comp. Sys. Sci.*, 12:108–121, 1976.

[Shashua and Levin, 2001] A. Shashua and A. Levin. Linear image coding for regression and classification using the tensor-rank principle. *Proceedings of CVPR*, 2001.

[Turk and Pentland, 1991] M. Turk and A. Pentland. Face recognition using eigenfaces. *Proceedings of CVPR*, 1991.

[Vasilescu and Terzopoulos, 2003] M. Vasilescu and D. Terzopoulos. Multilinear subspace analysis for image ensembles. *Proceedings of CVPR*, 2003.

[Yan *et al.*, 2005] S. Yan, D. Xu, Q. Yang, L. Zhang, X. Tang, and H. Zhang. Discriminant analysis with tensor representation. *Proceedings of CVPR*, 2005.

[Yan *et al.*, 2007] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin. Graph embedding and extension: A general framework for dimensionality reduction. *IEEE TPAMI*, 2007.

[Ye *et al.*, 2004] J. Ye, R. Janardan, and Q. Li. Two-dimensional linear discriminant analysis. *Advances in Neural Information Processing Systems*, 2004.

[Ye, 2005] J. Ye. Generalized low rank approximations of matrices. *Machine Learning Journal*, 2005.