Multiplicative Nonnegative Graph Embedding

Changhu Wang[1]*, Zheng Song[2], Shuicheng Yan[2], Lei Zhang[3], Hong-Jiang Zhang[4]
[1]MOE-MS Key Lab of MCC, University of Science and Technology of China
[2]Department of Electrical and Computer Engineering, National University of Singapore
[3]Microsoft Research Asia, [4]Microsoft Advanced Technology Center, Beijing, China
wch@ustc.edu, {zheng.s,eleyans}@nus.edu.sg, {leizhang,hjzhang}@microsoft.com

Abstract

In this paper, we study the problem of nonnegative graph embedding, originally investigated in [14] for reaping the benefits from both nonnegative data factorization and the specific purpose characterized by the intrinsic and penalty graphs [13]. Our contributions are two-fold. On the one hand, we present a multiplicative iterative procedure for nonnegative graph embedding, which significantly reduces the computational cost compared with the iterative procedure in [14] involving the matrix inverse calculation of an M-matrix. On the other hand, the nonnegative graph embedding framework is expressed in a more general way by encoding each datum as a tensor of arbitrary order, which brings a group of byproducts, e.g.*, nonnegative discriminative tensor factorization algorithm, with admissible time and memory cost. Extensive experiments compared with the state-of-the-art algorithms on nonnegative data factorization, graph embedding, and tensor representation demonstrate the algorithmic properties in computation speed, sparsity, discriminating power, and robustness to realistic image occlusions.*

1. Introduction

Motivated by the psychological and physiological evidence on parts-based representations in human vision system [2][5], recently techniques for nonnegative representations have been extensively studied for finding sparse nonnegative bases, such that an image could be formed from these nonnegative bases in a non-subtractive way. Nonnegative matrix factorization (NMF) [5] is the pioneering work for such a purpose, and by following NMF, many algorithms have been proposed for nonnegative data factorization and classification. Li *et al.* [6] imposed extra constraints to reinforce the basis sparsity of NMF; also matrix-based NMF has been extended to nonnegative tensor factorization (NTF) [3][9] for handling the data encoded as general high-order tensors. Wang *et al.* proposed the Fisher-NMF [11], which was further studied by Kotsia *et al.* [7], by adding an extra term of scatter difference to the objective function of NMF. Tao *et al.* [10] proposed to employ local rectangle binary features for image reconstruction.

Beyond the initiative single purpose of nonnegative data factorization, Yang *et al.* [14] proposed a unified formulation, called *nonnegative graph embedding* (NGE), for obtaining customized nonnegative data factorization by simultaneously realizing the specific purpose characterized by the intrinsic and penalty graphs, which can be used to instantiate certain dimensionality reduction algorithm [13]. Despite of the mathematic soundness of NGE, it has two limitations: 1) the time complexity of NGE is very high due to the matrix inverse calculation for the so-called M-matrix in each iteration, and will be prohibitively high when the sample number is too large; and 2) NGE is formulated based on data with vector representations, and its general formulation with tensor representations, coinciding with the general graph embedding framework proposed in [13], however was not investigated. Therefore, a natural question to ask is whether we can obtain such a generalized nonnegative graph embedding framework with the following three characteristics: 1) the derived solution is nonnegative; 2) the procedure to obtain the solution is efficient, ideally again based on the elegant multiplicative iterative procedure as in NMF [5]; and 3) the formulation is general and applicable for data encoded as tensors of arbitrary order.

This work is dedicated to designing such a generalized nonnegative graph embedding framework. First, we present a generalized graph embedding formulation based on tensor representation for reaping the benefits from both nonnegative data factorization and the specific purpose characterized by the intrinsic and penalty graphs. Then, an efficient iterative procedure is proposed to obtain the nonnegative solution, where the basis matrices and coefficient ma-

*Changhu Wang performed this work while being a Research Engineer at the Department of Electrical and Computer Engineering, National University of Singapore.

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

trix are updated in a multiplicative manner without matrix inverse calculation. By inheriting the unification property of the graph embedding framework [13], this new formulation brings a group of novel nonnegative data factorization algorithms, *e.g.*, the nonnegative discriminative tensor factorization algorithm (NDTF) which performs nonnegative data factorization based on tensor data and with the purpose of high discriminating power.

2. Formulation for Generalized Nonnegative Graph Embedding

In this section, we introduce the math formulation for the generalized nonnegative graph embedding by encoding each datum as a tensor of arbitrary order instead of a vector. Let the training data $\mathcal{A} = [\mathcal{X}_1, ..., \mathcal{X}_N]$ be an *n*-th order tensor, where each datum $\mathcal{X}_i \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_{n-1}}$ is represented as an (*n*-1)th order tensor, and N is the total number of training samples. For a general classification problem, the datum $\mathcal{X}_i$ is labeled as $c_i \in \{1, ..., N_c\}$, where N_c is the class number. Denote the sample number of the cth class as n_c. Note that we utilize in this work the following rule to facilitate presentation: for any matrix A, its corresponding lowercase version a_i means the ith column vector of A, and A_{ij} denotes the element of A at the ith row and jth column.

2.1. Objective for Nonnegative Data Factorization

For tensor data, the nonnegative tensor factorization [3] of $\mathcal{A}$ can be represented as the sum of k rank-1 tensors $\mathcal{A} = \sum_{m=1}^{k} (u_m^b \otimes)_{b=1}^{n-1} v_m$. The corresponding objective function to optimize is,

$$\min_{U^b, V: 1 \le b \le n-1} \|\mathcal{A} - \sum_{m=1}^{k} (u_m^b \otimes)_{b=1}^{n-1} v_m\|^2 \quad s.t.\ U^b, V \ge 0,\ \ 1 \le b \le n-1, \tag{1}$$

where $\otimes$ is the outer product operator. $U^b = [u_1^b, ..., u_k^b] \in \mathbb{R}^{d_b \times k}$, $\forall 1 \le b \le n-1$, and $V = [v_1, ..., v_k]$. The relationship between the *n*-th order data and the rank-1 tensor factorization is captured by the set of rank-1 tensors $\mathcal{T}_m = u_m^1 (\otimes u_m^b)_{b=2}^{n-1}$ such that each datum $\mathcal{X}_t$ is represented by a superposition of $\mathcal{T}_1, ..., \mathcal{T}_k$ with the reconstruction coefficients taken from $v_1, ..., v_k$.

Usually, $k < \min(\prod_{b=1}^{n-1} d_b, N)$, and thus we could consider V as the low-dimensional representations for the training data $\mathcal{A}$ with the objective of *best reconstruction* under nonnegative constrains. However, the coefficient matrix derived based on the *best reconstruction* is unnecessarily good at discriminating power, since no label information is leveraged in nonnegative tensor factorization.

2.2. Objective for Purpose of Graph Embedding

In order to reinforce the certain purpose of graph embedding [13], *e.g.*, the separability of the labeled data, without the loss of data reconstruction capability, we divide the data *reconstruction* representations V into two parts, namely,

$$V = [V^1, V^2], \tag{2}$$

where $V^1 = [v_1^1, v_2^1, ..., v_q^1] \in \mathbb{R}^{N \times q}$ ($q < k$), which serves for the certain purpose of graph embedding, and $V^2 = [v_1^2, v_2^2, ..., v_{k-q}^2] \in \mathbb{R}^{N \times (k-q)}$, which contains the additional information together with V^1 for data reconstruction. Note that V^1 is expected to be good for the purpose for graph embedding, while the whole V is used for data reconstruction purpose, and hence the targets of data reconstruction and the purpose of graph embedding coexist harmoniously, and do not mutually compromise as in conventional formulations with two objectives. Similarly, the basis matrix $\{U^b\}_{b=1}^{n-1}$ is also divided into two parts,

$$U^b = [{U^b}^1, {U^b}^2], \tag{3}$$

where ${U^b}^1 \in \mathbb{R}^{d_b \times q}$ and ${U^b}^2 \in \mathbb{R}^{d_b \times (k-q)}$.

There exist varieties of formulations for dimensionality reduction, and Yan *et al.* [13] claimed that most of them can be explained within a unified framework, call graph embedding. Let $G = \{\mathcal{A}, S\}$ be an undirected weighted graph with vertex set $\mathcal{A}$ and similarity matrix $S \in \mathbb{R}^{N \times N}$. Each element of the real symmetric matrix S measures for a pair of vertices the similarity, which is assumed to be nonnegative in this work. The diagonal matrix D and Laplacian matrix L of a graph are defined as,

$$L = D - S,\ \ D_{ii} = \sum_{j \ne i} S_{ij},\quad \forall\, i. \tag{4}$$

Graph embedding generally involves an intrinsic graph G, which characterizes the favorite relationship among the data, and a penalty graph $G^p = \{\mathcal{A}, S^p\}$, which characterizes the unfavorable relationship among the data, with $L^p = D^p - S^p$, where D^p is the diagonal matrix as defined in Eqn. (4). Thus two targets of graph-preserving are given as follows,

$$\begin{cases} \max_{V^1} \sum_{i \ne j} \|V_i^1 - V_j^1\|^2 S_{ij}^p, \\ \min_{V^1} \sum_{i \ne j} \|V_i^1 - V_j^1\|^2 S_{ij}, \end{cases} \tag{5}$$

where V_i^1 is the ith rows of V^1. As aforementioned, ${U^b}^2$ is considered as the complementary space of ${U^b}^1$, and thus the first objective in Eqn. (5) can be approximately transformed into,

$$\min_{V^2} \sum_{i \ne j} \|V_i^2 - V_j^2\|^2 S_{ij}^p. \tag{6}$$

2.3. Unified Formulation

To achieve the above two objectives required for generalized nonnegative graph embedding, we can have a unified objective function as,

$$\min_{U^b, V: 1 \le b \le n-1} \|\mathcal{A} - \sum_{m=1}^{k} (u_m^b \otimes)_{b=1}^{n-1} v_m\|^2 + \alpha Tr({V^1}^T L V^1) + \alpha Tr({V^2}^T L^p V^2), \quad s.t.\ U^b, V \ge 0,\ \ 1 \le b \le n-1, \tag{7}$$

where α is a positive parameter for balancing the aforementioned two objectives.

Note that the above formulation is ill-posed, and the objective has the trend to drive the V^1 to be zero. This issue is also suffered by the formulation for Fisher-NMF [11]. As aforementioned, U^b is the basis matrix and hence it is natural to require that the column vectors of U^b are normalized, namely,

$$\|u_i^b\| = 1,\ i = 1, 2, \cdots, k. \tag{8}$$

This extra constraint makes the optimization problem more complicated, and we compensate the norms of the bases into the coefficient matrix V and get the final object function for generalized nonnegative graph embedding as,

$$\begin{aligned} \min_{U^b, V: 1 \le b \le n-1} & \|\mathcal{A} - \sum_{m=1}^{k} (u_m^b \otimes)_{b=1}^{n-1} v_m\|_F^2 \\ & + \alpha Tr(Q^1 V^{1^T} L V^1 Q^{1^T}) + \alpha Tr(Q^2 V^{2^T} L^p V^2 Q^{2^T}), \\ & s.t.\ U^b, V \ge 0,\ \ 1 \le b \le n-1, \end{aligned} \tag{9}$$

where Q^1 and Q^2 are given by $Q^1 = \prod_{b=1}^{n-1} Q_b^1$ and $Q^2 = \prod_{b=1}^{n-1} Q_b^2$, where

$$\begin{aligned} Q_b^1 &= diag\{\|u_1^b\|, ..., \|u_q^b\|\}, \\ Q_b^2 &= diag\{\|u_{q+1}^b\|, ..., \|u_k^b\|\}. \end{aligned} \tag{10}$$

Note that as the matrices S and S^p are symmetric, thus the matrices L and L^p are also symmetric. This objective function is biquadratic, and generally there does not exist a closed-form solution. We present in the next section an efficient multiplicative iterative procedure for computing the nonnegative solution, which is much faster than the iterative procedure presented in [14] involving the matrix inverse calculation for the so-called M-matrix in each iteration.

3. Multiplicative Iterative Procedure

Most iterative procedures for solving high-order optimization problems transform the original intractable problem into a set of tractable sub-problems, and finally obtain the convergence to a local optimum. Our proposed iterative procedure also follows this philosophy and optimizes $\{U^b\}_{b=1}^{n-1}$ and V alternately.

3.1. Preliminaries

Before formally describing the iterative procedure for generalized nonnegative graph embedding, we first introduce the concept of auxiliary function, and the lemma which shall be used for the algorithmic deduction.

Definition 1 Function $G(A, A')$ is an auxiliary function for function $F(A)$ if the conditions

$$G(A, A') \ge F(A), \quad G(A, A) = F(A), \tag{11}$$

are satisfied.

From the above definition, we have the following lemma with proofs omitted.

Lemma 1 If G is an auxiliary function, then F is non-increasing under the update

$$A^{t+1} = \arg\min_A G(A, A^t), \tag{12}$$

where t means the tth iteration.

3.2. Optimize $\mathbf{U^b}$ for Given V and $\{\mathbf{U^p}\}_{\mathbf{p=1, p\neq b}}^{\mathbf{n-1}}$

For fixed V and $\{U^p\}_{p=1, p\neq b}^{n-1}$, the objective function in Eqn. (9) with respect to U^b can be written as

$$\begin{aligned} & F(U^b) = \|\mathcal{A} - \sum_{m=1}^{k} (u_m^b \otimes)_{b=1}^{n-1} v_m\|_F^2 \\ + \ & \alpha Tr(Q^1 V^{1^T} L V^1 Q^{1^T}) \\ + \ & \alpha Tr(Q^2 V^{2^T} L^p V^2 Q^{2^T}) \\ = \ & \|A_{(b)} - \sum_{m=1}^{k} u_m^b [(\otimes_{p=b+1}^{n-1} u_m^p) \otimes v_m (\otimes_{p=1}^{b-1} u_m^p)]^T\|_F^2 \\ + \ & \alpha Tr(Q^1 V^{1^T} L V^1 Q^{1^T}) \\ + \ & \alpha Tr(Q^2 V^{2^T} L^p V^2 Q^{2^T}) \\ = \ & \|A_{(b)} - U^b Z_u^T\|_F^2 + Tr(U^b Y_u U^{b^T}), \end{aligned} \tag{13}$$

where $[\ \cdot\ ]$ in this equation represents to transform a tensor to a vector. $A_{(b)} \in \mathbb{R}^{d_b \times (d_{b+1} \times \ldots \times d_n \times d_1 \times \ldots \times d_{b-1})}$, which results from flattening the tensor $\mathcal{A}$. Z_u is a matrix, in which the mth column is $[(\otimes_{p=b+1}^{n-1} u_m^p) \otimes v_m (\otimes_{p=1}^{b-1} u_m^p)]$. Y_u is given as

$$\begin{aligned} Y_u &= \alpha \begin{bmatrix} (\prod_{p\neq b} Q_p^1) V^{1^T} L V^1 (\prod_{p\neq b} Q_p^1)^T & 0 \\ 0 & (\prod_{p\neq b} Q_p^2) V^{2^T} L^p V^2 (\prod_{p\neq b} Q_p^2)^T \end{bmatrix} \cdot I \\ &= Y_{u+} - Y_{u-}, \end{aligned}$$

with the matrices Y_{u+} and Y_{u-} defined as,

$$Y_{u+} = \alpha \begin{bmatrix} (\prod_{p\neq b} Q_p^1) V^{1^T} D V^1 (\prod_{p\neq b} Q_p^1)^T & 0 \\ 0 & (\prod_{p\neq b} Q_p^2) V^{2^T} D^p V^2 (\prod_{p\neq b} Q_p^2)^T \end{bmatrix} \cdot I,$$

$$Y_{u-} = \alpha \begin{bmatrix} (\prod_{p\neq b} Q_p^1) V^{1^T} S V^1 (\prod_{p\neq b} Q_p^1)^T & 0 \\ 0 & (\prod_{p\neq b} Q_p^2) V^{2^T} S^p V^2 (\prod_{p\neq b} Q_p^2)^T \end{bmatrix} \cdot I.$$

Note the operator $\cdot$ means that each element of the output matrix is the multiplication of the corresponding elements of two input matrices.

To integrate the nonnegative constraints into the objective function, we set $(\Phi_u)_{ij}$ as the Lagrange multiplier for

constraint $U^b_{ij} \geq 0$, and the matrix $\Phi_u = [(\Phi_u)_{ij}]$. Then the Lagrange $\mathcal{L}(U^b)$ with respect to U^b is defined as,

$$\begin{aligned}\mathcal{L}^b &= \|A_{(b)} - U^b Z_u^T\|_F^2 + Tr(U^b Y_u U^{b^T}) + Tr(\Phi_u U^{b^T}) \\ &= Tr(A_{(b)} A_{(b)}^T) - 2Tr(A_{(b)} Z_u U^{b^T}) + \\ &\quad Tr(U^b Z_u^T Z_u U^{b^T}) + Tr(U^b Y_u U^{b^T}) + Tr(\Phi_u U^{b^T}).\end{aligned}$$

By setting the partial derivation of $\mathcal{L}^b$ with respect to U^b as zero, namely,

$$\frac{\partial \mathcal{L}^b}{\partial U^b} = -2A_{(b)} Z_u + 2U^b Z_u^T Z_u + 2U^b Y_u + \Phi_u = 0,$$

and then along with the Karush-Kuhn-Tucker (KKT) condition [8] of $(\Phi_u)_{ij} U^b_{ij} = 0$, we have

$$\begin{aligned}& -(A_{(b)} Z_u)_{ij} U^b_{ij} + (U^b Z_u^T Z_u)_{ij} U^b_{ij} + (U^b Y_u)_{ij} U^b_{ij} \\ = \;& -(A_{(b)} Z_u)_{ij} U^b_{ij} + (U^b Z_u^T Z_u)_{ij} U^b_{ij} \\ & + (U^b Y_{u+})_{ij} U^b_{ij} - (U^b Y_{u-})_{ij} U^b_{ij} = 0 \qquad (14)\end{aligned}$$

Then for the final solution, the following relation should be satisfied,

$$U^b_{ij} \leftarrow U^b_{ij} \frac{(A_{(b)} Z_u + U^b Y_{u-})_{ij}}{(U^b Z_u^T Z_u + U^b Y_{u+})_{ij}}. \qquad (15)$$

We shall prove afterward that the above updating rule could result in a convergent iterative procedure to obtain a local optimum of the solution. Obviously this updating rule is multiplicative and the non-negativity of the solution is guaranteed.

3.3. Convergence of the Update Rule for $\mathbf{U^b}$

Here, we denote F_{ij} as the part of $F(U^b)$ in Eqn. (13) relevant to U_{ij}, and then we have

$$\begin{aligned}F'_{ij} &= (-2A_{(b)} Z_u + 2U^b Z_u^T Z_u + 2U^b Y_u)_{ij}, \qquad (16) \\ F''_{ij} &= (2Z_u^T Z_u + 2Y_u)_{jj}. \qquad (17)\end{aligned}$$

Then the auxiliary function of F_{ij} is designed as

$$\begin{aligned}G(U^b_{ij}, U^{b\,(t)}_{ij}) &= F_{ij}(U^{b\,(t)}_{ij}) + F'_{ij}(U^{b\,(t)}_{ij})(U^b_{ij} - U^{b\,(t)}_{ij}) \\ &+ \frac{(U^{b^{(t)}} Z_u^T Z_u)_{ij} + (U^{b^{(t)}} Y_{u+})_{ij}}{U^{b\,(t)}_{ij}} (U^b_{ij} - U^{b\,(t)}_{ij})^2. \qquad (18)\end{aligned}$$

Lemma 2 Eqn. (18) is an auxiliary function for F_{ij}.

Proof: Since $G(U^b_{ij}, U^b_{ij}) = F_{ij}(U^b_{ij})$ is obvious, we need only show that $G(U^b_{ij}, U^{b\,(t)}_{ij}) \geq F_{ij}(U^b_{ij})$. To do this, we compare the Taylor series expansion of $F_{ij}(U^b_{ij})$,

$$\begin{aligned}F_{ij}(U^b_{ij}) = F_{ij}(U^{b\,(t)}_{ij}) &+ F'_{ij}(U^{b\,(t)}_{ij})(U^b_{ij} - U^{b\,(t)}_{ij}) \\ &+ \frac{1}{2} F''_{ij}(U^b_{ij} - U^{b\,(t)}_{ij})^2, \qquad (19)\end{aligned}$$

with Eqn. (18). We can see that $G(U^b_{ij}, U^{b\,(t)}_{ij}) \geq F_{ij}(U^b_{ij})$ is equivalent to

$$\frac{(U^{b^{(t)}} Z_u^T Z_u)_{ij} + (U^{b^{(t)}} Y_{u+})_{ij}}{U^{b\,(t)}_{ij}} \geq (Z_u^T Z_u)_{jj} + (Y_u)_{jj}. \qquad (20)$$

It is easy to verify that

$$(U^{b^{(t)}} Z_u^T Z_u)_{ij} = \sum_{m=1}^{k} U^{b\,(t)}_{im} (Z_u^T Z_u)_{mj} \geq U^{b\,(t)}_{ij} (Z_u^T Z_u)_{jj},$$

and

$$\begin{aligned}(U^{b^{(t)}} Y_{u+})_{ij} &= \sum_{m=1}^{k} U^{b\,(t)}_{im} (Y_{u+})_{mj} \\ &\geq U^{b\,(t)}_{ij} (Y_{u+})_{jj} \\ &\geq U^{b\,(t)}_{ij} (Y_{u+} - Y_{u-})_{jj} \\ &= U^{b\,(t)}_{ij} (Y_u)_{jj}. \qquad (21)\end{aligned}$$

Thus, Eqn. (20) holds and $G(U^b_{ij}, U^{b\,(t)}_{ij}) \geq F_{ij}(U^b_{ij})$. □

Lemma 3 Eqn. (15) could be obtained by minimizing the auxiliary function $G(U^b_{ij}, U^{b\,(t)}_{ij})$, where $U^{b\,(t)}_{ij}$ is the iterative solution at the t-th step.

Proof: To obtain the minimum, we only need set the partial derivative $\frac{\partial G(U^b_{ij}, U^{b\,(t)}_{ij})}{\partial U^b_{ij}} = 0$, and have

$$\begin{aligned}&\frac{\partial G(U^b_{ij}, U^{b\,(t)}_{ij})}{\partial U^b_{ij}} = F'_{ij}(U^{b\,(t)}_{ij}) \\ &+ \frac{2(U^{b^{(t)}} Z_u^T Z_u + U^{b^{(t)}} Y_{u+})_{ij}}{U^{b\,(t)}_{ij}} (U^b_{ij} - U^{b\,(t)}_{ij}) = 0. \qquad (22)\end{aligned}$$

Then we can obtain the iterative updating rule for U^b as,

$$U^{b^{(t+1)}}_{ij} \leftarrow U^{b^{(t)}}_{ij} \frac{(A_{(b)} Z_u + U^{b^{(t)}} Y_{u-})_{ij}}{(U^{b^{(t)}} Z_u^T Z_u + U^{b^{(t)}} Y_{u+})_{ij}}, \qquad (23)$$

and the lemma is proved. □

3.4. Optimize V for Given $\{\mathbf{U^b}\}_{\mathbf{b=1}}^{\mathbf{n-1}}$

After updating the matrices $\{U^b\}_{b=1}^{n-1}$, we normalize the column vectors of them and consequently convey the norm to the coefficient matrix V, namely,

$$\begin{aligned}v_m &\leftarrow v_m \prod_{b=1}^{n-1} \|u^b_m\|, \forall\, m, \qquad (24) \\ u^b_m &\leftarrow u^b_m / \|u^b_m\|, \forall\, m, b. \qquad (25)\end{aligned}$$

Then based on the normalized $\{U^b\}_{b=1}^{n-1}$ in Eqn. (25), the objective function in Eqn. (9) with respect to V is then simplified to be,

$$
\begin{aligned}
F(V) &= \|\mathcal{A} - \sum_{m=1}^{k} (u_m^b \otimes)_{b=1}^{n-1} v_m\|_F^2 \\
&\quad + \alpha Tr(V^{1^T} L V^1) + \alpha Tr(V^{2^T} L^p V^2) \\
&= \|A_{(n)} - \sum_{m=1}^{k} v_m [u_m^1 (\otimes u_m^b)_{b=2}^{n-1}]^T\|_F^2 \\
&\quad + \alpha Tr(V^{1^T} L V^1) + \alpha Tr(V^{2^T} L^p V^2) \\
&= \|A_{(n)} - V Z_v^T\|_F^2 + Tr(V^{1^T} Y_v^1 V^1) \\
&\quad + Tr(V^{2^T} Y_v^2 V^2), \qquad (26)
\end{aligned}
$$

where $[\ \cdot\ ]$ in this equation represents to transform a tensor to a vector. Z_v is a matrix, where the mth column is $[u_m^1 (\otimes u_m^b)_{b=2}^{n-1}]$. Y_v^1 and Y_v^2 are given as,

$$
\begin{aligned}
Y_v^1 &= \alpha L \ = Y_{v+}^1 - Y_{v-}^1, \\
Y_v^2 &= \alpha L^p = Y_{v+}^2 - Y_{v-}^2,
\end{aligned}
$$

with the matrices defined as,

$$
\begin{aligned}
Y_{v+}^1 &= \alpha D, \quad Y_{v+}^2 = \alpha D^p, \\
Y_{v-}^1 &= \alpha S, \quad Y_{v-}^2 = \alpha S^p.
\end{aligned}
$$

To integrate the nonnegative constraints into the objective function, we set $(\Phi_v)_{ij}$ as the Lagrange multiplier for constraint $V_{ij} \geq 0$, and the matrix $\Phi_v = [(\Phi_v)_{ij}]$. Then the Lagrange $\mathcal{L}^v$ with respect to V is defined as,

$$
\begin{aligned}
\mathcal{L}^v &= \|A_{(n)} - V Z_v^T\|_F^2 + Tr(V^{1^T} Y_v^1 V^1) \\
&\quad + Tr(V^{2^T} Y_v^2 V^2) + Tr(\Phi_v V^T) \\
&= Tr(A_{(n)} A_{(n)}^T) - 2Tr(A_{(n)} Z_v V^T) \\
&\quad + Tr(V Z_v^T Z_v V^T) + Tr(V^{1^T} Y_v^1 V^1) \\
&\quad + Tr(V^{2^T} Y_v^2 V^2) + Tr(\Phi_v V^T). \qquad (27)
\end{aligned}
$$

By setting the partial derivation of $\mathcal{L}^v$ with respect to V as zero,

$$
\begin{aligned}
\frac{\partial \mathcal{L}^v}{\partial V} &= -2A_{(n)} Z_v + 2V Z_v^T Z_v \\
&\quad + 2[Y_v^1 V^1, Y_v^2 V^2] + \Phi_v = 0,
\end{aligned}
$$

along with the Karush-Kuhn-Tucker (KKT) condition [8] of $(\Phi_v)_{ij} V_{ij} = 0$, we can have

$$
\begin{aligned}
& -(A_{(n)} Z_v)_{ij} V_{ij} + (V Z_v^T Z_v)_{ij} V_{ij} \\
+ \ & [Y_v^1 V^1, Y_v^2 V^2]_{ij} V_{ij} \\
= \ & -(A_{(n)} Z_v)_{ij} V_{ij} + (V Z_v^T Z_v)_{ij} V_{ij} \\
+ \ & [Y_{v+}^1 V^1, Y_{v+}^2 V^2]_{ij} V_{ij} - [Y_{v-}^1 V^1, Y_{v-}^2 V^2]_{ij} V_{ij} \\
= \ & 0.
\end{aligned}
$$

Then the final relation on the solution should be satisfied,

$$
V_{ij} \leftarrow V_{ij} \frac{(A_{(n)} Z_v + [Y_{v-}^1 V^1, Y_{v-}^2 V^2])_{ij}}{(V Z_v^T Z_v + [Y_{v+}^1 V^1, Y_{v+}^2 V^2])_{ij}}, \qquad (28)
$$

which offers an updating rule for a convergent iterative procedure to obtain a local optimum solution for V.

3.5. Convergence of the Update Rule for V

Here, we denote F_{ij} as the part of $F(V)$ in Eqn. (26) relevant to V_{ij}, and then we have,

$$
\begin{aligned}
F'_{ij} &= (-2A_{(n)} Z_v + 2V Z_v^T Z_v + 2[Y_v^1 V^1, Y_v^2 V^2])_{ij}, \\
F''_{ij} &= \begin{cases} 2(Z_v^T Z_v)_{jj} + 2(Y_v^1)_{ii} & \text{if } j \leq q; \\ 2(Z_v^T Z_v)_{jj} + 2(Y_v^2)_{ii} & \text{otherwise.} \end{cases}
\end{aligned}
$$

Then the auxiliary function of F_{ij} is designed as

$$
\begin{aligned}
G(V_{ij}, V_{ij}^t) &= F_{ij}(V_{ij}^t) + F'_{ij}(V_{ij}^t)(V_{ij} - V_{ij}^t) \\
&+ \frac{(V^t Z_v^T Z_v)_{ij} + [Y_{v+}^1 V^{1^t}, Y_{v+}^2 V^{2^t}]_{ij}}{V_{ij}^t} (V_{ij} - V_{ij}^t)^2. \qquad (29)
\end{aligned}
$$

Lemma 4 Eqn. (29) is an auxiliary function for F_{ij}.

Proof: Since $G(V_{ij}, V_{ij}) = F_{ij}(V_{ij})$ is obvious, we need only show that $G(V_{ij}, V_{ij}^t) \geq F_{ij}(V_{ij})$. To do this, we compare the Taylor series expansion of $F_{ij}(V_{ij})$,

$$
\begin{aligned}
F_{ij}(V_{ij}) &= F_{ij}(V_{ij}^t) + F'_{ij}(V_{ij}^t)(V_{ij} - V_{ij}^t) \\
&\quad + \frac{1}{2} F''_{ij} (V_{ij} - V_{ij}^t)^2, \qquad (30)
\end{aligned}
$$

with Eqn. (29), and then $G(V_{ij}, V_{ij}^t) \geq F_{ij}(V_{ij})$ is equivalent to

$$
\begin{aligned}
& \frac{(V^t Z_v^T Z_v)_{ij} + [Y_{v+}^1 V^{1^t}, Y_{v+}^2 V^{2^t}]_{ij}}{V_{ij}^t} \\
\geq \ & \begin{cases} (Z_v^T Z_v)_{jj} + (Y_v^1)_{ii} & \text{if } j \leq q; \\ (Z_v^T Z_v)_{jj} + (Y_v^2)_{ii} & \text{otherwise.} \end{cases} \qquad (31)
\end{aligned}
$$

It is easy to verify that

$$
(V^t Z_v^T Z_v)_{ij} = \sum_{m=1}^{k} V_{im}^t (Z_v^T Z_v)_{mj} \geq V_{ij}^t (Z_v^T Z_v)_{jj}, \qquad (32)
$$

and

$$
\begin{aligned}
& [Y_{v+}^1 V^{1^t}, Y_{v+}^2 V^{2^t}]_{ij} \\
= \ & \begin{cases} \sum_{m=1}^{d_n} (Y_{v+}^1)_{im} V_{mj}^t & \text{if } j \leq q; \\ \sum_{m=1}^{d_n} (Y_{v+}^2)_{im} V_{mj}^t & \text{otherwise.} \end{cases} \\
\geq \ & \begin{cases} (Y_{v+}^1)_{ii} V_{ij}^t & \text{if } j \leq q; \\ (Y_{v+}^2)_{ii} V_{ij}^t & \text{otherwise.} \end{cases} \\
\geq \ & \begin{cases} (Y_{v+}^1 - Y_{v-}^1)_{ii} V_{ij}^t & \text{if } j \leq q; \\ (Y_{v+}^2 - Y_{v-}^2)_{ii} V_{ij}^t & \text{otherwise.} \end{cases} \\
= \ & \begin{cases} (Y_v^1)_{ii} V_{ij}^t & \text{if } j \leq q; \\ (Y_v^2)_{ii} V_{ij}^t & \text{otherwise.} \end{cases} \qquad (33)
\end{aligned}
$$

Thus, Eqn. (31) holds and $G(V_{ij}, V_{ij}^t) \geq F_{ij}(V_{ij})$. □

Lemma 5 Eqn. (28) could be obtained by minimizing the auxiliary function $G(V_{ij}, V_{ij}^t)$.

Proof: To obtain the minimum, we only need set the partial derivative $\frac{\partial G(V_{ij}, V_{ij}^t)}{\partial V_{ij}} = 0$, and have

$$
\begin{aligned}
& \frac{\partial G(V_{ij}, V_{ij}^t)}{\partial V_{ij}} = F'_{ij}(V_{ij}^t) \\
& + \frac{2(V^t Z_v^T Z_v + [Y_{v+}^1 V^{1^t}, Y_{v+}^2 V^{2^t}])_{ij}}{V_{ij}^t} (V_{ij} - V_{ij}^t) = 0. \qquad (34)
\end{aligned}
$$

Then we can obtain the iterative updating rule for V as,

$$V_{ij}^{t+1} \leftarrow V_{ij}^{t} \frac{(A_{(n)} Z_v + [Y_{v-}^1 V^{1^t}, Y_{v-}^2 V^{2^t}])_{ij}}{(V Z_v^T Z_v + [Y_{v+}^1 V^{1^t}, Y_{v+}^2 V^{2^t}])_{ij}}, \quad (35)$$

and the lemma is proved. □

4. Experiments

In this section, we take the intrinsic and penalty graphs from the marginal fisher analysis [13] algorithm to instantiate the proposed generalized nonnegative graph embedding framework, and the corresponding algorithm is called nonnegative discriminative tensor factorization (NDTF). We systematically evaluate its effectiveness in terms of computation speed, basis sparsity, discriminating power, and robustness to realistic image occlusions by comparing with the state-of-the-art algorithms on nonnegative data factorization, graph embedding, and tensor representation.

4.1. Experiment Setup

Several popular subspace learning and nonnegative learning algorithms are evaluated for comparison purpose: four unsupervised ones including principal component analysis (PCA) [4], nonnegative matrix factorization (NMF) [5], localized nonnegative matrix factorization (LNMF) [6], and nonnegative tensor factorization (NTF) [3], two supervised ones including linear discriminant analysis (LDA) [1] and marginal fisher analysis (MFA) [13], three nonnegative graph embedding methods, including M-matrix based nonnegative graph embedding (NGE) [14], the proposed 2D nonnegative discriminative tensor factorization (2D-NDTF), and the proposed 3D nonnegative discriminative tensor factorization (3D-NDTF). In this work, among the above nine algorithms, NTF and 3D-NDTF consider an image as a matrix, while the other ones consider an image as a vector. The matrix versions of other supervised algorithms are not further evaluated in this work due to their inherent convergency issue and the comparable performances with their vector versions as shown in [15].

For the NDTF algorithms, the intrinsic graph and penalty graph are set the same as those for MFA, where the number of nearest neighbors of each sample is fixed to be $min(n_c - 1, 3)$ and the number of shortest pairs from different classes is set as 20 for each class in this work. For nonnegative data factorization related algorithms, the reconstruction coefficients for a new datum is computed as in [6].

Three benchmark face database[1], *i.e.* ORL, FERET, and CMU PIE, are used. All images are aligned by fixing the locations of two eyes. The ORL database contains 40 persons, each with 10 images. For the FERET database, we use 70 people with six images for each person. 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 (C27, C05, C29, C09, and C07) and illuminations indexed as 08 and 11 is used, and therefore each person has ten images. For the ORL database, the images are normalized to 64-by-64 pixels; for the FERET database, the images are normalized to 56-by-46 pixels; and for the PIE database, the images are normalized to 32-by-32 pixels. For all databases, half of the images for each person are randomly selected as training data, and the other half for testing. The reported accuracy is averaged over five random splits of all data.

[1] http://www.face-rec.org/databases/

Table 1. The time cost per iteration on average (seconds) for NGE [14] and 2D-NDTF on ORL, FERERT, and PIE databases.

Algorithm	ORL	FERET	PIE
NGE [14]	16.90	15.71	62.97
2D-NDTF	**0.35**	**0.22**	**0.16**

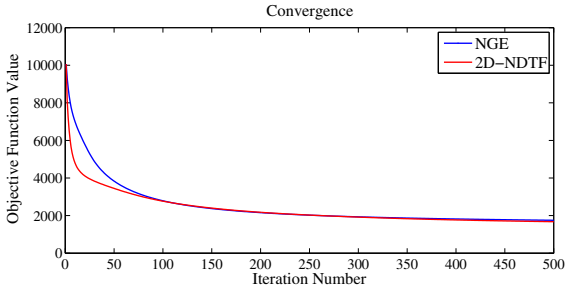

Figure 1. Comparison objective function value vs. iteration number for NGE and 2D-NDTF on the ORL database.

4.2. Computation Speed

As aforementioned, the M-matrix based nonnegative graph embedding algorithm (NGE) proposed in [14] suffers from high time complexity caused by the matrix inverse calculation for the M-matrix in each iteration, and will be prohibitively high when the sample number is too large.

The proposed NDTF is a multiplicative nonnegative graph embedding algorithm as shown in Eqn. (15) and (28). It is obvious that the multiplicative update rule is much more efficient than the M-matrix based update rule. The time cost per iteration on average for NGE and 2D-NDTF is listed in Table 1. The two algorithms used the same initialization matrices, and they were implemented using Matlab 2008a on a computer with Intel (R)Core (TM) 2 Duo 2.66GHz CPU and 4GB of RAM. From Table 1 we can see that the speed of 2D-NDTF is overwhelmingly faster than NGE. More specifically, 2D-NDTF is about 47 times on ORL, 70 times on FERET, and 393 times on PIE faster than NGE. The slow speed of NGE algorithm is caused by the computation of k times of the matrix inverse calculation for the M-matrix in each iteration, where k is the number of bases introduced in Eqn. (7). We also compared the con-

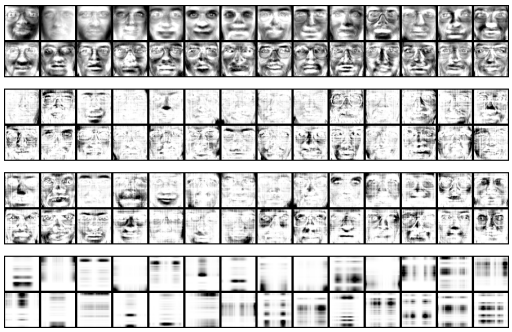

Figure 2. Basis matrix visualization of the algorithms PCA (1st row), NMF (2nd row), 2D-NDTF (3nd row), and 3D-NDTF (4th row, based on $\mathcal{T}_m$ as described in Section 2.1) based on the training data of the ORL database.

vergency speed of these two algorithms. Figure 1 shows the objective function value decreases with the iterations on the ORL database. We can observe that 2D-NDTF is a little better than NGE on algorithmic convergency when the iteration number is small. As the cost functions of NGE and 2D-NDTF are the same, we will only show the performance of 2D-NDTF afterwards.

4.3. Sparsity Analysis

In this subsection, we examine the sparsity property of the NDTF related algorithms. The basis matrices of 2D-NDTF and 3D-NDTF compared with those from PCA and NMF on the ORL database are depicted in Figure 2, from which we can observe that: 1) the bases of 2D-NDTF, 3D-NDTF and NMF are much sparser than those of PCA, and 2) 3D-NDTF is also sparser than 2D-NDTF. On the one hand, by leveraging the labeled and unlabeled data, 2D-NDTF and 3D-NDTF may have superior discriminative capability over those unsupervised nonnegative algorithms such as NMF and LNMF; on the other hand, the sparsity property of 2D-NDTF and 3D-NDTF algorithms makes them potentially more robust to image occlusions than PCA and other related algorithms. We will validate these points in the next subsections.

4.4. Classification Capability

In this subsection, we evaluate the discriminating power of the 2D-NDTF and 3D-NDTF algorithms by comparing them with three popular subspace learning algorithms: PCA, LDA, and MFA, as well as three nonnegative data factorization algorithms: NMF, LNMF, and NTF. For LDA and MFA, we first reduce the data to the dimension of N-N_c using PCA, where N is the number of training data and N_c is the number of classes, for avoiding the singular value issue as conventionally. To be fair, all other algorithms also use the same PCA preprocessing. For all nonnegative algorithms, the parameter k is set as $N \times m/(N+m)$ in all the experiment settings, where m is the feature dimension

Table 2. Face recognition accuracies (%) of different algorithms on three databases. Notice that the values in parentheses are the standard deviations of five rounds.

Algorithm	ORL	FERET	PIE
PCA	87.10 (±2.46)	79.81 (±3.73)	80.89 (±1.64)
NMF	86.90 (±3.78)	69.43 (±3.85)	80.57 (±2.99)
LNMF	87.00 (±1.50)	84.19 (±2.85)	85.84 (±3.02)
NTF	88.10 (±2.38)	82.67 (±2.51)	80.44 (±2.79)
LDA	94.30 (±1.15)	**89.91 (±4.42)**	95.11 (±1.20)
MFA	**95.30 (±1.15)**	89.33 (±4.36)	95.30 (±1.18)
2D-NDTF	95.10 (±1.34)	89.81 (±2.75)	96.00 (±0.66)
3D-NDTF	**95.30 (±1.60)**	89.81 (±3.46)	**96.57 (±1.20)**

of the vector representation of an image. q is simply set to be N_c for NDTF related algorithms. The parameter α in 2D-NDTF and 3D-NDTF is selected from [10, 100, 1000]. We report the best results by exploring all possible feature dimensions for all algorithms as conventionally [13].

The comparison results of different algorithms on the ORL, FERET, and PIE databases are listed in Table 2, from which we could draw the following conclusions. First, the performances of nonnegative algorithms NMF, LNMF, and NTF are much worse than supervised algorithms LDA, MFA, 2D-NDTF, and 3D-NDTF, which shows that without considering the labeled data, nonnegative algorithms could not guarantee good discriminating power. Second, the performances of NDTF related algorithms slightly outperform MFA and LDA on average, since they all fully utilize the label information, and the nonnegative property is limited in bringing much greater classification power. However, as shown in the next subsection, due to the sparsity property, NDTF related algorithms are more robust than MFA and LDA to image occlusions.

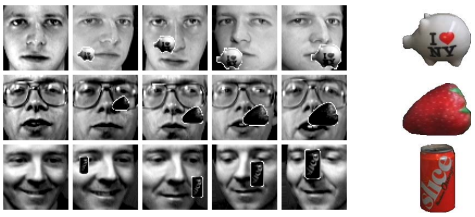

Figure 3. Sample images from the ORL database with occlusion patch sizes as 0-by-0, 16-by-16, 20-by-20, 24-by-24, and 28-by-28 pixels respectively. The original occlusion objects are also listed. Different occlusion types from top row to bottom row are: 1) piggy bank, 2) strawberry, and 3) pop can.

4.5. Robustness to Realistic Image Occlusions

As aforementioned, the bases from NDTF algorithms are sparse, localized, and discriminative, which indicates that NDTF related algorithms are potentially more robust to image occlusions compared with other subspace learning algorithms. To verify this point, we randomly add realistic image occlusions of different sizes to the testing im-

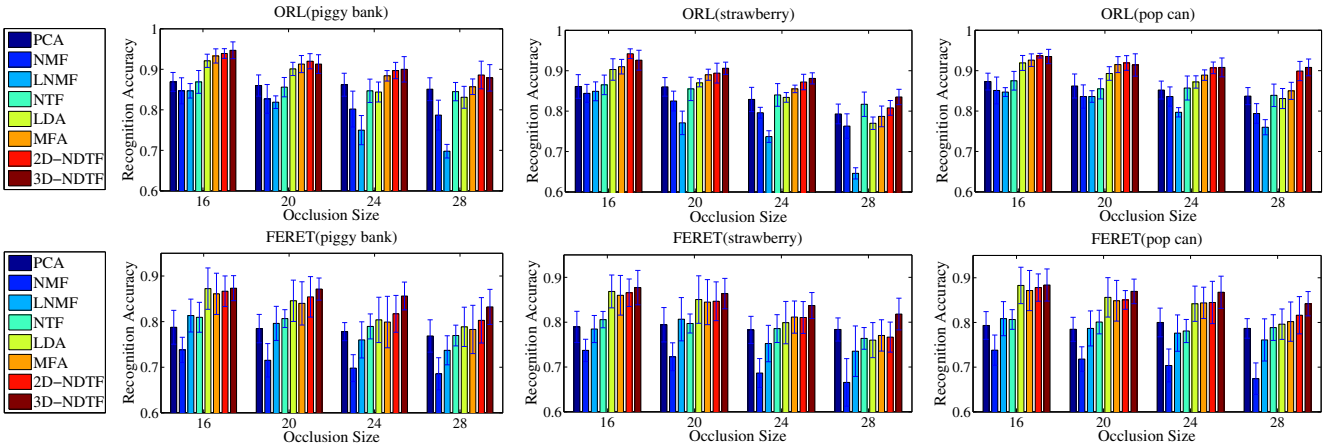

Figure 4. Face recognition accuracy vs. occlusion patch size on the ORL face database (top row) and FERET face database (bottom row). For better viewing, please see the color pdf file.

ages. Several different objects are used as occlusions, *i.e.* 1) "piggy bank", 2) "strawberry", and 3) "pop can". Several example faces with occlusions of different sizes and different types are depicted in Figure 3. Figure 4 presents the face recognition accuracies in the cases with occlusion on ORL and FERET databases. From these results, we can have the following observations: 1) the performances of unsupervised algorithms are much lower than supervised algorithms when occlusion size is small; 2) the gap between unsupervised algorithms and supervised algorithms is becoming smaller when the occlusion size is increasing, since the larger occlusion could weaken the effect of supervised learning; 3) the superiority of NDTF algorithms over LDA and MFA becomes more and more clear with the increase of occlusion size, which shows that NDTF algorithms are more robust to image occlusions compared with other subspace learning algorithms such as LDA and MFA; and 4) 3D-NDTF outperforms 2D-NDTF on average. It should be noted that due to the space limitation, we have omitted the occlusion results on the PIE database, which are consistent with the above observations.

5. Conclusions and Future Works

In this paper, we studied the generalized framework of nonnegative graph embedding, the vector version of which was initially studied in [14]. We presented a multiplicative update rule for solving this general problem, and also generalized the framework to the cases with data encoded as tensors of arbitrary order. This generalized framework is a tool and can be used to design new nonnegative data factorization algorithms for specific purpose, such as the nonnegative discriminative tensor factorization algorithm (NDTF) discussed in the experimental section. We are planning to further explore this generalized nonnegative graph embedding framework from three aspects: 1) to design semi-supervised nonnegative data factorization algorithm based on this framework for harnessing the unlabeled data in model learning stage, 2) to explore solution for online learning [12], and 3) to explore other ways to harmoniously achieve the two objectives, *i.e.*, good reconstruction capability and good discriminating power.

Acknowledgement

This work is supported by NRF/IDM Program, under research Grant NRF2008IDM-IDM004-029.

References

[1] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *TPAMI*, 2002.

[2] D. Field. What is the Goal of Sensory Coding? *Neural Computation*,1994.

[3] T. Hazan, S. Polak, and A. Shashua. Sparse image coding using a 3d nonnegative tensor factorization. *ICCV*, 2005.

[4] I. Joliffe. Principal component analysis. *Springer-Verlag, New York*, 1986.

[5] D. Lee and H. Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature*, 1999.

[6] S. Li, X. Hou, H. Zhang, and Q. Cheng. Learning spatially localized, parts-based representation. *CVPR*, 2001.

[7] I. Kotsia, S. Zafeiriou, and I. Pitas. A Novel Discriminant Nonnegative Matrix Factorization Algorithm With Applications to Facial Image Characterization Problems. *TIFS*, 2007.

[8] H. Kuhn, and A. Tucker. Nonlinear programming. *Proceedings of 2nd Berkeley Symposium*, 1951.

[9] A. Shashua and T. Hazan. Nonnegative Tensor Factorization with Applications to Statistics and Computer Vision. *ICML*, 2005.

[10] H. Tao, R. Crabb, and F. Tang. Non-orthogonal binary subspace and its applications in computer vision. *CVPR*, 2005.

[11] Y. Wang, Y. Jiar, C. Hu, and M. Turk. Fisher nonnegative matrix factorization for learning local features. *ACCV*, 2004.

[12] J. Yan, B. Zhang, S. Yan, Q. Yang, H. Li, Z. Chen, W. Xi, W. Fan, W. Ma, and Q. Cheng IMMC: Incremental Maximum Margin Criterion. *SIGKDD*, 2004.

[13] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang and S. Lin. Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. *TPAMI*, 2007.

[14] J. Yang, S. Yan, Y. Fu, X. Li, and T. Huang. Nonnegative Graph Embedding. *CVPR*, 2008.

[15] J. Ye, R. Janardan, and Q. Li. Two-Dimensional Linear Discriminant Analysis. *NIPS*, 2004.