Multimed Tools Appl (2014) 72:2679–2699
DOI 10.1007/s11042-013-1572-z

Graph regularized discriminative non-negative matrix factorization for face recognition

Xianzhong Long · Hongtao Lu · Yong Peng · Wenbin Li

Published online: 6 July 2013

Abstract Non-negative matrix factorization (NMF) has been widely employed in computer vision and pattern recognition fields since the learned bases can be interpreted as a natural parts-based representation of the input space, which is consistent with the psychological intuition of combining parts to form a whole. In this paper, we propose a novel constrained nonnegative matrix factorization algorithm, called the graph regularized discriminative non-negative matrix factorization (GDNMF), to incorporate into the NMF model both intrinsic geometrical structure and discriminative information which have been essentially ignored in prior works. Specifically, both the graph Laplacian and supervised label information are jointly utilized to learn the projection matrix in the new model. Further we provide the corresponding multiplicative update solutions for the optimization framework, together with the convergence proof. A series of experiments are conducted over several benchmark face datasets to demonstrate the efficacy of our proposed GDNMF.

Keywords Non-negative matrix factorization · Graph Laplacian · Discriminative information · Face recognition

X. Long (✉) · H. Lu · Y. Peng
Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai, China
e-mail: lxz85@sjtu.edu.cn

H. Lu
e-mail: lu-ht@sjtu.edu.cn

Y. Peng
e-mail: pengyong851012@sjtu.edu.cn

W. Li
Department of Diagnostic and Interventional Radiology,
Affiliated Sixth People's Hospital, Shanghai Jiao Tong University, Shanghai, China
e-mail: liwenbin@sh163.net

1 Introduction

Non-negative matrix factorization (NMF) was originally proposed to incorporate the non-negative constraints into general matrix factorization [13, 14]. It aims to find two non-negative matrices whose product provides a good approximation to the original matrix. The nonnegativity properties lead to a parts-based representation because they allow only additive, not subtractive, combinations. More particularly, NMF represents data as a linear combination of a set of basis vectors, in which both the combination coefficients and the basis vectors are non-negative. This property is consistent with the psychological and physical evidence of the parts-based representation in the human brain [16, 17, 22]. The advantages of this parts-based representation have been observed in many real world problems such as face recognition [15], document clustering [23] and DNA gene expression analysis [5].

As one of the most challenging classification tasks in computer vision and pattern recognition, face recognition has drawn many researchers' attentions; many techniques have been proposed in the past few decades. A face image of size $p \times q$ pixels is usually represented by a $p \times q$ dimensional vector. However, these $p \times q$ dimensional vectors are too large to allow fast face recognition and the recognition accuracy is usually low. In order to deal with the high dimensional data quickly and improve the recognition rate, some dimensionality reduction techniques have been proposed. Among these dimensionality reduction methods, NMF and its variants have been widely utilized in face recognition field. Here we give a brief review on these models for their advantages as well as disadvantages. More specifically, Donoho et al. [9] proved that NMF does not necessarily decompose an object into parts, i.e., NMF can not obtain the parts-based representation on some datasets. In order to overcome this problem and learn spatially localized, parts-based representation of visual patterns, local non-negative matrix factorization (LNMF) was proposed in [15]. The purpose of the LNMF is to impose the locality of features on bases and make the representation suitable for tasks where the feature localization is more important. However, LNMF ignores the data geometric structure and the discriminative information although it could learn a more robust parts-based representation than NMF. To use the data geometric structure, Cai et al. [6] proposed another variant of NMF, which is called graph regularized non-negative matrix factorization (GNMF). In the GNMF algorithm, the geometrical structure of data is encoded by a k nearest neighbor graph. GNMF was specially designed for clustering tasks, so it may not perform well for classification problems. By imposing manifold regularization and margin maximization on NMF, another variant of NMF called manifold regularized discriminative non-negative matrix factorization (MD-NMF) was introduced in [11]. MD-NMF considered both the local geometry of data and the discriminative information of different classes simultaneously. In summary, these variants of NMF have their own properties in classification or cluster tasks.

Recently, many researchers hold the idea that the data is more likely to reside on a low-dimensional submanifold embedded in the high-dimensional ambient space. In order to detect the underlying manifold structure, many manifold learning algorithms have been proposed, such as locally linear embedding (LLE) [18], isometric mapping (ISOMAP) [21], laplacian eigenmap (LE) [2]. All these algorithms made use of the

so-called locally invariant idea [12], i.e., the nearby points were likely to have similar embeddings. It has been shown that learning performance could be significantly enhanced if the geometrical structure was exploited and the local invariance was considered. In addition, the authors incorporated the labels into the dictionary learning stage and obtained a discriminative dictionary, which was used in face recognition and good performance was achieved in [24].

Inspired by the success in non-negative matrix factorization based on graph regularized [6] and discriminative dictionary learning [24], in this paper, we put forward a novel algorithm, called graph regularized discriminative non-negative matrix factorization (GDNMF), which explicitly considers the local invariance and label information. We encode the geometrical structure of data space by constructing a k nearest neighbor graph and increase the discriminative ability of different classes by considering the label information. Our goal is to find a parts-based representation discriminative space in which two data points are sufficiently close to each other if they are connected in the graph. To this end, we present a new matrix decomposition objective function by integrating the graph structure and label information. We also develop an optimization scheme to solve this objective function based on multiplicative iterative updates of these three factor matrices. This generates a new parts-based data representation which takes into account the geometrical structure and discriminative information of the input space simultaneously. Moreover, the convergence proof of our optimization framework is provided in Appendix. Experiments show that our proposed approach achieves better recognition accuracy than some recent variants of NMF.

The remainder of the paper is organized as follows: Section 2 introduces the basic idea of existing variants of NMF. The proposed GDNMF model as well as the optimization method are described in detail in Section 3. In Section 4, the comparative results of face recognition on four widely used face datasets are reported. Finally, conclusions are given in Section 5.

2 Related work

Let $\mathbf{X}$ be a data matrix of n m-dimensional samples $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$, i.e., $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$. Each column of $\mathbf{X}$ represents a face image with m dimensions. Usually, m is very large and this may result in slow recognition speed and low recognition accuracy. Thus, dimensionality reduction is necessary before recognition and some newest variants of NMF are used as dimensionality reduction techniques. This section first introduces the original NMF algorithm and then reviews three variants of NMF.

2.1 Non-negative matrix factorization (NMF)

Non-negative matrix factorization (NMF) [13] decomposes a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ into a product of two matrices $\mathbf{W} \in \mathbb{R}^{m \times r}$ and $\mathbf{H} \in \mathbb{R}^{r \times n}$ $(r \ll \min(m, n))$, i.e., $\mathbf{X} \approx \mathbf{WH}$. Some algorithms for NMF have been raised in [4, 14], such as multiplicative update algorithms, gradient descent algorithms and alternating least squares algorithms.

Multiplicative update rules were firstly considered in [14]. The first criterion function of NMF is based on minimizing the Euclidean distance between $\mathbf{X}$ and $\mathbf{WH}$. The corresponding optimization problem is as follows:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{H}} \|\mathbf{X} - \mathbf{WH}\|_F^2 \\ &\text{s.t.} \mathbf{W} \geq 0, \mathbf{H} \geq 0 \end{aligned} \tag{1}$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm, $\mathbf{X} \in \mathbb{R}^{m\times n}$ is a training sample matrix, $\mathbf{W} \in \mathbb{R}^{m\times r}$ is called basis matrix and $\mathbf{H} \in \mathbb{R}^{r\times n}$ is coefficient matrix. All the entries of $\mathbf{X}$, $\mathbf{W}$ and $\mathbf{H}$ are non-negative. The well-known multiplicative update rules are as follows:

$$\begin{aligned} \mathbf{H}_{qj} &\longleftarrow \mathbf{H}_{qj} \frac{(\mathbf{W}^T\mathbf{X})_{qj}}{(\mathbf{W}^T\mathbf{WH})_{qj}} \\ \mathbf{W}_{iq} &\longleftarrow \mathbf{W}_{iq} \frac{(\mathbf{XH}^T)_{iq}}{(\mathbf{WHH}^T)_{iq}} \end{aligned} \tag{2}$$

Another criterion function of NMF is based on the Kullback-Leibler divergence, and the corresponding optimization problem is as follows:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{H}} D(\mathbf{X} \parallel \mathbf{WH}) \\ &\text{s.t.} \mathbf{W} \geq 0, \mathbf{H} \geq 0 \end{aligned} \tag{3}$$

where $D(\mathbf{X} \parallel \mathbf{WH}) = \sum_{i=1}^{m}\sum_{j=1}^{n}(\mathbf{X}_{ij} \ln \frac{\mathbf{X}_{ij}}{(\mathbf{WH})_{ij}} - \mathbf{X}_{ij} + (\mathbf{WH})_{ij})$ is the Kullback-Leibler divergence between matrices $\mathbf{X}$ and $\mathbf{WH}$.

The corresponding multiplicative update rules for solving (3) are as follows:

$$\begin{aligned} \mathbf{H}_{qj} &\longleftarrow \mathbf{H}_{qj} \frac{\sum_{i=1}^{m} \mathbf{W}_{iq}\mathbf{X}_{ij}/(\mathbf{WH})_{ij}}{\sum_{i=1}^{m} \mathbf{W}_{iq}} \\ \mathbf{W}_{iq} &\longleftarrow \mathbf{W}_{iq} \frac{\sum_{j=1}^{n} \mathbf{H}_{qj}\mathbf{X}_{ij}/(\mathbf{WH})_{ij}}{\sum_{j=1}^{n} \mathbf{H}_{qj}} \end{aligned} \tag{4}$$

The convergence of these multiplicative update rules (2) and (4) have been proved in [14].

2.2 Local non-negative matrix factorization (LNMF)

By adding penalties to NMF, LNMF was introduced in [15] to obtain a robust part-based representation. The corresponding optimization problem is as follows:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{H}} D(\mathbf{X} \parallel \mathbf{WH}) + \alpha \sum_{i,j} \mathbf{U}_{ij} - \beta \sum_{i,i} \mathbf{V}_{ii} \\ &\text{s.t.} \mathbf{W} \geq 0, \mathbf{H} \geq 0 \end{aligned} \tag{5}$$

where $\mathbf{U} = \mathbf{W}^T\mathbf{W}$, and $\mathbf{V} = \mathbf{H}\mathbf{H}^T$, α, β >0 are some constants. Minimizing $\sum_{i,j}\mathbf{U}_{ij}$ suppresses over decomposition of the basis matrix $\mathbf{W}$, while maximizing $\sum_{i,i}\mathbf{V}_{ii}$ encourages retaining components with important information.

2.3 Graph regularized non-negative matrix factorization (GNMF)

Recently, the graph regularized non-negative matrix factorization (GNMF) was proposed in [6] to encode the data geometric structure in a nearest neighbor graph. GNMF solved the following optimization problem:

$$\min_{\mathbf{W},\mathbf{H}} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \mathrm{Tr}(\mathbf{H}\mathbf{L}\mathbf{H}^T)$$
$$\text{s.t.}\mathbf{W} \geq 0, \mathbf{H} \geq 0 \tag{6}$$

where $\mathbf{L}$ is the graph Laplacian matrix [3], regularization parameter $\lambda \geq 0$ controls the smoothness of the new representation, and the Tr is the trace of matrix, i.e., the sum of matrix diagonal entries.

2.4 Manifold regularized discriminative non-negative matrix factorization (MD-NMF)

In order to incorporate the manifold regularization and the margin maximization into NMF, MD-NMF was proposed in [11]. MD-NMF solved the following optimization problem:

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{X} \parallel \mathbf{W}\mathbf{H}) + \frac{\alpha}{2}\mathrm{Tr}(\mathbf{W}\mathbf{e}\mathbf{W}^T) + \frac{\beta}{2}\mathrm{Tr}(\mathbf{H}\mathbf{H}^T)$$
$$+ \frac{\gamma}{2}\mathrm{Tr}(\mathbf{H}(\mathbf{L}_c^{-1/2})^T\mathbf{L}_g\mathbf{L}_c^{-1/2}\mathbf{H}^T)$$
$$\text{s.t.}\mathbf{W} \geq 0, \mathbf{H} \geq 0 \tag{7}$$

where α, β and γ are the trade-off parameters and they are non-negative constants. $\mathbf{e} = \bar{\mathbf{1}} - \mathbf{I}$, where $\bar{\mathbf{1}}$ signifies the matrix whose elements are all one and $\mathbf{I}$ is an identity matrix with appropriate dimensionality. $\mathbf{L}_g$ and $\mathbf{L}_c$ are two different Laplacian matrices with respect to two different adjacent graphs.

3 Graph regularized discriminative non-negative matrix factorization (GDNMF)

Inspired by the work in [6, 24], we formulate our optimization problem by adding supervised label information to the objective function of GNMF. The definition and update rules of GDNMF are given below.

3.1 GDNMF model

Recent studies in spectral graph theory [7] and manifold learning theory [2] have proved that the local geometric structure can be effectively modeled through a k nearest neighbor graph on a scatter of data points. Considering a graph with n vertices where each vertex corresponds to a data point, we seek its k nearest neighbors and put edges between each data point $\mathbf{x}_i$ and its neighbors. There are

many choices to define a weight matrix $\mathbf{C}$ on the graph, such as '0-1 weighting', 'Heat kernel weighting' and 'Dot-product weighting'. We construct our weight matrix with respect to training set by using the '0-1 weighting'. The entries of our weight matrix are defined by:

$$\mathbf{C}_{i,j} = \begin{cases} 1, & \text{if } \mathbf{x}_i \in N_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_k(\mathbf{x}_i) \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

where $N_k(\mathbf{x}_i)$ consists of k NNs of $\mathbf{x}_i$ and they have the same label as $\mathbf{x}_i$, the choices of k are discussed in our experiment studies section. $\mathbf{C} \in \mathbb{R}^{n\times n}$, and a diagonal matrix $\mathbf{B} \in \mathbb{R}^{n\times n}$ is obtained according to weight matrix $\mathbf{C}$. $\mathbf{B}_{ii}$ is the row (or equivalently column, since $\mathbf{C}$ is symmetrical) sum of the weight matrix $\mathbf{C}$, i.e., $\mathbf{B}_{ii} = \sum_{j=1}^{n} \mathbf{C}_{ij}$. $\mathbf{L} = \mathbf{B} - \mathbf{C}$ is called graph Laplacian matrix and $\mathbf{L} \in \mathbb{R}^{n\times n}$.

We also define another class indicator matrix $\mathbf{S} \in \mathbb{R}^{c\times n}$ and give the definition:

$$\mathbf{S}_{i,j} = \begin{cases} 1, & \text{if } y_j = i, \ \ j = 1, 2, \cdots, n, \ \ i = 1, 2, \cdots, c \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where $y_j \in \{1, 2, \cdots, c\}$ denotes the class label of the *jth* sample $\mathbf{x}_j$ and c is the total number of classes in training set $\mathbf{X}$.

The GDNMF solves the following optimization problem:

$$\min_{\mathbf{W},\mathbf{H},\mathbf{A}} \|\mathbf{X} - \mathbf{WH}\|_F^2 + \lambda \mathrm{Tr}(\mathbf{HLH}^T) + \gamma \|\mathbf{S} - \mathbf{AH}\|_F^2$$
$$\text{s.t.} \mathbf{W} \geq 0, \mathbf{H} \geq 0, \mathbf{A} \geq 0 \tag{10}$$

where $\mathbf{A} \in \mathbb{R}^{c\times r}$ is a non-negative matrix and is initialized randomly in our algorithm, λ and γ are the non-negative regularization parameters.

In essence, GDNMF is a kind of novel supervised nonnegative matrix factorization. We explicitly incorporate the graph Laplacian and label information into the cost function of NMF. The second term and third term in (10) guarantee that the learned bases can retain the intrinsic geometrical structure of data and have discriminative ability respectively. Thus the learned bases are not only consistent with the intrinsic geometrical structure but also with the discriminative power.

3.2 The Update Rules of GDNMF

Though the objective function in (10) is not jointly convex in the pair $(\mathbf{W}, \mathbf{H}, \mathbf{A})$, it is convex with respect to one variable in the $(\mathbf{W}, \mathbf{H}, \mathbf{A})$ while fixing the others. Thus, we extend the multiplicative update rules based on Frobenius norm of the original NMF [14], aiming to find a local optimum. The objective function (10) can be written as:

$$\begin{aligned} J &= \mathrm{Tr}((\mathbf{X} - \mathbf{WH})^T(\mathbf{X} - \mathbf{WH})) \\ &\quad + \lambda \mathrm{Tr}(\mathbf{HLH}^T) + \gamma \,\mathrm{Tr}((\mathbf{S} - \mathbf{AH})^T(\mathbf{S} - \mathbf{AH})) \\ &= \mathrm{Tr}(\mathbf{X}^T\mathbf{X}) - 2\mathrm{Tr}(\mathbf{X}^T\mathbf{WH}) + \mathrm{Tr}(\mathbf{H}^T\mathbf{W}^T\mathbf{WH}) \\ &\quad + \gamma \,\mathrm{Tr}(\mathbf{S}^T\mathbf{S}) - 2\gamma \,\mathrm{Tr}(\mathbf{S}^T\mathbf{AH}) + \gamma \,\mathrm{Tr}(\mathbf{H}^T\mathbf{A}^T\mathbf{AH}) \\ &\quad + \lambda \mathrm{Tr}(\mathbf{HLH}^T), \end{aligned} \tag{11}$$

where we apply the matrix properties $\mathrm{Tr}(\mathbf{XY}) = \mathrm{Tr}(\mathbf{YX})$ and $\mathrm{Tr}(\mathbf{X}) = \mathrm{Tr}(\mathbf{X}^T)$. Let Φ, Ψ and Ω be the Lagrange multiplier for $\mathbf{W}$, $\mathbf{H}$ and $\mathbf{A}$, respectively. Then the corresponding Lagrange function form of J is:

$$\begin{aligned} L_f = & \mathrm{Tr}(\mathbf{X}^T\mathbf{X}) - 2\mathrm{Tr}(\mathbf{X}^T\mathbf{WH}) + \mathrm{Tr}(\mathbf{H}^T\mathbf{W}^T\mathbf{WH}) \\ & + \gamma\,\mathrm{Tr}(\mathbf{S}^T\mathbf{S}) - 2\gamma\,\mathrm{Tr}(\mathbf{S}^T\mathbf{AH}) + \gamma\,\mathrm{Tr}(\mathbf{H}^T\mathbf{A}^T\mathbf{AH}) \\ & + \lambda\mathrm{Tr}(\mathbf{HLH}^T) + \mathrm{Tr}(\Phi^T\mathbf{W}) + \mathrm{Tr}(\Psi^T\mathbf{H}) + \mathrm{Tr}(\Omega^T\mathbf{A}) \end{aligned} \tag{12}$$

The partial derivatives of L_f with respect to $\mathbf{H}$, $\mathbf{W}$ and $\mathbf{A}$ respectively are:

$$\frac{\partial L_f}{\partial \mathbf{H}} = 2\mathbf{W}^T(\mathbf{WH} - \mathbf{X}) + 2\lambda\mathbf{HL} + 2\gamma\,\mathbf{A}^T(\mathbf{AH} - \mathbf{S}) + \Psi = 0 \tag{13}$$

$$\frac{\partial L_f}{\partial \mathbf{W}} = 2\mathbf{WHH}^T - 2\mathbf{XH}^T + \Phi = 0 \tag{14}$$

$$\frac{\partial L_f}{\partial \mathbf{A}} = -2\gamma\,\mathbf{SH}^T + 2\gamma\,\mathbf{AHH}^T + \Omega = 0 \tag{15}$$

From the KKT conditions $\Psi_{qj}\mathbf{H}_{qj} = 0$, $\Phi_{iq}\mathbf{W}_{iq} = 0$ and $\Omega_{kq}\mathbf{A}_{kq} = 0$, we can get the following equations:

$$[2\mathbf{W}^T(\mathbf{WH} - \mathbf{X}) + 2\lambda\mathbf{H}(\mathbf{B} - \mathbf{C}) + 2\gamma\,\mathbf{A}^T(\mathbf{AH} - \mathbf{S})]_{qj}\mathbf{H}_{qj} + \Psi_{qj}\mathbf{H}_{qj} = 0 \tag{16}$$

$$(2\mathbf{WHH}^T - 2\mathbf{XH}^T)_{iq}\mathbf{W}_{iq} + \Phi_{iq}\mathbf{W}_{iq} = 0 \tag{17}$$

$$(-2\gamma\,\mathbf{SH}^T + 2\gamma\,\mathbf{AHH}^T)_{kq}\mathbf{A}_{kq} + \Omega_{kq}\mathbf{A}_{kq} = 0 \tag{18}$$

Therefore, we have the following updating rules for $\mathbf{H}$, $\mathbf{W}$, $\mathbf{A}$.

$$\mathbf{H}_{qj} \longleftarrow \mathbf{H}_{qj}\frac{(\gamma\,\mathbf{A}^T\mathbf{S} + \mathbf{W}^T\mathbf{X} + \lambda\mathbf{HC})_{qj}}{(\mathbf{W}^T\mathbf{WH} + \gamma\,\mathbf{A}^T\mathbf{AH} + \lambda\mathbf{HB})_{qj}} \tag{19}$$

$$\mathbf{W}_{iq} \longleftarrow \mathbf{W}_{iq}\frac{(\mathbf{XH}^T)_{iq}}{(\mathbf{WHH}^T)_{iq}} \tag{20}$$

$$\mathbf{A}_{kq} \longleftarrow \mathbf{A}_{kq}\frac{(\mathbf{SH}^T)_{kq}}{(\mathbf{AHH}^T)_{kq}} \tag{21}$$

Table 1 The algorithm of graph regularized discriminative non-negative matrix factorization (GDNMF)

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$, graph Laplacian matrix $\mathbf{L} \in \mathbb{R}^{n\times n}$, indicator matrix $\mathbf{S} \in \mathbb{R}^{c\times n}$, parameters λ, γ, r

Initialization: Randomly initialize three non-negative matrices $\mathbf{W} \in \mathbb{R}^{m\times r}$, $\mathbf{H} \in \mathbb{R}^{r\times n}$ and $\mathbf{A} \in \mathbb{R}^{c\times r}$

Repeat

1. Update $\mathbf{H}$ by rule (19)
2. Update $\mathbf{W}$ by rule (20)
3. Update $\mathbf{A}$ by rule (21)

Until Convergence

Output: Basis matrix $\mathbf{W}$

Table 2 Statistics of the four datasets

Dataset	Number of samples	Dimensionality	Number of classes
Yale	165	1,600	15
ORL	400	1,024	40
UMIST	575	1,600	20
PIE	1,428	1,024	68

Regarding these three updating rules, we have the following theorem and this theorem is proved in our Appendix.

Theorem *The objective function* $O_1 = \|\mathbf{X} - \mathbf{WH}\|_F^2 + \lambda \mathrm{Tr}(\mathbf{HLH}^T) + \gamma \|\mathbf{S} - \mathbf{AH}\|_F^2$ *in* (10) *is nonincreasing under the updating rules in* (19), (20) *and* (21).

We can iteratively update **W**, **H**, and **A** until the objective value of O_1 does not change or the number of iteration exceed the maximum value. The procedure is depicted in Table 1.

4 Experimental results

In this section, we compare the proposed GDNMF with four representative algorithms, which are NMF [14], LNMF [15], GNMF [6] and MD-NMF [11], on four face datasets, i.e., Yale [1], ORL [19], UMIST [10] and PIE [20]. The important statistics of the four datasets are summarized in Table 2. Figure 1 shows example images of Yale, ORL, UMIST and PIE datasets. All the face images used in our experiments are manually aligned and cropped. Each face image is represented as a column vector and the features (pixel values) are then scaled to [0,1] (divided by 255). We randomly select (3, 5, 7) images from each subject as training set and the rest as test set. The training set is used to learn basis matrix for the low-dimensional space. The test set is utilized to report the accuracy of face recognition in the obtained low-dimensional space. The accuracy is calculated as the percentage of samples in the test set that are correctly classified using the k nearest neighbor rule. In all our experiments, we employ 1-nearest neighbor classifier. During the process of calculating the basis matrix **W**, all the variants of NMF and original NMF

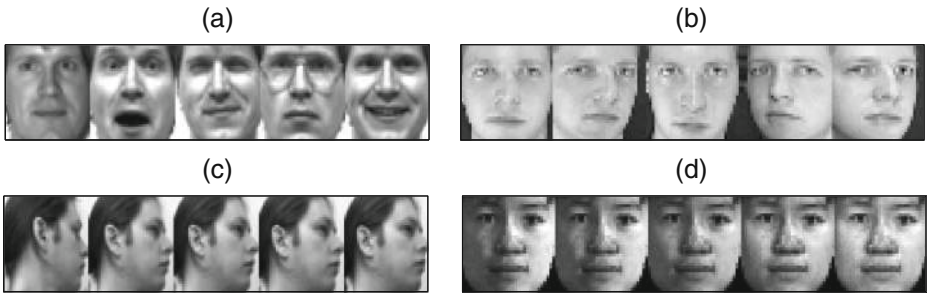

Fig. 1 Face image examples of the **a** Yale, **b** ORL, **c** UMIST, and **d** PIE datasets

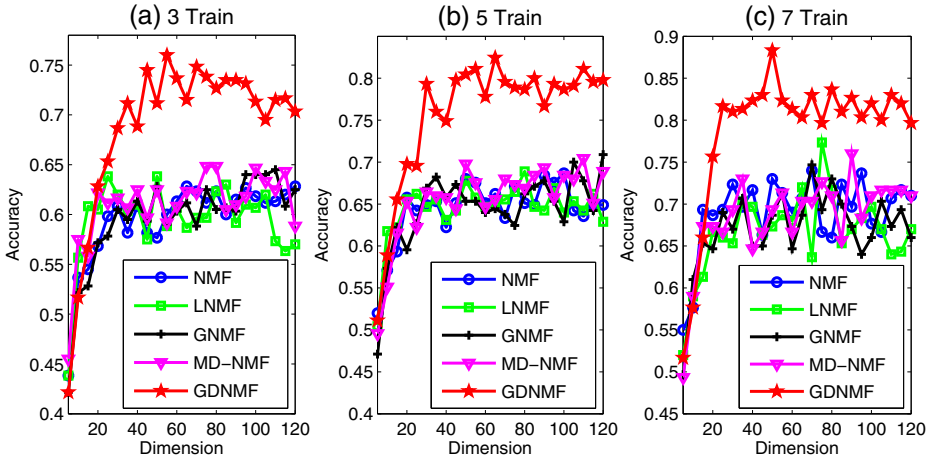

Fig. 2 Face recognition accuracy on the Yale dataset. We randomly selected **a** three, **b** five, and **c** seven images from each person to train a model and used the rest images in the dataset for test, so we had three types of partitions on this dataset. We conducted five trials for each partition and compared the performance of different algorithms based on the averaged accuracy of the five trials on each dimension for each type of the partition

algorithms are repeated 300 iterations. Furthermore, similar to Guan et al. [11], the projection matrices in all algorithms based on NMF are $(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T$ rather than $\mathbf{W}$. These trials are independently conducted five times and the averaged accuracy is reported. In addition, it is worth noting that we use multiplicative update rules for all the variants of NMF and original NMF in order to get a fair time comparison. All experiments are conducted in MATLAB, which is executed on a server with an Intel X5650 CPU (2.66GHz and 12 cores) and 32GB RAM.

4.1 Yale face dataset

The Yale face dataset [1] contains 165 grayscale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. Each image is normalized to 40×40 pixel

Table 3 Best accuracy on the Yale dataset

Algorithm	3 Train	5 Train	7 Train
NMF [13]	0.6283(65)	0.6867(100)	0.7400(70)
LNMF [15]	0.6383(50)	0.6889(80)	0.7733(75)
GNMF [6]	0.6450(110)	0.7089(120)	0.7467(70)
MD-NMF [11]	0.6483(75)	0.7044(110)	0.7600(90)
GDNMF	**0.7600(55)**	**0.8244(65)**	**0.8833(50)**

Bold entries denote the correct classification accuracy and the corresponding dimensionality of our GDNMF algorithm. The values outside the brackets represent the classification accuracy and the values in the brackets show the corresponding dimensionality. The bold fonts are used to emphasize that our GDNMF algorithm is better than other several methods in most cases

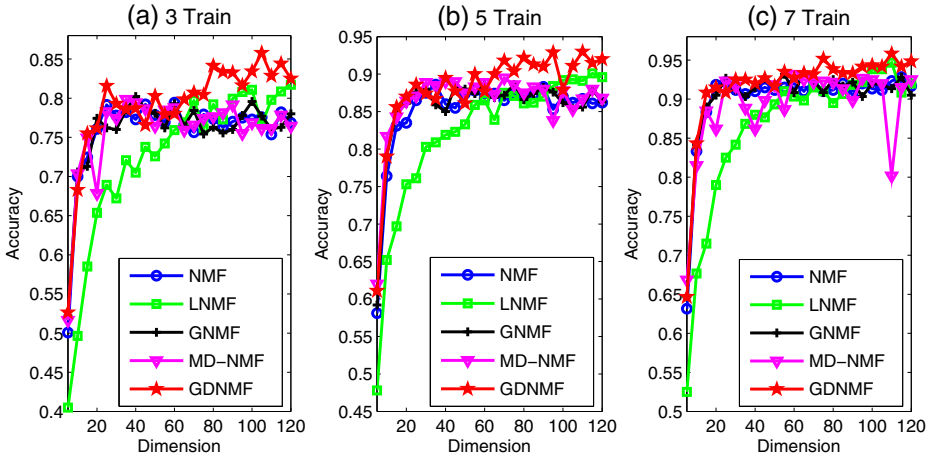

Fig. 3 Face recognition accuracy on the ORL dataset. We randomly selected **a** three, **b** five, and **c** seven images from each person to train a model and used the rest images in the dataset for test, so we had three types of partitions on this dataset. We conducted five trials for each partition and compared the performance of different algorithms based on the averaged accuracy of the five trials on each dimension for each type of the partition

array and reshaped to a vector. Figure 2 gives that the average accuracy versus the dimension of the subspace. According to Fig. 2a–c, it can be seen that GDNMF outperforms all other algorithms in most situations. Table 3 shows the best accuracy and corresponding dimension of all the algorithms. From Table 3, we can see that our GDNMF is ten percentage points higher than the other best algorithm.

4.2 ORL face dataset

The ORL face dataset [19] consists 400 images of 40 different subjects in PGM format. Each subject has 10 images. Subjects were asked to face the camera and no restrictions were imposed on expression; only limited side movement and limited tilt were tolerated. For most subjects, the images were shot at different times and with different lighting conditions, but all the images were taken against a dark homogeneous background. Some subjects were captured with and without glass.

Table 4 Best accuracy on the ORL dataset

Algorithm	3 Train	5 Train	7 Train
NMF [13]	0.7943(60)	0.8860(35)	0.9283(115)
LNMF [15]	0.8171(120)	0.9010(115)	0.9467(110)
GNMF [6]	0.8021(40)	0.8870(60)	0.9333(65)
MD-NMF [11]	0.7986(35)	0.8950(60)	0.9317(60)
GDNMF	**0.8579(105)**	**0.9300(110)**	**0.9583(110)**

Bold entries denote the correct classification accuracy and the corresponding dimensionality of our GDNMF algorithm. The values outside the brackets represent the classification accuracy and the values in the brackets show the corresponding dimensionality. The bold fonts are used to emphasize that our GDNMF algorithm is better than other several methods in most cases

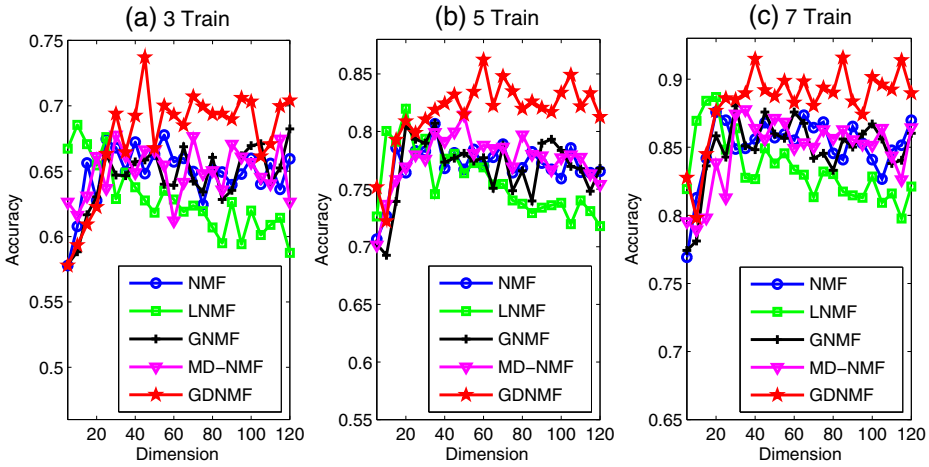

Fig. 4 Face recognition accuracy on the UMIST dataset. We randomly selected **a** three, **b** five, and **c** seven images from each person to train a model and used the rest images in the dataset for test, so we had three types of partitions on this dataset. We conducted five trials for each partition and compared the performance of different algorithms based on the averaged accuracy of the five trials on each dimension for each type of the partition

Each image is normalized to 32×32 pixel array and reshaped to a vector. Figure 3 displays that the average accuracy versus the dimension of the subspace. Table 4 illustrates the best accuracy and corresponding dimension of all the algorithms.

4.3 UMIST face dataset

The UMIST face dataset [10] contains 575 images in PGM format of 20 people. Each people covering a range of poses from profile to frontal views. Each image is normalized to 40×40 pixel array and reshaped to a vector. Figure 4 shows that the average accuracy versus the dimension of the subspace. From Fig. 4a–c, we can find that our GDNMF outperforms all other algorithms in most situations. Table 5 depicts the best accuracy and corresponding dimension of all the algorithms.

Table 5 Best accuracy on the UMIST dataset

Algorithm	3 Train	5 Train	7 Train
NMF [13]	0.6777(55)	0.8067(35)	0.8759(20)
LNMF [15]	0.6854(10)	0.8198(20)	0.8869(20)
GNMF [6]	0.6823(120)	0.8072(35)	0.8805(30)
MD-NMF [11]	0.6777(30)	0.8143(50)	0.8777(35)
GDNMF	**0.7371(45)**	**0.8623(60)**	**0.9159(85)**

Bold entries denote the correct classification accuracy and the corresponding dimensionality of our GDNMF algorithm. The values outside the brackets represent the classification accuracy and the values in the brackets show the corresponding dimensionality. The bold fonts are used to emphasize that our GDNMF algorithm is better than other several methods in most cases

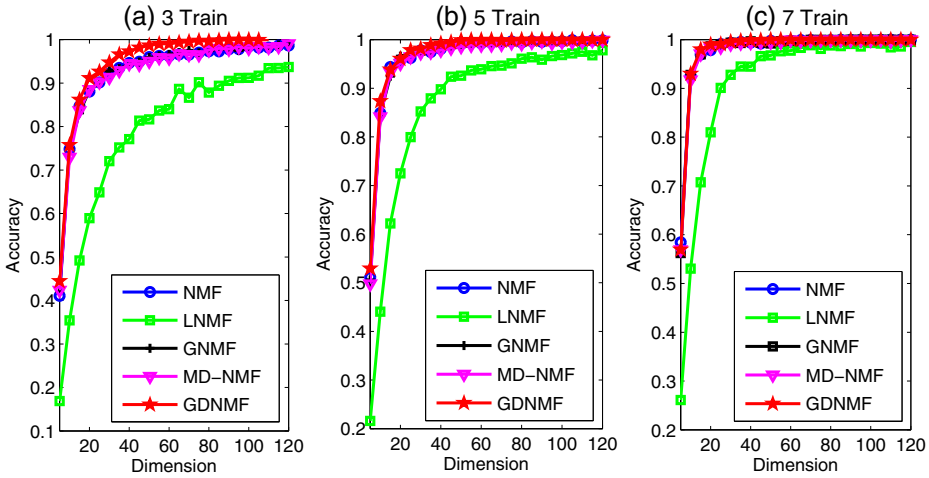

Fig. 5 Face recognition accuracy on the PIE dataset. We randomly selected **a** three, **b** five, and **c** seven images from each person to train a model and used the rest images in the dataset for test, so we had three types of partitions on this dataset. We conducted five trials for each partition and compared the performance of different algorithms based on the averaged accuracy of the five trials on each dimension for each type of the partition

4.4 PIE face dataset

The CMU PIE face dataset [20] contains 68 human subjects with 41,368 face images as a whole. Each person under 13 different poses, 43 different illumination conditions, and with 4 different expressions. In our experiments, a subset of images with pose ID C27 and different illumination conditions is used, and thus, there are 21 images for each subject. Each image is normalized to 32×32 pixel array and reshaped to a vector. The average accuracy versus the dimension of the subspace is shown in Fig. 5. Table 6 gives the best accuracy and corresponding dimension of all the algorithms.

Table 6 Best accuracy on the PIE dataset

Algorithm	3 Train	5 Train	7 Train
NMF [13]	0.9864(120)	0.9989(105)	**1.00(110)**
LNMF [15]	0.9369(120)	0.9781(120)	0.9952(120)
GNMF [6]	0.9884(120)	0.9983(105)	**1.00(90)**
MD-NMF [11]	0.9907(120)	0.9989(120)	**1.00(120)**
GDNMF	**1.00(110)**	**1.00(80)**	**1.00(50)**

Bold entries denote the correct classification accuracy and the corresponding dimensionality of our GDNMF algorithm. The values outside the brackets represent the classification accuracy and the values in the brackets show the corresponding dimensionality. The bold fonts are used to emphasize that our GDNMF algorithm is better than other several methods in most cases

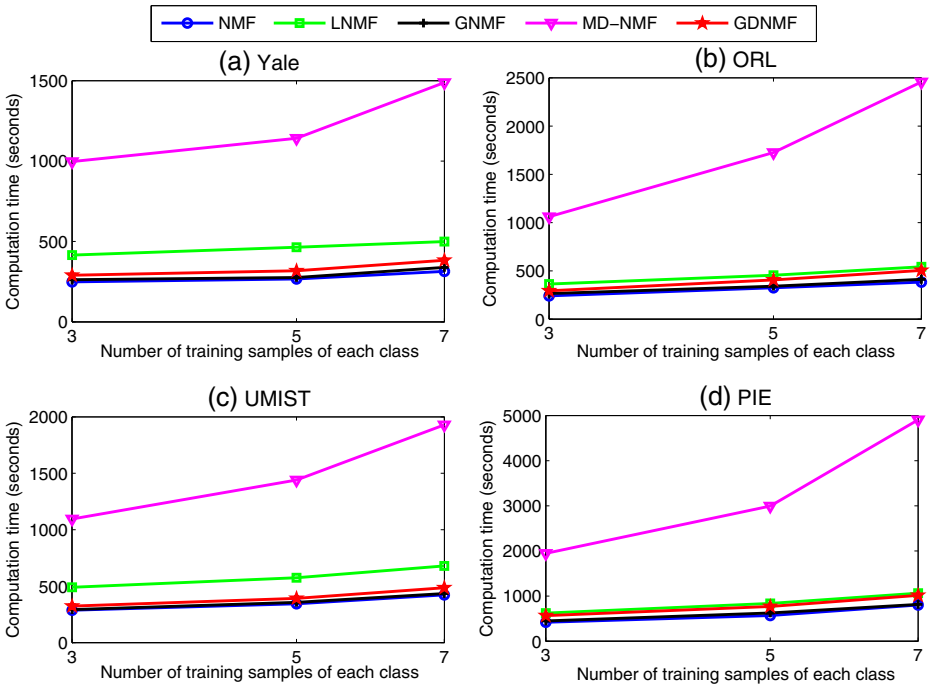

Fig. 6 Computation time versus the number of training samples of each class on the **a** Yale, **b** ORL, **c** UMIST, and **d** PIE

4.5 Computation time and computational complexity analysis

In order to have a fair time comparison, in our experiments, we only utilize multiplicative update rules to solve all the optimization problems appeared in several variants of NMF including NMF, LNMF, GNMF, MD-NMF and GDNMF. Figure 6 illustrates that the computation time versus the number of training samples of each class on the four datasets. The computation time is the total time of five trials for each types of partitions. In each trial, the dimensionality of basis matrix changes from 5 to 120 and the increasing step size is 5. From the comparison results reported in Fig. 6,

Table 7 Computational operation counts for each iteration in NMF and GDNMF

Algorithm	NMF
Addition	$2mnr + 2(m+n)r^2$
Multiplication	$2mnr + 2(m+n)r^2 + (m+n)r$
Division	$(m+n)r$
Overall	$O(mnr)$
Algorithm	GDNMF
Addition	$2mnr + (2m+4n+2c)r^2 + (2n+kn+cp)r$
Multiplication	$2mnr + (2m+4n+2c)r^2 +$ $(3n+kn+m+cp+c)r$
Division	$(m+n+c)r$
Overall	$O(mnr)$

Table 8 Parameters used in computational complexity analysis

Parameters	Description
m	Number of features
n	Number of data points
r	Number of factors
c	Number of class
p	Number of training samples obtained from each class
k	Number of nearest neighbor

we can see that the computation time of GDNMF is slightly higher than NMF and GNMF, but much lower than MD-NMF and slightly lower than LNMF. This reason is that the model of MD-NMF is more complex than our model.

Furthermore, we give the computational complexity analysis of standard NMF and our GDNMF based on the multiplicative update rules. Specifically, we analyze the computational operation counts for each iteration in formulas (2) and (19)–(21). According to the update rules, we can easily count the arithmetic operations of

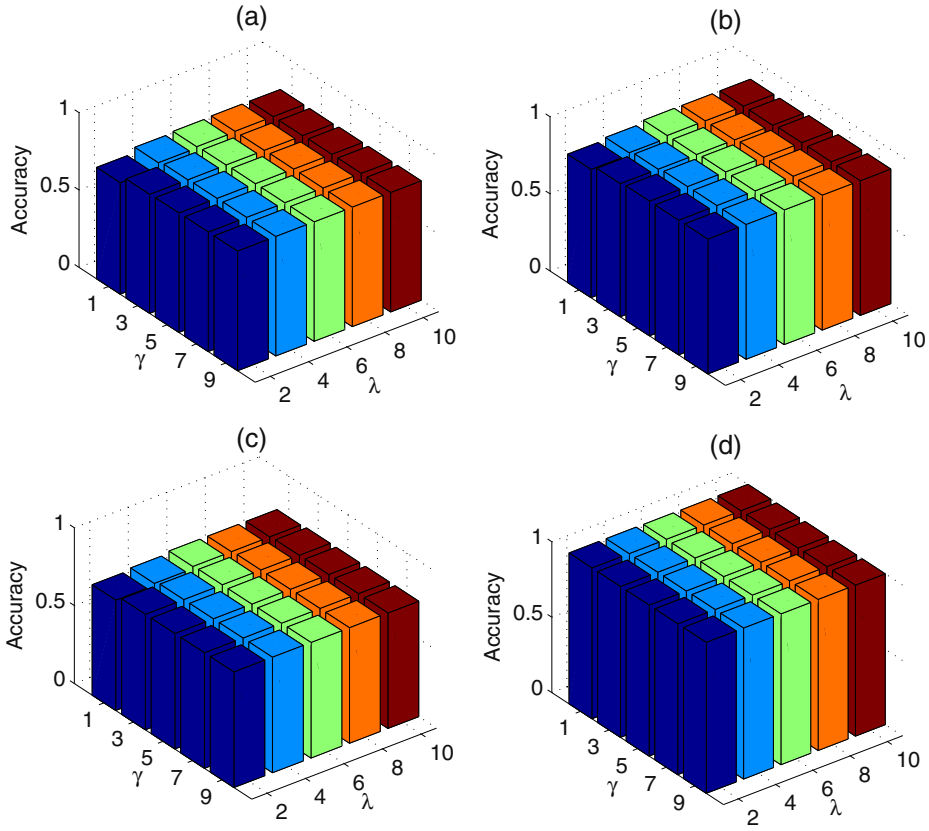

Fig. 7 Face recognition accuracy of our GDNMF under different λ and γ on the **a** Yale, **b** ORL, **c** UMIST, and **d** PIE

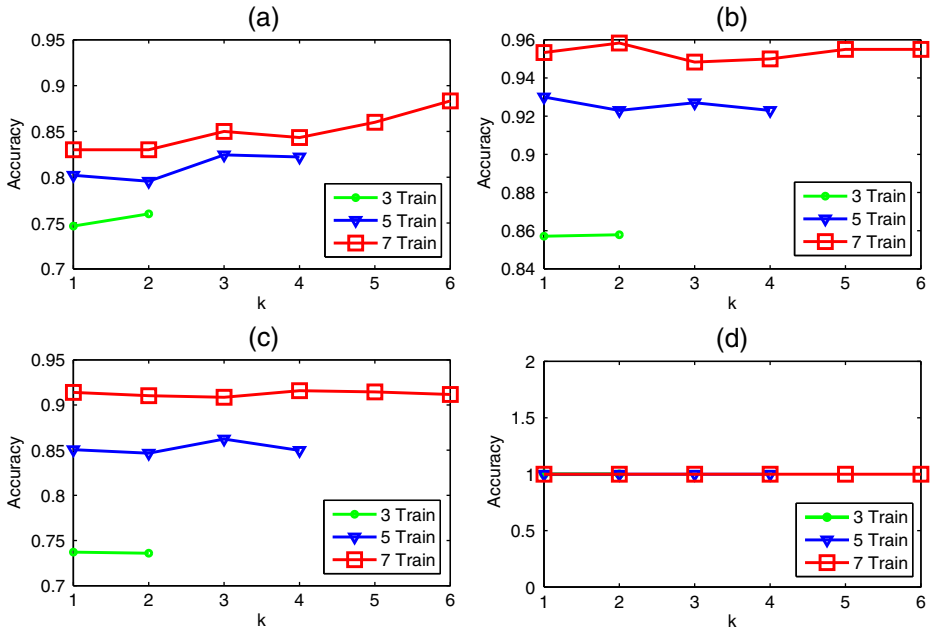

Fig. 8 Face recognition accuracy of our GDNMF under different k on the **a** Yale, **b** ORL, **c** UMIST, and **d** PIE

each iteration in NMF. The results are summarized in Table 7. For GDNMF, it is noteworthy that $\mathbf{C}$ is a sparse matrix. If we use a k nearest neighbor graph, there are k nonzero elements on each column of $\mathbf{C}$. Thus, we only need knr addition and multiplication to compute $\mathbf{HC}$. For indicator matrix $\mathbf{S}$, there are one nonzero element on each column of $\mathbf{S}$ and p nonzero elements on each row of $\mathbf{S}$. So, we need nr addition and multiplication to compute $\mathbf{A}^T\mathbf{S}$, and need cpr addition and multiplication to compute $\mathbf{SH}^T$. The arithmetic operations of GDNMF are also given in Table 7. Meanwhile, the explanation of parameters used in Table 7 are presented in Table 8.

Besides the multiplicative updates, GDNMF also needs $O(p^2m)$ to construct the weight matrix $\mathbf{C}$ and needs $O(pc)$ to construct the indicator matrix $\mathbf{S}$. Suppose the multiplicative updates stop after t iterations, the overall cost for NMF is $O(tmnr)$ and the overall cost for GDNMF is $O(tmnr + p^2m + pc)$.

4.6 Parameter selection

We also utilize the same way proposed in Guan et al. [11] to select our parameters in our proposed algorithm. More specifically, in the proposed GDNMF (10), there are

Table 9 k setting on Yale, ORL, UMIST, and PIE datasets

Dataset	Yale			ORL			UMIST			PIE		
Partition	3	5	7	3	5	7	3	5	7	3	5	7
k	2	3	6	2	1	2	1	3	4	2	4	6

two regularization parameters, i.e., λ and γ. It is time-consuming to select all these parameters based on the grid search. Fortunately, λ and γ affect the performance slightly if they are set in feasible ranges. In our experiments, for the Yale, ORL, UMIST and PIE datasets, we randomly select three images from each class to form the training set and the rest as test set. These trails are independently performed five times, and the average recognition accuracy is reported. Figure 7 presents the face recognition accuracy of our GDNMF under different λ and γ on the four face datasets. The classification accuracies do not change when each of λ and γ is selected from a range. The proposed GDNMF model (10) is stable when varying λ and γ within (4, 10) and (3, 9) respectively. In all the above experiments, we set $\lambda = 6$ and $\gamma = 5$ in our GDNMF algorithm.

Another important parameter in our GDNMF algorithm is the k used in (8) to construct the weight matrix $\mathbf{C}$. We study its effect on classification accuracy on the four datasets, we randomly select (3, 5, 7) images from each class as training set and use the remainder images for test. Similar to Guan et al. [11], k varies from 1 to $(TN)/(CN) - 1$ wherein TN is the training samples number and CN is the class number. Figure 8 demonstrates that the face recognition accuracy of our GDNMF under different k on the four face datasets. Table 9 presents the optimal value of k with which the best classification accuracy is achieved on the Yale, ORL, UMIST and PIE datasets respectively. It is worth noting that the face recognition accuracies of our GDNMF under different k are equal to 1 on the PIE dataset, in our proposed algorithm, we set $k = (TN)/(CN) - 1$ for PIE.

5 Conclusions

By introducing the geometrical structure and discriminative information of data, we have presented an efficient graph regularized discriminative non-negative matrix factorization. As a result, GDNMF can have more discriminative power than the conventional NMF and its several variants. Further, we show the corresponding multiplicative update rules and convergence studies. Evaluations on four datasets have revealed both higher recognition accuracy and lower time complexity of the proposed algorithm in comparison to those of the state-of-the-art algorithms.

Acknowledgements This work is supported in part by the National Basic Research Program of China (973 program) under Grant 2009CB320901, the National Natural Science Foundation of China under Grant 61272247, the National High Technology Research and Development Program of China (863 program) under Grant 2008AA02Z310, the European Union Seventh Framework Programme under Grant 247619, the Shanghai Committee of Science and Technology under Grant 08411951200, and the Innovation Ability Special Fund of Shanghai Jiao Tong University under Grant Z030026.

Appendix (Proof of Theorem)

In order to prove Theorem, we need to show that O_1 is non-increasing under the updating steps in (19), (20) and (21). For the objective function O_1, we need to fix $\mathbf{H}$ and $\mathbf{A}$ if we update $\mathbf{W}$, so, the first term of O_1 exists. Similarly, we need to fix $\mathbf{W}$ and $\mathbf{H}$ if we update $\mathbf{A}$, the third term of O_1 exists. Therefore, we have exactly the same update formula for $\mathbf{W}$ and $\mathbf{A}$ in GDNMF as in the original NMF. Thus, we can use

the convergence proof of NMF to show that O_1 is nonincreasing under the update step in (20) and (21). These details can be found in [14].

Hence, we only need to prove that O_1 is non-increasing under the updating step in (19). We follow the similar process depicted in [14]. Our proof make use of an auxiliary function similar to that used in the Expectation-Maximization algorithm [8]. We first give the definition of the auxiliary function.

Definition $G(h, h^{'})$ is an auxiliary function of $F(h)$ if the following conditions are satisfied.

$$G(h, h^{'}) \geq F(h), \quad G(h, h) = F(h) \tag{22}$$

The above auxiliary function is very important because of the following lemma.

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

$$h^{(t+1)} = \arg\min_{h} G(h, h^{(t)}) \tag{23}$$

Proof $F(h^{(t+1)}) \leq G(h^{(t+1)}, h^{(t)}) \leq G(h^{(t)}, h^{(t)}) = F(h^{(t)})$

Now, we show that the update step for $\mathbf{H}$ in (19) is exactly the update in (23) with a proper auxiliary function.

Considering any element h_{ab} in $\mathbf{H}$, we use F_{ab} to denote the part of O_1 which is only relevant to h_{ab}. It is easy to obtain the following derivatives.

$$F^{'}_{ab} = \left(\frac{\partial O_1}{\partial \mathbf{H}}\right)_{ab} = [2\mathbf{W}^T(\mathbf{WH} - \mathbf{X}) + 2\lambda\mathbf{H}(\mathbf{B} - \mathbf{C}) + 2\gamma\,\mathbf{A}^T(\mathbf{AH} - \mathbf{S})]_{ab} \tag{24}$$

$$F^{''}_{ab} = 2(\mathbf{W}^T\mathbf{W})_{aa} + 2\lambda\mathbf{B}_{bb} - 2\lambda\mathbf{C}_{bb} + 2\gamma\,(\mathbf{A}^T\mathbf{A})_{aa} \tag{25}$$

It is enough to show that each F_{ab} is nonincreasing under the update step of (19) because our update is essentially element-wise. Consequently, we introduce the following lemma. □

Lemma 2 *Function*

$$\begin{aligned} G(h, h^{(t)}_{ab}) = {} & F_{ab}(h^{(t)}_{ab}) + F^{'}_{ab}(h^{(t)}_{ab})(h - h^{(t)}_{ab}) \\ & + \frac{(\mathbf{W}^T\mathbf{WH} + \gamma\,\mathbf{A}^T\mathbf{AH} + \lambda\mathbf{HB})_{ab}}{h^{(t)}_{ab}}(h - h^{(t)}_{ab})^2 \end{aligned} \tag{26}$$

is an auxiliary function of F_{ab}.

Proof We only need to prove that $G(h, h^{(t)}_{ab}) \geq F_{ab}(h)$ because $G(h, h) = F_{ab}(h)$ is obvious. Therefore, we first consider the Taylor series expansion of $F_{ab}(h)$.

$$\begin{aligned} F_{ab}(h) = {} & F_{ab}(h^{(t)}_{ab}) + F^{'}_{ab}(h^{(t)}_{ab})(h - h^{(t)}_{ab}) \\ & + [(\mathbf{W}^T\mathbf{W})_{aa} + \lambda\mathbf{B}_{bb} - \lambda\mathbf{C}_{bb} + \gamma\,(\mathbf{A}^T\mathbf{A})_{aa}](h - h^{(t)}_{ab})^2 \end{aligned} \tag{27}$$

We compare the (27) with (26) to find that $G(h, h_{ab}^{(t)}) \geq F_{ab}(h)$ is equivalent to

$$\frac{(\mathbf{W}^T\mathbf{W}\mathbf{H} + \gamma\,\mathbf{A}^T\mathbf{A}\mathbf{H} + \lambda\mathbf{H}\mathbf{B})_{ab}}{h_{ab}^{(t)}} \geq (\mathbf{W}^T\mathbf{W})_{aa} + \lambda\mathbf{B}_{bb} - \lambda\mathbf{C}_{bb} + \gamma\,(\mathbf{A}^T\mathbf{A})_{aa} \tag{28}$$

In fact, we have

$$\begin{aligned}(\mathbf{W}^T\mathbf{W}\mathbf{H} + \gamma\,\mathbf{A}^T\mathbf{A}\mathbf{H})_{ab} &= \sum_{q=1}^{r}(\mathbf{W}^T\mathbf{W})_{aq}h_{qb}^{(t)} + \gamma\sum_{q=1}^{r}(\mathbf{A}^T\mathbf{A})_{aq}h_{qb}^{(t)} \\ &\geq (\mathbf{W}^T\mathbf{W})_{aa}h_{ab}^{(t)} + \gamma\,(\mathbf{A}^T\mathbf{A})_{aa}h_{ab}^{(t)}\end{aligned} \tag{29}$$

and

$$(\lambda\mathbf{H}\mathbf{B})_{ab} = \lambda\sum_{j=1}^{n}h_{aj}^{(t)}\mathbf{B}_{jb} \geq \lambda h_{ab}^{(t)}\mathbf{B}_{bb} \geq \lambda h_{ab}^{(t)}\mathbf{B}_{bb} - \lambda h_{ab}^{(t)}\mathbf{C}_{bb} \tag{30}$$

Thus, (28) holds and $G(h, h_{ab}^{(t)}) \geq F_{ab}(h)$. We can now demonstrate the convergence of Theorem: □

Proof of Theorem Replacing $G(h, h_{ab}^{(t)})$ in (23) by (26) results in the following update rule:

$$\begin{aligned}h_{ab}^{(t+1)} &= h_{ab}^{(t)} - h_{ab}^{(t)}\frac{F_{ab}^{'}(h_{ab}^{(t)})}{2(\mathbf{W}^T\mathbf{W}\mathbf{H} + \gamma\,\mathbf{A}^T\mathbf{A}\mathbf{H} + \lambda\mathbf{H}\mathbf{B})_{ab}} \\ &= h_{ab}^{(t)}\frac{(\gamma\,\mathbf{A}^T\mathbf{S} + \mathbf{W}^T\mathbf{X} + \lambda\mathbf{H}\mathbf{C})_{ab}}{(\mathbf{W}^T\mathbf{W}\mathbf{H} + \gamma\,\mathbf{A}^T\mathbf{A}\mathbf{H} + \lambda\mathbf{H}\mathbf{B})_{ab}}\end{aligned} \tag{31}$$

Since (26) is an auxiliary function and F_{ab} is nonincreasing under this update rule. □

References

1. Belhumeur PN, Hespanha JP, Kriegman DJ (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell 19(7):711–720
2. Belkin M, Niyogi P (2001) Laplacian eigenmaps and spectral techniques for embedding and clustering. Adv Neural Inform Process Syst 14:585–591
3. Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15(6):1373–1396
4. Berry M, Browne M, Langville A, Pauca V, Plemmons R (2007) Algorithms and applications for approximate nonnegative matrix factorization. Comput Stat Data Anal 52(1):155–173
5. Brunet J-P, Tamayo P, Golub TR, Mesirov JP (2004) Metagenes and molecular pattern discovery using matrix factorization. Proc Natl Acad Sci 101(12):4164–4169
6. Cai D, He X, Han J, Huang T (2011) Graph regularized nonnegative matrix factorization for data representation. IEEE Trans Pattern Anal Mach Intell 33(8):1548–1560
7. Chung FRK (1997) Spectral graph theory (Conference Board of the Mathematical Sciences, Regional Conference Series in Mathematics, No. 92), American Mathematical Society
8. Demrsrsa A, Lamb N, Rubin D (1977) Maximum likelihood from incomplete data via the em algorithm. J Roy Stat Soc Ser B (Methodological) 39(1):1–38
9. Donoho DL, Stodden VC (2004) When does non-negative matrix factorization give a correct decomposition into parts? In: Advances in neural information processing systems 16: proceedings of the 2003 conference. MIT Press

10. Graham DB, Allinson NM (1998) Characterising virtual eigensignatures for general purpose face recognition. In: Face recognition. Springer, pp 446–456
11. Guan N, Tao D, Luo Z, Yuan B (2011) Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent. IEEE Trans Image Process 20(7):2030–2048
12. Hadsell R, Chopra S, LeCun Y (2006) Dimensionality reduction by learning an invariant mapping. In: IEEE conference on computer vision and pattern recognition, vol 2, pp 1735–1742
13. Lee D, Seung H (1999) Learning the parts of objects by non-negative matrix factorization. Nature 401(6755):788–791
14. Lee D, Seung H (2001) Algorithms for non-negative matrix factorization. Adv Neural Inform Process Syst 13:556–562
15. Li S, Hou X, Zhang H, Cheng Q (2001) Learning spatially localized, parts-based representation. In: IEEE conference on computer vision and pattern recognition, vol 1, pp I–207
16. Logothetis N, Sheinberg D (1996) Visual object recognition. Ann Rev Neurosci 19(1):577–621
17. Palmer S (1977) Hierarchical structure in perceptual representation. Cognitive Psychol 9(4):441–474
18. Roweis S, Saul L (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326
19. Samaria FS, Harter AC (1994) Parameterisation of a stochastic model for human face identification. In: Proceedings of the second IEEE workshop on applications of computer vision, pp 138–142
20. Sim T, Baker S, Bsat M (2003) The cmu pose, illumination, and expression database. IEEE Trans Pattern Anal Mach Intell 25(12):1615–1618
21. Tenenbaum J, De Silva V, Langford J (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323
22. Wachsmuth E, Oram M, Perrett D (1994) Recognition of objects and their component parts: responses of single units in the temporal cortex of the macaque. Cereb Cortex 4(5):509–522
23. Xu W, Liu X, Gong Y (2003) Document clustering based on non-negative matrix factorization. In: Proceedings of the 26th annual international ACM SIGIR conference on research and development in informaion retrieval, pp 267–273
24. Zhang Q, Li B (2010) Discriminative k-svd for dictionary learning in face recognition. In: IEEE conference on computer vision and pattern recognition, pp 2691–2698

Xianzhong Long is currently a Ph.D. Candidate in the Department of Computer Science and Engineering, Shanghai Jiao Tong University. His research interests include Computer Vision and Pattern Recognition.

Hongtao Lu got his Ph.D. degree in Electronic Engineering from Southeast University, Nanjing, in 1997. After graduation he became a postdoctoral fellow in Department of Computer Science, Fudan University, Shanghai, China, where he spent two years. In 1999, he joined the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, where he is now a professor. His research interest includes machine learning, computer vision and pattern recognition, and information hiding. He has published more than sixty papers in international journals such as IEEE Transactions, Neural Networks and in international conferences. His papers got more than 400 citations by other researchers.

Yong Peng received the B.S degree in computer science from Hefei New Star Research Institute of Applied Technology, the M.S degree from Graduate University of Chinese Academy of Sciences. Now he is working towards his PhD degree in Shanghai Jiao Tong University. His research interests include machine learning, pattern recognition and evolutionary computation.

Wenbin Li got his Ph.D. degree from Shanghai Medical University, Shanghai, in 1995. In 2004, he joined the Department of Diagnostic and Interventional Radiology, Affiliated Sixth People's Hospital of Shanghai Jiao Tong University, Shanghai, where he is now a professor. His research interest includes medical image processing, medical imaging exam, and pattern recognition. He has published more than fifty papers in international journals conferences.