Contents lists available at ScienceDirect

Pattern Recognition

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

Nonlinear nonnegative matrix factorization based on Mercer kernel construction

Binbin Pan [a], Jianhuang Lai [b],*, Wen-Sheng Chen [c]

[a] School of Mathematics and Computational Science, Sun Yat-Sen University, Guangzhou, China
[b] School of Information Science and Technology, Sun Yat-Sen University, Guangzhou, China
[c] College of Mathematics and Computational Science, Shenzhen University, Shenzhen, China

ARTICLE INFO

Article history:
Received 28 May 2010
Received in revised form
10 March 2011
Accepted 16 March 2011
Available online 6 April 2011

Keywords:
Nonnegative matrix factorization
Mercer kernel
Kernel mapping
Face recognition

ABSTRACT

Generalizations ofnonnegative matrix factorization (NMF) in kernel feature space, such as projected gradient kernel NMF (PGKNMF) and polynomial Kernel NMF (PNMF), have been developed for face and facial expression recognition recently. However, these existing kernel NMF approaches cannot guarantee the nonnegativity of bases in kernel feature space and thus are essentially semi-NMF methods. In this paper, we show that nonlinear semi-NMF cannot extract the localized components which offer important information in object recognition. Therefore, nonlinear NMF rather than semi-NMF is needed to be developed for extracting localized component as well as learning the nonlinear structure. In order to address the nonlinear problem of NMF and the semi-nonnegative problem of the existing kernel NMF methods, we develop the nonlinear NMF based on a self-constructed Mercer kernel which preserves the nonnegative constraints on both bases and coefficients in kernel feature space. Experimental results in face and expressing recognition show that the proposed approach outperforms the existing state-of-the-art kernel methods, such as KPCA, GDA, PNMF and PGKNMF.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Nonnegative matrix factorization (NMF) [1,2], which aims to find part-based representation of nonnegative data, is an unsupervised subspace method. It decomposes the data into two nonnegative matrices,[1] the bases and the coefficients, in which the data are represented as a non-subtractive combination of bases. The nonnegativity constraints are compatible with the intuitive notion of combining parts to form a whole. For example, the bases represent the parts of face (nose, eyes, etc.) in face representation, and these parts are combined together to compose the face. Therefore, NMF is a promising approach for the extraction of localized components. Due to intuitive interpretability and part-based representation, NMF and its alternatives have been widely applied to face recognition [3], multimedia signal processing [4], document clustering [5], environmetrics [6], chemometrics [7], and bioinformatics [8].

Classical NMF is a linear model and it may fail to discover the nonlinearities of data. However, many real-life data have latent nonlinear structures. For example, the distribution of face image variations under different pose and illumination is complex and nonlinear. Therefore, the performance of traditional NMF is limited. Accordingly, it is necessary to develop the nonlinear NMF. We suggest here using the combination of kernel technology and NMF for this purpose. Kernel method is a powerful technique in handling nonlinear correlations, and it has been widely used for the extension of linear method to nonlinear version. The idea of kernel method is to map the data nonlinearly into a kernel feature space, where the nonlinearities will be linearized. Then, the linear method can be performed in the kernel feature space to process the nonlinear data. The great success of kernel to model the nonlinearities attracts many researchers for in-depth exploring [9]. For example, kernel principal component analysis (KPCA) [10] and generalized discriminant analysis (GDA) [11] were found to outperform their linear versions in different applications.

Recently, generalizations of NMF in kernel feature space, namely polynomial kernel NMF (PNMF) [12] and projected gradient kernel NMF (PGKNMF) [13], have been introduced to model NMF nonlinearly. They learn more useful latent features. Similar to NMF, PNMF approximates the embedded data as the linear combination of bases in kernel feature space by minimizing the squared Euclidean distance. It develops multiplicative updating rules that guaranteed the non-increasing evolution of the cost function. But the updating algorithm is not guaranteed to converge to the stationary points. Moreover, only the polynomial kernels are usable in PNMF. Using projected gradient method, PGKNMF successively optimizes two subproblems [14], which ensures that

* Corresponding author. Tel./fax: +86 20 84110175.
E-mail addresses: alt26cn@gmail.com (B. Pan), stsljh@mail.sysu.edu.cn, sunnyweishi@gmail.com (J. Lai), chenws@szu.edu.cn (W.-S. Chen).

[1] A nonnegative matrix means each entry of the matrix is nonnegative.

0031-3203/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2011.03.023

the limit point is a stationary point and that arbitrary positive kernels can be used. However, neither PNMF nor PGKNMF guaranteed the nonnegative constraints on bases in kernel feature space (cf. Section 3.1). Therefore, PNMF and PGKNMF are essentially semi-nonnegative matrix factorization (semi-NMF) [15][2] other than NMF. As shown in [15], semi-NMF performs worse than NMF in terms of clustering accuracies. Furthermore, to the best of our knowledge, no literature has studied the part-based representation or the classification power of semi-NMF.

Thus, we pose the following two questions:

1. Is semi-NMF fit for object recognition?
2. If not, how to develop nonlinear NMF?

The above two questions will be answered in this paper in the application of face recognition. For the first question, we theoretically illustrate that the bases of semi-NMF do not exhibit the characteristics of sparsity (cf. Section 3.2) which is important for localized component extraction and object recognition [3,16,17]. We also empirically compare the subspace representation and the classification power of semi-NMF with those of NMF. The empirical findings are in accord with the theoretical analysis.

For the second question, different from the previous kernelized methods, this paper develops a novel kernel mapping to impose the nonnegativity. The kernel function induced by the mapping is proven to be a Mercer one. The effectiveness of the proposed algorithm is evaluated by face and facial expression recognition. The results are encouraging, manifesting that nonlinear NMF has superiority over nonlinear semi-NMF.

The rest of this paper is organized as follows. Section 2 introduces the related work briefly. Section 3 presents the shortcomings of PNMF and PGKNMF, which motivates the work of this paper. The details of the proposed algorithm are described in Section 4. Experimental comparisons are given in Section 5, and the conclusions are drawn in Section 6.

2. Related work

In this section, we briefly describe the original NMF method [1] and the related extending works [12,13].

2.1. Nonnegative matrix factorization (NMF)

Give a set of facial images $\{I_i\}_{i=1}^{n}$, where n is the number of images. By stacking pixels of each image into a column vector x_i, one can get a training set $X=\{x_1,\ldots,x_n\}\subset\Gamma$, where $x_i\in\mathbb{R}^m$, m is the number of pixels for an image, Γ denotes the input space. The x_i's are concatenated to form a matrix $V=[x_1,\ldots,x_n]\in\mathbb{R}^{m\times n}$. NMF [1] aims to find an approximate decomposition

$$V\approx WH \tag{1}$$

by imposing the nonnegative constraints on W and H, where $W\in\mathbb{R}^{m\times r}$ is the bases, $H\in\mathbb{R}^{r\times n}$ is the coefficients, r is the number of bases. These constraints offer some degree of sparsity in bases and coefficients which will be illustrated in Section 3.2. The (1) can be casted as the problem of minimizing the reconstruction error under the nonnegative constraints:

$$E_{NMF}(W,H)=\|V-WH\|_F^2=\sum_{ij}(V_{ij}-(WH)_{ij})^2$$
$$\text{s.t. } W,H\geq 0, \tag{2}$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

It is unrealistic to find the global minima of problem (2) since this problem is not convex for variables W and H. Lee and Seung [2] employed coordinate-descent method and ensured that the objective function was non-increasing after each iteration by choosing appropriate steps. The NMF algorithm is described in Algorithm 1.

Algorithm 1. NMF algorithm.

```
Initialize $W_{ij}\geq 0, H_{ij}\geq 0, \forall i,j$.
**for** $k=1,2,\ldots$ until convergence **do**
    $H_{ij}\leftarrow H_{ij}\frac{(W^TV)_{ij}}{(W^TWH)_{ij}}$,
    $W_{ij}\leftarrow W_{ij}\frac{(VH^T)_{ij}}{(WHH^T)_{ij}}$,
    $W_{ij}\leftarrow \frac{W_{ij}}{\sum_k W_{kj}}$.
**end for**
```

Comparing with other subspace methods with holistic components, such as PCA and LDA, NMF can extract localized components which offer advantages in object recognition, including stability to local deformations, lighting variations, and partial occlusion [3]. In face recognition, NMF has shown to be superior to PCA and LDA [18]. Thus, NMF has been widely investigated recently. In real world, many data exhibit nonlinear structure, however, the linear NMF method cannot learn the nonlinear relations between the data. Therefore, nonlinear NMF should be developed. Two kernel-based nonlinear NMF algorithms have been proposed and are introduced as follows.

2.2. Polynomial NMF (PNMF)

PNMF [12] is the variant of NMF in kernel feature space, aiming at representing the images in a nonlinear way. Each image is firstly embedded in a polynomial feature space P via a polynomial kernel-induced nonlinear mapping

$$\phi: x\in\Gamma\mapsto\phi(x)\in P. \tag{6}$$

The dot product in P can be written by means of polynomial kernel $\langle\phi(x),\phi(y)\rangle=k(x,y)=(x^Ty)^d$, where d is an integer.

The idea of PNMF is to find a set of bases W^ϕ in P to approximate the embedded images, i.e., $\phi(x_i)\approx W^\phi h_i, i=1,2,\ldots,n$. The problem is formulated as minimizing the reconstruction error in P:

$$E_{PNMF}(W^\phi,H)=\|V^\phi-W^\phi H\|_F^2$$
$$\text{s.t. } w_i,H\geq 0, i=1,\ldots,r, \tag{7}$$

where $V^\phi=[\phi(x_1),\ldots,\phi(x_n)], W^\phi=[\phi(w_1),\ldots,\phi(w_n)]$. Vectors w_i are called the pre-images of the bases. The detailed algorithm to solve problem (7) is referred to [12].

2.3. Projected gradient kernel NMF (PGKNMF)

PGKNMF [13] is developed to remedy the limitations of PNMF: (1) PNMF cannot guarantee that the limit point is a stationary point, (2) Only polynomial kernel can be used. It solves problem (7) by successively optimizing two subproblems:

$$E_{PNMF}(W^\phi)$$
$$\text{s.t. } w_i\geq 0, i=1,\ldots,r, \text{ with } H \text{ fixed} \tag{8}$$

and

$$E_{PNMF}(H)$$
$$\text{s.t. } H\geq 0 \text{ with } W^\phi \text{ fixed}. \tag{9}$$

[2] Semi-NMF decomposes a matrix into a mixed-sign bases and a nonnegative coefficients.

Each subproblem is solved by using projected gradient method [14]. PGKNMF ensures that the limit point is a stationary point and arbitrary kernel can be used.

PNMF and PGKNMF tackle the original NMF in a nonlinear way, by embedding the images in the kernel feature space. The merits of them are that they learn the nonlinear relations between the data. But they are essential nonlinear semi-NMF methods since they cannot guarantee the nonnegativity of bases, thus, they cannot take full advantage of NMF. We will show both theoretically and experimentally that unlike NMF, semi-NMF does not have the ability of extracting localized components.

3. Analysis and motivation

In this section, we will show that PNMF and PGKNMF are semi-NMF methods, and then compare the NMF with semi-NMF in theory and applications. We firstly describe the problem of generalizing NMF in kernel feature space F as follows. In the kernel method, the input data are embedded into F by a kernel mapping

$$\phi : x \in \Gamma \mapsto \phi(x) \in F. \tag{10}$$

The ϕ induces a kernel function k with entries

$$k(x,y) = \langle \phi(x),\phi(y) \rangle = \phi^T(x)\phi(y). \tag{11}$$

Each $\phi(x_i)$ is represented as a linear combination of a set of bases $w_i^\phi \in F$ with weights $h_{ij} \geq 0$, i.e.

$$\phi(x_i) \approx \sum_{j=1}^{r} w_j^\phi h_{ji}, \tag{12}$$

In matrix notation, (12) can be represented as

$$V^\phi \approx W^\phi H, \tag{13}$$

where $V^\phi = [\phi(x_1) \ \cdots \ \phi(x_n)]$, $W^\phi = [w_1^\phi \ \cdots \ w_r^\phi]$ and $[H]_{ji} = h_{ji}$, with $W^\phi \in \mathbb{R}_+^{m\times r}$ and $H \in \mathbb{R}_+^{r\times n}$.

The existing methods, PNMF and PGKNMF, tackled the problem (13) by preserving the nonnegativity of H, but they do not guarantee w_i^ϕ to be nonnegative. We illustrate that w_i^ϕ, learned by PNMF and PGKNMF, are mix-sign, and thus PNMF and PGKNMF are essential semi-NMF. We further show that the bases of semi-NMF are non-sparse. The details of the above discussions are shown as follows.

3.1. Demonstration of PNMF and PGKNMF being semi-NMF

In order to show that PNMF and PGKNMF are semi-NMF, we only need to prove that the embedded data are mixed-sign, thus, the representing bases are naturally mixed-sign. The embedded point is nonnegative if and only if the angle between the point and each positive axis is equal to or less than 90°. More formally, we introduce the following theorem.

Theorem 1. *Given a vector x, $x \geq 0$ iff $\angle(x,e_i) \leq \pi/2$, for $\forall i$, where e_i is the unit vector of the ith axis, $\angle(x,e_i)$ denotes the angle between x and e_i.*

Proof. We consider the sufficiency. Suppose for contradiction that $x^{(j)} < 0$, where $x^{(j)}$ is the jth element of x. Then we have

$$\cos(\angle(x,e_j)) = \frac{\langle x,e_j \rangle}{\|x\|_2 \cdot \|e_j\|_2} = \frac{x^{(j)}}{\|x\|_2} < 0, \tag{14}$$

which shows that $\angle(x,e_j) > \pi/2$. This is a contradiction.

The necessity is straightforward. □

In Theorem 1, to judge whether the embedded data are nonnegative, the angle information between embedded data and axis is necessary. But this is lacked in kernel function. To illustrate it, we assume that there exists an implicit mapping ϕ which embeds the data nonnegatively. The ϕ induces a kernel function k. If rotation transformation R is imposed on ϕ, it will yield another mapping $\psi = R \cdot \phi$ which rotates the original nonnegative embedded points to new embedded points with negative entries. However, both ϕ and ψ correspond to the same k.

Both PNMF and PGKNMF adopt implicit mapping determined by kernels. As discussed above, implicit mapping makes the embedded points impossible to maintain the nonnegativity. Therefore, they both essentially exploit semi-NMF in kernel feature space.

3.2. NMF versus semi-NMF in sparseness

In order to get the localized components of the data, these components should be sparse, which has been extensively investigated in the literatures [19,17]. To examine the sparseness property of NMF and semi-NMF, we make use of the optimization theory. The minimization problems of NMF and semi-NMF are

$$\min_{W \geq 0, H \geq 0} E_{NMF}(W,H) = \|V - WH\|_F^2, \tag{15}$$

$$\min_{H \geq 0} E_{semi-NMF}(W,H) = \|V - WH\|_F^2. \tag{16}$$

Almost all the existing algorithms are developed to find the stationary point of the minimization problems. If (W^*,H^*) is a stationary point of (15), then it satisfies the following part of Karush–Kuhn–Tucker (KKT) optimality conditions [20]:

$$W_{ij}^* \cdot \nabla_W E_{NMF}(W^*,H^*)_{ij} = 0, \quad H_{ij}^* \cdot \nabla_H E_{NMF}(W^*,H^*)_{ij} = 0, \ \forall i,j. \tag{17}$$

The condition (17) is the complementary slackness condition, which enforces the sparse solution. This is because that if $\nabla_W E_{NMF}(W^*,H^*)_{ij}$ is not zero, then W_{ij}^* is forced to zero. Alternatively, in the language of optimization, the reason for sparsity is that the stationary point (W^*,H^*) of NMF will be typically located at the boundary of the feasible domain $\mathbb{R}_+^{m\times r} \times \mathbb{R}_+^{r\times n}$, hence will feature zero entries. This is similar to the support vector machine (SVM) [21], where the Lagrangian multipliers produce sparse support vectors.

Similarly, if (W^*,H^*) is a stationary point of (16), we get the following optimality conditions:

$$H_{ij}^* \cdot \nabla_H E_{semi-NMF}(W^*,H^*)_{ij} = 0, \forall i,j. \tag{18}$$

Note that there is no complementary slackness condition for W. The unconstraints on W make no sparsity in the solution W^*. Only H^* have some zero entries on the boundary of the feasible domain. Therefore, an important property of NMF is that its nonnegative constraints typically induce sparse solution.

3.3. Empirical comparisons of NMF and semi-NMF

We empirically compare the performance of NMF and semi-NMF in face recognition to support the theoretical findings in Section 3.2. The standard FERET face database [22], well known as variants in pose, illumination and expression, was selected for validation.

Learning bases. Similar to [1], semi-NMF and NMF representations with 49 bases were learned from the data. Fig. 1 shows the resulting semi-NMF and NMF bases.

As showed in Fig. 1, the semi-NMF bases are holistic, like PCA bases (eigenfaces) [23]. On the contrary, NMF learns both localized and holistic bases. This is in accord with the theoretical analysis in Section 3.2. Thus, the bases of semi-NMF cannot capture the notion of the parts of face.

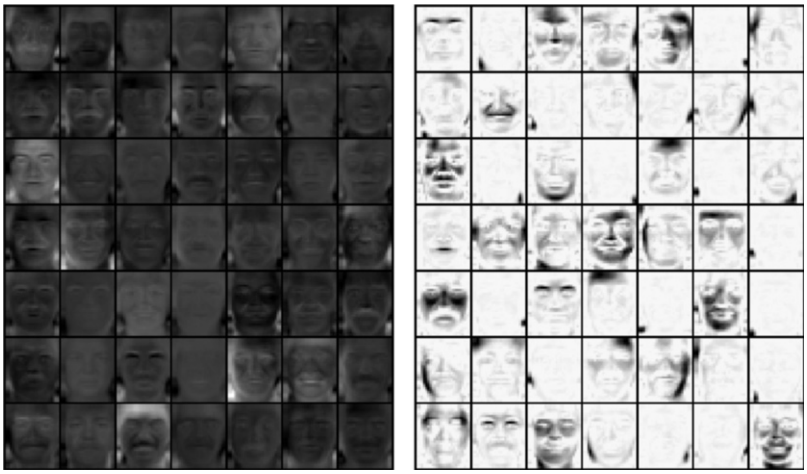

Fig. 1. Semi-NMF (left) and NMF (right) bases.

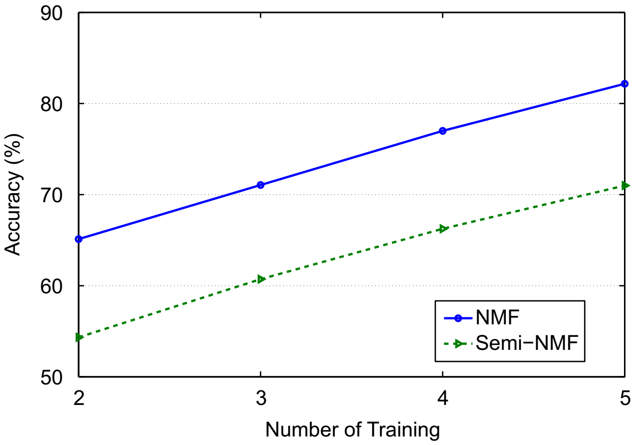

Fig. 2. Performance on FERET database.

Face recognition. The semi-NMF and NMF representations were comparatively evaluated for face recognition. We randomly selected j (j=2,3,4,5) samples of each individual for training and the rest $(6-j)$ images for testing. The experiments were conducted 10 times repetitively and the average results were recorded. The accuracy is defined as the ratio of the number of correctly classified data to the number of compared data. The NMF formula is from [1] and the semi-NMF formula is from [15]. Fig. 2 shows the recognition accuracies.

As shown, semi-NMF provides dissatisfactory performance, while NMF yields more favorable results. In short, comparing with semi-NMF, NMF is more suitable for face recognition due to its part-based representation property.

As discussed above, it is expected that nonlinear NMF would be superior to nonlinear semi-NMF in face recognition, which motivates our work.

4. Proposed approach

In this section, we show how to develop nonlinear NMF with a nonnegative kernel mapping and self-constructed Mercer kernel.

4.1. Nonnegative kernel mapping construction

Our goal is to construct a kernel mapping ϕ^n which keeps nonnegative constraints on the embedded training data. Moreover, the corresponding kernel function k_M induced by ϕ^n should approximate a desired kernel function. Frequently used kernel functions are the polynomial ones and the radial basis function (RBF) ones. In this paper, we adopt the widely used RBF kernel

$$k_{RBF}(x,y) = \exp\left(-\frac{\|x-y\|_2^2}{\sigma}\right), \tag{19}$$

where $\exp(\cdot)$ is the exponential function and $\sigma > 0$ is the width of kernel. It has been extensively validated that this RBF kernel is suitable for many types of data. The width σ, crucial for the performance of the kernel-based methods, determines the geometrical structure of the embedded data.

The mapping ϕ^n is constructed by the following two steps:

1. ϕ^n defined on X is firstly constructed.
2. Extend ϕ^n to the whole input space.

4.1.1. Step 1: defining ϕ^n on X

We solve the problem formulated as

$$E(\phi^n) = \sum_{i=1}^{n}\sum_{j=1}^{n}(\langle \phi^n(x_i), \phi^n(x_j)\rangle - k_{RBF}(x_i,x_j))^2$$
$$\text{s.t. } \phi^n(x_i) \geq 0,\ 1 \leq i \leq n. \tag{20}$$

Cost function (20) measures the difference between the k_M and the RBF kernel. ϕ^n is derived by minimizing this cost function subject to the nonnegative constraints. If we denote $K_{RBF} = (k_{RBF}(x_i,x_j))_{n\times n}$, $x_i,x_j \in X$, then (20) can be simplified in matrix notation:

$$E(Q) = \frac{1}{4}\|Q^TQ - K_{RBF}\|_F^2$$
$$\text{s.t. } Q \geq 0, \tag{21}$$

where $Q = (\phi^n(x_1), \ldots, \phi^n(x_n)) \in \mathbb{R}^{s\times n}$, with s the dimensionality of F. The constant $\frac{1}{4}$ is to make the updating formulations more neatly.

We call (21) the symmetric NMF. Motivated by [2], we devise the coordinate-descent scheme to minimize (21) efficiently.

Taking the derivative of (21) with respect to Q, it gives that

$$\frac{\partial E(Q)}{Q_{ab}} = -(QK_{RBF} - QQ^TQ)_{ab}. \tag{22}$$

The update rule then states

$$Q_{ab} \leftarrow Q_{ab} + \eta_{ab}(QK_{RBF} - QQ^TQ)_{ab} \tag{23}$$

for some step length $\eta_{ab} > 0$. The searching direction chosen in (23) is indeed a descent direction.

In order to restrict Q to nonnegativity, we set

$$\eta_{ab} = \beta \frac{(QK_{RBF})_{ab}}{(QQ^TQ)_{ab}}, \tag{24}$$

where $0 < \beta \le 1$. The update rule becomes

$$Q_{ab} \leftarrow Q_{ab}\left(1-\beta+\beta\frac{(QK_{RBF})_{ab}}{(QQ^TQ)_{ab}}\right). \tag{25}$$

Searching β is the most time-consuming operation, we use a simple and efficient "backtracking line search" method [20]. The procedure is illustrated in Algorithm 2.

Algorithm 2. Symmetric NMF $K \approx Q^TQ$ $(Q \ge 0)$.

```
Input $\beta = 0.1$, $\sigma \in (0,1)$, maxiter, $Q^1$.
for $k=1$ : maxiter do
  $Q^{k+1} \leftarrow Q^k \otimes ((1-\beta^{n_k})\mathbf{1} + \beta^{n_k}((Q^kK) \oslash (Q^k(Q^k)^TQ^k)))$,
    where $n_k$ is the first nonnegative integer satisfying
  $E(Q^{k+1}) \le E(Q^k) - \sigma\|\nabla E(Q^k)\| \cdot \|Q^{k+1}-Q^k\|$,

  $\mathbf{1}$ is a matrix with all entries equal to 1, $\otimes$ and $\oslash$ denote
  elementwise multiplication and division, respectively.
end for
```

The "Armijo condition" (27) is a popular choice in line search, which ensures sufficient decrease in the objective function per iteration [20]. By trying $1, \beta, \beta^2, \ldots$, the β^{n_k} satisfying (27) always exists because Q^k is moving along the descent direction. The "Armijo condition" implies the convergence of Algorithm 2, since $E(Q^k)$ is non-increasing and bounded below, it is a convergent sequence.

By applying Algorithm 2, a nonnegative matrix Q, representing the embedded training data, is obtained. Write Q as block form $Q = [q_1, q_2, \ldots, q_n]$, where q_i is the ith column of Q. The kernel mapping ϕ^n, defined on X, is

$$\phi^n|_X : x_i \mapsto q_i, \quad i = 1,2,\ldots,n. \tag{28}$$

4.1.2. Step 2: extending ϕ^n to Γ

The second step is to extend the domain of ϕ^n from subspace X to the whole input space Γ. For any $x \in \Gamma \backslash X$, we assume that $\phi^n(x)$ lies in the space $span\{\phi^n(x_1), \ldots, \phi^n(x_n)\}$:

$$\phi^n(x) = \sum_{i=1}^{n} \alpha_i(x)\phi^n(x_i) = Q \cdot \alpha(x), \tag{29}$$

where $\alpha(x) = (\alpha_1(x), \alpha_2(x), \ldots, \alpha_n(x))^T$ is a vector of combination coefficients.

The problem of finding $\phi^n(x)$ is converted to seeking the coefficients $\alpha(x)$. As we will indicate soon, this conversion makes the problem more easy to solve. $\alpha(x)$ is found by satisfying $\langle \phi^n(x_i), \phi^n(x) \rangle \approx k_{RBF}(x_i, x)$, $i = 1,2,\ldots,n$. In vector notation, we need to solve the following optimization problem:

$$\alpha^*(x) = \underset{\alpha(x)}{\operatorname{argmin}} \|\hat{\Phi}^n(x) - \hat{K}_{RBF}(x)\|_2^2, \tag{30}$$

where $\hat{\Phi}^n(x) = (\langle \phi^n(x_1), \phi^n(x) \rangle, \ldots, \langle \phi^n(x_n), \phi^n(x) \rangle)^T$ and $\hat{K}_{RBF}(x) = (k_{RBF}(x_1, x), \ldots, k_{RBF}(x_n, x))^T$.

From (28) and (29), we have

$$\hat{\Phi}^n(x) = (q_1^T(Q \cdot \alpha(x)), \ldots, q_n^T(Q \cdot \alpha(x)))^T = Q^TQ \cdot \alpha(x) = K_{RBF} \cdot \alpha(x). \tag{31}$$

Problem (30) can be turned into the following:

$$\alpha^*(x) = \underset{\alpha(x)}{\operatorname{argmin}} \|K_{RBF} \cdot \alpha(x) - \hat{K}_{RBF}(x)\|_2^2. \tag{32}$$

Eq. (32) is an unconstrained convex quadratic programming, where its solution is obtained by setting the derivative to zero:

$$\alpha^*(x) = K_{RBF}^{-1} \cdot \hat{K}_{RBF}(x). \tag{33}$$

What deserves to be mentioned is that the matrix inverse in (33) always exists in real application. If $x_1, \ldots, x_n$ are distinct points, the matrix K_{RBF} has full rank.[3]

Therefore, ϕ^n defined on Γ is

$$\phi^n(x) = Q \cdot \alpha(x) = Q \cdot K_{RBF}^{-1} \cdot \hat{K}_{RBF}(x). \tag{34}$$

4.1.3. Nonnegative kernel mapping

By performing the above two steps, we have constructed a kernel mapping which preserves the nonnegativity on the training data. More generally, we give the definition of nonnegative kernel mapping.

Definition 1 (*Nonnegative kernel mapping*). Give a set $\{z_1, \ldots, z_n\} \subset \Gamma$, $n \in \mathbb{N}$, kernel matrix $K = [k(z_i, z_j)]_{n \times n}$ and nonnegative matrix $Q \in \mathbb{R}_+^{s \times n}$. We call

$$\phi^n(x) = Q \cdot K^{-1} \cdot \hat{K}(x), \tag{35}$$

where $x \in \Gamma$, $\hat{K}(x) = (k(z_1, x), \ldots, k(z_n, x))^T$, the nonnegative kernel mapping with respect to $\{z_1, \ldots, z_n\}$.

To see why we call (35) the nonnegative kernel mapping, we present the following theorem.

Theorem 2. *If ϕ^n is a nonnegative kernel mapping with respect to $\{z_1, \ldots, z_n\}$, then $\phi^n(z_i) \ge 0$, $i = 1,2,\ldots,n$.*

Proof. For any $z_i \in \{z_1, \ldots, z_n\}$, we have

$$\phi^n(z_i) = Q \cdot K^{-1} \cdot \hat{K}(z_i) = Q \cdot K^{-1} \cdot Ke_i = Q \cdot e_i = q_i \ge 0, \tag{36}$$

where e_i is the ith column of identity matrix and q_i is the ith column of Q, proving our claim. □

Notably, if Q is chosen the solution to (21), the nonnegative kernel mapping will reduce to (34), used in the nonlinear NMF. How to find other appropriate Q is not within the scope of this article, left for future research.

Remark 1. The empirical kernel mapping [24] has the similar form as the nonnegative kernel mapping

$$\phi^e(x) = \Lambda^{-1/2}P \cdot \hat{K}(x) = \Lambda^{1/2}PK^{-1} \cdot \hat{K}(x) = Q \cdot K^{-1} \cdot \hat{K}(x), \tag{37}$$

where $Q = \Lambda^{1/2}P$, Λ is a diagonal matrix containing the eigenvalues of K, and P consists of the eigenvectors corresponding to the eigenvalues. However, the Q defined in (37) is not always nonnegative.

In some applications, if original data are nonnegative, we want the embedded data to be nonnegative as well. For example, NMF is only applied to nonnegative data [1], some sources must be either zero or positive to be physically meaningful in source-separation problems [25] (i.e., the amount of pollutant emitted by a factory is nonnegative [6], the probability of a particular topic appearing in a linguistic document is nonnegative [26] and note volumes in musical audio are nonnegative [27]). Thus, nonnegative kernel mapping is more suitable than empirical kernel mapping for these applications.

[3] If the inverse does not exist, we use the pseudo-inverse instead.

4.2. Mercer kernel construction

Based on (34), we construct a function defined on $\Gamma \times \Gamma$:

$$k_M(x,y) = \langle \phi^n(x), \phi^n(y) \rangle = \langle Q \cdot \alpha(x), Q \cdot \alpha(y) \rangle = \alpha^T(x) \cdot K_{RBF} \cdot \alpha(y), \tag{38}$$

where $x,y \in \Gamma$ and $\alpha(x)$, $\alpha(y)$ are defined in (33). The following theorem demonstrates k_M is a Mercer kernel.

Lemma 1 (*Taylor and Cristianini* [28]). *A function $k : \Gamma \times \Gamma \to \mathbb{R}$ is a Mercer kernel if and only if the matrix K yielded from any finite subset $\{y_1, y_2, \ldots, y_n\} \subset \Gamma$ with entries $K_{ij} = k(y_i, y_j)$ is positive semi-definite.*

Theorem 3. *The function k_M defined in* (38) *is a Mercer kernel.*

Proof. For any given finite subset $\{y_1, y_2, \ldots, y_n\} \subset \Gamma$, $\tilde{K}$ is yielded from the subset with entries $\tilde{K}_{ij} = k_M(y_i, y_j) = \alpha^T(y_i) \cdot K_{RBF} \cdot \alpha(y_j)$. For any column vector $u \in \mathbb{R}^n$,

$$\begin{aligned} u^T \tilde{K} u &= u^T \begin{bmatrix} \alpha^T(y_1) \cdot K_{RBF} \cdot \alpha(y_1) & \cdots & \alpha^T(y_1) \cdot K_{RBF} \cdot \alpha(y_n) \\ \cdots & \cdots & \cdots \\ \alpha^T(y_n) \cdot K_{RBF} \cdot \alpha(y_1) & \cdots & \alpha^T(y_n) \cdot K_{RBF} \cdot \alpha(y_n) \end{bmatrix} u \\ &= u^T \begin{pmatrix} \alpha^T(y_1) \\ \vdots \\ \alpha^T(y_n) \end{pmatrix} K_{RBF} (\alpha(y_1) \ \cdots \ \alpha(y_n)) u \\ &= v^T K_{RBF} v, \end{aligned} \tag{39}$$

where $v = (\alpha(y_1)\ \alpha(y_2)\ \cdots\ \alpha(y_n))u$. Since K_{RBF} is positive semi-definite, $v^T K_{RBF} v \geq 0$, namely, $u^T \tilde{K} u \geq 0$. This proves that k_M is a Mercer kernel according to Lemma 1. □

We have constructed ϕ^n to embed the data into F and k_M to approximate the RBF kernel. Performing NMF on Q which represents the embedded training data learns the nonlinear NMF.

4.3. Algorithm design

The proposed approach is named Mercer Kernel NMF (mkNMF). We incorporate our previous block trick [29] into mkNMF to reduce the computational complexity. Suppose the cost of NMF-based method m is $O(T(m))$, where $T(\cdot)$ is a function with respect to times of multiplicative operations per iteration. Then the complexity with block trick reduces to $O(T(m)/c)$, where c is the number of class. The algorithm is designed below:

Training stage.

*Step*1: Generate K_{RBF} from X using RBF kernel.

*Step*2: Perform symmetric NMF

$$K_{RBF} \approx Q^T Q. \tag{40}$$

*Step*3: Written Q as block form $Q = [Q_1 \ \cdots \ Q_c]$, where $Q_i \in \mathbb{R}^{s \times n_i}$ $(i=1,2,\ldots,c)$ consists of n_i embedded training data of the ith class and c is the number of class, employ $Q \overset{BNMF}{\approx} W^\phi H$, namely

$$[Q_1 \ \cdots \ Q_c] \approx [W_1^\phi \ \cdots \ W_c^\phi] \begin{bmatrix} H_1 & & \\ & \ddots & \\ & & H_c \end{bmatrix}, \tag{41}$$

where $(Q_i)_{s \times n_i} \overset{NMF}{\approx} (W_i^\phi)_{s \times r_0} (H_i)_{r_0 \times n_i}, i = 1,2,\ldots,c$, $W^\phi \in \mathbb{R}^{s \times r}, H \in \mathbb{R}^{r \times n}$, $r = cr_0, n = \sum_{i=1}^{c} n_i$.

Recognition stage.

*Step*4. Project $\phi^n(x)$, where $x \in \Gamma$, onto the subspace spanned by W^ϕ:

$$h_x = (W^\phi)^+ \phi^n(x) = (W^\phi)^+ Q K_{RBF}^{-1} \cdot \hat{K}_{RBF}(x). \tag{42}$$

*Step*5: Employ the nearest neighbor classification. x is classified to class j, if $x_k = \mathrm{argmin}_{x_i \in X} d(h_x, h_{x_i})$, where x_k belongs to class j and $d(\cdot,\cdot)$ denotes the distance between two inputs.

Remark 2. Dimensionality selection of F is an important issue in the proposed algorithm. Many literatures have claimed that the dimensionality should be very high to linearize the embedded data [28,24]. However, the selection of kernel is more important than the setting of dimensionality. The performances of the algorithm may not be better in a higher dimensional space if the kernels are homologous. In Section 5.2, this argument will be illustrated by experimental results.

4.4. Computational complexity

The complexity of mkNMF method is analyzed. The cost of forming the kernel matrix is $O(mn^2)$. The symmetric NMF (40) requires $O(n^2 s)$ per iteration, and the Block NMF (41) needs $O(nsr/c)$ per iteration. If #iter1 and #iter2 iterations are executed in (40) and (41), respectively, the total cost is

$$\#iter1 \times O(n^2 s) + iter2 \times O(nsr/c) + O(mn^2). \tag{43}$$

The (40) runs costly, since it embeds the data into a higher dimensional space. In practical, #iter1 is set less than #iter2.

5. Experimental results

The effectiveness of the proposed nonlinear NMF is evaluated by face and facial expression recognition, along with NMF [1], nonlinear semi-NMF (PNMF [12], PGKNMF [13]), nonlinear PCA (KPCA [10]) and nonlinear LDA (GDA [11]). The selected face databases involve pose, expression, illumination and occlusion variations which are known to be the main challenges of face recognition. The parameters of each evaluated method are set via grid search technique to provide the best performance on all the databases. Specifically, the RBF kernel parameter δ is selected 6000 in mkNMF and PGKNMF, 40 in KPCA, 40,000 in GDA. The polynomial kernel parameter d is set to 3 in PNMF. For the sake of a fair comparison, the block strategy is used in mkNMF, PNMF and PGKNMF. The methods with block trick give improvements of accuracies slightly. More importantly, the computational complexity can be reduced efficiently. Each dataset is randomized 10 times. The results are the average of the 10 runs.

In many linear subspace methods (e.g., PCA, LDA, NMF), the learned base can be represented as a face (eigenface or Fisherface) or a part of face (NMF face) intuitively, and the reconstruction of original facial images is allowed by expansion of bases in subspace. However, in kernel methods, the intuitive representation and reconstruction are no longer possible [10]. The reason includes two aspects: on one hand, the embedded images no longer represent true faces in F, and thus the learned base has no intuitive notion of exhibiting a face or a part of face; on the other hand, the learned bases in F are unknown if an implicit mapping is applied, moreover, a vector in F may not have a pre-image in input space. But we can anticipate the sparsity of learned bases by NMF technique in the light of the analysis in Section 3.2. The improvements of discriminative power in virtue of sparsity can be verified in face recognition.

5.1. Face databases

Three face databases, namely FERET [22], CMU PIE [30] and AR [31], are selected for evaluations.

The FERET database contains of 14,126 images that includes 1199 individuals. We select a subset including 720 images of 120 individuals (six images for each) to evaluate the algorithms. This subset involves variations in facial expression, illumination, and pose. In our experiment, the facial portion of each original image was automatically cropped based on the location of eyes. Images from one individual are shown in Fig. 3.

The CMU PIE database contains 41,368 facial images of 68 people. A subset including 3808 images of 68 individuals (56 images per individual) is selected to test the algorithms. The facial portion of each original image was automatically cropped based on the location of eyes. Part images of one people are shown in Fig. 4.

The AR database contains over 4000 images corresponding to 126 people's faces. A subset including 119 individuals (26 images for each) is selected to validate the methods. All images have been cropped based on the location of eyes. Cropped images from one individual are shown in Fig. 5.

Fig. 3. Images of one individual in the FERET database.

Fig. 4. Part images of one individual in the CMU PIE database.

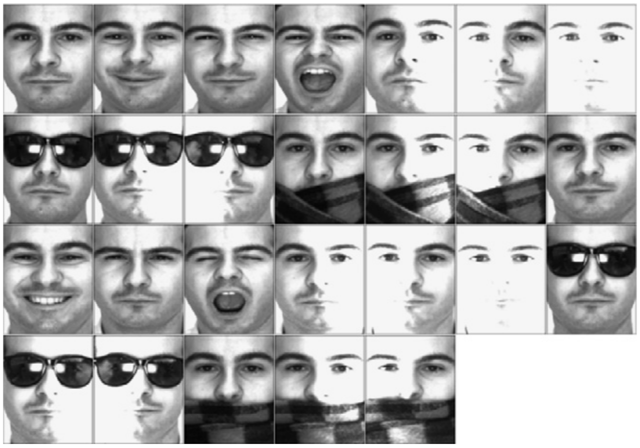

Fig. 5. Images of one individual in the AR database.

The evaluated methods, especially the kernel-based methods, run costly on the original facial images. In order to reduce the computational expense, the original facial images with resolution 112×92 are reduced to LL-images (LL represents the low frequency components) with resolution 30×25 using Daubechies D4 wavelet transforms. The rank of NMF equals to the number of individual, namely $r=120$ in FERET, $r=68$ in CMU PIE and $r=119$ in AR. Block parameter $r0$ is set to 4. Nearest neighbor classifier based on Euclidean measure (EM) is used.

5.2. Experiments on FERET database

We randomly select j $(j=2,3,4,5)$ training samples from each individual and the rest $(6-j)$ images for testing. The mkNMF algorithm under different dimensionality $s\in\{1500,3000,4500,6000\}$ of F is evaluated. The chosen dimensionality is just a multiple of the dimensionality of the input space (=750). The mean accuracies, associated with the running time, are tabulated in Tables 1 and 2, respectively (NT is short for number of training).

Table 1 shows that the accuracies under different s are almost the same. The entire mean varies slightly while enlarging s. It is verified experimentally that the dimensionality is little influential while the kernels are homologous. However, when s enlarges, the running time increases remarkably, raising from 181.61 s with dimensionality 1500 to 394.39 s with dimensionality 6000. Considering the computational efficiency, we choose $s=1500$ for the following experiments.

We also evaluate our algorithm with other kernels — the polynomial and hyperbolic ones [32] of the forms

$$k_p(x,y)=(\langle x,y\rangle+t)^n, \tag{44}$$

$$k_{Hy}^n(x,y)=[\mathrm{sech}(\beta\|x-y\|)]^n, \tag{45}$$

where $t\geq0$. We adopt the first, second and third-order hyperbolic kernels and find the best parameters via grid search. Concretely, the parameters are set as following: $\beta=$ 0.025, 0.015 and 0.012 for k_{Hy}^1, k_{Hy}^2 and k_{Hy}^3, $t=50{,}000$ and $n=2$ for k_p. The accuracies are shown in Fig. 6. These kernels give close results, although the polynomial one presents lower accuracies at 4 and 5 training data, it still significantly outperforms the original NMF. Thus, our algorithm is also valid with other kernels.

Fig. 7 gives the average accuracies with different algorithms. The entire mean of NMF, PNMF, PGKNMF, KPCA, GDA and mkNMF are 75.29%, 70.82%, 69.44%, 56.79%, 81.03% and 83.81%, respectively. Although PNMF and PGKNMF are the nonlinear variants of

Table 1
Accuracy (%) of mkNMF under different dimensionality.

NT	$s=1500$	$s=3000$	$s=4500$	$s=6000$
2	74.71	74.33	73.67	74.33
3	82.67	82.69	83.06	82.81
4	86.67	86.88	87.17	87.04
5	90.58	91.33	91.08	91.42
Entire mean	83.66	83.81	83.75	83.90

Table 2
Running time (s) of mkNMF under different dimensionality.

$s=1500$	$s=3000$	$s=4500$	$s=6000$
181.61	252.47	321.97	394.39

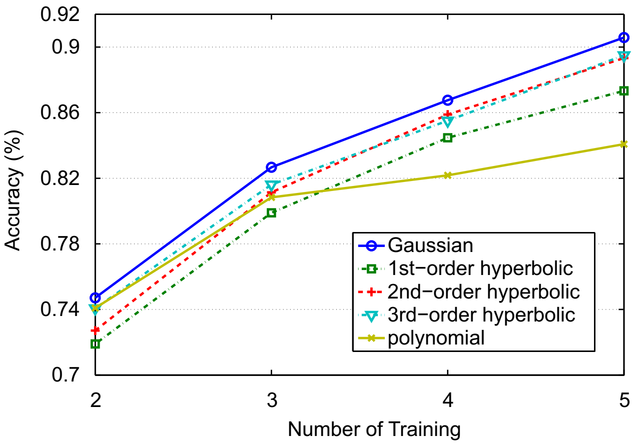

Fig. 6. Performance of mkNMK with different kernels.

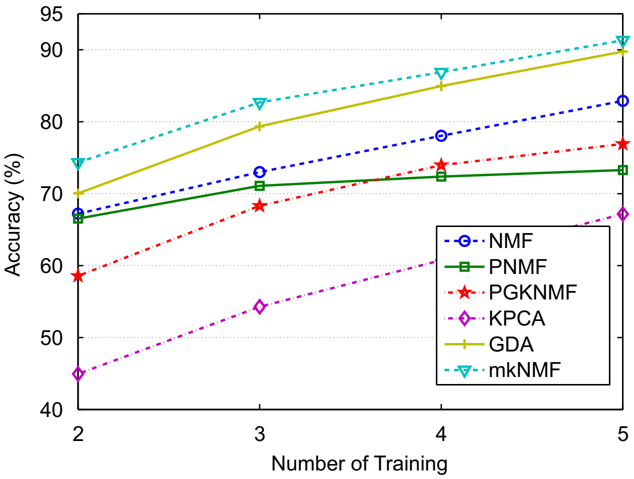

Fig. 7. Performance on FERET database.

NMF, they perform poorer than NMF. The proposed method outperforms other algorithms.

5.3. Experiments on CMU PIE database

We arbitrarily select j (j=7,14,21,28) training samples from each individual while the rest ($56-j$) images for testing. The mean results are plotted in Fig. 8. Unsupervised KPCA still performs the poorest, only with accuracy 51.53%. The mkNMF algorithm, with accuracy 75.68%, still gives the best performance.

5.4. Experiments on AR database

We randomly choose j (j=3,6,9,12) training samples from each individual while the rest ($26-j$) images for testing. The average results are shown in Fig. 9. KPCA performs very badly on this more complicated database. Nonlinear semi-NMF (PNMF and PGKNMF) also gives dissatisfactory performance. The proposed nonlinear NMF, giving the best results, is robust to changes in lighting, facial expression, pose and occlusion.

5.5. ROC curve analysis

To better compare the generalization performances of various algorithms, we draw the receiver operating characteristic (ROC) curves [33] for FERET, CMU PIE and AR databases in Figs. 10–12, respectively. As shown in the three figures, while guaranteeing a false positive rate of 20%, mkNMF achieves the best true positive rate. The areas under the ROC curves (AUC) of mkNMF are the largest.

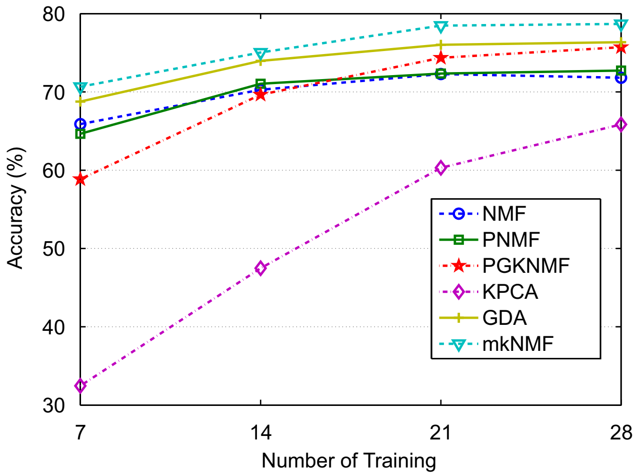

Fig. 8. Performance on CMU PIE database.

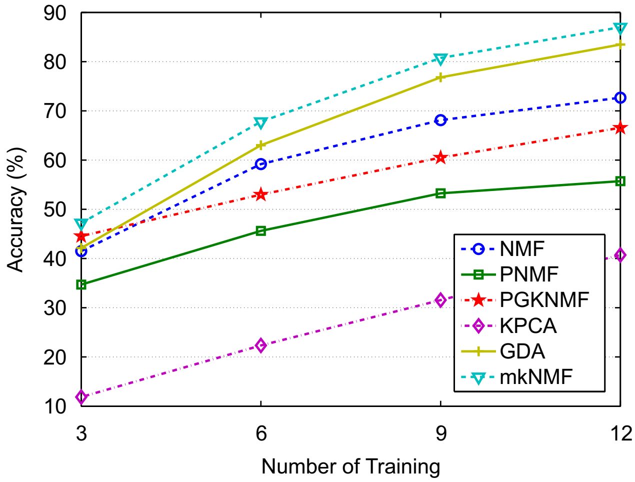

Fig. 9. Performance on AR database.

5.6. Facial expression database

A database of Japanese female facial expressions (JAFFE), used in [12,13], was collected. Ten people posed three or four examples of each of the six basic facial expressions (happiness, sadness, surprise, anger, disgust, fear) and a neutral face for a total of 213 images of facial expressions [34]. Seven facial expressions of one individual are shown in Fig. 13.

The face region is cropped based on the location of eyes, then resized to 38 × 32 pixels by Daubechies wavelet. The difference images, obtained by subtracting each expression image from its corresponding neutral pose, are used instead of the original facial expressive images, due to the fact that in the difference images, the facial parts in motion are emphasized. Fig. 14 shows the difference images of Fig. 13.

5.7. Experiments on JAFFE database

The experiments are conducted similarly as [12]. Comparing with the former face databases, this database is relatively simple, having the variation only in expressions. However, it is a different

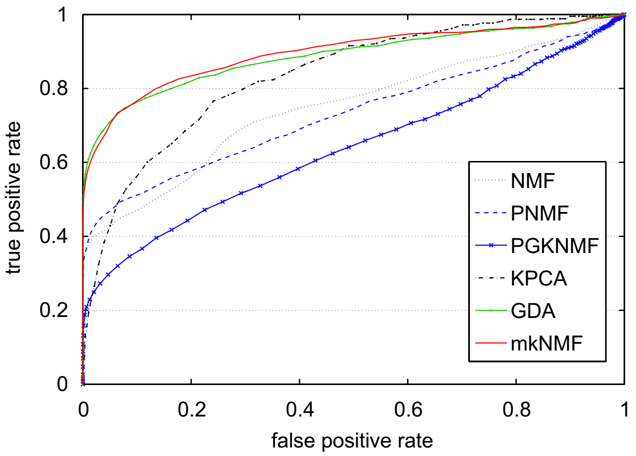

Fig. 10. ROC curve comparisons for FERET database.

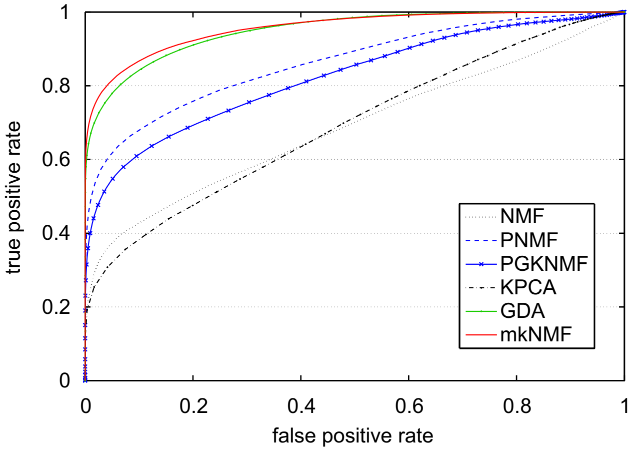

Fig. 11. ROC curve comparisons for CMU PIE database.

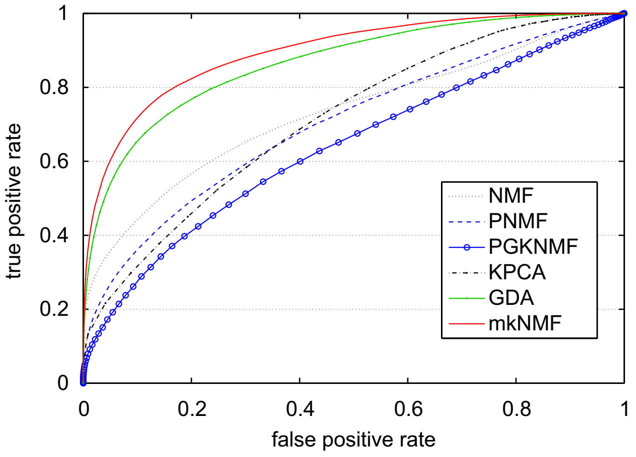

Fig. 12. ROC curve comparisons for AR database.

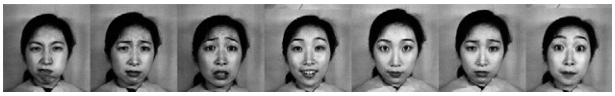

Fig. 13. Seven facial expressions of one individual.

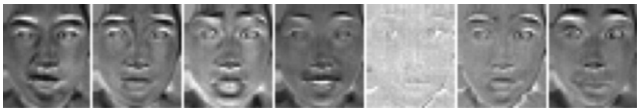

Fig. 14. Difference images of Fig. 13.

Table 3
Accuracy (%) comparison on JAFFE database.

Classifier	NMF	PNMF	PGKNMF	KPCA	GDA	mkNMF
EM	90.96	88.90	88.77	83.56	91.10	91.78
CSM	89.86	89.32	87.53	87.67	87.40	90.68

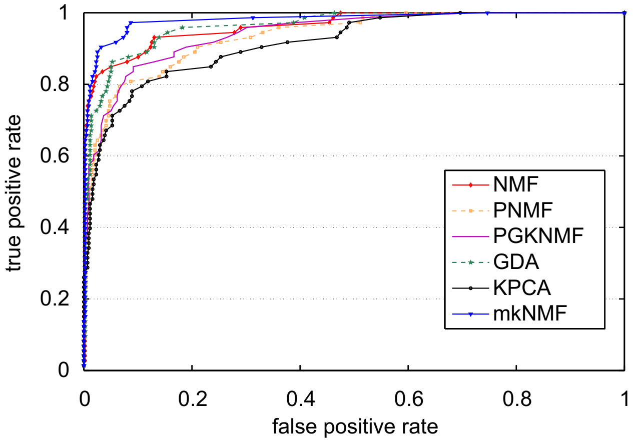

Fig. 15. ROC curve comparisons for JAFFE database.

task. The subspace methods are also suitable for this kind of task. We randomly select 140 images including each individual's seven facial expressions for training and the rest for testing. The rank of NMF is set to 49. Block parameter $r0=7$. Nearest neighbor classifiers based on Euclidean measure (EM) and cosine similarity measure (CSM) are adopted in the experiments. Table 3 gives the accuracies and Fig. 15 shows the ROC curve. The results indicate that mkNMF gives the best performance as well.

5.8. *Running time*

We compare the running time of the evaluated methods. Since we only focus on the computational efficiency, we use the whole FERET database as a training set. The NMF-based methods are the iterative procedures, of which the complexity depends on the times of iteration. The setting is the same as that in the foregoing experiments — we choose 1000 iterations for NMF, PNMF and mkNMF ((40) executes 50 iterations and (41) runs 1000 iterations), while the running time of PGKNMF is computed to give the same ending cost of PNMF. KPCA and GDA require to compute the eigendecomposition of the kernel matrix, of which the complexity is cubic in the number of data points. One advantage of KPCA and GDA is that the global minimum is found in closed form, therefore they do not suffer from local minima. The hardware configuration is *Intel Core*2.40-GHz CPU and 2 G RAM. All codes are run in Matlab environment. The running times of various algorithms are listed in Table 4.

GDA and KPCA calculate the eigendecomposition of a 720 × 720 matrix, which is not time-consuming using Jacobi methods [35]. The complexity of original PNMF [12] is higher

Table 4
Running time (s) of different methods.

NMF	PNMF	PGKNMF	KPCA	GDA	mkNMF
48.72	10.27	6.11	8.85	5.75	80.64

than that of NMF. However, it reduces to a large extent by incorporating the block strategy. As shown in [13], PGKNMF reaches the minimum faster than PNMF, therefore it needs less time to arrive the ending cost. The computational expense of mkNMF is higher than other methods. This is because it explicitly maps the data to a higher dimensional space. A faster algorithm of mkNMF would be a topic of future work.

5.9. Discussions

The main challenges of face recognition include pose, expression, occlusion and illumination variations. An efficient way for tackling this problem is using nonlinear methods. Three face databases selected in the paper are well known as these nonlinear variations. The experimental results show that the proposed method performs the best, validating that it is a promising method in solving the nonlinear variations in face recognition.

Remarkably, mkNMF method is marginal better than the GDA algorithm. GDA is regarded as the state-of-the-art kernel method in face recognition. Furthermore, GDA is a supervised method which exploits the label information to improve the performance. As an unsupervised method, mkNMF achieves a comparable, even slightly better, performance, showing the power of this method in face recognition. How to utilize the label information efficiently in mkNMF to improve the performance is an interesting topic in the future work.

From the ROC curves, the AR database is the most challenging one among the three face databases. This is because the variations of AR database is more complicated, containing the occlusions which do not exist in other two databases. The performances of many methods are degraded. However, the proposed method still gives satisfactory performance, demonstrating its robustness.

We also evaluate the methods on a different task — facial expression recognition. The purpose of this comparison is to show that the proposed method can deal with other similar task. Such comparison is also executed in [12,13]. Again, the mkNMF algorithm outperforms other methods.

6. Conclusions

In this paper, we have presented a nonlinear NMF to tackle complex and nonlinear data distribution. The proposed algorithm remedies the drawback of semi-nonnegativity of the existing kernel NMF methods. The nonnegative constraints on both bases and coefficients are compatible with part-based representation, and thus offer advantages in object recognition. According to the experimental results, the proposed method outperforms KPCA, GDA, NMF, PNMF and PGKNMF algorithms in face and facial expression recognition.

There are still some significant issues for future investigation. How to find an appropriate kernel is crucial for the performance of kernel NMF method. Dimensionality selection of kernel feature space is another important problem deserving to be further researched.

Acknowledgments

This paper is supported by the NSFC (U0835005, 60803083, 60873168), GuangDong Program (2010B031000004), and Science & Technology Planning Project of Shenzhen City (JC200903130300A, JYC200903250175A).

References

[1] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.
[2] D.D. Lee, H.S. Seung, Algorithms for non-negative matrix factorization, in: Advances in Neural Information Processing Systems, 2001, pp. 556–562.
[3] S.Z. Li, X. Hou, H. Zhang, Q. Cheng, Learning spatially localized, parts-based representation, in: Proceedings of International Conference on Computer Vision and Pattern Recognition, Kauai, HI, 2001, pp. 606–610.
[4] S. Bucak, B. Gunsel, Incremental subspace learning via non-negative matrix factorization, Pattern Recognition 42 (2009) 788–797.
[5] W. Xu, X. Liu, Y. Gong, Document clustering based on non-negative matrix factorization, in: Proceedings of 26th Annual International ACM SIGIR Conference, Toronto, Canada, 2003, pp. 267–273.
[6] P. Paatero, U. Tapper, Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values, Environmetrics 5 (1994) 111–126.
[7] Y.-L. Xie, P. Hopke, P. Paatero, Positive matrix factorization applied to a curve resolution problem, J. Chemometrics 12 (1999) 357–364.
[8] P. Fogel, S.S. Young, D.M. Hawkins, N. Ledirac, Inferential robust non-negative matrix factorization analysis of microarray data, Bioinformatics 23 (2007) 44–49.
[9] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, B. Schölkopf, An introduction to kernel-based learning algorithms, IEEE Trans. Neural Networks 12 (2001) 181–201.
[10] B. Schölkopf, A.J. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (1998) 1299–1319.
[11] G. Baudat, F. Anouar, Generalized discriminant analysis using a kernel approach, Neural Comput. 12 (2000) 2385–2404.
[12] I. Buciu, N. Nikolaidis, I. Pitas, Nonnegative matrix factorization in polynomial feature space, IEEE Trans. Neural Networks 19 (2008) 1090–1100.
[13] S. Zafeiriou, M. Petrou, Nonlinear nonnegative component analysis algorithms, IEEE Trans. Image Process. 19 (2010) 1050–1066.
[14] C.J. Lin, Projected gradient methods for nonnegative matrix factorization, Neural Comput. 19 (2007) 2756–2779.
[15] C. Ding, T. Li, M.I. Jordan, Convex and semi-nonnegative matrix factorizations, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 45–55.
[16] B. Klingenberg, J. Curry, A. Dougherty, Non-negative matrix factorization: ill-posedness and a geometric algorithm, Pattern Recognition 42 (2009) 918–928.
[17] N. Gillis, F. Glineur, Using underapproximations for sparse nonnegative matrix factorization, Pattern Recognition 43 (2010) 1676–1687.
[18] T. Zhang, B. Fang, Y.Y. Tang, G. He, J. Wen, Topology preserving non-negative matrix factorization for face recognition, IEEE Trans. Image Process. 17 (2008) 574–584.
[19] A. Pascual-Montano, J.M. Carazo, K. Kochi, D. Lehmann, R.D. Pascual-Marqui, Nonsmooth nonnegative matrix factorization (nsnmf), IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 403–415.
[20] J. Nocedal, S.J. Wright, Numerical Optimization, Springer-Verlag, 1999.
[21] V.N. Vapnik, Statistical Learning Theory, Wiley, New York, NY, USA, 1998.
[22] P.J. Phillips, H. Moon, S.A. Rizvi, P.J. Rauss, The Feret evaluation methodology for face-recognition algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 1090–1104.
[23] M. Turk, A.P. Pentland, Eigenfaces for recognition, J. Cogn. Neurosci. 3 (1991) 71–86.
[24] B. Schölkopf, A.J. Smola, Learning with Kernels, MIT Press, Cambridge, MA, USA, 2002.
[25] D. Martinez, A. Bray, Nonlinear blind source separation using kernels, IEEE Trans. Neural Networks 14 (2003) 228–235.
[26] M. Novak, R. Mammone, Use of nonnegative matrix factorization for language model adaptation in a lecture transcription task, in: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 1, Salt Lake City, UT, 2001, pp. 541–544.
[27] M.D. Plumbley, S.A. Abdallah, J.P. Bello, M.E. Davies, G. Monti, M.B. Sandler, Automatic music transcription and audio source separation, Cybern. Syst. 33 (2002) 603–627.
[28] J.S. Taylor, N. Cristianini, Kernel Method for Pattern Analysis, Cambridge University Press, Cambridge, U.K., 2004.
[29] W.S. Chen, B.B. Pan, B. Fang, M. Li, J.L. Tang, Incremental nonnegative matrix factorization for face recognition, Math. Probl. Eng. (2008) 17 pp, Article ID 410674.
[30] T. Sim, S. Baker, M. Bsat, The CMU pose, illumination, and expression database, IEEE Trans. Pattern Anal. Mach. Intell. 25 (2003) 1615–1618.
[31] A.M. Martinez, R. Benavente, The AR face database, CVC Technical Report # 24, 1998.

[32] K. Le, K. Dabke, G. Egan, Hyperbolic wavelet family, Rev. Sci. Instrum. 75 (2004) 4678–4693.

[33] M. Pepe, Receiver operating characteristic methodology, J. Am. Stat. Assoc. 95 (2000) 308–311.

[34] M.J. Lyons, S. Akamatsu, M. Kamachi, J. Gyoba, Coding facial expressions with Gabor wavelets, in: Proceedings of Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998, pp. 200–205.

[35] G. Golub, C. Loan, Matrix Computations, third ed., Johns Hopkins, Baltimore, 1996.

Binbin Pan received his B.Sc. and M.Sc. degrees in Applied Mathematics from Shenzhen University, China in 2006 and 2009, respectively. He is currently pursuing a Ph.D. in the School of Mathematics and Computational Science at Sun Yat-Sen University. His research interests include matrix factorization, sparse representations and kernel methods with application to pattern recognition and machine learning.

Jianhuang Lai received his M.Sc. degree in Applied Mathematics in 1989 and his Ph.D. in Mathematics in 1999 from Sun Yat-Sen University, China. He joined Sun Yat-Sen University in 1989 as an Assistant Professor, where currently, he is a Professor with the Department of Automation of School of Information Science and Technology and vice dean of School of Information Science and Technology.

Dr. Lai had successfully organized the International Conference on Advances in Biometric Personal Authentication'2004, which was also the Fifth Chinese Conference on Biometric Recognition (Sinobiometrics'04), Guangzhou, in December 2004. He has taken charge of more than five research projects, including NSF-Guangdong (no. U0835005), NSFC (nos. 60144001, 60373082, 60675016), the Key (Keygrant) Project of Chinese Ministry of Education (no. 105134), and NSF of Guangdong, China (nos. 021766, 06023194). He has published over 80 scientific papers in the international journals and conferences on image processing and pattern recognition. His current research interests are in the areas of digital image processing, pattern recognition, multimedia communication, wavelet and its applications.

Prof. Lai serves as a standing member of the Image and Graphics Association of China and also serves as a standing director of the Image and Graphics Association of Guangdong.

Wen-Sheng Chen received the B.Sc. and Ph.D. degrees in Mathematics from Sun Yat-Sen (Zhongshan) University, Guangzhou, China, in 1989 and 1998, respectively. He is currently a Professor in the Institute of Intelligent Computing Science, College of Mathematics and Computational Science, Shenzhen University, Shenzhen, China. His current research interests include pattern recognition, kernel methods, and wavelet analysis and its applications. Dr. Chen is a member of the Chinese Mathematical Society.