Research Article

A Constrained Algorithm Based NMF$_\alpha$ for Image Representation

Chenxue Yang,[1] Tao Li,[1] Mao Ye,[1] Zijian Liu,[2,3] and Jiao Bao[1]

[1] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*
[2] *School of Science, Chongqing Jiaotong University, Chongqing 400074, China*
[3] *Department of Mathematics, Hangzhou Normal University, Hangzhou 310036, China*

Correspondence should be addressed to Mao Ye; cvlab.uestc@gmail.com

Nonnegative matrix factorization (NMF) is a useful tool in learning a basic representation of image data. However, its performance and applicability in real scenarios are limited because of the lack of image information. In this paper, we propose a constrained matrix decomposition algorithm for image representation which contains parameters associated with the characteristics of image data sets. Particularly, we impose label information as additional hard constraints to the α-divergence-NMF unsupervised learning algorithm. The resulted algorithm is derived by using Karush-Kuhn-Tucker (KKT) conditions as well as the projected gradient and its monotonic local convergence is proved by using auxiliary functions. In addition, we provide a method to select the parameters to our semisupervised matrix decomposition algorithm in the experiment. Compared with the state-of-the-art approaches, our method with the parameters has the best classification accuracy on three image data sets.

1. Introduction

Learning an efficient representation of image information is a key problem in machine learning and computer vision. Efficiency of the representation refers to the ability to capture significant information from a high dimensional image space. Such a high dimensional problem is difficult to manipulate and compute; therefore dimension reduction becomes the crucial method to cope with this problem. Fortunately, matrix factorization is a valid approach to solve the dimension reduction problem, and it has a long and successful history in dealing with image representation [1–3]. Some methods of matrix factorization can be referred to as principal component analysis (PCA) [4], singular value decomposition (SVD) [5], vector quantization (VQ) [6], and nonnegative matrix factorization (NMF) [7].

Among all techniques for matrix factorization, NMF is distinguished from others by its use of nonnegative constraints in learning a basis representation of image data [8] and has been applied in face recognition [9–11], medical imaging [12, 13], electroencephalogram (EEG) classification for brain computer interface [14], and many other areas. However, NMF is an unsupervised learning algorithm

and inapplicable to learning a basic representation from limited image information. Thus, to make up for this deficiency, extra constraints are implicitly or explicitly incorporated into NMF to derive some semisupervised matrix decomposition algorithms. In [15], the authors impose label information as additional hard constraints to NMF based on the squared Euclidean distance and Kulback-Leibler divergence. Such a representation encodes the data points from the same class using the indicator matrix in a new representation space, where the obtained part-based representation is more discriminating.

However, none of the semisupervised NMF algorithms mentioned above contain parameters associated with the characteristics of image data sets. In this paper, we introduce α-divergence-NMF algorithm [16], where α is a parameter. We impose the labeled constraints to the α-divergence-NMF algorithm to derive a generic constraint matrix decomposition algorithm which includes some existing algorithms as their special cases: one of them is CNMF$_{KL}$ [15] with $\alpha = 1$. Then, we obtain the proposed algorithm using Karush-Kuhn-Tucker (KKT) method as well as the projected gradient method and prove its monotonic local convergence using an auxiliary function. Comparing to the current semisupervised

NMF algorithms, we analyze the classification accuracy for two fixed values of α ($\alpha = 0.5$, 2) on three image data sets.

The CNMF$_\alpha$ algorithm does not work well for a fixed value of α. Since the parameter α is associated with the characteristics of a learning machine, the model distribution is more inclusive when α goes to $+\infty$ and is more exclusive when α approaches to $-\infty$. The selection of the optimal value of α plays a critical role in determining the discriminative basis vectors. In this paper, we provide a method to select parameters α for our semisupervised CNMF$_\alpha$ algorithm. The variation of α is associated with the characteristics of image data sets. Compared with the algorithms in [15, 16], our algorithm is more complete and systemic.

The rest of the paper is organized as follows. In Section 2, we make a brief overview on standard NMF algorithm and constraint NMF algorithm. The detailed algorithms with labeled constraints and theoretical proof of the convergence of the algorithms are provided in Sections 3 and 4 separately. Section 5 presents some experimental results to show the advantages of our algorithm. Finally, a conclusion is given in Section 6.

2. Related Work

NMF, proposed by Lee and Seung [7], is considered to provide a part-based representation and applied in diverse examples of nonnegative data [17–21] including text data mining, subsystem identification, spectral data analysis, audio and sound processing, and document clustering.

Suppose $X = [x_1, \ldots, x_n] \in R^{m \times n}$ is a set of n training images, where $\{x_t\}_{t=1}^n$ is a column vector and consists of m nonnegative pixel values of a training image. NMF is to find two nonnegative matrix factors $W \in R^{m \times r}$ and $H \in R^{n \times r}$ to approximate the original image matrix

$$X \approx WH^T, \tag{1}$$

where the positive integer r is smaller than m or n.

NMF uses nonnegative constraints to make the representation purely adapted to an unsupervised way. It is inapplicable to learn a basis representation to the limited image information. To make up for this deficiency, extra constraints such as locality [22], sparseness [9], and orthogonality [23] were implicitly or explicitly incorporated into NMF to identify better local features or provide more sparse representation.

In [15], the authors impose label information as additional hard constraints to NMF unsupervised learning algorithm to derive a semisupervised matrix decomposition algorithm, which makes the obtained representation more discriminating. The label information is incorporated as follows.

Suppose $X = \{x_i\}_{i=1}^n$ is a data set, which consists of n training images. Set that the first s images $\{x_1, \ldots, x_s\}$ ($s \leq n$) are represented by the label information, and the remaining $n - s$ images $\{x_{s+1}, \ldots, x_n\}$ ($s \leq n$) are represented by the unlabeled. Assume there exist c classes and each image from

$\{x_1, \ldots, x_s\}$ is designated one class. Then we have an $s \times c$ indicator matrix S, which can be represented as

$$s_{ij} = \begin{cases} 1, & \text{if } x_i \text{ is designated the } j\text{th class,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

From the indicator matrix S, a label constraint matrix U can be defined as

$$U = \begin{pmatrix} S_{s \times c} & 0 \\ 0 & I_{n-s} \end{pmatrix}, \tag{3}$$

where I_{n-s} denotes an $(n - s) \times (n - s)$ identity matrix.

Imposing the label information as additional hard constraint by an auxiliary matrix V, there is $H = UV$. It verifies that $h_i = h_j$ if x_i and x_j have the same labels. With the label constraints, the standard NMF is transformed into factorizing a large-size matrix X into the product of three small-size matrices W, U, and V:

$$X \approx W(UV)^T. \tag{4}$$

Such a representation encodes the data points from the same class using the indicator matrix in a new representation space.

3. A Constrained Algorithm Based NMF$_\alpha$

The exact form of the error measure of (1) is as crucial as the nonnegative constraints in the success of the NMF algorithm in learning a useful representation of image data. In the researches on NMF, there are quite a large number of investigations for error measure, such as Csiszár's f-divergences [24], Amari's α-divergence [25], and Bregman divergences [26]. Here, we introduce a genetic multiplicative updating algorithm [16] which iteratively minimizes the α-divergence between X and WH^T. We define the α-divergence as

$$D_\alpha \left[X \| WH^T \right]$$
$$= \frac{1}{\alpha(1-\alpha)} \sum_{i=1}^m \sum_{j=1}^n \left\{ \alpha X_{ij} + (1-\alpha) \left[WH^T \right]_{ij} - X_{ij}^\alpha \left[WH^T \right]_{ij}^{1-\alpha} \right\}, \tag{5}$$

where α is a positive parameter. We combine the labeled constraints with (4) to derive the following objective function, which is based on the α-divergence between X and $W(UV)^T$,

$$D_\alpha \left[X \| W(UV)^T \right]$$
$$= \frac{1}{\alpha(1-\alpha)}$$
$$\times \sum_{i=1}^m \sum_{j=1}^n \left\{ \alpha X_{ij} + (1-\alpha) \left[WV^T U^T \right]_{ij} - X_{ij}^\alpha \left[WV^T U^T \right]_{ij}^{1-\alpha} \right\}. \tag{6}$$

With the constraints $W_{ij} \geq 0$, $U_{ij} \geq 0$, and $V_{ij} \geq 0$, the minimization of $D_\alpha[X \| W(UV)^T]$ can be formulated as a constrained minimization problem with inequality constraints. In the following, we will present two methods to find a local minimum of (6).

3.1. KKT Method. Let $\lambda_{ij} \geq 0$ and $\mu_{ij} \geq 0$ be the Lagrangian multipliers associated with constraints $V_{ij} \geq 0$ and $W_{ij} \geq 0$, respectively. The Karush-Kuhn-Tucker conditions require that both the optimality conditions

$$\frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial V_{ij}}$$

$$= \frac{1}{\alpha} \left\{ \sum_k [U^T W]_{ki} - \sum_k [U^T W]_{ki} \left(\frac{X_{kj}}{[WV^T U^T]_{kj}} \right)^\alpha \right\} = \lambda_{ij}, \tag{7}$$

$$\frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial W_{ij}}$$

$$= \frac{1}{\alpha} \left\{ \sum_k [V^T U^T]_{jk} - \sum_k [V^T U^T]_{jk} \left(\frac{X_{ik}}{[WV^T U^T]_{ik}} \right)^\alpha \right\} = \mu_{ij} \tag{8}$$

and the complementary slackness conditions

$$\lambda_{ij} V_{ij} = 0,$$
$$\mu_{ij} W_{ij} = 0 \tag{9}$$

are satisfied. If $V_{ij} = 0$ and $W_{ij} = 0$, then either λ_{ij} μ_{ij} can have any values. At the same time, if $\lambda_{ij} = 0$ and $\mu_{ij} = 0$, then $V_{ij} \geq 0$ and $W_{ij} \geq 0$. Hence we need both $\lambda_{ij} \neq 0$ $\mu_{ij} \neq 0$ and $V_{ij} \neq 0$ $W_{ij} \neq 0$. It follows from (9) that

$$\lambda_{ij} V_{ij}^\alpha = 0,$$
$$\mu_{ij} W_{ij}^\alpha = 0. \tag{10}$$

We multiply both sides of (7) and (8) by V_{ij} and W_{ij}, respectively, and incorporate with (10), and then we obtain the following updating rules:

$$V_{ij} \longleftarrow V_{ij} \left[\frac{\sum_k [U^T W]_{ki} \left(X_{kj} / [WV^T U^T]_{kj} \right)^\alpha}{\sum_l [U^T W]_{li}} \right]^{1/\alpha}, \tag{11}$$

$$W_{ij} \longleftarrow W_{ij} \left[\frac{\sum_k [V^T U^T]_{jk} \left(X_{ik} / [WV^T U^T]_{ik} \right)^\alpha}{\sum_l [V^T U^T]_{jl}} \right]^{1/\alpha}. \tag{12}$$

3.2. Projected Gradient Method. Considering the gradient descent algorithm [24, 25], the updating rules for the objective function (6) can be also derived by using the projected gradient method [27] and have the form

$$\phi \left(V_{ij} \right) \longleftarrow \phi \left(V_{ij} \right) + \delta_{ij} \frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial V_{ij}},$$

$$\phi \left(W_{ij} \right) \longleftarrow \phi \left(W_{ij} \right) + \gamma_{ij} \frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial W_{ij}}, \tag{13}$$

where $\phi(\cdot)$ is a suitably chosen function and δ_{ij} and γ_{ij} are two parameters to control the step size of gradient descent. Then, we have

$$V_{ij} \longleftarrow \phi^{-1} \left(\phi \left(V_{ij} \right) + \delta_{ij} \frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial V_{ij}} \right),$$

$$W_{ij} \longleftarrow \phi^{-1} \left(\phi \left(W_{ij} \right) + \gamma_{ij} \frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial W_{ij}} \right). \tag{14}$$

Setting $\phi(\Omega) = \Omega^\alpha$, to guarantee that the updating rules (11) and (12) hold, we need

$$\delta_{ij} = -\frac{\alpha V_{ij}^\alpha}{\sum_k [U^T W]_{ki}}, \qquad \gamma_{ij} = -\frac{\alpha W_{ij}^\alpha}{\sum_k [V^T U^T]_{jk}}. \tag{15}$$

From (7) and (8), the updating rules become

$$V_{ij}^\alpha \longleftarrow V_{ij}^\alpha + \delta_{ij} \frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial V_{ij}}$$

$$= V_{ij}^\alpha \frac{\sum_k [U^T W]_{ki} \left(X_{kj} / [WV^T U^T]_{kj} \right)^\alpha}{\sum_l [U^T W]_{li}},$$

$$W_{ij}^\alpha \longleftarrow W_{ij}^\alpha + \gamma_{ij} \frac{\partial D_\alpha \left[X \| W(UV)^T \right]}{\partial W_{ij}}$$

$$= W_{ij}^\alpha \frac{\sum_k [V^T U^T]_{jk} \left(X_{ik} / [WV^T U^T]_{ik} \right)^\alpha}{\sum_l [V^T U^T]_{jl}}, \tag{16}$$

which are the same as the updating rules (11) and (12). We have shown that the algorithm can be derived using Karush-Kuhn-Tucker conditions and presented alternative of the algorithm using the projected gradient. The use of the two methods guarantees the correctness of the algorithm theoretically.

In the following, we will give a theorem to guarantee the convergence of the iterations in updates (11) and (12).

Theorem 1. *For the objective function (6), $D_\alpha[X\|W(UV)^T]$ is nonincreasing under the updating rules (11) and (12). The objective function D_α is invariant under these updates if and only if V and W are at a stationary point.*

Multiplicative updates for our constrained algorithm based NMF_α are given in (11) and (12). These updates find a local minimum of $D_\alpha[X\|W(UV)^T]$, which is the final solution of (11) and (12). Note that, when $\alpha = 1$, the updates (11) and (12) are the same with CNMF_{KL} algorithm [15], which is a special case included in our generic constraint matrix decomposition algorithm. In the following, we will give the proof of Theorem 1.

4. Convergence Analysis

To prove Theorem 1, we will make use of an auxiliary function that was used in the expectation-maximization algorithm [28, 29].

Definition 2. A function $G(x, \hat{x})$ is defined as an auxiliary function for $F(x)$ if the following two conditions are both satisfied:

$$G(x, x) = F(x), \qquad G(x, \hat{x}) \geq F(x). \tag{17}$$

Lemma 3. *Assume that the function $G(x, \hat{x})$ is an auxiliary function for $F(x)$; then $F(x)$ is nonincreasing under the update*

$$x^{t+1} = \arg\min_{x} G\left(x, x^t\right). \tag{18}$$

Proof. Consider $F(x^{t+1}) \leq G(x^{t+1}, x^t) \leq G(x^t, x^t) = F(x^t)$. ◻

It can be observed that the equality $F(x^{t+1}) = F(x^t)$ holds only if x^t is a local minimum of $G(x, \hat{x})$. We iterate the update in (18) to obtain a sequence of estimates that converge to a local minimum $x_{\min} = \arg\min_{x} F(x)$ of the objective function given by

$$F\left(x_{\min}\right) \leq \cdots \leq F\left(x^{t+1}\right) \leq F\left(x^t\right)$$
$$\leq \cdots \leq F\left(x_2\right) \leq F\left(x_1\right) \leq F\left(x_0\right). \tag{19}$$

In the following, we will show that the objective function (6) is nonincreasing under the updating rules (11) and (12) by defining the appropriate auxiliary functions with respect to V_{ij} and W_{ij}.

Lemma 4. *Function*

$$G\left(V, \widehat{V}\right)$$

$$= \frac{1}{\alpha(1-\alpha)} \sum_{i,j,k} X_{ij} \xi_{ijk}\left(\widehat{V}\right)$$

$$\times \left\{ \alpha + (1-\alpha) \frac{W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij} \xi_{ijk}\left(\widehat{V}\right)} - \left(\frac{W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij} \xi_{ijk}(\widehat{V})}\right)^{1-\alpha} \right\}, \tag{20}$$

where $\xi_{ijk}(\widehat{V}) = W_{ik}[\widehat{V}^T U^T]_{kj} / \sum_l W_{il}[\widehat{V}^T U^T]_{lj}$, is an auxiliary function for

$$F(V) = \frac{1}{\alpha(1-\alpha)}$$

$$\times \sum_{i,j} \alpha X_{ij} + (1-\alpha)\left[W V^T U^T\right]_{ij} - X_{ij}^{\alpha}\left[W V^T U^T\right]_{ij}^{1-\alpha}. \tag{21}$$

Proof. Obviously, $G(V, V) = F(V)$. According to the definition of auxiliary function, we only need to prove $G(V, \widehat{V}) \geq F(V)$. To do this, we use the convex function $f(\cdot)$ for positive α to rewrite the α-divergence function $F(V)$ as

$$F(V) = \sum_{i,j} X_{ij} f\left(\frac{\sum_k W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij}}\right), \tag{22}$$

where

$$f(x) = \frac{1}{\alpha(1-\alpha)}\left[\alpha - (1-\alpha)x - x^{1-\alpha}\right]. \tag{23}$$

Note that $\sum_k \xi_{ijk}(\widehat{V}) = 1$ and $\xi_{ijk}(\widehat{V}) \geq 0$ from the definition of $\xi_{ijk}(\widehat{V})$. Applying Jensen's inequality [30], it leads to

$$f\left(\sum_k W_{ik}\left[V^T U^T\right]_{kj}\right) \leq \sum_k \xi_{ijk}\left(\widehat{V}\right) f\left(\frac{W_{ik}\left[V^T U^T\right]_{kj}}{\xi_{ijk}\left(\widehat{V}\right)}\right). \tag{24}$$

From the above inequality, it follows that

$$F(V) = \sum_{i,j} X_{ij} f\left(\frac{\sum_k W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij}}\right)$$

$$\leq \sum_{i,j,k} X_{ij} \xi_{ijk}\left(\widehat{V}\right) f\left(\frac{W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij} \xi_{ijk}\left(\widehat{V}\right)}\right) \tag{25}$$

$$= G\left(V, \widehat{V}\right),$$

which satisfies the condition of auxiliary function. ◻

Reversing the rules of V_{ij} and W_{ij} in Lemma 4, we define an auxiliary function for the update (12).

Lemma 5. *Function*

$$G\left(W, \widehat{W}\right)$$

$$= \frac{1}{\alpha(1-\alpha)} \sum_{i,j,k} X_{ij} \eta_{ijk}\left(\widehat{W}\right)$$

$$\times \left\{ \alpha + (1-\alpha) \frac{W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij} \eta_{ijk}\left(\widehat{W}\right)} - \left(\frac{W_{ik}\left[V^T U^T\right]_{kj}}{X_{ij} \eta_{ijk}\left(\widehat{W}\right)}\right)^{1-\alpha} \right\}, \tag{26}$$

where $\eta_{ijk}(\widehat{W}) = \widehat{W}_{ik}[V^T U^T]_{kj} / \sum_l \widehat{W}_{il}[V^T U^T]_{lj}$, is an auxiliary function for

$$F(W)$$

$$= \frac{1}{\alpha(1-\alpha)}$$

$$\times \sum_{i,j} \alpha X_{ij} + (1-\alpha)\left[W V^T U^T\right]_{ij} - X_{ij}^{\alpha}\left[W V^T U^T\right]_{ij}^{1-\alpha}. \tag{27}$$

This can be easily proved in the same way as Lemma 4. From Lemmas 4 and 5, now we can demonstrate the convergence of Theorem 1.

Proof. To guarantee the stability of $F(V)$, from Lemma 3, we just need to obtain the minimum of $G(V, \widehat{V})$ with respect to V_{ij}. Set the gradient of (20) to zero; there is

$$
\frac{\partial G\left(V, \widehat{V}\right)}{\partial V_{ij}}
$$

$$
= \frac{1}{\alpha} \sum_k \left[U^T W\right]_{ki} \left\{1 - \left(\frac{W_{ki}\left[V^T U^T\right]_{ij}}{X_{kj} \xi_{kji}\left(\widehat{V}\right)}\right)^{-\alpha}\right\} = 0. \tag{28}
$$

Then, it follows that

$$
V_{ij} = \widehat{V}_{ij} \left[\frac{\sum_k \left[U^T W\right]_{ki} \left(X_{kj} / \sum_l W_{kl}\left[\widehat{V}^T U^T\right]_{lj}\right)^{\alpha}}{\sum_k \left[U^T W\right]_{ki}}\right]^{1/\alpha}, \tag{29}
$$

which is similar to the form of the updating rule (11). Similarly, to guarantee the updating rule (12) holds, the minimum of $G(W, \widehat{W})$, which can be determined by setting the gradient of (26) to zero, must exist. □

Since G is an auxiliary function, according to Lemma 4, F in (21) is nonincreasing under the update (11). Multiplying updates (11) and (12), we can find a local minimum of $D_{\alpha}[X \| W(UV)^T]$.

5. Experiments

In this section, the CNMF$_{\alpha}$ algorithm is systematically compared with the current constrained NMF algorithms on three image data sets, named ORL Database [31], Yale Database [32], and Caltech 101 Database [33]. The details of the above three databases will be described individually later. We introduce the evaluated algorithms firstly.

(i) Constrained nonnegative matrix factorization algorithm in [15] aims to minimize the F-norm cost.

(ii) Constrained nonnegative matrix factorization algorithm with parameter $\alpha = 0.5$ in this paper aims to minimize the Hellinger divergence cost.

(iii) Constrained nonnegative matrix factorization algorithm with parameter $\alpha = 1$ in [15], aiming at minimizing the KL-divergence cost, is the best reported algorithm in image representation.

(iv) Constrained nonnegative matrix factorization algorithm with parameter $\alpha = 2$ in this paper aims to minimize the χ^2-divergence cost.

(v) Constrained nonnegative matrix factorization algorithm with parameter $\alpha = \alpha_k$ in this paper aims to minimize the α-divergence cost, where the parameters α_k are associated with the characteristics of the image database and designed by the presented method. CNMF$_{KL}$ algorithm is a special case of our CNMF$_{\alpha_k}$ algorithm with $\alpha = \alpha_k = 1$.

We apply these algorithms to a problem of classification and evaluate their performance on three image data sets which contain a number of different categories of image. For each date set, the evaluations are conducted with different numbers of clusters; here the number of clusters k varies from 2 to 10. We randomly choose k categories from one image data set and mix the images of these k categories as the collection X. Then, for the semisupervised algorithms, we randomly pick up 10 percent images from each category in X and record their category number as the available label information to obtain the label matrix U. For some special data sets, the label process is different and we will describe the details later.

Suppose a data set has N categories $C_1, C_2 \ldots, C_N$, and the cardinalities of these labeled images are $L_1, L_2 \ldots, L_N$, respectively. Since the label constraint matrix U is composed of the indicator matrix S and the identity matrix I, the indicator matrix S plays a critical role in classification performance for different categories in X. To determine the effectiveness of S, we define a measure to represent the relationship between the cardinalities of labeled samples and the total samples,

$$
r(k) = \frac{L_{\max}(k) - L_{\min}(k)}{\sum_{i=1}^{k} C_i} k, \tag{30}
$$

where $L_{\max}(k)$ and $L_{\min}(k)$ denote the maximum and minimum labeled cardinalities of k categories. For the fixed cluster number k in the data set, $r(k)$ is different if the number of samples in each category is different. Then, we compute α as follows:

$$
\alpha_k = \frac{\alpha_N}{r(N)} \left(|r(k) - r(N)| \right.
$$

$$
\times \left(\left| \frac{L_{\max}(k) k}{\sum_{i=1}^{k} C_i} - \frac{\sum_{i=1}^{N} L_i}{\sum_{i=1}^{N} C_i} \right| \right. \tag{31}
$$

$$
\left. \left. + \left| \frac{L_{\min}(k) k}{\sum_{i=1}^{k} C_i} - \frac{\sum_{i=1}^{N} L_i}{\sum_{i=1}^{N} C_i} \right| \right)^{-1} \right),
$$

where

$$
\alpha_N = \left(L_{\max}(N) - L_{\min}(N)\right)
$$

$$
\times \left(\left| L_{\max}(N) - \frac{\sum_{i=1}^{N} L_i}{\sum_{i=1}^{N} C_i} \right| + \left| L_{\min}(N) - \frac{\sum_{i=1}^{N} L_i}{\sum_{i=1}^{N} C_i} \right| \right)^{-1}. \tag{32}
$$

The value of α_k computed by (31) is associated with the characteristics of image data sets, since its variation is caused by both the cardinalities of labeled samples in each category and the total samples. We can obtain both the cardinalities of labeled samples and the total samples in our semisupervised algorithms. However, we can not get the cardinalities of labeled images exactly in many real-word applications. Moreover, the value of α varies depending on data sets. It is still an open problem how to select the optimal α [16].

To evaluate the classification performance, we define classification accuracy as the first measure. Our CNMF$_{\alpha}$

algorithm described in (11) and (12) provides a classification label of each sample, marked s_{il_i}. Suppose $X = \{x_i\}_{i=1}^n$ is a data set, which consists of n training images. For each sample, let l_i be the true class label provided by the image data set. More specifically, if the image x_i is designated the l_ith class, we evaluate it as a correct label and set $s_{il_i} = 1$. Otherwise, it is counted as a false label and noted $s_{il_i} = 0$. Eventually, we compute the percentage of correct labels obtained by defining the accuracy measure as

$$\mathrm{Ac}\left(s_{il_i}\right) = \frac{\sum_{i=1}^n s_{il_i}}{n}. \tag{33}$$

To evaluate the classification performance, we carry out computation about the normalized mutual information, which is used to measure how similar two sets of clusters are, as the second measure. Given two data sets of clusters C_i and C_j, their normalized mutual information is defined as

$$\mathrm{NMI}\left(C_i, C_j\right)$$
$$= \frac{\sum_{c_i \in C_i, c_j \in C_j} p\left(c_i, c_j\right) \cdot \log\left\{p\left(c_i, c_j\right) / \left[p\left(c_i\right) p\left(c_j\right)\right]\right\}}{\max\left(H\left(C_i\right), H\left(C_j\right)\right)}, \tag{34}$$

which takes values between 0 and 1. Where $p(c_i)$, $p(c_j)$ denote the probabilities that an image arbitrarily chosen from the data set belongs to the clusters C_i and C_j, respectively, and $p(c_i, c_j)$ denotes the joint probability that this arbitrarily selected image belongs to the cluster C_i as well as C_j at the same time. $H(C_i)$ and $H(C_j)$ are the entropies of C_i and C_j, respectively.

Experimental results on each data set will be presented as classification accuracy and the normalized mutual information is in Tables 1 and 2.

5.1. ORL Database. The Cambridge ORL Face Database has 400 images for 40 different people, 10 images per person. The images of some people are taken at different times, varying lighting slightly, facial expressions (open/closed eyes, smiling/nonsmiling), and facial details (glasses/no glasses). All the images are taken against a dark homogeneous background with the subjects in an upright, frontal position and slight left-right out-of-plane rotation. To locate the faces, the input images are preprocessed. They are resized to 32×32 pixels with 256 gray levels per pixel and normalized in orientation so that two eyes in the facial areas are aligned at the same position.

There are 10 images for each category in ORL and 10 percent is just one image. For the fixed parameter α ($\alpha = 0.5, 1, 2$), we randomly choose two images from each category to provide the label information. Note that the same label is meaningless for (30). To obtain $r(k)$, we divide the 40 categories into 3 groups: 10 categories, 20 categories, and 10 categories. In the first 10 categories, pick up 1 image from each category to provide the label information; pick up 2 images from each category in the second 20 categories; and pick up 3 from each category in the remaining categories. The

dividing process is repeated 10 times and the obtained average classification accuracy is recorded as the final result.

Figure 1 shows the graphical classification accuracy rates and normalized mutual information on the ORL Database. Note that if the samples in the collection X come from the same group, we set $\alpha_k = 0.67$. Because of the same number of samples in each category, the variation of the α is small even though we label different cardinalities of samples. Compared to the constrained nonnegative matrix factorization algorithms with fixed parameters, our CNMF_{α_k} algorithm gives the best performance since the selection of α_k is suitable to the collection X. Table 1 summarizes the detailed classification accuracy and error bars of $\mathrm{CNMF}_{\alpha=0.5}$, $\mathrm{CNMF}_{\alpha=2}$, and CNMF_{α_k}. It shows that our CNMF_{α_k} algorithm achieves 1.92 percent improvement compared to the best reported $\mathrm{CNMF}_{\mathrm{KL}}$ algorithm ($\alpha = 1$) [15] in average classification accuracy. For normalized mutual information, the details and the error bars of our constrained algorithms with $\alpha = 0.5$, $\alpha = 2$, and α_k are listed in Table 2. Comparing to the best algorithm CNMF, our CNMF_{α_k} algorithm achieves 0.54 percent improvement.

5.2. Yale Database. The Yale Database consists of 165 grayscale images for 15 individuals, 11 images per person. One per image is taken from different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. We preprocess all the images of Yale Database in the same way as the ORL Database. Each image is linearly stretched to a 1,024-dimensional vector in image space.

The Yale Database also has the same number of samples in each category. To obtain an appropriate α, we do similar label processing to the ORL Database. Divide the 15 individuals into 3 groups averagely, choose 1 image from each category in the first group, choose 2 from each category in the second, and choose 3 from each category in the remaining group. We repeat the process 10 times and record the average classification accuracy as the final result.

Figure 2 shows the classification accuracy and normalized mutual information on the Yale Database. Set $\alpha_k = 0.55$ when $r(k) - r(N) = 0$. It indicates that the samples in the collection X just come from two groups; that is, choose 15 images from 10 categories in the first and second group. CNMF_{α_k} achieves an extraordinary performance for all the cases and $\mathrm{CNMF}_{\alpha=0.5}$ follows. This suggests that the constrained nonnegative matrix factorization algorithm has a higher classification accuracy when the value of α is close to α_k. Comparing to the best reported $\mathrm{CNMF}_{\mathrm{KL}}$ algorithm, CNMF_{α_k} achieves 2.42 percent improvement in average classification accuracy. For normalized mutual information, CNMF_{α_k} achieves 7.2 percent improvement compared to CNMF. The details of classification accuracy and normalized mutual information are provided in Tables 3 and 4, which contain the error bars of $\mathrm{CNMF}_{\alpha=0.5}$, $\mathrm{CNMF}_{\alpha=2}$, and CNMF_{α_k}.

5.3. Caltech 101 Database. The Caltech 101 Database created by Caltech University has images of 101 different object

TABLE 1: The comparison of classification accuracy rates on the ORL Database.

k	Classification accuracy $Ac(s_{il_i})$ (%)				
	CNMF	$CNMF_{\alpha=1}$	$CNMF_{\alpha=0.5}$	$CNMF_{\alpha=2}$	$CNMF_{\alpha_k}$
2	**93.90**	92.95	93.51 ± 0.09	92.14 ± 0.18	93.89 ± 0.91
3	86.20	84.33	85.71 ± 0.83	82.26 ± 1.23	$\mathbf{86.21} \pm 2.07$
4	84.55	84.05	84.74 ± 0.56	83.02 ± 0.47	$\mathbf{84.78} \pm 0.36$
5	**80.16**	78.38	80.10 ± 0.22	80.15 ± 0.18	$\mathbf{80.16} \pm 0.29$
6	76.72	76.73	77.13 ± 0.28	77.60 ± 0.19	$\mathbf{78.09} \pm 0.40$
7	81.14	79.04	80.34 ± 0.81	78.07 ± 0.88	$\mathbf{82.15} \pm 1.95$
8	78.08	77.02	78.66 ± 0.69	78.77 ± 0.94	$\mathbf{79.45} \pm 1.33$
9	83.19	82.23	83.57 ± 0.36	81.22 ± 0.21	$\mathbf{85.19} \pm 1.21$
10	77.03	76.88	77.69 ± 0.12	77.75 ± 0.19	$\mathbf{78.96} \pm 0.73$
Avg.	82.33	81.29	82.38	81.22	**83.21**

TABLE 2: The comparison of normalized mutual information on the ORL Database.

k	Normalized mutual information (%)				
	CNMF	$CNMF_{\alpha=1}$	$CNMF_{\alpha=0.5}$	$CNMF_{\alpha=2}$	$CNMF_{\alpha_k}$
2	**78.24**	75.82	76.01 ± 0.14	73.94 ± 0.11	77.67 ± 0.58
3	**76.61**	75.36	75.38 ± 0.41	74.50 ± 0.53	75.92 ± 1.55
4	78.69	77.82	77.96 ± 0.47	77.04 ± 0.54	$\mathbf{79.27} \pm 1.26$
5	**76.69**	74.27	75.32 ± 0.80	72.62 ± 0.87	75.35 ± 1.67
6	76.52	76.09	76.21 ± 0.12	76.09 ± 0.67	$\mathbf{77.60} \pm 1.82$
7	82.45	81.35	82.36 ± 0.42	81.35 ± 0.26	$\mathbf{83.08} \pm 1.59$
8	80.57	79.92	80.11 ± 0.73	79.91 ± 0.68	$\mathbf{82.49} \pm 1.97$
9	86.03	85.67	86.73 ± 0.70	85.67 ± 0.37	$\mathbf{88.05} \pm 1.15$
10	81.77	81.07	82.07 ± 0.12	81.07 ± 0.12	$\mathbf{82.97} \pm 1.71$
Avg.	79.73	78.60	79.13	78.02	**80.27**

categories. Each category contains about 31 to 800 images with a total of 9,144 samples of size roughly 300×300 pixels. This database is particularly challenging for learning a basis representation of image information, because the number of training samples per category is exceedingly small. In our experiment, we select the 10 largest categories (3,044 images in total), except the background category. To represent the input images, we do the preprocessing by using the codewords generated by SIFT features [34]. Then we obtain 555,292 SIFT descriptors and generate 500 codewords. By assigning the descriptors to the closest codewords, each image in Caltech 101 database is represented by a 500-dimensional frequency histogram.

We randomly select k categories from Faces-Easy category in Caltech 101 database and convert them to gray-scale of 32×32. The label process is repeated 10 times and the obtained values of α_k computed by (31) are listed in Table 7. The variation of the α in the same k categories is great. That is, selecting an appropriate α plays a critical role for the mixture of different categories in one image data set, especially in the case that the number of samples in each category is different. The choice of α_k can fully reflect the effectiveness of the indicator matrix S.

Figure 3 shows the effect that derived from the using of proposed algorithm with α_k. The upper part of figure is the original samples which contain 26 images, the middle part is their gray images, and the lower is the combination of the basis vectors learned by $CNMF_{\alpha_k}$.

The classification accuracy results and normalized mutual information for Faces-Easy category in Caltech 101 database are detailed in Tables 5 and 6, which contain the error bars of $CNMF_{\alpha=0.5}$, $CNMF_{\alpha=2}$, and $CNMF_{\alpha_k}$. The graphical results of classification performance are shown in Figure 4. The best performance in this experiment is achieved when the parameters α_k listed in Table 7 were selected. In general, our method demonstrates much better effectiveness in classification by choosing α_k. Comparing to the best reported algorithm other than our $CNMF_\alpha$ algorithm, $CNMF_{\alpha_k}$ achieves 2.4 percent improvement in average classification accuracy, and comparing to the CNMF algorithm [15] other than $CNMF_\alpha$ algorithm, $CNMF_{\alpha_k}$ achieves 8.77 percent improvement in average classification accuracy. For normalized mutual information, $CNMF_{\alpha_k}$ achieves 2.29 percent improvement and consistently outperforms the other algorithms.

6. Conclusion

In this paper, we present a generic constraint nonnegative matrix factorization algorithm by imposing label information as additional hard constraint to the α-divergence-NMF algorithm. The proposed algorithm can be derived using

TABLE 3: The comparison of classification accuracy rates on the Yale Database.

k	$\text{Ac}(s_{il_i})$ (%)				
	CNMF	CNMF$_{\alpha=1}$	CNMF$_{\alpha=0.5}$	CNMF$_{\alpha=2}$	CNMF$_{\alpha_k}$
2	81.64	85.23	85.93 ± 0.65	84.18 ± 0.81	**86.27 ± 2.46**
3	68.48	76.91	77.14 ± 0.86	77.26 ± 0.42	**79.01 ± 1.09**
4	64.25	66.70	68.11 ± 0.67	66.12 ± 0.16	**69.33 ± 2.23**
5	65.82	67.71	69.14 ± 1.21	66.54 ± 1.01	**70.57 ± 2.22**
6	60.12	64.26	65.62 ± 0.94	63.70 ± 0.94	**66.92 ± 1.41**
7	59.25	62.47	63.97 ± 0.72	59.83 ± 0.38	**65.23 ± 1.77**
8	54.40	59.26	60.12 ± 0.31	56.76 ± 0.07	**60.89 ± 1.28**
9	54.85	57.30	58.08 ± 0.12	54.88 ± 0.14	**58.98 ± 0.62**
10	52.68	56.20	**57.56 ± 0.03**	53.83 ± 0.01	**57.56 ± 0.73**
Avg.	62.39	65.89	67.30	64.79	**68.31**

TABLE 4: The comparison of normalized mutual information on the Yale Database.

k	Normalized mutual information (%)				
	CNMF	CNMF$_{\alpha=1}$	CNMF$_{\alpha=0.5}$	CNMF$_{\alpha=2}$	CNMF$_{\alpha_k}$
2	48.99	59.66	59.73 ± 0.39	58.17 ± 0.17	**61.11 ± 1.82**
3	42.90	53.52	53.58 ± 0.73	52.75 ± 0.93	**54.49 ± 1.74**
4	47.80	53.60	53.66 ± 0.18	52.99 ± 0.65	**54.58 ± 1.64**
5	53.99	57.49	57.56 ± 0.07	56.21 ± 0.06	**58.58 ± 2.32**
6	50.63	55.88	56.15 ± 0.07	55.40 ± 0.08	**57.62 ± 1.09**
7	53.03	57.60	57.88 ± 0.22	57.27 ± 0.16	**59.05 ± 1.89**
8	50.83	56.52	56.93 ± 0.13	56.20 ± 0.13	**58.98 ± 1.46**
9	53.08	55.30	55.98 ± 0.73	54.81 ± 0.81	**56.15 ± 1.20**
10	52.84	56.73	57.43 ± 0.16	56.41 ± 0.41	**58.30 ± 1.11**
Avg.	50.45	56.26	56.54	55.58	**57.65**

(a) Classification accuracy rates

(b) Normalized mutual information
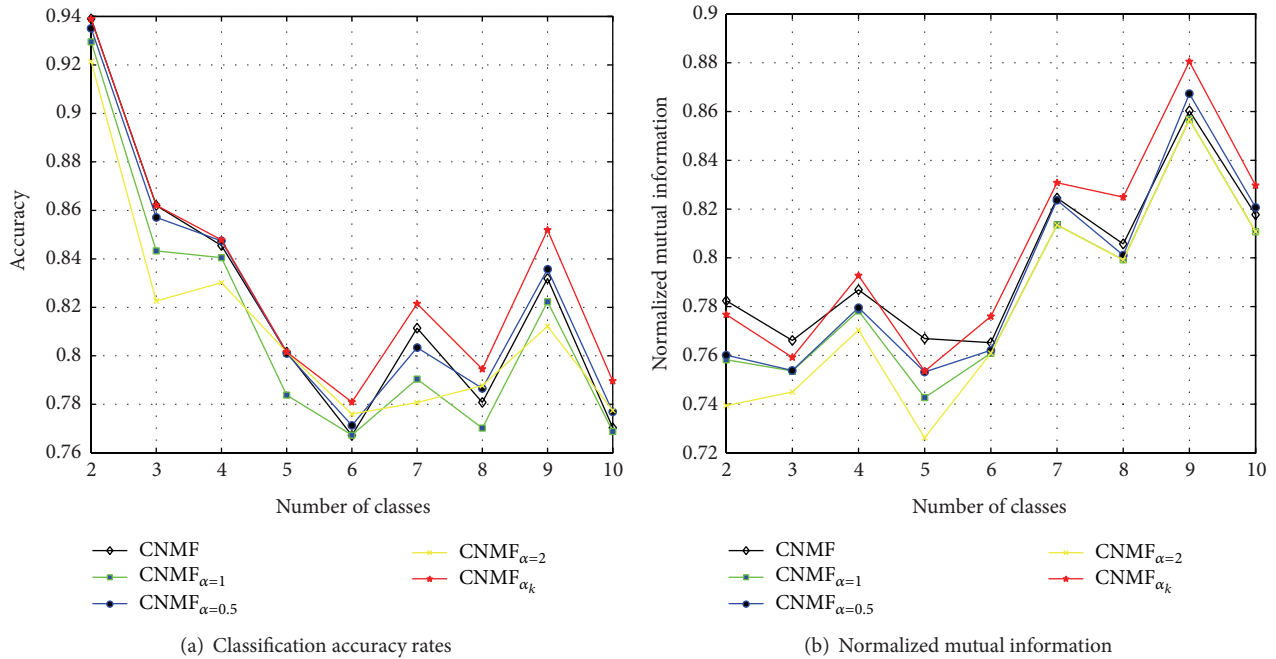
FIGURE 1: Classification performance on the ORL Database.

(a) Classification accuracy rates

(b) Normalized mutual information
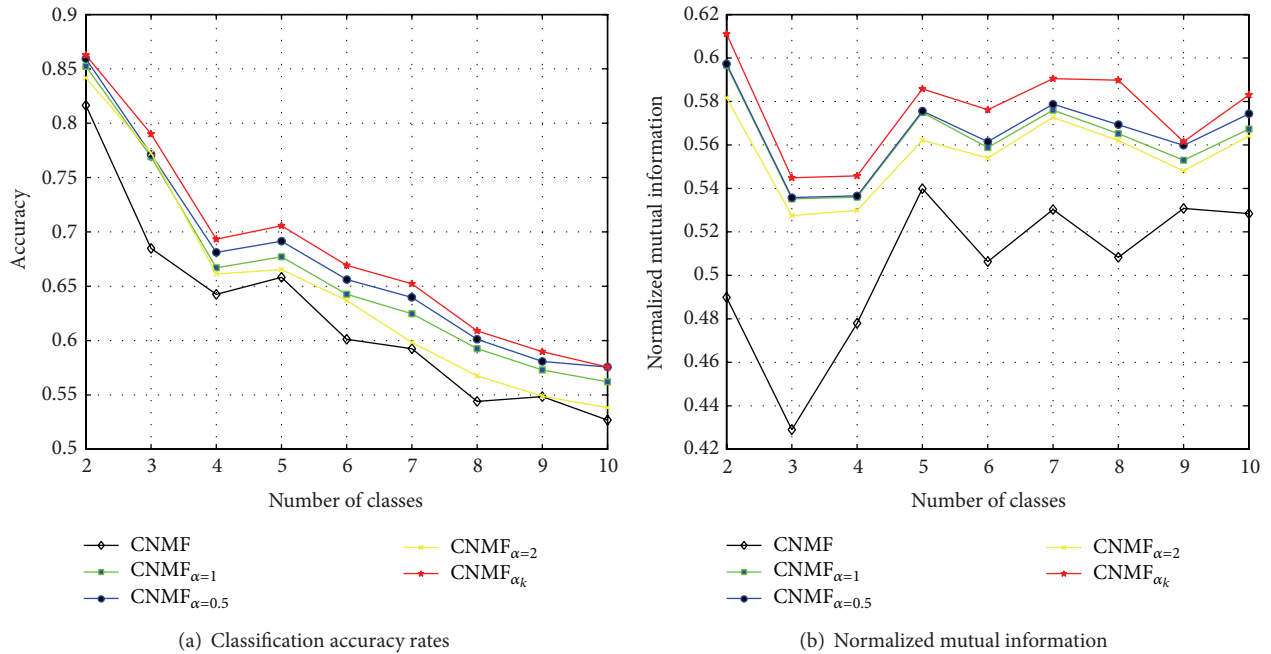
FIGURE 2: Classification performance on the Yale Database.
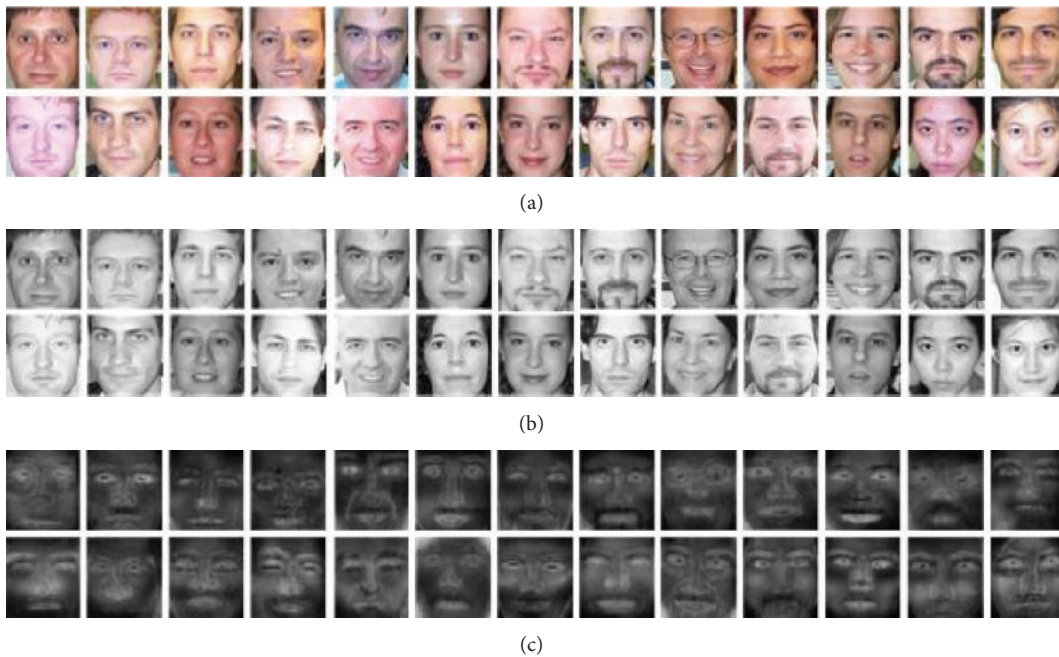

(a)

(b)

(c)

FIGURE 3: Sample images used in YaleB Database. As shown in the three 2×13 montages, (a) is the original images, (b) is the gray images, and (c) is the basis vectors learned by CNMF_{α_k}. Positive values are illustrated with white pixels.

Karush-Kuhn-Tucker conditions and presented alternative of the algorithm using the projected gradient. The use of the two methods guarantees the correctness of the algorithm theoretically. The image representation learned by our algorithm contains a parameter α. Since α-divergence is a parametric discrepancy measure and the parameter α is associated with the characteristics of a learning machine, the

model distribution is more inclusive when α goes to $+\infty$ and is more exclusive when α approaches $-\infty$. The selection of the optimal value of α plays a critical role in determining the discriminative basis vectors. We provide a method to select the parameters α for our semisupervised CNMF_{α} algorithm. The variation of the α is caused by both the cardinalities of labeled samples in each category and the total

(a) Classification accuracy rates
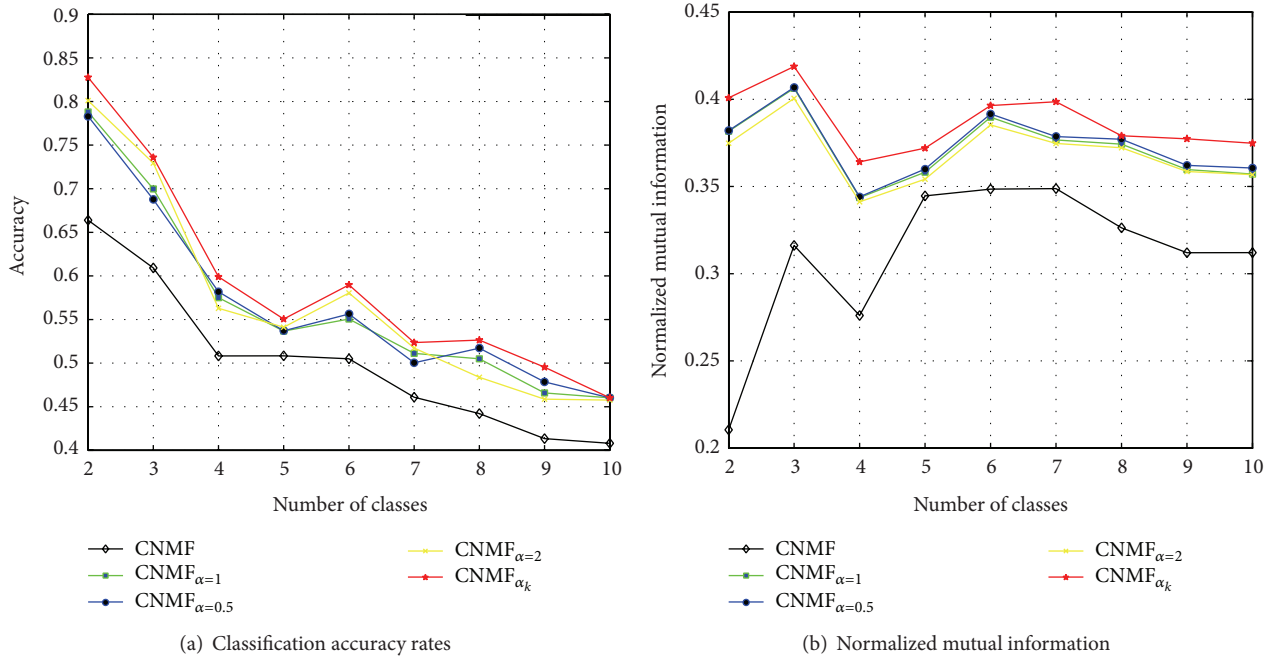
(b) Normalized mutual information

FIGURE 4: Classification performance on the Caltech 101 Database.

TABLE 5: The comparison of classification accuracy rates on the Caltech 101 Database.

k	$Ac(s_{il_i})$ (%)				
	CNMF	CNMF$_{\alpha=1}$	CNMF$_{\alpha=0.5}$	CNMF$_{\alpha=2}$	CNMF$_{\alpha_k}$
2	66.39	78.80	78.31 ± 1.69	80.10 ± 2.07	**82.76** ± 3.56
3	60.91	69.98	68.77 ± 1.57	72.94 ± 1.51	**73.59** ± 3.51
4	50.82	57.50	58.19 ± 1.98	56.28 ± 1.96	**59.89** ± 2.82
5	50.83	53.67	53.71 ± 1.02	54.12 ± 1.44	**55.06** ± 2.63
6	50.49	55.05	55.63 ± 0.89	58.04 ± 1.47	**58.97** ± 1.97
7	46.09	51.12	50.04 ± 0.11	51.73 ± 0.25	**52.36** ± 1.81
8	44.21	50.50	51.72 ± 0.51	48.37 ± 1.59	**52.64** ± 1.52
9	41.34	46.58	47.85 ± 0.28	45.87 ± 0.30	**49.54** ± 1.23
10	40.80	**46.02**	**46.02** ± 0.05	45.76 ± 0.08	**46.02** ± 0.26
Avg.	50.21	56.58	56.69	57.02	**58.98**

TABLE 6: The comparison of normalized mutual information on the Caltech 101 Database.

k	Normalized mutual information (%)				
	CNMF	CNMF$_{\alpha=1}$	CNMF$_{\alpha=0.5}$	CNMF$_{\alpha=2}$	CNMF$_{\alpha_k}$
2	21.05	38.17	38.20 ± 0.18	37.48 ± 0.21	**40.08** ± 2.19
3	31.62	40.63	40.68 ± 0.15	40.05 ± 0.14	**41.88** ± 1.24
4	27.60	34.36	34.41 ± 0.79	34.12 ± 0.91	**36.41** ± 1.17
5	34.46	35.82	36.00 ± 0.99	35.42 ± 1.01	**37.20** ± 1.78
6	34.85	38.97	39.16 ± 0.86	38.53 ± 1.11	**39.64** ± 1.83
7	34.88	37.67	37.86 ± 0.47	37.47 ± 0.49	**39.86** ± 2.58
8	32.63	37.43	37.71 ± 0.39	37.22 ± 0.42	**37.91** ± 2.18
9	31.20	35.97	36.21 ± 0.60	35.86 ± 0.63	**37.73** ± 0.67
10	31.21	35.71	36.06 ± 0.02	35.68 ± 0.03	**37.48** ± 0.22
Avg.	31.05	37.19	37.37	36.87	**38.69**

TABLE 7: The value of α in k categories.

k	α_1	α_2	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}
2	2.07	2.08	2.22	2.27	2.57	3.52	3.97	3.98	4.37	8.96
3	0.45	0.55	0.66	1.07	1.95	2.31	2.31	3.54	4.24	7.54
4	0.30	0.32	0.35	0.86	0.98	1.27	1.65	2.10	3.27	5.80
5	0.45	0.49	0.51	0.71	0.80	1.13	2.05	2.55	2.82	3.89
6	0.07	0.56	0.73	1.08	1.26	1.46	2.01	2.12	3.91	8.52
7	0.71	0.94	1.12	1.14	1.16	1.26	1.27	1.37	1.57	8.97
8	0.12	0.30	0.50	0.64	0.66	0.69	0.78	0.79	0.97	1.11
9	0.78	0.79	2.01	0.10	0.17	0.28	0.29	0.30	0.31	0.39
10	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

samples. In the experiments, we apply the fixed parameters α and α_k to analyze the classification accuracy on three image databases. The experimental results have demonstrated that the CNMF_{α_k} algorithm has best classification accuracy. However, we can not get the cardinalities of labeled images exactly in many real-word applications. Moreover, the value of α varies depending on data sets. It is still an open problem how to select the optimal α for a specific image data set.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] I. Biederman, "Recognition-by-components: a theory of human image understanding," *Psychological Review*, vol. 94, no. 2, pp. 115–147, 1987.

[2] S. Ullman, *High-Level Vision: Object Recognition and Visual Cognition*, MIT Press, Cambridge, Mass, USA, 1996.

[3] P. Paatero and U. Tapper, "Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[4] I. T. Jolliffe, *Principle Component Analysis*, Springer, Berlin, Germany, 1986.

[5] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley-Interscience, New York, NY, USA, 2001.

[6] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.

[7] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[8] M. Chu, F. Diele, R. Plemmons, and S. Ragni, "Optimality, computation, and interpretation of nonnegative matrix factorizations," *SIAM Journal on Matrix Analysis*, pp. 4–8030, 2004.

[9] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. S. Cheng, "Learning spatially localized, parts-based representation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, pp. I207–I212, Kauai, Hawaii, USA, December 2001.

[10] Y. Wang, Y. Jia, H. U. Changbo, and M. Turk, "Non-negative matrix factorization framework for face recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, no. 4, pp. 495–511, 2005.

[11] Y. Wang, Y. Jia, C. Hu, and M. Turk, "Fisher non-negative matrix factorization for learning local features," in *Proceedings of the Asian Conference on Computer Vision*, Jeju Island, Republic of Korea, 2004.

[12] J. H. Ahn, S. Kim, J. H. Oh, and S. Choi, "Multiple nonnegative-matrix factorization of dynamic PET images," in *Proceedings of the Asian Conference on Computer Vision*, 2004.

[13] J. S. Lee, D. D. Lee, S. Choi, and D. S. Lee, "Application of n onnegative matrix factorization to dynamic positron emission tomography," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation*, pp. 629–632, San Diego, Calif, USA, 2001.

[14] H. Lee, A. Cichocki, and S. Choi, "Nonnegative matrix factorization for motor imagery EEG classification," in *Proceedings of the International Conference on Artificial Neural Networks*, Springer, Athens, Greece, 2006.

[15] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1299–1311, 2012.

[16] A. Cichocki, H. Lee, Y.-D. Kim, and S. Choi, "Non-negative matrix factorization with α-divergence," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1433–1440, 2008.

[17] L. K. Saul, F. Sha, and D. D. Lee, "Tatistical signal processing with nonnegativity constraints," *Proceedings of EuroSpeech*, vol. 2, pp. 1001–1004, 2003.

[18] P. M. Kim and B. Tidor, "Subsystem identification through dimensionality reduction of large-scale gene expression data," *Genome Research*, vol. 13, no. 7, pp. 1706–1718, 2003.

[19] V. P. Pauca, J. Piper, and R. J. Plemmons, "Nonnegative matrix factorization for spectral data analysis," *Linear Algebra and its Applications*, vol. 416, no. 1, pp. 29–47, 2006.

[20] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proceedings of*

IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 177–180, New Paltz, NY, USA, 2003.

[21] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, "Document clustering using nonnegative matrix factorization," *Information Processing and Management*, vol. 42, no. 2, pp. 373–386, 2006.

[22] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[23] S. Choi, "Algorithms for orthogonal nonnegative matrix factorization," in *Proceedings of the International Joint Conference on Neural Networks*, pp. 1828–1832, June 2008.

[24] A. Cichocki, R. Zdunek, and S. Amari, "Csiszar's divergences for nonnegative matrix factorization: family of new algorithms," in *Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation*, Charleston, SC, USA, 2006.

[25] A. Cichocki, R. Zdunek, and S.-I. Amari, "New algorithms for non-negative matrix factorization in applications to blind source separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '06)*, pp. V621–V624, Toulouse, France, May 2006.

[26] I. S. Dhillon and S. Sra, "Generalized nonnegative matrix approximations with Bregman divergences," in *Advances in Neural Information Processing Systems 8*, pp. 283–290, MIT Press, Cambridge, Mass, USA, 2006.

[27] A. Cichocki, S. Amari, and R. Zdunek, "Extended SMART algorithms for non-negative matrix factorization," in *Proceedings of the 8th International Conference on Artificial Intelligence and Soft Computing*, Zakopane, Poland, 2006.

[28] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society Series B: Methodological*, vol. 39, no. 1, pp. 1–38, 1977.

[29] L. Saul and F. Pereira, "Aggregate and mixed-order Markov models for statistical language processing," in *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing*, C. Cardie and R. Weischedel, Eds., pp. 81–89, ACL Press, 1997.

[30] F. Hansen and G. Kjærgård Pedersen, "Jensen's inequality for operators and Löwner's theorem," *Mathematische Annalen*, vol. 258, no. 3, pp. 229–241, 1982.

[31] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase .html.

[32] http://vision.ucsd.edu/content/yale-face-database.

[33] http://www.vision.caltech.edu/Image_Datasets/Caltech101/.

[34] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.