Topographic NMF for Data Representation

Yanhui Xiao, Zhenfeng Zhu, Yao Zhao, *Senior Member, IEEE,* Yunchao Wei, Shikui Wei, and Xuelong Li, *Fellow, IEEE*

***Abstract*—Nonnegative matrix factorization (NMF) is a useful technique to explore a parts-based representation by decomposing the original data matrix into a few parts-based basis vectors and encodings with nonnegative constraints. It has been widely used in image processing and pattern recognition tasks due to its psychological and physiological interpretation of natural data whose representation may be parts-based in human brain. However, the nonnegative constraint for matrix factorization is generally not sufficient to produce representations that are robust to local transformations. To overcome this problem, in this paper, we proposed a topographic NMF (TNMF), which imposes a topographic constraint on the encoding factor as a regularizer during matrix factorization. In essence, the topographic constraint is a two-layered network, which contains the square nonlinearity in the first layer and the square-root nonlinearity in the second layer. By pooling together the structure-correlated features belonging to the same hidden topic, the TNMF will force the encodings to be organized in a topographical map. Thus, the feature invariance can be promoted. Some experiments carried out on three standard datasets validate the effectiveness of our method in comparison to the state-of-the-art approaches.**

***Index Terms*—Data clustering, dimension reduction, feature invariance, machine learning, nonnegative matrix factorization.**

I. Introduction

DATA representation is a fundamental problem in image processing and pattern recognition tasks. A good representation can typically reveal the latent structure of data, and further facilitate these tasks in terms of learnability and computational complexity [1]–[7]. However, in many real applications, the input data matrix is generally of very high dimension, which brings the curse of dimensionality for further data processing [8]. To solve this problem, matrix factorization approaches, such as Cholesky decomposition and singular value decomposition (SVD), have been used to explore two or more lower dimensional matrices whose product provides a good approximation for the original data matrix.

Among matrix factorization methods, nonnegative matrix factorization (NMF) [9] has recently become popular for data representation owning to its psychological and physiological interpretation of natural data whose representation may be parts-based in human brain [10], [11]. Since there is only additive, not subtractive, combinations, NMF with nonnegative constraints will obtain a parts-based representation. In essence, it models data as a linear combination of a set of basis vectors, and both the combination encodings and the basis vectors are nonnegative. That is, a face image can be represented by an additive combination of several versions of mouths, noses, eyes, and other facial parts. What's more, in many real-world applications, the components must be either zero or positive. For instance, the probability of a given document belonging to a particular group is nonnegative [12]. In addition, NMF has shown superior performance to PCA and SVD in face recognition [13] and document clustering [12].

To obtain the desired characteristics like preserving local structure, minimizing prediction error etc., many NMF variants have been developed by modifying the objective function or constraint conditions of the original NMF. For example, to consider the geometric structure in the data, several graph regularized NMF methods [14]–[17] were presented to learn a new parts-based data representation, which respected the graph structure. Ding *et al*. [18] developed a NMF-like algorithm that yielded nonnegative factors but allowed the data matrix to have mixed signs. Based on linear programming, a NMF algorithm with earth mover's distance was presented in [19].

Liu *et al*. [20] proposed an A-optimal nonnegative projection (ANP) method by imposing prediction error constraint on the encoding factor, which results in a data representation with the smaller prediction error. Ding *et al*. [21] presented that NMF is roughly equivalent to the classical K-means clustering, and the nonnegative constraint results in sparseness that could lead to better approximations of the cluster indicators than direct optimization in the discrete space. Thus, Hoyer [22] introduced a nonnegative sparse coding algorithm (NNSC), which explicitly incorporated a sparseness constraint based on the relationship between the L_1 norm and L_2 norm. However, the above extensions may not be sufficient to produce

Manuscript received March 21, 2013; revised November 8, 2013; accepted November 20, 2013. Date of publication December 19, 2013; date of current version September 12, 2014. This work was supported in part by the National Basic Research Program of China under Grant 2012CB316400, in part by the National Natural Science Foundation of China under Grant 61025013, Grant 61125106, Grant 61202240, and Grant 61202241, in part by the Program for Changjiang Scholars and Innovative Research Team in University under Grant IRT201206, and in part by the Shaanxi Key Innovation Team of Science and Technology under Grant 2012KCT-04. This paper was recommended by Associate Editor D. Goldgof.

Y. Xiao, Z. Zhu, Y. Wei, and S. Wei are with the Institute of Information Science, Beijing Jiaotong University, Beijing, China, and with the Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China (e-mail: xiaoyanhui@gmail.com; zhfzhu@bjtu.edu.cn; 11112065@bjtu.edu.cn; shkwei@bjtu.edu.cn).

Y. Zhao is with the Institute of Information Science, Beijing Jiaotong University, and also with State Key Laboratory of Rail Traffic Control and Safety, Beijing 100044, China (e-mail: yzhao@bjtu.edu.cn).

X. Li is with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: xuelong_li@opt.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2013.2294215

(a) (b)

Fig. 1. Learned 16 basis vectors on the ORL dataset by the NNSC and the proposed TNMF, respevtively. (a) NNSC. (b) TNMF.

representations that are robust to local transformations, such as scale and rotational invariance.

In order to learn the invariance of feature representation, we propose a topographic NMF algorithm (TNMF). Specifically, a topographic constraint is imposed on the encoding factor as a regularizer for matrix factorization. In essence, the topographic constraint is a two-layered network, which contains the square nonlinearity in the first layer and the square-root nonlinearity in the second layer. Our TNMF is inspired by reconstruction topographic independent component analysis (RICA) [23]–[25], which has demonstrated that the topographic constraint can be helpful to learn invariance on input data. In particular, this constraint forces encodings to be organized in a topographical map by pooling together structure-correlated features belonging to the same hidden topic. By pooling over related features, the proposed topographic architecture can learn complex invariances, e.g., scale and rotational invariance. Fig. 1 shows the learned 16-basis vectors on the ORL dataset by NNSC and TNMF, respectively, and the TNMF can obtain better basis vectors than NNSC by achieve invariance.

To outline the workflow of our proposed TNMF, the overview is illustrated in Fig. 2. Specifically, we firstly extract the raw image features for input images. Then, by pooling related features together, we can obtain the encodings that are grouped by hidden topics in a structure-correlated feature space. Finally, K-means can be applied on new representations for clustering.

II. Brief Review of NMF

As a matrix factorization algorithm, NMF [9] is utilized to decompose the original data matrix into a set of bases and encodings where the basis and encodings are assumed to be nonnegative. Mathematically, given a data matrix $X = [x_{ij}] = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in R^{m\times n}$, NMF aims to find two nonnegative matrices $W = [w_{ik}] = [W_1, ..., W_t] \in R^{m\times t}$ and $S = [s_{jk}] = [\mathbf{s}_1, \ldots, \mathbf{s}_n]^T \in R^{n\times t}$ to approximate the original matrix as follows:

$$X \approx WS^T$$

where each column of X is a sample vector. Each data point $\mathbf{x}_j$ is approximated by a linear combination of the columns of W with the coefficient $\mathbf{s}_j$. Thus, W and S can be regarded as a basis set and encodings, respectively. To quantify the quality of the approximation, a cost function can be constructed by some measures of distance. One popular measure is the Euclidean distance (i.e., Frobenius norm)

$$O_F = ||X - WS^T||_F^2. \tag{1}$$

Although the objective function O_F in (1) is not convex with respect to both W and S together, the following alternating algorithm [26] converged to a local minimum:

$$\begin{aligned} w_{ik} &\leftarrow w_{ik}\frac{(XS)_{ik}}{(WS^TS)_{ik}} \\ s_{jk} &\leftarrow s_{jk}\frac{(X^TW)_{jk}}{(SW^TW)_{jk}}. \end{aligned} \tag{2}$$

In many real applications, we generally have $t \ll \min(m, n)$. Thus, NMF is to explore a compressed approximation of the original data matrix.

III. Topographic NMF

Reconstruction topographic ICA (RICA) [23]–[25] is an unsupervised learning algorithm that can learn complex invariant features from unlabeled image patches by using a topographic network. This network can be described as a two-layered network, with the square nonlinearity in the first layer ($(.)^2$) and the square-root nonlinearity in the second layer ($\sqrt{(.)}$), respectively. Mathematically, given a data matrix $X = [x_{ij}] \in R^{m\times n}$, the topographic network is performed by minimizing the following objective function

$$p = \sqrt{\varepsilon + H \cdot ((W^TX) \odot (W^TX))} \tag{3}$$

where $W = [W_1, ..., W_t] \in R^{m\times t}$ is the basis matrix of the first layer, $H = [h_{ij}] \in R^{t\times t}$ is the spatial pooling matrix in the second layer, $\odot$ denotes the element-wise multiplication and ε is a small positive constant. In addition, the pooling matrix H is hard-coded to represent the topographical structure of the neurons in the first layer as in [25], generally fixed to uniform weights, i.e., each element h_{ij} is 1.

In fact, W^TX could be regarded as encodings corresponding to data X [24]. Thus, we denote $S = (W^TX)^T$ where $S = [s_{jk}] \in R^{n\times t}$ for simplicity. Then, we rewrite the function p as

$$p(S) = \sqrt{\varepsilon + H \cdot (S \odot S)^T}. \tag{4}$$

For convenience, we term the $p(S)$ as topographic constraint. Equation (4) forces encodings to be organized in a topographical map by pooling together structure-correlated features belonging to the same hidden topic. More specifically, features that are near to each other in the topographic map are relatively strongly dependent in the sense of mutual information [23].

By incorporating the topographic constraint into the original NMF model, the optimization problem (1) becomes

$$O_F = ||X - WS^T||_F^2 + \lambda p(S) \tag{5}$$

where λ is a tradeoff parameter. As pointed out in [23] and [24], such topographic pooling architecture results in pooling units that are robust to local transformations of their

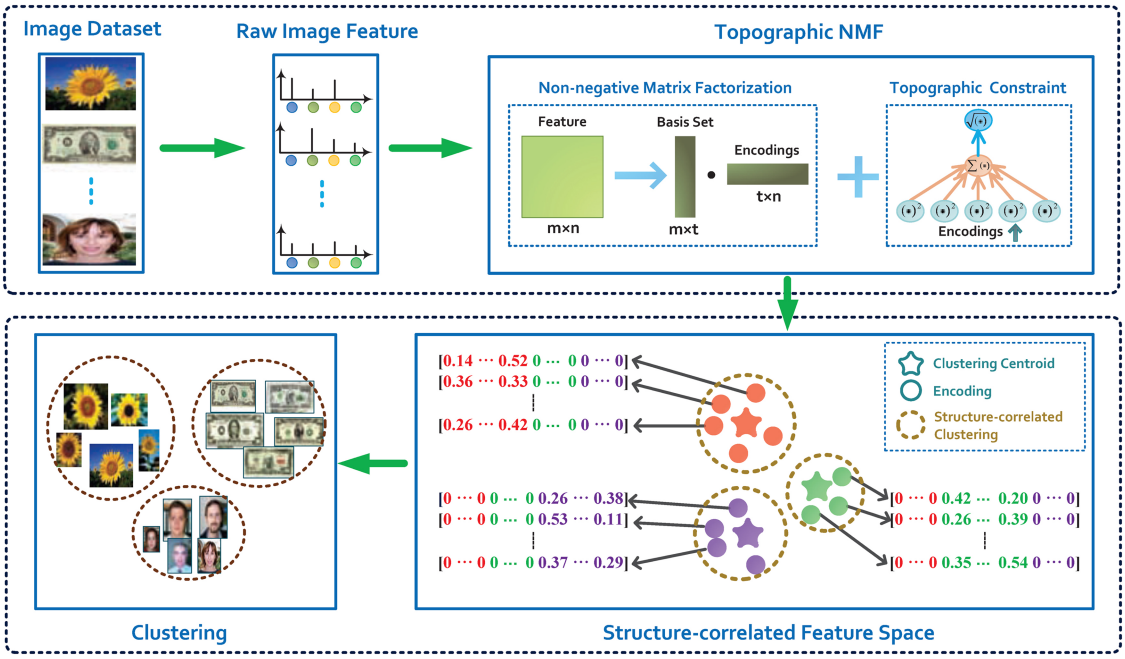

Fig. 2. Method overview for image clustering. (a) Raw image features are extracted and then decomposed into a nonnegative basis set and the corresponding nonnegative encodings with topographic constraint. Note that topographic constraint is a two-layered network, which contains square nonlinearity in the first layer and square-root nonlinearity in the second layer. (b) Obtained encodings are grouped in a structure-correlated feature space, which will be beneficial to clustering tasks.

inputs, and meanwhile promotes feature selectivity by allowing the reconstruction error and minimizing the encoding energy. As we known, the combination of robustness and selectivity is central to feature invariance [27].

To solve the problem (5), the objective function can be rewritten as

$$\begin{aligned} O_F &= ||X - WS^T||_F^2 + \lambda p(S) \\ &= \mathbf{Tr}((X - WS^T)(X - WS^T)^T) + \lambda p(S) \\ &= \mathbf{Tr}(XX^T) + \mathbf{Tr}(WS^T SW^T) - 2\mathbf{Tr}(XSW^T) \\ &\quad + \lambda p(S) \end{aligned} \tag{6}$$

where $\mathbf{Tr}(\cdot)$ denotes the trace of a matrix, and the steps of derivation employ the matrix property $\mathbf{Tr}(AB) = \mathbf{Tr}(BA)$ and $\mathbf{Tr}(B) = \mathbf{Tr}(B^T)$.

A. Multiplicative Update Rules Formulation

The objective function of TNMF in (6) is not convex with respect to both variables W and S. Thus, it is unrealistic to explore the global optima for the objective. In the following, we describe an alternative update scheme to obtain the local optima.

Given $\Phi = [\phi_{ik}] \in R^{m\times t}$ and $\Psi = [\varphi_{jk}] \in R^{n\times t}$, denote ϕ_{ik} and φ_{jk} as the Lagrange multipliers for constraint $w_{ik} \geq 0$ and $s_{jk} \geq 0$. Thus, the Lagrange L is as follows:

$$\begin{aligned} L = &\mathbf{Tr}(XX^T) + \mathbf{Tr}(WS^T SW^T) - 2\mathbf{Tr}(XSW^T) \\ &+ \lambda p(S) + \mathbf{Tr}(\Phi W^T) + \mathbf{Tr}(\Psi S^T). \end{aligned} \tag{7}$$

With respect to W and S, the partial derivatives of L are

$$\frac{\partial L}{\partial W} = -2XS + 2WS^T S + \Phi \tag{8}$$

$$\frac{\partial L}{\partial S} = -2X^T W + 2SW^T W + \lambda p'(S) + \Psi. \tag{9}$$

By utilizing the KKT conditions $\phi_{ik} w_{ik} = 0$ and $\varphi_{jk} s_{jk} = 0$, we obtain the following equations for w_{ik} and s_{jk}:

$$-(XS)_{ik} w_{ik} + (WS^T S)_{ik} w_{ik} = 0 \tag{10}$$

$$-(X^T W)_{jk} s_{jk} + (SW^T W)_{jk} s_{jk} + \frac{1}{2}\lambda p'_{jk}(S) s_{jk} = 0. \tag{11}$$

Since the update is essentially element-wise, we use $p_{jk}(S)$ to denote the part of function $p(S)$, which is only relevant to the element s_{jk} in S. Equations (10) and (11) lead to the following update rules:

$$w_{ik} \leftarrow w_{ik} \frac{(XS)_{ik}}{(WS^T S)_{ik}} \tag{12}$$

$$s_{jk} \leftarrow s_{jk} \frac{(X^T W)_{jk}}{(SW^T W)_{jk} + \frac{1}{2}\lambda p'_{jk}(S)}. \tag{13}$$

Thus, the update rules (12) and (13) can be used to solve the optimization problem (6). Note that the W and S are randomly initialized in our experiments.

Regarding the update rules (12) and (13), we have the following theorem.

Theorem 1: The objective function O_F of TNMF in (6) is nonincreasing under the update rules in (12) and (13). The objective function is invariant under these updates if and only if W and S are at a stationary point.

Theorem 1 grantees the convergence under the update rules of W and S, i.e., (12) and (13), and the final solution will be a local optimum. The proof of Theorem 1 is given in the following section.

B. Proof of Convergence

In order to prove Theorem 1, the cost function O_F of TNMF should be demonstrated to be nonincreasing under the update steps in (12) and (13). While we have exactly the same update formula for W in (12) as the original NMF [26]. In addition, (13) is only related to S. Thus, we just consider that O_F is nonincreasing under the second update step in (13).

To prove the convergence of O_F with (13), we employ the following property of an auxiliary function similar to that used in the expectation maximization algorithm [28].

Lemma 1: If G is an auxiliary function of F, i.e., $G(s, s') \geq F(s)$ and $G(s, s) = F(s)$, then F is nonincreasing under the update

$$s^{(q+1)} = \arg\min_{s} G(s, s^{(q)}). \tag{14}$$

Proof:

$$F(s^{(q+1)}) \leq G(s^{(q+1)}, s^{(q)}) \leq G(s^{(q)}, s^{(q)}) = F(s^{(q)}).$$

Notice that $F(s^{(q+1)}) = F(s^{(q)})$ holds only if $s^{(q)}$ is a local minimum of $G(s, s^{(q)})$.

Now, we will show that the update step for S in (13) is exactly the update in (14) with a proper auxiliary function G.

We rewrote the objective function O_F of TNMF in (6) as follows:

$$\begin{aligned} O_F &= ||X - WS^T||_F^2 + \lambda p(S) \\ &= \sum\nolimits_{i,j} \Big(x_{ij} - \sum\nolimits_k w_{ik} s_{jk}\Big)^2 + \\ &\quad \lambda \sum\nolimits_{j,l} \sqrt{\varepsilon + \sum\nolimits_k h_{lk} s_{jk}^2} \end{aligned} \tag{15}$$

where $1 \leq l \leq t$. In addition, we use F_{jk} to denote the part of O_F, which is only relevant to the element s_{jk} in S. Thus, we have

$$F'_{jk} = \left(\frac{\partial O_F}{\partial S}\right)_{jk} = (-2X^T W + 2SW^T W)_{jk} + \lambda p'_{jk}(S) \tag{16}$$

and

$$F''_{jk} = (2W^T W)_{kk} + \lambda p''_{jk}(S). \tag{17}$$

■

Lemma 2: Function

$$\begin{aligned} G(s, s_{jk}^{(q)}) &= F_{jk}(s_{jk}^{(q)}) + F'_{jk}(s_{jk}^{(q)})(s - s_{jk}^{(q)}) \\ &+ \frac{(SW^T W)_{jk} + \frac{1}{2}\lambda p'_{jk}(S)}{s_{jk}^{(q)}}(s - s_{jk}^{(q)})^2 \end{aligned} \tag{18}$$

is an auxiliary function for F_{jk}, the part of O_F which is only relevant to s_{jk}.

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

$$\begin{aligned} F_{jk}(s) &= F_{jk}(s_{jk}^{(q)}) + F'_{jk}(s_{jk}^{(q)})(s - s_{jk}^{(q)}) \\ &+ ((W^T W)_{kk} + \frac{1}{2}\lambda p''_{jk}(S))(s - s_{jk}^{(q)})^2 \end{aligned} \tag{19}$$

with (18) to find that $G(s, s_{jk}^{(q)}) \geq F_{jk}(s)$ is equivalent to

$$\frac{(SW^T W)_{jk} + \frac{1}{2}\lambda p'_{jk}(S)}{s_{jk}^{(q)}} \geq (W^T W)_{kk} + \frac{1}{2}\lambda p''_{jk}(S). \tag{20}$$

It is easy to check that

$$(SW^T W)_{jk} = \sum\nolimits_l s_{jl}^{(q)} (W^T W)_{lk} \geq s_{jk}^{(q)} (W^T W)_{kk}.$$

In addition, we have p'_{jk} and p''_{jk} as follows:

$$p'_{jk}(S) = \sum\nolimits_l \frac{h_{lk} s_{jk}^{(q)}}{\sqrt{\varepsilon + \sum_k h_{lk} (s_{jk}^{(q)})^2}} \tag{21}$$

$$\begin{aligned} p''_{jk}(S) = \sum\nolimits_l \Bigg(&\frac{h_{lk}}{\sqrt{\varepsilon + \sum_k h_{lk} (s_{jk}^{(q)})^2}} \\ &- \frac{(h_{lk} s_{jk}^{(q)})^2}{(\varepsilon + \sum_k h_{lk} (s_{jk}^{(q)})^2)^{3/2}} \Bigg). \end{aligned} \tag{22}$$

Similarly, it is easy to verify that $\frac{1}{2}\lambda p'_{jk}(S) \geq \frac{1}{2}\lambda p''_{jk}(S) \cdot s_{jk}^{(q)}$. Therefore, (20) holds and we have $G(s, s_{jk}^{(q)}) \geq F_{jk}(s)$. Now, we can prove the convergence of Theorem 1. ■

Proof of Theorem 1 Replacing $G(s, s_{jk}^{(q)})$ in (14) by (18) leads to the update rule

$$\begin{aligned} s_{jk}^{(q+1)} &= s_{jk}^{(q)} - s_{jk}^{(q)} \frac{F'_{jk}(s_{jk}^{(q)})}{2(SW^T W)_{jk} + \lambda p'_{jk}(S)} \\ &= s_{jk}^{(q)} \frac{(X^T W)_{jk}}{(SW^T W)_{jk} + \frac{1}{2}\lambda p'_{jk}(S)}. \end{aligned}$$

According to Lemma 2, F_{jk} is nonincreasing under this update rule.

C. Connection to RICA

The optimization problem in RICA [23], [24] is defined as

$$O_F = ||X - WW^T X||_F^2 + \lambda p(W^T X). \tag{23}$$

Similar to TNMF, RICA also attempts to find a matrix factorization, and simultaneously forces the pooling features to group similar features together to achieve invariance. According to (5) and (23), it is clear there is a close connection between the proposed TNMF and RICA. However, TNMF has three major differences from RICA.

1) As an independent competent analysis method, the prewhitening on the input data in RICA should be carried out to satisfy the orthogonality constraint, i.e., $WW^T = I$. But for TNMF, the explicit encoding as $S = (W^T X)^T$ is employed without the requirement of data prewhitening.
2) Nonnegative constraints are imposed on both basis matrix W and encoding matrix S in TNMF, which leads to a parts-based representation with only additive, not subtractive, combinations.
3) The goal of TNMF is to obtain a low-rank approximation ($t \ll m$) to the input data matrix with each column of encoding matrix S corresponding to a hidden topic, which is roughly equivalent to the classical K-means. However, RICA as in sparse coding [29] tries to learn highly over-complete features ($t \gg m$).

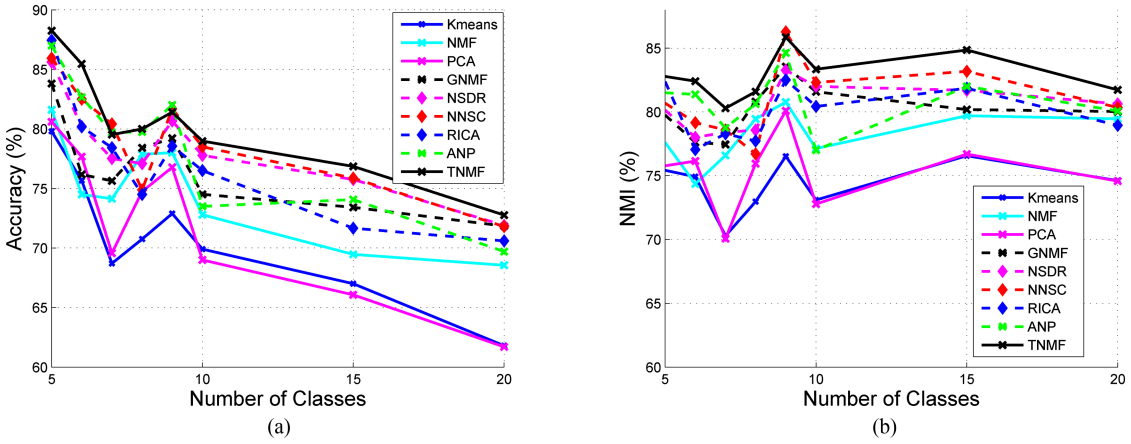

Fig. 3. Clustering results on ORL dataset. (a) Accuracy. (b) NMI.

TABLE I
CLUSTERING RESULTS ON ORL DATASET

N	Accuracy (%)								
	Kmeans	NMF	PCA	GNMF	NSDR	NNSC	RICA	ANP	TNMF
5	79.80±2.80	81.60±1.18	80.60±2.64	83.80±1.75	85.61±1.66	85.94±1.57	87.44±1.04	87.00±0.66	**88.25±0.62**
6	75.67±1.63	74.50±1.21	77.67±0.92	76.17±1.26	80.20±1.17	82.50±1.28	80.17±0.45	82.67±0.87	**85.46±0.97**
7	68.71±1.55	74.14±1.55	69.57±1.91	75.64±1.17	77.52±1.10	**80.42±0.82**	78.42±1.12	79.71±1.02	79.52±0.90
8	70.75±0.81	77.88±0.75	74.75±1.16	78.41±0.57	77.09±0.59	74.88±0.17	74.50±1.11	79.75±0.42	**80.00±0.43**
9	72.89±0.47	78.00±1.32	76.78±1.73	79.22±0.52	80.67±0.49	81.70±0.62	78.56±0.50	**82.00±0.54**	81.41±0.53
10	69.90±0.52	72.80±0.59	69.00±0.49	74.51±0.47	77.78±0.63	78.50±0.43	76.50±0.67	73.50±0.47	**78.98±0.41**
15	67.00±0.36	69.47±0.16	66.07±0.16	73.42±0.31	75.75±0.52	75.87±0.37	71.67±0.37	74.07±0.22	**76.85±0.38**
20	61.80±0.29	68.55±0.33	61.70±0.34	71.84±0.23	71.89±0.22	71.80±0.08	70.60±0.15	69.70±0.46	**72.77±0.06**
Avg.	70.81±1.05	74.62±0.89	72.02±1.17	76.62±0.79	78.31±0.80	78.95±0.67	77.23±0.68	78.55±0.58	**80.41±0.54**
N	Normalized Mutual Information (%)								
5	75.42±3.60	77.60±1.45	75.78±3.50	79.75±1.33	80.14±1.02	80.68±1.28	82.37±1.24	81.50±0.85	**82.78±0.71**
6	74.92±1.44	74.36±1.07	76.15±0.69	77.61±0.89	78.00±0.81	79.16±0.55	77.05±0.46	81.37±0.68	**82.41±0.41**
7	70.29±1.01	76.59±1.29	70.07±1.48	77.45±0.73	78.40±0.87	78.58±0.93	78.29±1.01	78.75±0.86	**80.29±0.75**
8	72.97±0.61	79.45±0.57	75.95±0.88	80.81±0.62	78.57±0.56	76.71±0.11	77.72±0.87	80.63±0.39	**81.58±0.28**
9	76.52±0.21	80.77±0.65	80.08±0.76	83.54±0.52	83.27±0.36	**86.27±0.38**	82.51±0.25	84.63±0.27	85.85±0.22
10	73.10±0.41	77.13±0.49	72.78±0.32	81.58±0.39	82.00±0.48	82.31±0.41	80.43±0.34	77.00±0.25	**83.34±0.32**
15	76.57±0.15	79.70±0.07	76.69±0.08	80.18±0.23	81.69±0.28	83.18±0.24	81.85±0.27	82.00±0.08	**84.85±0.23**
20	74.61±0.14	79.46±0.14	74.58±0.13	80.03±0.14	80.63±0.23	80.35±0.07	78.95±0.08	80.07±0.20	**81.73±0.07**
Avg.	74.30±0.95	78.13±0.72	75.26±0.98	80.11±0.61	80.34±0.58	80.91±0.50	79.90±0.57	80.75±0.45	**82.85±0.37**

IV. EXPERIMENTS

In this section, the datasets and evaluation metrics are first introduced. Then, we evaluate the performance of the proposed TNMF model for image clustering over some previous state-of-the-art algorithms. Once the clustering results are obtained, we further analyze their statistical significance. Finally, we analyze the computational complexity of TNMF, and experimentally show the speed of its convergence. All experiments were conducted on a windows machine with Intel Core2 Duo 3 GHz CPU(E8400) and 3 GB RAM.

A. Datasets

The performance of TNMF is evaluated on three public image datasets: AT&T ORL,[1] Caltech 101 [30], and Yale.[2]

1) The ORL dataset contains ten different images in each of 40 distinct subjects, thus 400 images in total. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling), and facial details (glasses/no glasses). Each image is 32×32 pixels with 256 gray levels per pixel.
2) Caltech 101 dataset contains 9144 images, which belong to 101 object classes and 1 background class including animals, vehicles, etc. Following the same experiment setup in ANP [20], we choose the ten largest categories as our experimental data which consists of 3044 images in total, and extract the SIFT descriptors [31] to form a 500-D frequency histogram [32] for each image.
3) Yale Face dataset consists of 165 grayscale images with 15 subjects. Each subject has 11 images, which are different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. Each image is 32×32 pixels with 256 gray levels per pixel.

[1] http://www.uk.research.att.com/facedatabase.html
[2] http://cvc.yale.edu/projects/yalefaces/yalefaces.html

TABLE II
CLUSTERING RESULTS ON CALTECH 101 DATASET

N	Accuracy (%)								
	Kmeans	NMF	PCA	GNMF	NSDR	NNSC	RICA	ANP	TNMF
2	74.00±3.83	76.88±4.42	73.88±3.88	78.03±4.04	78.45±4.21	80.49±2.24	81.13±2.36	82.13±3.96	**83.34±2.32**
3	58.25±3.31	68.25±2.83	56.00±2.41	75.64±2.63	75.81±2.74	74.88±1.10	75.63±1.57	76.17±2.70	**78.78±1.54**
4	56.06±2.35	66.88±2.41	57.06±2.46	74.53±2.15	74.07±2.37	73.96±1.34	74.25±1.04	71.56±1.90	**75.25±1.12**
5	52.30±0.94	60.20±1.37	51.80±0.89	68.41±1.16	68.74±1.86	66.17±0.57	67.56±0.76	66.55±1.35	**70.04±0.54**
6	49.96±0.47	58.71±1.05	48.79±0.44	65.78±0.95	65.97±1.23	63.41±0.60	64.75±0.37	63.17±0.99	**66.06±0.66**
7	46.50±0.47	54.00±0.57	46.18±0.37	61.76±0.51	62.52±0.77	61.04±0.44	61.75±0.34	58.14±0.71	**64.13±0.40**
8	46.06±0.27	54.53±0.38	45.03±0.24	59.85±0.31	60.17±0.46	60.02±0.25	59.10±0.07	59.84±0.53	**62.23±0.14**
9	43.94±0.21	54.22±0.26	43.06±0.15	58.27±0.24	58.46±0.28	59.13±0.08	58.52±0.21	58.83±0.24	**60.54±0.12**
10	42.02±0.04	51.52±0.16	40.00±0.04	56.45±0.10	56.57±0.20	57.77±0.08	58.47±0.14	57.63±0.07	**58.97±0.06**
Avg.	52.12±1.32	60.58±1.49	51.31±1.21	66.52±1.34	66.75±1.57	66.32±0.74	66.79±0.69	66.00±1.38	**68.82±0.77**
N	Normalized Mutual Information (%)								
2	41.49±17.58	49.60±20.86	41.46±17.59	58.16±8.13	58.91±9.22	57.37±6.14	59.05±7.15	59.08±18.60	**61.54±5.48**
3	35.77±7.17	49.00±7.68	33.09±5.31	58.81±2.74	58.67±3.06	58.50±2.13	60.14±2.40	60.27±6.46	**60.46±2.55**
4	39.85±4.65	54.19±4.61	40.02±4.70	57.78±2.21	57.14±2.65	59.60±1.44	58.74±1.01	59.68±4.00	**59.88±1.63**
5	40.82±2.09	50.35±2.11	40.19±2.13	57.40±1.63	57.76±1.87	58.42±0.87	58.44±0.56	58.02±2.07	**59.50±0.86**
6	41.13±1.14	50.91±1.50	40.30±0.98	57.04±0.55	57.39±0.60	56.59±0.47	57.80±0.25	57.08±1.38	**58.07±0.52**
7	38.81±0.86	48.32±0.78	38.10±0.75	56.51±0.47	56.64±0.54	56.21±0.39	**56.79±0.27**	51.91±0.91	56.66±0.49
8	41.58±0.63	50.89±0.47	40.89±0.45	56.47±0.33	**56.57±0.36**	56.02±0.16	56.10±0.10	55.93±0.55	56.23±0.19
9	41.95±0.23	52.30±0.32	41.00±0.21	56.34±0.26	56.43±0.31	56.79±0.06	56.17±0.14	56.08±0.21	**56.83±0.18**
10	40.86±0.03	51.17±0.13	38.57±0.05	56.31±0.21	56.37±0.24	56.33±0.14	56.20±0.22	56.39±0.09	**56.92±0.11**
Avg.	40.25±3.82	50.75±4.27	39.29±3.57	57.20±1.84	57.32±2.09	57.31±1.30	57.71±1.34	57.16±3.81	**58.45±1.33**

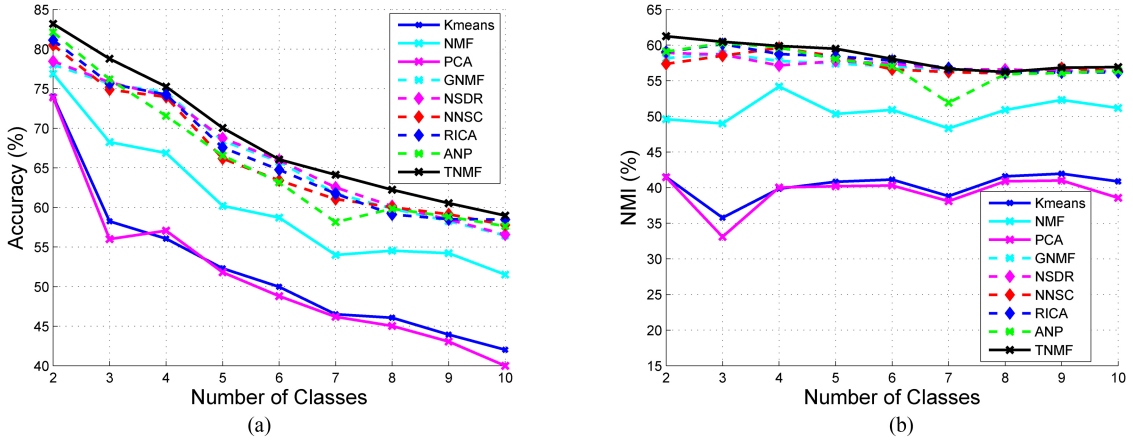

Fig. 4. Clustering results on the Caltech 101 dataset. (a) Accuracy. (b) NMI.

B. Evaluation Metrics

The clustering results are usually evaluated by comparing the cluster label of each sample with its label provided by the database. Similar to [20], two standard clustering metrics, the accuracy (AC) and normalized mutual information metric (NMI), are utilized to measure the clustering performance. Given a dataset with n images, for each image x_i, denote by e_i and r_i the cluster label and the ground truth provided by the database, respectively. The metric AC is defined as follows:

$$AC = \frac{\sum_{i=1}^{n} \delta(r_i, map(e_i))}{n} \tag{24}$$

where $\delta(x, y)$ is the delta function, which equals one if $x = y$ and equals zero otherwise, and $map(e_i)$ is the mapping function that maps each cluster label e_i to the best label from the database. The best mapping can be found by employing the Kuhn–Munkres algorithm [33].

Let C denote the set of clusters obtained from the ground truth and $\tilde{C}$ obtained from our algorithm. Their mutual information metric $MI(C, \tilde{C})$ is defined as follows:

$$MI(C, \tilde{C}) = \sum_{c_i \in C, \tilde{c}_j \in \tilde{C}} p(c_i, \tilde{c}_j) \cdot \log \frac{p(c_i, \tilde{c}_j)}{p(c_i) \cdot p(\tilde{c}_j)} \tag{25}$$

where $p(c_i)$ and $p(\tilde{c}_j)$ are the probabilities that an image arbitrarily selected from the dataset belongs to the clusters c_i and $\tilde{c}_j$, respectively, and $p(c_i, \tilde{c}_j)$ is the joint probability that the arbitrarily selected image belongs to the clusters c_i, as well as $\tilde{c}_j$ at the same time. In our experiments, we use the normalized mutual information NMI as follows:

$$NMI(C, \tilde{C}) = \frac{MI(C, \tilde{C})}{\max(H(C), H(\tilde{C}))} \tag{26}$$

where $H(C)$ and $H(\tilde{C})$ are the entropies of C and $\tilde{C}$, respectively. Note that $NMI(C, \tilde{C})$ ranges from 0 to 1. $NMI = 1$ if the two sets of clusters are identical, and $NMI = 0$ if the two sets are independent.

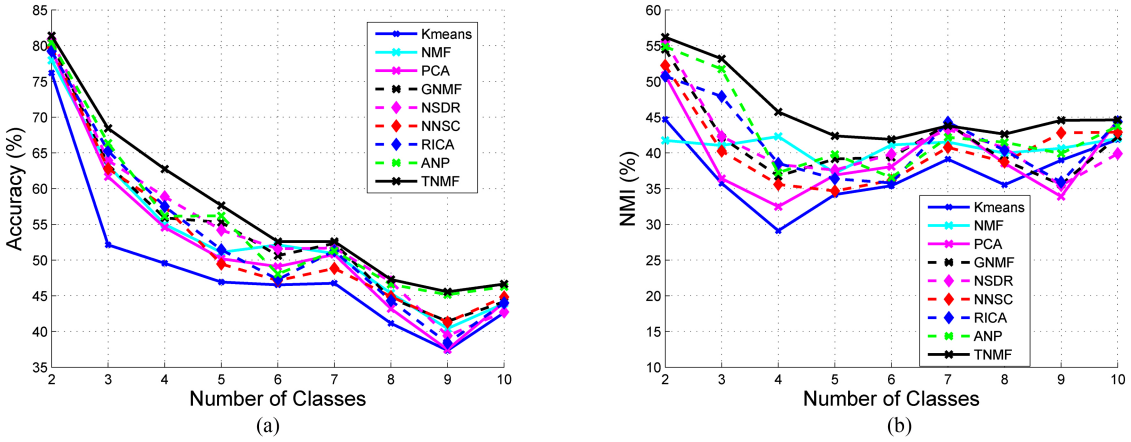

Fig. 5. Clustering results on the Yale dataset. (a) Accuracy. (b) NMI.

TABLE III
CLUSTERING RESULTS ON YALE DATASET

N	Accuracy (%)								
	Kmeans	NMF	PCA	GNMF	NSDR	NNSC	RICA	ANP	TNMF
2	76.18±2.74	77.95±2.29	79.36±1.76	80.35±2.09	80.91±2.40	79.73±2.15	79.24±2.73	80.45±2.29	**81.39±2.17**
3	52.12±1.15	63.21±0.82	61.67±0.85	63.33±0.96	63.94±1.00	62.79±0.85	65.26±1.31	66.36±0.94	**68.42±0.90**
4	49.55±0.57	55.00±0.71	54.55±0.69	55.91±0.54	58.86±0.80	57.50±0.83	57.50±1.03	56.14±0.89	**62.73±0.86**
5	46.91±0.50	51.09±0.66	50.18±0.35	55.27±0.48	54.18±0.78	49.45±0.37	51.45±0.73	56.18±0.65	**57.64±0.62**
6	46.52±0.37	52.12±0.22	49.09±0.31	50.61±0.09	51.58±0.32	47.12±0.25	47.27±0.24	48.03±0.43	**52.58±0.17**
7	46.75±0.39	50.91±0.58	50.78±0.33	52.34±0.45	51.69±0.68	48.83±0.31	51.30±0.31	51.29±0.44	**52.60±0.40**
8	41.14±0.19	45.34±0.32	43.18±0.29	44.66±0.27	46.93±0.41	44.89±0.42	44.32±0.25	46.59±0.58	**47.27±0.31**
9	37.34±0.25	40.41±0.11	37.48±0.15	41.42±0.07	39.40±0.10	41.31±0.25	38.38±0.26	45.15±0.19	**45.56±0.12**
10	42.64±0.24	43.91±0.30	44.09±0.08	44.18±0.24	42.73±0.32	44.82±0.17	44.00±0.07	46.27±0.27	**46.64±0.09**
Avg.	48.79±0.71	53.33±0.67	52.26±0.53	54.23±0.58	54.47±0.76	52.94±0.62	53.19±0.77	55.16±0.74	**57.20±0.63**
N	Normalized Mutual Information (%)								
2	44.68±3.47	41.74±2.54	50.88±2.96	54.47±3.59	55.59±3.48	52.23±3.26	50.73±3.86	54.86±3.19	**56.19±2.14**
3	35.74±2.56	41.02±2.20	36.41±2.21	42.17±2.72	42.30±2.42	40.25±2.22	47.89±3.02	51.72±2.23	**53.16±1.39**
4	29.11±1.38	42.29±1.41	32.48±1.51	36.74±0.98	38.37±1.67	35.57±2.02	38.53±2.97	37.23±0.70	**45.73±1.08**
5	34.17±1.15	37.16±0.75	36.89±0.75	39.14±0.71	37.59±0.82	34.66±0.63	36.37±1.13	39.78±0.82	**42.37±0.51**
6	35.40±0.81	41.11±0.35	38.07±0.64	39.37±0.23	39.83±0.50	36.14±0.53	35.79±0.68	36.55±0.75	**41.88±0.45**
7	39.12±0.57	41.51±0.51	44.07±0.38	44.04±0.60	43.28±0.64	40.77±0.37	**44.28±0.60**	42.18±0.29	43.79±0.40
8	35.52±0.40	39.95±0.42	38.54±0.43	38.94±0.36	40.70±0.61	38.72±0.74	40.34±0.49	41.48±0.37	**42.62±0.29**
9	38.96±0.39	40.63±0.15	33.89±0.15	35.56±0.28	35.43±0.06	42.79±0.41	35.92±0.40	39.91±0.20	**44.55±0.16**
10	41.87±0.29	42.02±0.28	44.60±0.17	42.26±0.31	39.89±0.26	42.89±0.24	44.35±0.11	43.76±0.25	**44.62±0.14**
Avg.	37.17±1.22	40.82±0.95	39.53±1.02	41.41±1.08	41.44±1.16	40.44±1.15	41.57±1.47	43.05±0.97	**46.10±0.72**

C. Clustering Results

To evaluate the clustering performance, we compare our TNMF with other state-of-the-art algorithms on the above two datasets. The evaluated algorithms are listed below:

1) traditional K-means on original data (Kmeans);.
2) nonnegative matrix factorization (NMF) [9];
3) principle component analysis (PCA);
4) graph regularized NMF (GNMF) [15];
5) nonnegative spectral clustering with discriminative regularization(NSDR) [4];
6) nonnegative sparse coding (NNSC) [22];
7) reconstruction ICA (RICA) [24];
8) A-optimal nonnegative projection (ANP) [20].

Following the same setting in ANP, N categories will be randomly picked up from the dataset with fixing cluster number N. All of these images are mixed into the collection X for clustering. To obtain the encodings, we set the dimensionality of the new space to be the same as the number of clusters N. Then, K-means is applied on new representations for clustering. The above process will be repeated ten times, and both

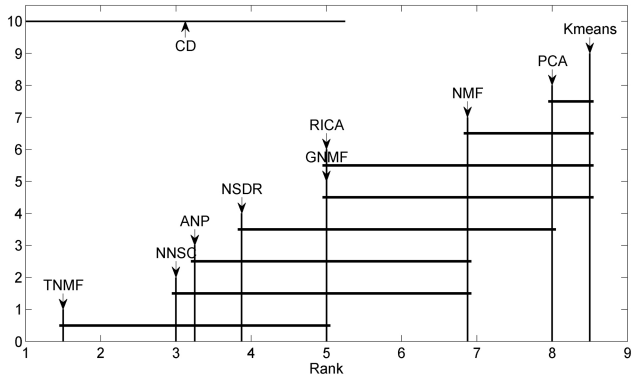

Fig. 6. Graphical representation of Nemenyi test on ORL data (CD=4.01).

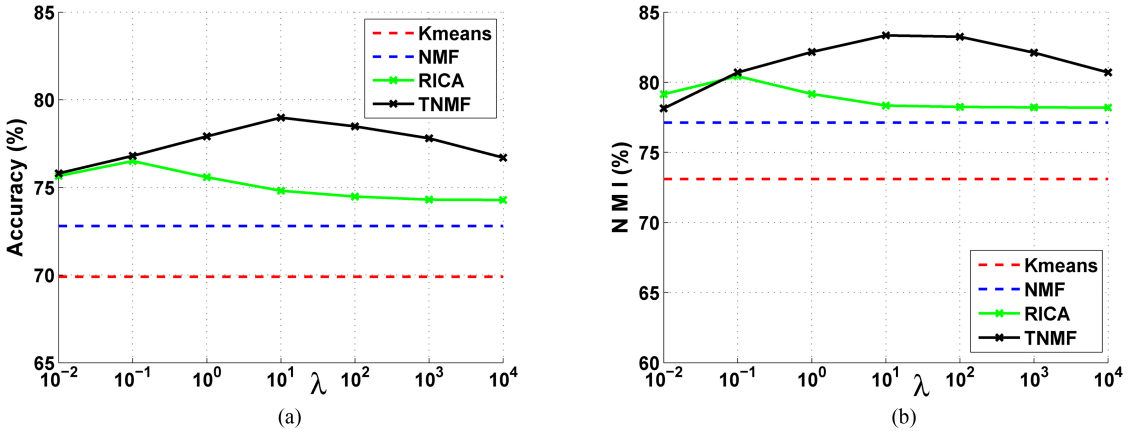

Fig. 7. Performance of TNMF versus parameter λ. (a) Accuracy. (b) NMI.

the average and variance of the performance are given as the final experiment result.

Fig. 3 shows the effectiveness of the proposed TNMF on ORL dataset. The detailed results are described in Table I. Our results are better than the best of the other algorithms, i.e., 1.46% improvement in accuracy averagely and 1.94% improvement in normalized mutual information averagely. Fig. 4 and Table II show the clustering results on the Caltech 101, respectively. Specifically, TNMF achieves 2.03% improvement in accuracy averagely and 0.74% improvement in normalized mutual information than the best of the other algorithms. Especially, TNMF outperforms all the other algorithms all the way in terms of clustering accuracy. In addition, Fig. 5 and Table III demonstrate the clustering results on the Yale dataset, respectively, and TNMF outperforms the other methods in most cases, i.e., 2.04% improvement in accuracy averagely and 3.05% improvement in normalized mutual information averagely.

D. Statistical Analysis

To evaluate the performance of our proposed TNMF, we use the Friedman test and Nemenyi test recommended in [34]. First, the average ranks of classifiers over all data are computed, and Friedman test is conducted to verify the null-hypothesis that all classifiers are equivalent in the respect of clustering performance. If the null hypothesis is rejected, then the Nemenyi test will proceed. In addition, if the average ranks of two classifiers differ by at least the critical difference (CD), then it can be concluded that their performances are significantly different.

In the Friedman test, we set the significant level $\alpha = 0.05$. Based on the accuracies of all methods over each cluster number on ORL dataset, we obtain the p-value $= 6.14 \times e^{-23}$ by performing the Friedman test. Since p-value is lower than α, we can reject the null hypothesis and Nemenyi test can proceed. As shown in Fig. 6 for CD diagram, TNMF achieves significant improvement over NMF, PCA, and Kmeans on ORL data. In addition, TNMF is much more competitive with some state-of-the-arts methods, NNSC, ANP, NSDR, GNMF, and RICA.

TABLE IV
RUNNING TIME (IN SECONDS)

N	5	10	15	20
NMF	0.13s	0.33s	0.49s	0.71s
RICA	4.37s	10.69s	13.68s	17.46s
TNMF	0.14s	0.36s	0.60s	0.97s

E. Tuning Parameter Selection

In the experiments, we experimentally set $\lambda = 10$ for the ORL data, $\lambda = 100$ for Caltech data, and $\lambda = 0.01$ for Yale data. Fig. 7 shows how the performance of TNMF varies with the parameter λ on ORL dataset for cluster number N=10. It is easy to find that the TNMF is stable with respect to the parameter λ.

F. Computational Complexity Analysis and Convergence Study

In this experiment, we test the speed performance of the original NMF, RICA, and our TNMF, and meanwhile investigate the effect over the cluster number N as 5, 10, 15, and 20. In addition, all the algorithms are terminated when changes of the parameter vector drop below 10^{-6}. We use the RICA implementation provided in [24]. Table IV shows the running times for NMF, RICA, and our TNMF. The results show that our method is much faster than the RICA and slightly slower than NMF. However, as shown in Section IV-C, TNMF achieves much better clustering performance than NMF. In addition, the overall cost of TNMF is same as the original NMF for each update step (i.e. $O(mnt)$).

In Section III-B, we have theoretically proved the convergence of TNMF, and now we experimentally show the speed of convergence of TNMF in comparison to the NMF in Fig. 8. Note that we set the cluster number $N = 10$. Fig. 8 demonstrates that TNMF converges as fast as NMF within 400 iterations.

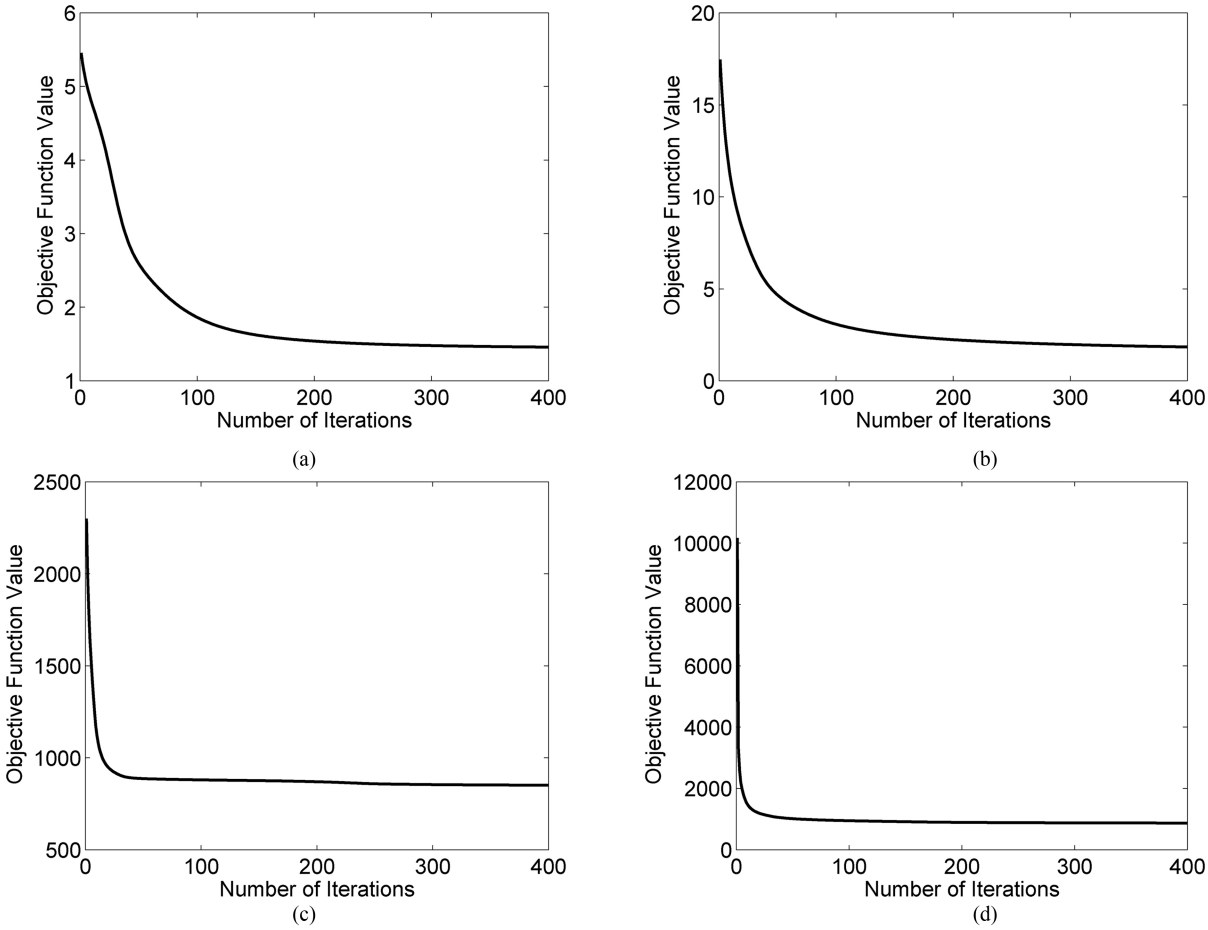

Fig. 8. Convergence of NMF and TNMF on ORL and Caltech 101 datasets. (a) ORL-NMF. (b) ORL-TNMF. (c) Caltech-NMF. (d) Caltech-TNMF.

V. Conclusion

In this paper, we propose a topographic NMF algorithm for data clustering, called TNMF. TNMF explicitly incorporates a topographic constraint to force encodings to be organized in a topographical map by pooling structure-correlated features together to achieve invariance. The experiments conducted on standardized datasets have demonstrated the effectiveness of the proposed method.

Acknowledgment

The authors would like to thank the reviewers for their valuable suggestions on the presentation and statistical comparison in this paper. They would also like to thank Q. V. Le for helpful discussions and providing the RICA code.

References

[1] Y. Sun, J. Paik, A. Koschan, D. L. Page, and M. A. Abidi, "Point fingerprint: A new 3-D object representation scheme," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 33, no. 4, pp. 712–717, Aug. 2003.

[2] K. Anderson and P. W. McOwan, "A real-time automated system for the recognition of human facial expressions," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 1, pp. 96–105, Feb. 2006.

[3] X. Li, S. Lin, S. Yan, and D. Xu, "Discriminant locally linear embedding with high-order tensor data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 2, pp. 342–352, Apr. 2008.

[4] Y. Yang, H. T. Shen, F. Nie, R. Ji, and X. Zhou, "Nonnegative spectral clustering with discriminative regularization." in *Proc. AAAI*, 2011, pp. 555–560.

[5] L. Liu, L. Shao, X. Zhen, and X. Li, "Learning discriminative key poses for action recognition," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1860–1870, Dec. 2013.

[6] H. Zhang, J. Ho, Q. Wu, and Y. Ye, "Multidimensional latent semantic analysis using term spatial information," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1625–1640, Dec. 2013.

[7] D. Tao, L. Jin, Z. Yang, and X. Li, "Rank preserving sparse learning for Kinect based scene classification," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1406–1417, Oct. 2013.

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

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

[10] S. Palmer, "Hierarchical structure in perceptual representation," *Cognitive Psychol.*, vol. 9, no. 4, pp. 441–474, 1977.

[11] E. Wachsmuth, M. Oram, and D. Perrett, "Recognition of objects and their component parts: Responses of single units in the temporal cortex of the macaque," *Cereb. Cortex*, vol. 4, no. 5, pp. 509–522, 1994.

[12] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in *Proc. 26th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2003, pp. 267–273.

[13] S. Li, X. Hou, H. Zhang, and Q. Cheng, "Learning spatially localized, parts-based representation," in *Proc. Comput. Vis. Pattern Recognit.*, vol. 1. 2001, pp. I–207.

[14] R. Zhi, M. Flierl, Q. Ruan, and W. B. Kleijn, "Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 1, pp. 38–52, Feb. 2011.

[15] D. Cai, X. He, J. Han, and T. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, Aug. 2011.
[16] Q. Gu and J. Zhou, "Local learning regularized nonnegative matrix factorization," in *Proc. IJCAI*, 2009, pp. 1046–1051.
[17] N. Guan, D. Tao, Z. Luo, and B. Yuan, "Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 2030–2048, Jul. 2011.
[18] C. Ding, T. Li, and M. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 32, no. 1, pp. 45–55, Jan. 2010.
[19] R. Sandler and M. Lindenbaum, "Nonnegative matrix factorization with earth mover's distance metric," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1873–1880.
[20] H. Liu, Z. Yang, Z. Wu, and X. Li, "A-optimal non-negative projection for image representation," in *Proc. Comput. Vis. Pattern Recognit.*, 2012, pp. 1592–1599.
[21] C. Ding, X. He, and H. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," in *Proc. SIAM Data Mining Conf.*, 2005, pp. 606–610.
[22] P. Hoyer, "Non-negative matrix factorization with sparseness constraints," *J. Mach. Learn. Res.*, vol. 5, pp. 1457–1469, Nov. 2004.
[23] A. Hyvärinen, P. Hoyer, and M. Inki, "Topographic independent component analysis," *Neural Comput.*, vol. 13, no. 7, pp. 1527–1558, 2001.
[24] Q. Le, A. Karpenko, J. Ngiam, and A. Ng, "ICA with reconstruction cost for efficient overcomplete feature learning," in *Proc. Adv. Neural Inform. Process. Syst.*, 2011, pp. 1017–1025.
[25] Q. Le, R. Monga, M. Devin, K. Chen, G. Corrado, J. Dean, *et al.*, "Building high-level features using large-scale unsupervised learning," in *Proc. Int. Conf. Mach. Learn.*, 2012, pp. 1–8.
[26] D. Seung and L. Lee, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 13. 2001, pp. 556–562.
[27] I. Goodfellow, Q. Le, A. Saxe, H. Lee, and A. Ng, "Measuring invariances in deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2009, pp. 646–654.
[28] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc. B*, vol. 39, no. 1, pp. 1–38, 1977.
[29] B. A. Olshausen, and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.
[30] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Comput. Vis. Image Und.*, vol. 106, no. 1, pp. 59–70, 2007.
[31] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. Int. Conf. Comput. Vis.*, vol. 2. 1999, pp. 1150–1157.
[32] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. Int. Conf. Comput. Vis.*, 2003, pp. 1470–1477.
[33] L. Lovász and M. Plummer, "Matching theory," *Akadémiai Kiadó*, 1986.
[34] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

Yanhui Xiao received the B.S. degree from Beijing Jiaotong University, Beijing, China, in 2007, and is currently pursuing the Ph.D degree at Beijing Jiaotong University.

His current research interests include sparse representation, independent component analysis, matrix factorization, computer vision, and machine learning.

Zhenfeng Zhu received the Ph.D degree in pattern recognition and intelligence system from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2005.

He is currently an Associate Professor with the Institute of Information Science, Beijing Jiaotong University, Beijing, China. He was a Visiting Scholar with the Department of Computer Science and Engineering, Arizona State University, Phoenix, AZ, USA, during 2010. His current research interests include image and video understanding, computer vision, and machine learning.

Yao Zhao (M'06–SM'12) received the B.S. degree from Fuzhou University, Fuzhou, China, in 1989 and the M.E. degree from Southeast University, Nanjing, China, in 1992, both from the Department of Radio Engineering, and the Ph.D. degree from the Institute of Information Science, Beijing Jiaotong University (BJTU), Beijing, China, in 1996.

He joined as an Associate Professor with BJTU in 1998 and became a Professor in 2001. From 2001 to 2002, he was a Senior Research Fellow with the Information and Communication Theory Group, Faculty of Information Technology and Systems, Delft University of Technology, Delft, The Netherlands. He is currently the Director of the Institute of Information Science, BJTU. Currently, he is leading several national research projects, such as the National Science Foundation of China 973 Program and 863 Program. His current research interests include image/video coding, digital watermarking and forensics, and video analysis and understanding.

Dr. Zhao serves on the editorial boards of several international journals. He is an Area Editor of *Processing: Image Communication* (Elsevier) and an Associate Editor of *Circuits, System & Signal Processing* (Springer). Dr. Zhao was the recipient of the National Science Foundation of China for Distinguished Young Scholars in 2010.

Yunchao Wei received the B.S. and M.S. degrees from the Hebei University of Economics and Business, Hebei, China, and Beijing Jiaotong University, Beijing, China, in 2009 and 2011, respectively. He is currently pursuing the Ph.D degree at Beijing Jiaotong University.

His current research interests include machine learning and its application to computer vision and multimedia analysis, e.g., image annotation and cross-media retrieval, etc.

Shikui Wei received the Ph.D. degree in signal and information processing from Beijing Jiaotong University (BJTU), Beijing, China, in 2010.

From 2010 to 2011, he was a Research Fellow with the School of Computer Engineering, Nanyang Technological University, Nanyang, Singapore. He is currently an Associate Professor with the Institute of Information Science, BJTU. His current research interests include computer vision, image/video analysis and retrieval, and copy detection.

Xuelong Li (M'02–SM'07–F'12) is a Full Professor with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China.