Non-Negative Matrix Factorization With Dual Constraints for Image Clustering

Zuyuan Yang, *Member, IEEE*, Yu Zhang, Yong Xiang, *Senior Member, IEEE*, Wei Yan, and Shengli Xie, *Senior Member, IEEE*

***Abstract*—How to learn dimension-reduced representations of image data for clustering has been attracting much attention. Motivated by that the clustering accuracy is affected by both the prior-known label information of some of the images and the sparsity feature of the representations, we propose a non-negative matrix factorization (NMF) method with dual constraints in this paper. In our model, one constraint is used to keep the label feature and the other constraint is utilized to enhance the sparsity of the representations. Notably that these two constraints are embedded naturally into the traditional NMF model, refraining from the usage of the balance parameters which are hard to choose. Meantime, for solving the proposed model, the alternative iteration scheme is employed, and an efficient algorithm based on convex optimization is designed to conduct each iteration operation. It is proved that this algorithm achieves a nonlinear convergence rate, much faster than existing methods with linear rate. Simulation results demonstrate the advantages of the proposed method.**

***Index Terms*—Clustering, non-negative matrix factorization (NMF), sparse representation.**

I. Introduction

Image clustering is an important issue in data mining, pattern recognition, and computer vision [1]–[3]. In practical applications, the dimensions of the initial images are often very high, leading to many obstacles in processing them. It is appealing to learn low-dimensional representations of the image data [4]. Regarding the dimensional reduction problem, there are lots of methods, such as the linear approach principal component analysis, the nonlinear approaches [5]–[7], etc. Among the existing methods, non-negative matrix factorization (NMF) plays a very important role.

Manuscript received March 27, 2017; revised July 26, 2017 and November 23, 2017; accepted March 14, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61722304, Grant 61333013, and Grant 61703113, in part by the Pearl River S&T Nova Program of Guangzhou under Grant 201610010196, in part by the Guangdong Natural Science Funds for Distinguished Young Scholar under Grant 2014A030306037, in part by the Science and Technology Plan Project of Guangdong under Grant 2014B090907010, Grant 2015B010131014, Grant 2017B010125002, and Grant 2015B010106010. This paper was recommended by Associate Editor K. Huang. *(Corresponding author: Shengli Xie.)*

Z. Yang, Y. Zhang, and S. Xie are with the Guangdong Key Laboratory of IoT Information Technology, School of Automation, Guangdong University of Technology, Guangzhou 510006, China (e-mail: yangzuyuan@aliyun.com; yuzhanggdut@aliyun.com; shlxie@gdut.edu.cn).

Y. Xiang is with the School of Information Technology, Deakin University, Melbourne, VIC 3125, Australia (e-mail: yxiang@deakin.edu.au).

W. Yan is with the Faculty of Science and Technology, University of Macau, Macau 999078, China (e-mail: yb67410@umac.mo).

Digital Object Identifier 10.1109/TSMC.2018.2820084

NMF aims to decompose a non-negative matrix into the product of two factor matrices which are both non-negative. It becomes a popular method for learning low-dimensional representations due to its meaningful interpretation of the learned results and the simple structure of the corresponding algorithms. Since the values of the entries of images are non-negative, NMF is widely used in image processing [8], particularly for clustering and classification [9]–[12].

In order to learn more efficient representations for clustering, one often adds some prior knowledge or constraint to standard NMF. For example, an interesting pairwise-constrained NMF model was presented in [13] and [14], where two kinds of constraints (must-link and cannot-link) are combined together to generate desired decompositions. Also, by restricting the pairwise constraints to parts of the data, a novel NMF framework was given in [10] to provide supervision for clustering. Furthermore, an orthogonal-property constrained NMF algorithm was designed in [11], and an inhomogeneous representation cost constrained NMF method was provided in [15].

Different from the constraints above, exact label information has attracted special attention in learning dimensional reduced representations for image clustering [16]. In many cases, we know in prior the exact labels of some of the images, although it is difficult to get the labels of all images. Even the number of the labeled image is not large, it is found that this information is very helpful to improve the clustering accuracy of unlabeled images [17]–[19]. Recently, it is shown that the accuracy can be further improved by enhancing the sparsity of the data [20]. However, it is hard for the existing methods with label constraint to generate sparse decompositions.

Motivated by the analysis above, we propose in this paper an efficient NMF method with dual constraints (NMF-DC) to learn low-dimensional representations of images for clustering. The main contribution of this paper is twofold.

1) A new NMF model is constructed, where two meaningful constraints are considered. One is the label matrix which is formed by the prior known label information of some of the images, and the other is the smoothness matrix which is used to enhance the sparsity of the learned representations. The constraints are embedded naturally into the traditional NMF model, instead of combining with the latter by using the balance parameters which are hard to choose.
2) By further exploring the proposed model, we prove that the cost function of the hidden subproblem is convex

and the corresponding gradient is Lipschitz continuous. Based on these findings, a convex optimization-based algorithm with nonlinear convergence rate is designed to solve the subproblem. It is much faster than the linear algorithms used for solving the similar subproblem in NMF-style methods.

The remainder of this paper is organized as follows. The traditional NMF model and the proposed NMF model are introduced in Section II, together with the corresponding NMF-DC algorithm and its analysis of computational complexity and convergence rate. In Section III, simulation results are presented and analyzed. Section IV concludes this paper.

II. Non-Negative Matrix Factorization With Dual Constraints

We first review the traditional NMF model, and then introduce our NMF model, the corresponding NMF-DC algorithm, and the convergence analysis to this algorithm.

A. Traditional NMF Model

For a given non-negative matrix $\mathbf{V} \in \mathbf{R}^{+}_{m\times n}$, NMF is aimed at finding two low-rank non-negative factor matrices $\mathbf{W} \in \mathbf{R}^{+}_{m\times r}$ and $\mathbf{Q} \in \mathbf{R}^{+}_{r\times n}$ with $r < \min\{m, n\}$, such that the product of these two factor matrices is an approximation of $\mathbf{V}$. Mathematically, it is represented by

$$\mathbf{V} \approx \mathbf{WQ}. \tag{1}$$

This approximation can be quantified by a cost function based on the Frobenius norm. Then, NMF can be achieved by minimizing the following cost function:

$$\min\ :\ \frac{1}{2}\|\mathbf{V} - \mathbf{WQ}\|_F^2 \tag{2}$$

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

B. Proposed NMF Model

The new NMF model will be constructed by adding two constraints, a label constraint and a sparsity constraint, to the traditional NMF model. To proceed, we first explain these two constraints.

Regarding the label constraint, many machine learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce considerable improvement in learning accuracy. And one can form a non-negative matrix to design the label matrix in NMF-style model. The process to generate the label matrix is as follows. Suppose that the data set $\mathbf{V}$ with n samples $\{\mathbf{v}_i\}_{i=1}^{n}$ can be classified into c categories. And suppose without loss of generality that the first l samples $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_l$ are labeled, and the rest $n-l$ samples $\mathbf{v}_{l+1}, \mathbf{v}_{l+2}, \ldots, \mathbf{v}_n$ are unlabeled. We also assume that each sample from $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_l$ is labeled with only one class. Then, a $c \times l$ indicator matrix $\mathbf{C}$ is formed, where $C_{i,j} = 1$ if $\mathbf{v}_i$ is labeled with the jth class; $C_{i,j} = 0$ otherwise. After that, the label matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \begin{pmatrix} \mathbf{C}_{c\times l} & 0 \\ 0 & \mathbf{I}_{n-l} \end{pmatrix} \tag{3}$$

where $\mathbf{I}_{n-l}$ denotes the $(n-l)\times(n-l)$ identity matrix and it shows that the $n-l$ unlabeled samples are not constrained. Interestingly, if $l = n$, all of data are labeled, and the learned representations can be able to have the consistent labels with all of original data.

As for the sparsity constraint, it is verified that the sparseness feature of data is beneficial for the improvement of the clustering accuracy. Meanwhile, it is known that in the NMF-style methods, because of the multiplicative nature, smoothness in one of the factors will almost certainly force sparseness in the other, in order to compensate for the final product to reproduce the data as best as possible. Therefore, we utilize a smooth matrix $\mathbf{S} \in \mathbf{R}^{+}_{r\times r}$ explicitly to constrain the factor to be sparse. This matrix $\mathbf{S}$ is square and its structure is [21]:

$$\mathbf{S} = (1-\delta)\mathbf{I}_r + \frac{\delta}{r}\mathbf{1}\mathbf{1}^{\mathrm{T}} \tag{4}$$

where $\mathbf{I}_r$ stands for the $r \times r$ identity matrix, $\mathbf{1}$ is the length-r vector of ones, $\delta \in [0, 1]$ is a parameter, and the superscript T denotes transpose. Clearly, in the case that $\delta = 0$, $\mathbf{S}$ is the identity matrix which is nonsmooth, then there is no sparse constraint. When δ tends to be 1, $\mathbf{S}$ will be very smooth, as all of the entries tend to be equal, then the sparse constraint will be extremely strong.

Combining the traditional NMF model with the constraints above, a new NMF model is proposed as follows:

$$\min:\ f(\mathbf{W}, \mathbf{H}) = \frac{1}{2}\|\mathbf{V} - \mathbf{WSHA}\|_F^2. \tag{5}$$

In this model, when the optimal $\mathbf{H}$ is obtained, $\mathbf{HA}$ will be used for clustering. It is worth noting that if $\mathbf{v}_i$ and $\mathbf{v}_j$ have the same label, then $\mathbf{a}_i = \mathbf{a}_j$ and thus $[\mathbf{HA}]_i = [\mathbf{HA}]_j$, i.e., the learned low-dimensional representations of $\mathbf{v}_i$ and $\mathbf{v}_j$ are the same. This shows that our method can keep the label features.

C. NMF-DC Algorithm

For the new model in (5), the variable matrices are $\mathbf{H}$ and $\mathbf{W}$. To optimize them, the widely used alternative update scheme is employed, i.e., optimizing one matrix while fixing the other. For the convenience of formula derivation, the cost function $f(\mathbf{W}, \mathbf{H})$ in (5) is rewritten as $f_1(\mathbf{H})$ when $\mathbf{W}$ is fixed and $f_2(\mathbf{W})$ when $\mathbf{H}$ is fixed, respectively. Since the optimizations to $f_1(\mathbf{H})$ and $f_2(\mathbf{W})$ are similar, we focus on the minimization of $f_1(\mathbf{H})$ here, and give the following lemma.

Lemma 1: The cost function $f_1(\mathbf{H})$ is convex.

Proof: Given any two matrices $\mathbf{H}_1, \mathbf{H}_2 \in \mathbf{R}^{+}_{r\times(c+n-l)}$ and a positive number $\lambda \in (0, 1)$, we have

$$\begin{aligned} &f_1(\lambda\mathbf{H}_1 + (1-\lambda)\mathbf{H}_2) - (\lambda f_1(\mathbf{H}_1) + (1-\lambda)f_1(\mathbf{H}_2)) \\ &\quad= \frac{1}{2}\mathrm{tr}\big((\mathbf{V} - \mathbf{WS}(\lambda\mathbf{H}_1 + (1-\lambda)\mathbf{H}_2)\mathbf{A})^{\mathrm{T}} \\ &\qquad\times (\mathbf{V} - \mathbf{WS}(\lambda\mathbf{H}_1 + (1-\lambda)\mathbf{H}_2)\mathbf{A})\big) \\ &\qquad- \frac{\lambda}{2}\mathrm{tr}\big((\mathbf{V} - \mathbf{WSH}_1\mathbf{A})^{\mathrm{T}}(\mathbf{V} - \mathbf{WSH}_1\mathbf{A})\big) \\ &\qquad- \frac{1-\lambda}{2}\mathrm{tr}\big((\mathbf{V} - \mathbf{WSH}_2\mathbf{A})^{\mathrm{T}}(\mathbf{V} - \mathbf{WSH}_2\mathbf{A})\big) \end{aligned} \tag{6}$$

where $\mathrm{tr}(\cdot)$ is the matrix trace operator. After some algebraic manipulations, it can be shown that (6) is equivalent to

$$\begin{aligned} &f_1(\lambda\mathbf{H}_1+(1-\lambda)\mathbf{H}_2)-(\lambda f_1(\mathbf{H}_1)+(1-\lambda)f_1(\mathbf{H}_2)) \\ &\quad=-\frac{\lambda(1-\lambda)}{2}\mathrm{tr}\big((\mathbf{WS}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{A})^{\mathrm{T}}(\mathbf{WS}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{A})\big) \\ &\quad=-\frac{\lambda(1-\lambda)}{2}\|\mathbf{WS}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{A}\|_F^2\le 0. \end{aligned} \tag{7}$$

Therefore, we have

$$f_1(\lambda\mathbf{H}_1+(1-\lambda)\mathbf{H}_2)\le\lambda f_1(\mathbf{H}_1)+(1-\lambda)f_1(\mathbf{H}_2). \tag{8}$$

According to (8) and the definition of convex function, one can conclude that $f_1(\mathbf{H})$ is convex. This completes the proof. ■

Moreover, we have the following lemma about the gradient of $f_1(\mathbf{H})$.

Lemma 2: The gradient of the cost function $f_1(\mathbf{H})$ is Lipschitz continuous and the Lipschitz constant is $L=\|\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}\|_2\times\|\mathbf{AA}^{\mathrm{T}}\|_2$, where $\|\mathbf{X}\|_2$ denotes the largest singular value of $\mathbf{X}$.

Proof: According to (5), we can calculate the gradient of $f_1(\mathbf{H})$ [i.e., $f(\mathbf{W},\mathbf{H})$ with fixed $\mathbf{W}$] by

$$\begin{aligned} \nabla_{\mathbf{H}}f_1(\mathbf{H})&=\frac{1}{2}\nabla_{\mathbf{H}}\mathrm{tr}\big(\mathbf{V}^{\mathrm{T}}\mathbf{V}\big)-\frac{1}{2}\nabla_{\mathbf{H}}\mathrm{tr}\big(\mathbf{V}^{\mathrm{T}}\mathbf{WSHA}\big) \\ &\quad-\frac{1}{2}\nabla_{\mathbf{H}}\mathrm{tr}\big(\mathbf{A}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{S}^{\mathrm{T}}\mathbf{V}\big) \\ &\quad+\frac{1}{2}\nabla_{\mathbf{H}}\mathrm{tr}\big(\mathbf{A}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{S}^{\mathrm{T}}\mathbf{WSHA}\big) \\ &=\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WSHAA}^{\mathrm{T}}-\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{VA}^{\mathrm{T}}. \end{aligned} \tag{9}$$

Then, for any two matrices $\mathbf{H}_1,\mathbf{H}_2\in\mathbf{R}^{+}_{r\times(c+n-l)}$, one can derive that

$$\begin{aligned} \|\nabla_{\mathbf{H}}f_1(\mathbf{H}_1)-\nabla_{\mathbf{H}}f_1(\mathbf{H}_2)\|_F^2&=\big\|\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{AA}^{\mathrm{T}}\big\|_F^2 \\ &=\mathrm{tr}\Big(\big(\mathbf{U}_1\Sigma\mathbf{U}_1^{\mathrm{T}}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{U}_2\Sigma\mathbf{U}_2^{\mathrm{T}}\big)^{\mathrm{T}} \\ &\quad\times\big(\mathbf{U}_1\Sigma\mathbf{U}_1^{\mathrm{T}}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{U}_2\Sigma\mathbf{U}_2^{\mathrm{T}}\big)\Big) \end{aligned} \tag{10}$$

where $\mathbf{U}_1\Sigma\mathbf{U}_1^{\mathrm{T}}$ and $\mathbf{U}_2\Sigma\mathbf{U}_2^{\mathrm{T}}$ are the singular value decompositions of $\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}$ and $\mathbf{AA}^{\mathrm{T}}$, respectively.

Let the largest singular values of $\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}$ and $\mathbf{AA}^{\mathrm{T}}$ be α and β, respectively. After some algebraic manipulations, it can be derived from (10) that

$$\begin{aligned} &\|\nabla_{\mathbf{H}}f_1(\mathbf{H}_1)-\nabla_{\mathbf{H}}f_1(\mathbf{H}_2)\|_F^2 \\ &\quad=\mathrm{tr}\Big(\mathbf{U}_2^{\mathrm{T}}(\mathbf{H}_1-\mathbf{H}_2)^{\mathrm{T}}\mathbf{U}_1\Sigma_1^2\mathbf{U}_1^{\mathrm{T}}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{U}_2\Sigma_2^2\Big) \\ &\quad\le\alpha^2\beta^2\mathrm{tr}\big(\mathbf{U}_2^{\mathrm{T}}(\mathbf{H}_1-\mathbf{H}_2)^{\mathrm{T}}\mathbf{U}_1\mathbf{U}_1^{\mathrm{T}}(\mathbf{H}_1-\mathbf{H}_2)\mathbf{U}_2\big) \\ &\quad=\alpha^2\beta^2\|\mathbf{H}_1-\mathbf{H}_2\|_F^2 \end{aligned} \tag{11}$$

where the last two equations come from the fact that $\mathbf{U}_1^{\mathrm{T}}\mathbf{U}_1,\mathbf{U}_1\mathbf{U}_1^{\mathrm{T}},\mathbf{U}_2^{\mathrm{T}}\mathbf{U}_2$, and $\mathbf{U}_2\mathbf{U}_2^{\mathrm{T}}$ are identity matrices. From (11), we can find a constant L (e.g., $L=\alpha\beta$), such that

$$\|\nabla_{\mathbf{H}}f_1(\mathbf{H}_1)-\nabla_{\mathbf{H}}f_1(\mathbf{H}_2)\|_F\le L\|\mathbf{H}_1-\mathbf{H}_2\|_F.$$

Therefore, $\nabla_{\mathbf{H}}f_1(\mathbf{H})$ is Lipschitz continuous and the Lipschitz constant is the product of the largest singular value of $\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}$ and $\mathbf{AA}^{\mathrm{T}}$, i.e., $L=\|\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}\|_2\times\|\mathbf{AA}^{\mathrm{T}}\|_2$. This completes the proof. ■

Based on Lemmas 1 and 2, the following proximal function is constructed to get the optimal $\mathbf{H}$ of $f_1(\mathbf{H})$:

$$\phi_1(\mathbf{Y},\mathbf{H})=f_1(\mathbf{Y})+\langle\nabla_{\mathbf{H}}f_1(\mathbf{Y}),\mathbf{H}-\mathbf{Y}\rangle+\frac{L}{2}\|\mathbf{H}-\mathbf{Y}\|_F^2 \tag{12}$$

where $L=\|\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{WS}\|_2\times\|\mathbf{AA}^{\mathrm{T}}\|_2$ is the Lipschitz constant and $\langle\cdot,\cdot\rangle$ denotes the matrix inner product. The Karush–Kuhn—Tucker conditions of (12) are as follows:

$$\begin{cases}\nabla_{\mathbf{H}}\phi_1(\mathbf{Y},\mathbf{H})\ge 0\\ \mathbf{H}\ge 0\\ \nabla_{\mathbf{H}}\phi_1(\mathbf{Y},\mathbf{H})\otimes\mathbf{H}=0\end{cases} \tag{13}$$

where $\otimes$ represents the component-wise multiplication. Then, we get a solution satisfying these conditions by

$$\mathbf{H}=\mathbf{P}\bigg(\mathbf{Y}-\frac{1}{L}\nabla_{\mathbf{H}}f_1(\mathbf{Y})\bigg) \tag{14}$$

where **P(Z)** projects all negative entries of $\mathbf{Z}$ to zero.

After that, by using the scheme in [22], the optimal $\mathbf{H}$ to minimize $f_1(\mathbf{H})$ is obtained by

$$\begin{cases}\mathbf{H}_k=\mathbf{P}\Big(\mathbf{Y}_{k-1}-\frac{1}{L}\nabla_{\mathbf{H}}f_1(\mathbf{Y}_{k-1})\Big)\\ \beta_k=\frac{1+\sqrt{4\beta_{k-1}^2+1}}{2}\\ \mathbf{Y}_k=\mathbf{H}_k+\frac{\beta_{k-1}-1}{\beta_k}(\mathbf{H}_k-\mathbf{H}_{k-1}).\end{cases} \tag{15}$$

For the initial values in (15), we set $\beta_0=1$ and $\mathbf{Y}_0=\mathbf{H}_0$, where $\mathbf{H}_0$ is a random non-negative matrix.

Similar to the scheme above, we can also construct the following proximal function $\phi_2(\mathbf{Z},\mathbf{W})$ of $f_2(\mathbf{W})$ to get an optimal $\mathbf{W}$:

$$\phi_2(\mathbf{Z},\mathbf{W})=f_2(\mathbf{Z})+\langle\nabla_{\mathbf{W}}f_2(\mathbf{Z}),\mathbf{W}-\mathbf{Z}\rangle+\frac{N}{2}\|\mathbf{W}-\mathbf{Z}\|_F^2 \tag{16}$$

where $N=\|\mathbf{SHAA}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{S}^{\mathrm{T}}\|_2$, and the following method is used to get the optimal $\mathbf{W}$ to minimize $f_2(\mathbf{W})$:

$$\begin{cases}\mathbf{W}_k=\mathbf{P}\Big(\mathbf{Z}_{k-1}-\frac{1}{N}\nabla_{\mathbf{W}}f_2(\mathbf{Z}_{k-1})\Big)\\ \alpha_k=\frac{1+\sqrt{4\alpha_{k-1}^2+1}}{2}\\ \mathbf{Z}_k=\mathbf{W}_k+\frac{\alpha_{k-1}-1}{\alpha_k}(\mathbf{W}_k-\mathbf{W}_{k-1}).\end{cases} \tag{17}$$

Finally, the optimal $\mathbf{H}$ and $\mathbf{W}$ in (5) are obtained by alternatively updating $\mathbf{H}_k$ in (15) and $\mathbf{W}_k$ in (17). After $\mathbf{H}$ is obtained, $\mathbf{HA}$ is used for clustering and traditional k-means method can be employed.

D. Analysis of Convergence Rate and Computational Complexity

In the proposed NMF-DC algorithm, the traditional alternative update method is utilized. We first focus on the convergence analysis in calculating the optimal $\mathbf{H}$ and give the following proposition.

Proposition 1: Given the initial matrix $\mathbf{H}_0$ and the sequence $\{\mathbf{H}_k\}_{k=2}^{\infty}$ generated by (15), it holds that

$$f_1(\mathbf{H}_k)-f_1(\mathbf{H}^*)\le\frac{2L\|\mathbf{H}_0-\mathbf{H}^*\|_F^2}{(k+1)^2} \tag{18}$$

where $\mathbf{H}^*$ denotes the optimal solution of (5) in the case that $\mathbf{W}$ is fixed and $L = \|\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{S}\|_2 \times \|\mathbf{A}\mathbf{A}^{\mathrm{T}}\|_2$ is the Lipschitz constant.

Proof: Based on [23, Th. 2.2.7], given $\hat{\mathbf{H}}$ which minimizes $\phi_1(\mathbf{Y}, \mathbf{H})$, for any $\mathbf{H} \in \mathbf{R}^+_{r\times(c+n-l)}$ and $\mathbf{Y} \in \mathbf{R}_{r\times(c+n-l)}$, it holds that

$$f_1(\mathbf{H}) \geq f_1\Big(\hat{\mathbf{H}}\Big) + L\Big\langle \hat{\mathbf{H}} - \mathbf{Y}, \mathbf{Y} - \mathbf{H}\Big\rangle + \frac{L}{2}\Big\|\mathbf{Y} - \hat{\mathbf{H}}\Big\|_F^2. \tag{19}$$

Let $\mathbf{Y} = \mathbf{Y}_k$, then based on the first equation in (15), we have $\hat{\mathbf{H}} = \mathbf{H}_{k+1}$. Furthermore, let $\mathbf{H} = \mathbf{H}_k$, then substituting $\mathbf{Y}$, $\hat{\mathbf{H}}$, and $\mathbf{H}$ into (19) yields

$$\begin{aligned} f_1(\mathbf{H}_k) \geq\ & f_1(\mathbf{H}_{k+1}) + \frac{L}{2}\|\mathbf{Y}_k - \mathbf{H}_{k+1}\|_F^2 \\ & + L\langle \mathbf{H}_{k+1} - \mathbf{Y}_k, \mathbf{Y}_k - \mathbf{H}_k\rangle. \end{aligned} \tag{20}$$

Similarly, substituting $\mathbf{Y} = \mathbf{Y}_k$, $\hat{\mathbf{H}} = \mathbf{H}_{k+1}$, and $\mathbf{H} = \mathbf{H}^*$ into (19) results in

$$\begin{aligned} f_1\big(\mathbf{H}^*\big) \geq\ & f_1(\mathbf{H}_{k+1}) + \frac{L}{2}\|\mathbf{Y}_k - \mathbf{H}_{k+1}\|_F^2 \\ & + L\big\langle \mathbf{H}_{k+1} - \mathbf{Y}_k, \mathbf{Y}_k - \mathbf{H}^*\big\rangle. \end{aligned} \tag{21}$$

Since $\beta_k > 1\ \forall k > 0$, we can multiply $\beta_k - 1$ to both sides of (20) and add it to (21). This gives

$$\begin{aligned} &(\beta_k - 1)f_1(\mathbf{H}_k) + f_1\big(\mathbf{H}^*\big) \\ &\quad \geq L\big\langle \mathbf{H}_{k+1} - \mathbf{Y}_k, \beta_k\mathbf{Y}_k - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\rangle \\ &\qquad + \beta_k f_1(\mathbf{H}_{k+1}) + \frac{L\beta_k}{2}\|\mathbf{Y}_k - \mathbf{H}_{k+1}\|_F^2. \end{aligned} \tag{22}$$

Further multiplying both sides of (22) by β_k, we have

$$\begin{aligned} &\Big(\beta_k^2 - \beta_k\Big)f_1(\mathbf{H}_k) + \beta_k f_1\big(\mathbf{H}^*\big) \\ &\quad \geq L\big\langle \beta_k(\mathbf{H}_{k+1} - \mathbf{Y}_k), \beta_k\mathbf{Y}_k - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\rangle \\ &\qquad + \beta_k^2 f_1(\mathbf{H}_{k+1}) + \frac{L\beta_k^2}{2}\|\mathbf{Y}_k - \mathbf{H}_{k+1}\|_F^2. \end{aligned} \tag{23}$$

From the second equation in (15), it holds $\beta_{k-1}^2 = \beta_k^2 - \beta_k$. Then, (23) can be rewritten as

$$\begin{aligned} &\beta_{k-1}^2 f_1(\mathbf{H}_k) + \Big(\beta_k^2 - \beta_{k-1}^2\Big)f_1\big(\mathbf{H}^*\big) - \beta_k^2 f_1(\mathbf{H}_{k+1}) \\ &\quad \geq L\big\langle \beta_k(\mathbf{H}_{k+1} - \mathbf{Y}_k), \beta_k\mathbf{Y}_k - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\rangle \\ &\qquad + \frac{L}{2}\|\beta_k\mathbf{H}_{k+1} - \beta_k\mathbf{Y}_k\|_F^2. \end{aligned} \tag{24}$$

Notably, for any matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, it holds $\|\mathbf{B}-\mathbf{A}\|_F^2 + 2\langle \mathbf{B} - \mathbf{A}, \mathbf{A} - \mathbf{C}\rangle = \|\mathbf{B} - \mathbf{C}\|_F^2 - \|\mathbf{A} - \mathbf{C}\|_F^2$. Besides, from the third equation in (15), we have $\beta_k(\mathbf{Y}_k - \mathbf{H}_k) = (\beta_{k-1} - 1)(\mathbf{H}_k - \mathbf{H}_{k-1})$. Then, (24) can be rewritten as

$$\begin{aligned} &\beta_{k-1}^2\big(f_1(\mathbf{H}_k) - f_1\big(\mathbf{H}^*\big)\big) - \beta_k^2\big(f_1(\mathbf{H}_{k+1}) - f_1\big(\mathbf{H}^*\big)\big) \\ &\quad \geq L\big\langle \beta_k(\mathbf{H}_{k+1} - \mathbf{Y}_k), \beta_k\mathbf{Y}_k - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\rangle \\ &\qquad + \frac{L}{2}\|\beta_k\mathbf{H}_{k+1} - \beta_k\mathbf{Y}_k\|_F^2 \\ &\quad = \frac{L}{2}\big\|\beta_k\mathbf{H}_{k+1} - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\|_F^2 \\ &\qquad - \frac{L}{2}\big\|\beta_k\mathbf{Y}_k - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\|_F^2 \\ &\quad = \frac{L}{2}\big\|\beta_k\mathbf{H}_{k+1} - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\|_F^2 \\ &\qquad - \frac{L}{2}\big\|(\beta_{k-1} - 1)(\mathbf{H}_k - \mathbf{H}_{k-1}) + \mathbf{H}_k - \mathbf{H}^*\big\|_F^2 \\ &\quad = \frac{L}{2}\big\|\beta_k\mathbf{H}_{k+1} - (\beta_k - 1)\mathbf{H}_k - \mathbf{H}^*\big\|_F^2 \\ &\qquad - \frac{L}{2}\big\|\beta_{k-1}\mathbf{H}_k - (\beta_{k-1} - 1)\mathbf{H}_{k-1} - \mathbf{H}^*\big\|_F^2. \end{aligned} \tag{25}$$

Since $\beta_0 = 1$, then by varying the subscript in (25) from 1 to $k-1$ where $k \geq 2$ and summing up all these inequalities, we obtain

$$\begin{aligned} &f_1(\mathbf{H}_1) - f_1\big(\mathbf{H}^*\big) - \beta_{k-1}^2\big(f_1(\mathbf{H}_k) - f_1\big(\mathbf{H}^*\big)\big) \\ &\quad \geq \frac{L}{2}\big\|\beta_{k-1}\mathbf{H}_k - (\beta_{k-1} - 1)\mathbf{H}_{k-1} - \mathbf{H}^*\big\|_F^2 - \frac{L}{2}\big\|\mathbf{H}_1 - \mathbf{H}^*\big\|_F^2 \\ &\quad \geq -\frac{L}{2}\big\|\mathbf{H}_1 - \mathbf{H}^*\big\|_F^2. \end{aligned} \tag{26}$$

Regarding $f_1(\mathbf{H}_1) - f_1(\mathbf{H}^*)$ on the left-hand side of (26), we substitute $\mathbf{Y} = \mathbf{Y}_0$, $\hat{\mathbf{H}} = \mathbf{H}_1$ and $\mathbf{H} = \mathbf{H}^*$ into (19) to get

$$\begin{aligned} f_1(\mathbf{H}_1) - f_1\big(\mathbf{H}^*\big) &\leq -\frac{L}{2}\|\mathbf{Y}_0 - \mathbf{H}_1\|_F^2 - L\big\langle \mathbf{H}_1 - \mathbf{Y}_0, \mathbf{Y}_0 - \mathbf{H}^*\big\rangle \\ &= \frac{L}{2}\big\langle \mathbf{Y}_0 - \mathbf{H}^*, 2\mathbf{Y}_0 - 2\mathbf{H}_1\big\rangle - \frac{L}{2}\|\mathbf{Y}_0 - \mathbf{H}_1\|_F^2 \\ &= \frac{L}{2}\big\langle \mathbf{Y}_0 - \mathbf{H}^*, \mathbf{Y}_0 - \mathbf{H}^* + \mathbf{H}^* - \mathbf{H}_1 + \mathbf{Y}_0 - \mathbf{H}_1\big\rangle \\ &\quad - \frac{L}{2}\langle \mathbf{Y}_0 - \mathbf{H}_1, \mathbf{Y}_0 - \mathbf{H}_1\rangle \\ &= \frac{L}{2}\big\langle \mathbf{Y}_0 - \mathbf{H}^*, \mathbf{Y}_0 - \mathbf{H}^*\big\rangle \\ &\quad + \frac{L}{2}\big\langle \mathbf{Y}_0 - \mathbf{H}^*, \mathbf{H}^* - \mathbf{H}_1\big\rangle \\ &\quad + \frac{L}{2}\big\langle \mathbf{Y}_0 - \mathbf{H}^*, \mathbf{Y}_0 - \mathbf{H}_1\big\rangle \\ &\quad - \frac{L}{2}\langle \mathbf{Y}_0 - \mathbf{H}_1, \mathbf{Y}_0 - \mathbf{H}_1\rangle \\ &= \frac{L}{2}\big\|\mathbf{Y}_0 - \mathbf{H}^*\big\|_F^2 + \frac{L}{2}\big\langle \mathbf{Y}_0 - \mathbf{H}^*, \mathbf{H}^* - \mathbf{H}_1\big\rangle \\ &\quad + \frac{L}{2}\big\langle \mathbf{H}_1 - \mathbf{H}^*, \mathbf{Y}_0 - \mathbf{H}_1\big\rangle \\ &= \frac{L}{2}\Big(\big\|\mathbf{Y}_0 - \mathbf{H}^*\big\|_F^2 - \big\|\mathbf{H}_1 - \mathbf{H}^*\big\|_F^2\Big). \end{aligned} \tag{27}$$

Substituting (27) into (26), we have

$$\begin{aligned} \beta_{k-1}^2\big(f_1(\mathbf{H}_k) - f_1\big(\mathbf{H}^*\big)\big) &\leq \frac{L}{2}\big\|\mathbf{H}_1 - \mathbf{H}^*\big\|_F^2 + f_1(\mathbf{H}_1) - f_1\big(\mathbf{H}^*\big) \\ &\leq \frac{L}{2}\big\|\mathbf{H}_1 - \mathbf{H}^*\big\|_F^2 \\ &\quad + \frac{L}{2}\Big(\big\|\mathbf{Y}_0 - \mathbf{H}^*\big\|_F^2 - \big\|\mathbf{H}_1 - \mathbf{H}^*\big\|_F^2\Big) \\ &= \frac{L}{2}\big\|\mathbf{Y}_0 - \mathbf{H}^*\big\|_F^2. \end{aligned} \tag{28}$$

Since $\mathbf{Y}_0 = \mathbf{H}_0$ and $\beta_{k-1} \geq (k+1)/2$, it follows from (28) that:

$$\begin{aligned} f_1(\mathbf{H}_k) - f_1\big(\mathbf{H}^*\big) &\leq \frac{L}{2\beta_{k-1}^2}\big\|\mathbf{H}_0 - \mathbf{H}^*\big\|_F^2 \\ &\leq \frac{2L\|\mathbf{H}_0 - \mathbf{H}^*\|_F^2}{(k+1)^2}. \end{aligned} \tag{29}$$

This completes the proof. ■

From Proposition 1, one can see that the convergence rate in calculating the optimal $\mathbf{H}$ is $O(1/k^2)$. Since the optimization schemes of $\mathbf{H}$ and $\mathbf{W}$ are similar, the convergence rate in calculating the optimal $\mathbf{W}$ is also $O(1/k^2)$.

Regarding the computational complexity, the majority of time cost is spent on computing the gradient $\nabla_{\mathbf{H}} f_1(\mathbf{Y}_{k-1})$ [or $\nabla_{\mathbf{W}} f_2(\mathbf{Z}_{k-1})$] in (15) [or (17)], which can be calculated by the method in (9). Based on the rule of the matrix multiplication, the time cost on getting $\mathbf{S}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}$, $\mathbf{A}\mathbf{A}^{\mathrm{T}}$, $\mathbf{V}\mathbf{A}^{\mathrm{T}}$ is $O(mr^2)$, $O(n(c+n-l)^2)$, $O(mn(c+n-l))$, respectively. Then, the time cost of (9) is $O(mrn)$ approximately. Thus, the computational complexity of our method is $K \times O(mrn)$, where K denotes the iteration number. Since the iterative method for updating $\mathbf{H}$ (or $\mathbf{W}$) achieves the nonlinear convergence rate $O(1/k^2)$, the needed iteration number K is often much smaller than other methods with linear rate.

III. Simulation Results

In this section, simulations are carried out to test the performance of the proposed NMF-DC algorithm in learning sparse and label-denoting representations for image clustering (where traditional k-means method is employed for clustering), with comparisons to related algorithms, including CNMF in [18], CNMF-KL in [18], GNMF in [24], SemiGNMF in [24], NsNMF in [21], AnsNMF in [8], and the traditional NMF. The following index is used to measure the sparseness of a matrix $\mathbf{V} \in \mathbf{R}^{+}_{m\times n}$ [25]:

$$S_{\mathbf{V}} = \frac{\sqrt{mn} - \left(\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbf{v}_{ij}\right)\Big/\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbf{v}_{ij}^2}}{\sqrt{mn}-1}. \tag{30}$$

With regard to the evaluation of the clustering performance, the widely used AC index is employed. Given a data set containing n images, let $l_1, l_2, \ldots, l_n$ be the cluster labels obtained by applying different algorithms and $r_1, r_2, \ldots, r_n$ be the cluster labels provided by the data set. Then, the AC index is defined as

$$\mathrm{AC} = \frac{\sum_{i=1}^{n}\delta(r_i, \mathrm{map}(l_i))}{n} \tag{31}$$

where $\mathrm{map}(l_i)$ is the mapping function that maps each cluster label l_i to the equivalent label from the data set, and $\delta(x, y)$ is the delta function which equals 1 if $x = y$ and equals 0 otherwise. Clearly, the AC index belongs to [0, 1]. The larger the AC, the better the clustering performance.

Besides, one can use the following adjusted rand index for the evaluation of the agreement between two partitions in clustering analysis with different numbers of clusters [26]. Given a set of n objects $O = \{O_1, O_2, \ldots, O_n\}$ and suppose that $U = \{U_1, U_2, \ldots, U_r\}$ and $Z = \{Z_1, Z_2, \ldots, Z_s\}$ represent two different partitions of the objects in O such that $\bigcup_{i=1}^{r} U_i = O = \bigcup_{j=1}^{s} Z_j$ and $U_i \bigcap U_{i'} = \emptyset = Z_j \bigcap Z_{j'}$ for $1 \le i \ne i' \le r$ and $1 \le j \ne j' \le s$. Given two partitions, U and Z, with r and s subsets, respectively, the contingency table can be formed to indicate group overlap between U and Z. Then, adjusted rand index is computed by

TABLE I
Contingency Table

$U\backslash Z$	Z_1	Z_2	$\cdots$	Z_s	Total
U_1	n_{11}	n_{12}	$\cdots$	n_{1s}	a_1
U_2	n_{21}	n_{22}	$\cdots$	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
U_r	n_{r1}	n_{r2}	$\cdots$	n_{rs}	a_r
Total	b_1	b_2	$\cdots$	b_s	n

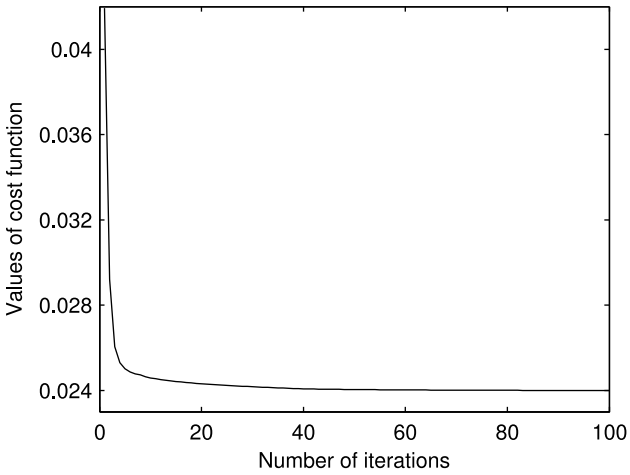


Fig. 1. Convergence curve of NMF-DC on the ORL database when $c = 3$.

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right]/\binom{n}{2}} \tag{32}$$

where n_{rs}, a_i, b_j are values from Table I. A generic entry, n_{rs}, represents the number of objects that were classified in the rth subset of partition R and in the sth subset of partition S. The notation $\binom{n}{2}$ indicates the total number of possible combinations of pairs from a given set. The larger the ARI, the better the clustering performance.

A. ORL Data

The ORL face database is provided by AT&T Laboratories Cambridge.[1] It contains 400 grayscale 32×32 face images from 40 people, i.e., ten images for each person. The images for each person were taken at different time, under varying lighting conditions, and with different facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All of the images were taken against a dark homogeneous background with the persons in an upright, frontal position (with tolerance for some side movement).

First, the convergence of the proposed NMF-DC algorithm is verified. The experiments are conducted under different parameters and different numbers of classes. It is found that the results are similar and all of them achieve good convergence performances. Here, the result in the case $c = 3$ is shown in Fig. 1, where the observed matrix $\mathbf{V}$ is normalized

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

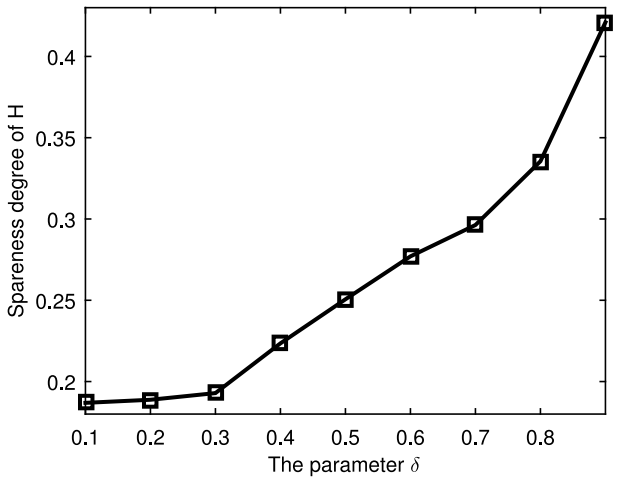


Fig. 2. Sparseness degree of **H** under different δ when $c = 3$.

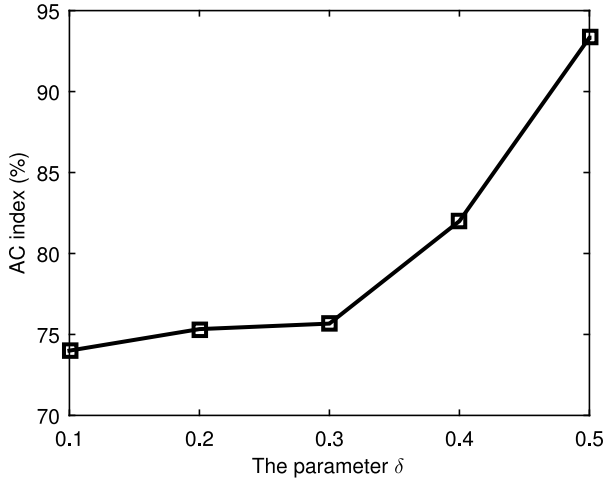


Fig. 3. AC index under different δ when $c = 3$.

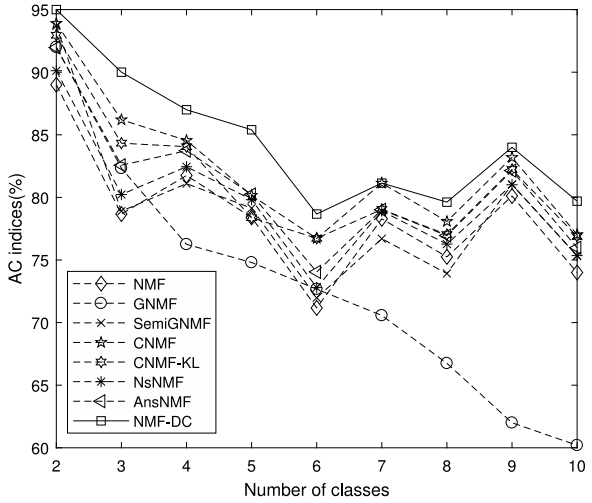


Fig. 4. AC indices of the compared algorithms on the ORL database.

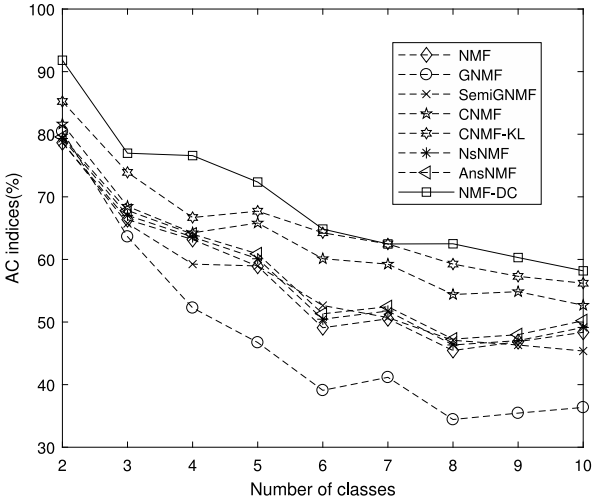


Fig. 5. AC indices of the compared algorithms on the Yale database.

between 0 and 1. From Fig. 1, one can see that it converges in only several iterations.

Then, we give the performance of NMF-DC in learning sparse representation. Notably, the sparseness degree of the learned **H** equals that of **HA** based on (30). We focus on **H**, although **HA** is used for clustering. Since the sparseness of **H** is controlled by the parameter δ in (4), we test it with different class number c under different δ. Here, the results for the case $c = 3$ are given. Fig. 2 shows the spareness degree of **H** when δ varies from 0.1 to 0.9. It can be seen that the proposed algorithm is able to learn sparse representation by using proper δ.

Furthermore, we show that the sparseness feature is helpful to the improvement of clustering accuracy. Fig. 3 shows the AC index when $c = 3$, where δ varies from 0.1 to 0.5. One can see that the clustering accuracy improves with the increase of δ, i.e., the increase of the sparseness degree of **H** or **HA**.

After that, we test the performance of NMF-DC in clustering **H**, and compare it with those of CNMF, CNMF-KL, GNMF, SemiGNMF, NsNMF, AnsNMF, and the traditional NMF. Fig. 4 shows the AC indices of these algorithms when the class number c varies from 2 to 10, with more detailed information being given in Table II, where the proportion of the labeled data in each class is 10%. It can be seen that the label-information-based algorithms CNMF, CNMF-KL, and our NMF-DC perform better than the other algorithms. This shows the importance of the label information. Among these algorithms, NMF-DC has the highest accuracy, due to the usage of the sparseness feature of **H**. Also, the corresponding ARI indices are provided in Table III. One can see that our method obtains the best ARI index. Notably that the averaged running time (Avg_t) of our method is the smallest as shown in Table II, implying that our method with nonlinear convergence rate is the most efficient one in computation.

B. Yale Data

The Yale database (see http://vision.ucsd.edu/content/yale-face-database) contains 165 grayscale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. Fig. 5 shows the AC indices of the compared algorithms versus the class number c and more detailed

TABLE II
AC INDICES(%) AND AVERAGED RUNNING TIME (S) ON THE ORL DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
2	89.00	92.00	93.70	93.90	92.95	90.12	91.97	95.00
3	78.67	82.33	78.90	86.20	84.33	80.23	82.57	90.00
4	81.75	76.25	81.10	84.55	84.05	82.45	83.73	87.00
5	78.40	74.80	79.14	80.16	78.38	79.82	80.23	85.40
6	71.17	72.67	71.98	76.72	76.73	72.80	74.05	78.67
7	78.29	70.57	76.70	81.14	79.04	78.91	79.01	81.14
8	75.25	66.75	73.92	78.08	77.03	76.27	76.95	79.62
9	80.11	62.00	81.04	83.19	82.23	81.01	82.14	84.00
10	74.00	60.20	75.33	77.03	76.88	75.35	75.98	79.70
Avg.	78.51	73.09	79.09	82.33	81.29	79.6622	80.7367	84.50
Avg_t	0.98	1.14	1.15	1.39	1.40	1.32	1.46	0.26

TABLE III
ARI INDICES(%) ON THE ORL DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
2	46.43	79.96	53.12	62.08	62.08	47.25	51.42	79.96
3	40.98	42.61	50.24	72.29	72.17	41.76	42.04	73.10
4	43.78	36.03	49.38	51.18	51.18	44.55	46.23	79.18
5	52.50	36.01	55.25	59.72	59.72	54.37	55.34	70.06
6	46.45	32.76	50.97	54.58	54.11	48.25	49.03	68.31
7	38.91	37.68	45.10	60.51	61.76	39.97	41.15	68.11
8	43.32	38.40	44.28	47.66	47.66	45.26	46.33	63.34
9	47.22	43.84	52.02	61.98	61.98	48.90	49.87	70.79
10	47.41	42.89	50.70	51.01	51.01	49.12	50.35	65.55
Avg.	45.22	43.35	50.12	57.89	57.96	46.6033	47.97	70.93

TABLE IV
AC INDICES(%) AND AVERAGED RUNNING TIME (S) ON THE YALE DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
2	78.64	80.45	78.73	81.64	85.23	79.32	79.57	91.82
3	66.36	63.64	65.73	68.48	73.91	67.04	67.83	76.97
4	63.18	52.27	59.25	64.25	66.70	63.59	64.04	76.59
5	58.91	46.73	58.96	65.82	67.71	60.11	60.89	72.36
6	49.09	39.09	52.64	60.12	64.26	50.45	51.32	64.85
7	50.52	41.17	50.71	59.25	62.47	51.76	52.45	62.47
8	45.45	34.43	47.08	54.40	59.26	46.34	47.27	62.50
9	46.87	35.45	46.33	54.85	57.30	47.03	47.98	60.30
10	48.36	36.36	45.37	52.68	56.20	49.14	50.21	58.18
Avg.	56.38	47.73	56.09	62.39	65.89	57.19	57.95	69.56
Avg_t	1.09	1.21	1.26	1.40	1.41	1.35	1.49	0.28

TABLE V
ARI INDICES(%) ON THE YALE DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
2	46.51	47.36	46.63	77.26	77.26	47.23	47.97	68.59
3	39.01	40.94	38.68	48.02	48.02	40.12	40.88	48.25
4	43.81	46.58	41.39	47.84	47.87	44.26	45.03	48.65
5	37.89	35.23	38.43	43.93	43.93	38.15	39.87	52.53
6	31.83	31.06	33.01	39.96	39.96	32.04	33.12	44.09
7	29.10	29.67	29.70	39.80	39.80	30.15	31.26	38.63
8	32.46	27.77	33.26	38.94	38.94	33.01	33.98	40.30
9	29.12	24.91	30.19	40.70	40.70	30.36	31.42	38.30
10	29.38	22.11	26.40	34.00	34.00	29.79	30.18	37.00
Avg.	35.46	33.96	35.30	45.61	45.61	36.12	37.07	46.26

information is given in Table IV, where the proportion of the labeled data in each class is 10%. And Table V provides the corresponding ARI indices of the compared algorithms. One can see that the averaged AC and ARI indices of the proposed NMF-DC algorithm are the best among the compared algorithms. Furthermore, the averaged running time in Table IV shows that our method is the fastest one.

C. Computer-Generated Data

In this part, we test the performance of the proposed NMF-DC algorithm using the dataset generated by the computer. The results are compared with those of NMF, GNMF, SemiGNMF, CNMF, CNMF-KL, NsNMF, and AnsNMF. The tested data is generated by using the rand() function in MATLAB software, plus a projection modification such that the Euclidean

TABLE VI
AC INDICES (%) AND AVERAGED RUNNING TIME (S) ON THE COMPUTER-GENERATED DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
5	87.04	92.91	93.20	94.10	95.19	87.99	88.75	96.25
10	85.91	90.27	91.08	92.34	92.84	86.22	87.18	95.97
15	84.80	88.26	90.55	91.10	91.17	85.17	86.40	95.05
20	82.77	87.24	88.16	89.53	90.92	83.96	84.82	94.51
25	81.13	87.07	87.37	88.12	88.52	82.03	83.16	92.08
30	80.35	84.89	85.18	86.27	87.08	81.37	82.53	91.32
Avg.	83.66	88.44	89.25	90.24	90.95	84.45	85.47	94.19
Avg_t	20.37	21.65	22.03	25.98	26.40	24.39	25.52	5.08

TABLE VII
ARI INDICES (%) ON THE COMPUTER-GENERATED DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
5	77.10	90.51	91.02	90.07	90.82	80.29	82.09	95.74
10	75.73	89.41	89.87	88.21	88.97	77.02	79.65	95.08
15	72.38	88.84	89.06	87.06	88.17	73.16	74.90	94.63
20	70.89	87.25	87.97	84.42	86.89	72.25	73.14	93.89
25	69.32	85.15	86.40	83.75	85.14	70.81	71.17	90.04
30	67.97	84.02	84.65	82.26	83.12	69.06	70.83	89.85
Avg.	72.23	87.53	88.16	85.96	87.18	73.76	75.29	93.20

TABLE VIII
AC INDICES (%) AND AVERAGED RUNNING TIME (S) ON THE LFW DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
2	66.34	66.57	68.23	68.49	68.99	66.89	67.01	70.50
3	65.26	66.42	67.50	66.30	65.63	66.17	66.98	68.57
4	61.80	62.34	63.65	63.21	64.27	62.09	63.14	66.25
5	60.76	61.08	61.96	62.17	62.90	61.21	61.95	66.51
6	54.21	53.87	54.20	56.12	57.52	55.93	56.21	58.58
7	50.28	49.48	51.71	51.05	50.98	51.31	52.44	53.64
8	45.31	44.19	45.85	46.40	46.26	46.04	46.97	50.37
9	45.87	46.21	47.38	45.29	45.89	46.13	47.28	50.17
10	41.36	42.64	44.79	42.88	43.49	42.27	43.18	46.20
Avg.	54.57	54.75	56.14	55.76	56.21	55.33	56.12	58.97
Avg_t	122.13	126.20	126.63	146.52	146.64	137.43	145.17	25.46

distances of the samples in the same class are smaller than that of the samples in different classes. It has 100 classes, and there are 20 samples with 2500 dimensions in each class. Table VI shows the AC indices of the compared algorithms when the class number varies from 5 to 30, together with the averaged running time, where the proportion of the labeled data is 10%. And the corresponding ARI indices are given in Table VII.

D. LFW Data

In this section, the labeled faces in the wild (LFW) dataset is tested [27]. This dataset contains 13 233 face images from 5749 people, where 1680 people have two or more distinct photographs.[2] The size of each image is 250 × 250. These images are obtained from the Web and they are collected in the unconstrained and natural circumstance, not for experiments on purpose, e.g., 35 images are shown in Fig. 6. For the convenience of comparison in a proper time, one thousand images are used in our experiments, including 50 people and 20 images for each people.

First, the performance of the proposed method under different number of classes and different proportions of the labeled data is tested, where the AC index is used for evaluation. Fig. 6 shows the variation of this index for each number of classes when the proportion varies from 10% to 80%.

Fig. 6. Thirty five images in LFW database, where five images in each row are the same people.

[2]http://vis-www.cs.umass.edu/lfw/

TABLE IX
ARI INDICES (%) ON THE LFW DATABASE

c	NMF	GNMF	SemiGNMF	CNMF	CNMF-KL	NsNMF	AnsNMF	NMF-DC
2	13.12	14.36	15.42	15.87	15.89	13.54	13.69	17.80
3	31.35	32.01	32.98	32.82	33.10	31.62	32.05	34.98
4	37.21	37.84	38.26	37.96	38.17	37.26	37.90	40.81
5	38.97	40.95	41.35	41.03	41.23	39.25	40.27	43.87
6	31.03	30.21	32.12	31.96	32.04	32.11	33.09	35.64
7	27.87	26.99	27.66	28.02	27.88	28.26	28.34	30.49
8	24.46	24.07	25.63	25.42	35.75	25.12	25.87	28.70
9	22.43	23.64	24.19	24.38	25.01	23.19	24.02	28.33
10	20.27	23.01	23.97	23.72	23.94	21.31	22.83	26.71
Avg.	35.46	33.96	35.30	45.61	45.61	36.12	37.07	46.26

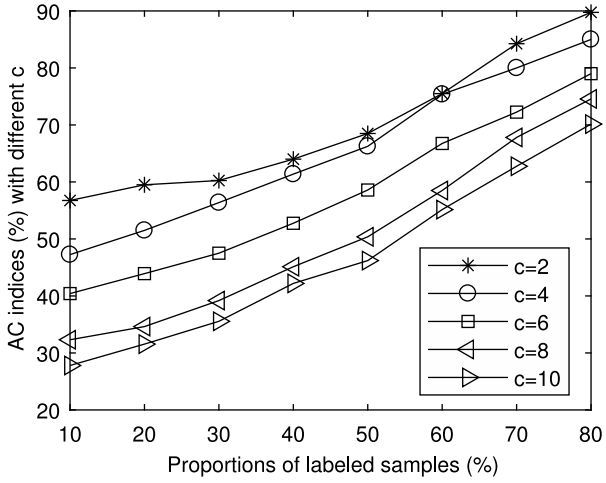


Fig. 7. AC indices of the proposed algorithm on the LFW database under different proportions of labeled samples.

Second, we compare the results of our method with that of the compared methods under different number of classes, where the proportion of the labeled data is 50%. Table VIII shows the AC indices of each compared algorithm when the number of the classes varies from 2 to 10, together with the averaged running time. And Table IX shows the corresponding ARI indices.

IV. CONCLUSION

In this paper, a new NMF method with label and sparseness constraints is proposed to learn low-dimensional representations of images for clustering. These two constraints are embodied by two non-negative matrices which can be combined with the non-negative factor matrices in traditional NMF. An efficient algorithm, called NMF-DC, is developed to solve the proposed model. Finally, one computer-generated database and three real image databases are used to test the performance of the new NMF-DC algorithm, together with several other benchmark algorithms. Simulation results show that the representations learned by our method have the best clustering performance.

REFERENCES

[1] Q. Wu, Z. Wang, F. Deng, Z. Chi, and D. D. Feng, "Realistic human action recognition with multimodal feature selection and fusion," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 4, pp. 875–885, Jul. 2013.

[2] C. Ding, C. Xu, and D. Tao, "Multi-task pose-invariant face recognition," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 980–993, Mar. 2015.

[3] G. Xu, D. Tao, and C. Xu, "Large-margin multi-view information bottleneck," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1559–1572, Aug. 2014.

[4] S. Wang and W. Zhu, "Sparse graph embedding unsupervised feature selection," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 3, pp. 329–341, Mar. 2018.

[5] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[6] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[7] Y. Sun, J. Gao, X. Hong, Y. Guo, and C. J. Harris, "Dimensionality reduction assisted tensor clustering," in *Proc. Int. Joint Conf. Neural Netw.*, Beijing, China, 2014, pp. 1565–1572.

[8] Z. Yang, Y. Xiang, K. Xie, and Y. Lai, "Adaptive method for nonsmooth nonnegative matrix factorization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 4, pp. 948–960, Apr. 2017.

[9] D. Kuang, S. Yun, and H. Park, "SymNMF: Nonnegative low-rank approximation of a similarity matrix for graph clustering," *J. Glob. Optim.*, vol. 62, no. 3, pp. 545–574, Jul. 2015.

[10] Y. Chen, M. Rege, M. Dong, and J. Hua, "Non-negative matrix factorization for semi-supervised data clustering," *Knowl. Inf. Syst.*, vol. 17, no. 3, pp. 355–379, Dec. 2008.

[11] H. Ma, W. Zhao, and Z. Shi, "A nonnegative matrix factorization framework for semi-supervised document clustering with dual constraints," *Knowl. Inf. Syst.*, vol. 36, no. 3, pp. 629–651, 2013.

[12] S. Hong, J. Choi, J. Feyereisl, B. Han, and L. S. Davis, "Joint image clustering and labeling by matrix factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 7, pp. 1411–1424, Jul. 2016.

[13] L. Jing, J. Yu, T. Zeng, and Y. Zhu, "Semi-supervised clustering via constrained symmetric non-negative matrix factorization," in *Proc. Int. Conf. Brain Informat.*, 2012, pp. 309–319.

[14] G. Li, X. Zhang, S. Zheng, and D. Li, "Semi-supervised convex nonnegative matrix factorizations with graph regularized for image representation," *Neurocomputing*, vol. 237, pp. 1–11, May 2017.

[15] Y.-H. Xiao, Z.-F. Zhu, Y. Zhao, and Y.-C. Wei, "Class-driven nonnegative matrix factorization for image representation," *J. Comput. Sci. Technol.*, vol. 28, no. 5, pp. 751–761, Sep. 2013.

[16] Y. Yi, Y. Shi, H. Zhang, J. Wang, and J. Kong, "Label propagation based semi-supervised non-negative matrix factorization for feature extraction," *Neurocomputing*, vol. 149, pp. 1021–1037, Feb. 2015.

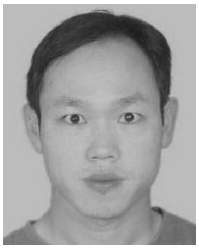
[17] M. Belkin, V. Sindhwani, and P. Niyogi, "Manifold regularization: A geometric framework for learning from examples," *J. Mach. Learn. Res.*, vol. 7, no. 11, pp. 2399–2434, 2006.

[18] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1299–1311, Jul. 2012.

[19] Z. Shu, C. Zhao, and P. Huang, "Constrained sparse concept coding algorithm with application to image representation," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 9, pp. 3211–3230, Sep. 2014.

[20] J. Kim and H. Park, "Sparse nonnegative matrix factorization for clustering," Georgia Inst. Technol., Atlanta, GA, USA, Rep. GT-CSE-08-01, 2008.

[21] A. Pascual-Montano, J. M. Carazo, K. Kochi, D. Lehmann, and R. D. Pascual-Marqui, "Nonsmooth nonnegative matrix factorization (nsNMF)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 403–415, Mar. 2006.
[22] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Comput.*, vol. 19, no. 10, pp. 2756–2779, 2007.
[23] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course.* Boston, MA, USA: Kluwer Acad., 2004.
[24] D. Cai, X. He, X. Wu, and J. Han, "Non-negative matrix factorization on manifold," in *Proc. 18th IEEE Int. Conf. Data Min.*, Pisa, Italy, 2008, pp. 63–72.
[25] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *J. Mach. Learn. Res.*, vol. 5, pp. 1457–1469, Dec. 2004.
[26] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, 1985.
[27] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," in *Advances in Face Detection and Facial Image Analysis*, M. Kawulok, M. E. Celebi, and B. Smolka, Eds. Cham, Switzerland: Springer, 2016, pp. 189–248.



Zuyuan Yang (M'15) received the B.E. degree from the Hunan University of Science and Technology, Xiangtan, China, in 2003 and the Ph.D. degree from the South China University of Technology, Guangzhou, China, in 2010.

He joined the National Program for New Century Excellent Talents in University and won the National Science Foundation for Excellent Young Scholars. He is currently a Researcher with the School of Automation, Guangdong University of Technology, Guangzhou. His current research interests include machine learning, nonnegative matrix factorization, and image processing.

Dr. Yang was a recipient of the excellent Ph.D. Thesis Award Nomination of China.

Yu Zhang was born in Hubei, China. He received the B.E. degree from the School of Automation, Guangdong University of Technology, Guangzhou, China, in 2015, where he is currently pursuing the master's degree.

His current research interests include machine learning and non-negative matrix factorization.

Yong Xiang (SM'12) received the Ph.D. degree in electrical and electronic engineering from the University of Melbourne, Melbourne, VIC, Australia.

He is a Professor and the Director of the Artificial Intelligence and Image Processing Research Cluster, School of Information Technology, Deakin University, Geelong, VIC, Australia. His current research interests include signal and system estimation, information and network security, multimedia (speech/image/video) processing, and wireless sensor networks. He has published over 130 refereed journal and conference papers in the above areas.

Dr. Xiang is an Associate Editor of the IEEE SIGNAL PROCESSING LETTERS and IEEE ACCESS. He has served as the program chair, the TPC chair, the symposium chair, and the session chair for a number of international conferences.

Wei Yan was born in Jiangxi, China. He received the master's degree from the School of Automation, Guangdong University of Technology, Guangzhou, China, in 2016. He is currently pursuing the Ph.D. degree with the Faculty of Science and Technology, University of Macau, Macau, China.

His current research interests include machine learning, non-negative signal processing, and blind signal processing.

Shengli Xie (M'01–SM'02) was born in Hubei, China, in 1958. He received the M.S. degree in mathematics from Central China Normal University, Wuhan, China, in 1992 and the Ph.D. degree in control theory and applications from South China University of Technology, Guangzhou, China, in 1997.

He is the Director of the Laboratory for Intelligent Information Processing and a Full Professor with the School of Automation, Guangdong University of Technology, Guangzhou. He has authored or co-authored two monographs, a dozen of patents and over 80 papers in journals and conference proceedings. His current research interests include automatic controls, signal processing, blind signal processing, and image processing.