Maximum Correntropy Criterion based Robust Semi-supervised Concept Factorization for Image Representation

Nan Zhou, Badong Chen, *Senior Member, IEEE*, Yuanhua Du*, Tao Jiang, *Member, IEEE,* Jun Liu, and Yangyang Xu

***Abstract*—Concept Factorization (CF) has shown its great advantage for both clustering and data representation and is particularly useful for image representation. Compared with Nonnegative Matrix Factorization (NMF), CF can be applied to data containing negative values. However, the performance of CF method and its extensions will degenerate a lot due to the negative effects of outliers, and CF is an unsupervised method that cannot incorporate label information. In this paper, we propose a novel concept factorization method, with a novel model built based on the Maximum Correntropy Criterion (MCC). In order to capture the local geometry information of data, our method integrates the robust adaptive embedding and concept factorization into a unified framework. The label information is utilized in adaptive learning process. Furthermore, an iterative strategy based on the accelerated block coordinate update is proposed. The convergence property of the proposed method is analyzed to ensure that the algorithm converges to a reliable solution. The experimental results on 4 real-world image datasets show that the new method can almost always filter out the negative effects of the outliers and outperform several state-of-the-art image representation methods.**

***Index Terms*—machine learning, concept factorization, maximum correntropy criterion, nonnegative matrix factorization, semi-supervised learning.**

I. Introduction

DIGITAL IMAGE is one of the most important data for pattern recognition and computer vision tasks. By the wide use of the cameras, images are much easier to be obtained than before. With the increasing size of the image data, we need more efficient methods to process the data. A good image representation technique can not only decrease computation time for some specific tasks by alleviating the effects of curse of dimensionality, but also improve the performance of algorithms.

There are many methods for image representation, especially linear data representation, such as Principal Component Analysis (PCA) [1], Independent Component Analysis (ICA) [2], Vector Quantization (VQ) [3], Nonnegative Matrix Factorization (NMF) [4], [5] and Concept Factorization (CF) [6]. These methods model the representation as matrix factorization problems, which use two or more matrix factors to approximate the data. They have strong connection with clustering problem [7], because one of the matrix factors can be regarded as cluster prototypes to interpret the hidden semantics and the other viewed as the coefficients.

PCA [1] is an effective unsupervised data representation method. It attempts to learn orthogonal bases that can best reconstruct the original data. Compared with PCA, NMF aims at finding two nonnegative low-rank matrices to approximate the nonnegative data matrix. The nonnegativity constraints in NMF only allow additive combination, and NMF leads to parts-based representation, which is similar to object representation in human brains [8]. Different from PCA, which can contain negative values in orthogonal bases and coefficients, NMF has better interpretability in general. However, NMF has some limitations. First, NMF requires nonnegativity of the bases. Although, this requirement may be appropriate for some kinds of data whose features are all nonnegative, such as image and text document, it is not desirable for the data that contain negative values, such as agriculture and economic data. If data contain negative numbers, the data need to be preprocessed before applying NMF, such as, by uniform shifting the data points. However, this way, it will destroy not only the interpretability of the data, but also the linearilty between data. This is illustrated by the following toy example. Suppose the data contain three samples $\mathbf{a} = [-1, 0, 1]^\top$, $\mathbf{b} = [0, 0, 1]^\top$ and $\mathbf{c} = [-2, 0, 3]^\top$. It is easy to verify that $\mathbf{c} = 2\mathbf{a} + \mathbf{b}$. Shifting the vectors along any vector $\mathbf{d}$, the three vectors are transformed to $\mathbf{a}' = \mathbf{a}+\mathbf{d}$, $\mathbf{b}' = \mathbf{b}+\mathbf{d}$, and $\mathbf{c}' = \mathbf{c}+\mathbf{d}$, and we cannot have $2\mathbf{a}'+\mathbf{b}' = \mathbf{c}'$ unless $\mathbf{d} = \mathbf{0}$. Hence, shifting data to be nonnegative could destroy the original linearity information. Second, because of the nonnegativity constraints on the matrix factors, the NMF should be applied on original data space, thus it cannot be kernelized. In order to utilize the power of kernel methods and extend NMF to the data with negative values, CF [6] was proposed. CF inherits all the strengths of NMF. In addition, it addresses the limitations of NMF by modeling

This work was made possible by support from National Key R&D Program of China (No. 2017YFB1002501), NSFC (No. 61802036, No. 91648208, No. 11901063 and No. 61703060), NSF Award DMS-1719549, National Natural Science Foundation Shenzhen Joint Research Program (No. U1613219), Key Research Project of Sichuan Province (No. 2017GZ0431), Sichuan Science and Technology Program (No. 2018GZ0385, No. 2019YFG0198 and No. 2017TD0019), Scientific Research Fund of Chengdu Science and Technology Burea (No. 2017-GH02-00049-HZ and No. 2018-YF05-00981-GX).

Nan Zhou, Tao Jiang and Jun Liu are with the College of Control Engineering, Chengdu University of Information Technology, Chengdu, Sichuan, 610225 China. (e-mail: nzhouuestc@126.com) Badong Chen is with Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shanxi, 710049, China. Yuanhua Du is with the College of Applied Mathematics, Chengdu University of Information Technology, Chengdu, Sichuan, 610225 China. Yangyang Xu is with the Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY, 12180, USA. *Corresponding author: Yuanhua Du

each basis as a linear combination of the data points, and also modeling each data point as a linear combination of bases.

Label is a very important discriminative information in machine learning tasks. If the label information is utilized properly, it can significantly improve the performance of machine learning and pattern recognition. However, compared to obtaining unlabeled data, it usually costs much more to acquire labeled data. In some cases, only a few data are labeled. Semi-supervised learning algorithms [9] were proposed to address this problem. Nevertheless, both NMF and CF are unsupervised learning methods, thus they could not take the advantage of the label information.

In this paper, we proposed a novel robust semi-supervised image representation method called MCC based robust semi-supervised concept factorization (MRSCF). The new method establishes the following benefits:

1) A novel maximum correntropy criterion (MCC) based robust concept factorization model is proposed. It can eliminate the negative effects of outliers.
2) In order to utilize the label information and data's local structure information, a novel MCC based robust non-negative adaptive embedding framework is incorporated into the model.
3) The sparsity and sum-to-one constraints are added into the model to increase the discrimination of the learned representations and avoid trivial solution.
4) A novel Fenchel conjugate and accelerated BCU combination strategy is proposed to solve the CF model. Different from the existing CF methods, which utilize the multiplicative strategy to solve the problem and only apply the convergence analysis for nonnegative data, the proposed algorithm is amenable to the data with no matter negative values or not, and convergence properties hold for both kinds of data.
5) Extensive experiments are conducted on four real-world image datasets, and seven state-of-the-art image representation methods are compared with the proposed method. The representation performance is evaluated on the clustering tasks. The results demonstrate the superiority of the new method over other compared methods.

The rest of the paper is organized as follows. Section II gives a brief review of some image representation methods and Maximum Correntropy Criterion (MCC). Section III revisits the Sparsity Induced Similarity (SIS) and proposes the MCC based Robust Nonnegative Adaptive Embedding regularization term and the MCC based robust semi-supervised CF model. Section IV develops an iterative algorithm, called MRSCF, to search the solution of the model and analyzes its convergence. Section V illustrates the experimental results. Finally, Section VI concludes the paper.

To facilitate the presentation of the paper, the notations are listed in Table I.

TABLE I
NOTATIONS

Notation	Description
n	Number of instances
d	Number of features
$A_{i\cdot}$	The i^{-th} row of the matrix A
k	Dimension of representation
$\|W\|_1$	$\sum_{i,j}\|W_{i,j}\|$, sum of absolute value of all elements of W

II. Related Works

A. NMF and CF

Given n data samples $\{\mathbf{x}_i\}_{i=1}^n$ in a d-dimensional space, the data matrix is represented as $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$. Let $k \ll \min(n, d)$. NMF aims to find two nonnegative matrix $U \in \mathbb{R}^{d\times k}$ and $H \in \mathbb{R}^{k\times n}$ to approximate the original nonnegative data matrix X by solving the following problem:

$$\min_{U,H} \|X - UH\|_F^2, \text{ s.t. } U \geq 0, H \geq 0, \tag{1}$$

where $\|\cdot\|_F$ denotes the matrix *Frobenius norm*. Each column of U can be regarded as the feature basis and each column of H is the k-dimensional representation under the basis.

It is shown that the objective function in (1) is nonconvex jointly with respect to U and H, but it is convex in one of them with the other fixed. Usually, it is very difficult to find the global minimum of (1). Lee et al. [4] proposed a multiplicative strategy to search for the local minimum iteratively as follows:

$$\begin{aligned} U_{i,m}^{t+1} &= U_{i,m}^t \frac{[X(H^t)^\top]_{i,m}}{[(U^t)(H^t)(H^t)^\top]_{i,m}}, \\ H_{m,j}^{t+1} &= H_{m,j}^t \frac{[X^\top(U^t)]_{m,j}^t}{[(H^t)^\top(U^t)^\top(U^t)]_{m,j}}. \end{aligned} \tag{2}$$

The most significant difference between NMF and other matrix factorization methods is that the substraction is not allowed in NMF. Therefore, NMF can learn a parts-based representation [5]. This parts-based representation has shown its desirable advantage in many applications, such as spectral data analysis [10], patch alignment [11], multi-view representation [12], document clustering [13], image representation [14], [15], and DNA gene expression analysis [16].

Although NMF is an efficient tool to represent some kinds of data and obtain good semantic interpretation and clustering results, it has limitations. First, it is desirable only for non-negative data. Second, it can only be conducted in original feature space of the data points. For this reason, the powerful kernel method could not be applied to NMF. To address these limitations, Xu et al. [6] extended NMF to CF. The core idea of CF is that each basis (cluster center) $\mathbf{u}_j$ is constructed as a linear combination of the data points $\mathbf{x}_i$

$$\mathbf{u}_j = \sum_{i=1}^n w_{i,j}\mathbf{x}_i, \tag{3}$$

where $w_{i,j} \geq 0$. Let $W = [w_{ij}] \in \mathbb{R}^{n\times k}$. The CF problem is formulated as follows:

$$\min_{W,H} \|X - XWH\|_F^2, \text{ s.t. } W \geq 0, H \geq 0. \tag{4}$$

Similar to NMF, if data are nonnegative, the multiplicative strategy can also be applied to solve (4) by the iterative updates:

$$\begin{aligned} W_{i,m}^{t+1} &= W_{i,m}^t \frac{[K(H^t)^\top]_{i,m}}{[K(W^t)(H^t)(H^t)^\top]_{i,m}^t}, \\ H_{m,j}^{t+1} &= H_{m,j}^t \frac{[K(W^t)]_{m,j}}{[(H^t)^\top(W^t)^\top K(W^t)]_{m,j}}, \end{aligned} \tag{5}$$

where $K = X^\top X$. In (5), the variable update strategy only involves the inner product of each samples, thus CF can be easily kernelized.

CF has shown some merits over NMF. However, the conventional CF only utilizes the global reconstruction information without considering any other discriminative information. To utilize local structure information of samples, Cai et al. [17] proposed Locally Consistent Concept Factorization (LCCF) to take advantage of samples' local structure. For the reason that the bases learned by CF may be relatively far away from the original data, Liu et al. [18] proposed Local-Coordinate CF (LCF). It assumes that each original data point should be close to only a few anchor points. This way, the learned bases can be made to be close to the original data points. However, LCCF may lead to trivial solution [19]. To address these issues, Guo et al. [20] added an orthogonality constraint and a sparse component to the LCCF model and proposed a Robust and Discriminative Concept Factorization (RDCF) method. However, none of the methods mentioned above consider how to utilize labeled data to improve the clustering performance, i.e., they are unsupervised. In order to utilize the labeled data, Cai et al. [21] revised the similarity matrix of conventional graph regularization term. If two samples share the same label, an edge is assigned a large weight to connect them. Nonetheless, there is no theoretical guidance about how to construct the graph regularization term and how to select the weights. Liu et al. [22] proposed a constrained concept factorization (CCF) that enforces the same labeled data to have the same low-dimensional representation. All the above mentioned methods used the Frobenius-norm as distance measure. When data contain outliers, their performance will degenerate a lot. Furthermore, to the best of our knowledge, all existing CF based methods only have convergence results on nonnegative data, and thus are not reliable to handle negative-valued data.

B. Maximum Correntropy Criterion

Maximum Correntropy Criterion (MCC) originates from Information Theoretic Learning (ITL) [23] and is based on Renyi's entropy [24]. It is used to handle non-Gaussian noise and outliers [25]. Because of its nice robust properties, MCC has been used in many fields, such as feature extraction [26], [27], [28], signal processing [29], [30], [31], [32], computer vision [33] and document clustering[34]. The correntropy is related to Renyi's quadratic entropy that is used to measure similarity between two random variables in a local range. Traditional correntropy is defined to measure similarity between two random variables x and y

$$V_\sigma(x, y) = E[k_\sigma(x - y)], \tag{6}$$

where $k_\sigma(\cdot)$ is a shift invariant mercer kernel [35], $\sigma > 0$ is the kernel bandwidth to present the correlation in kernel space, and $E[\cdot]$ denotes the mathematical expectation. For the reason that the joint probability density function is hard to be obtained in practice, the correntropy is defined by sample estimator as follows:

$$\hat{V}_\sigma(x, y) = \frac{1}{n}\sum_{i=1}^{n} k_\sigma(x_i - y_i), \tag{7}$$

where $k_\sigma(x) = g(x, \sigma) \triangleq \exp(-x^2/2\sigma^2)$ is Gaussian kernel and $\{x_i, y_i\}_{i=1}^n$ is the sample set. The maximum of (7) is called Maximum Correntropy Criterion (MCC).

There are some other metrics that also have better outlier robustness than traditional Frobenius norm, such as L21-norm. We choose MCC as the robust metric mainly for two reasons. First, MCC has nice robustness properties. It is a local measure whose value mainly depends on the probability along $x = y$, thus, it can truncate the effects of large error. However, L21-norm is a global measure, which can also be affected by large error outliers. It is shown in [25] that MCC is equivalent to L2-norm distance if points are close, behaves like the L1-norm distance as points get further apart, and eventually approaches the L0-norm distance as points get further apart. Second, the model based on MCC is easier to be solved. Therefore, we choose MCC as the robust metric to construct CF model.

III. MCC based Robust Semi-supervised Concept Factorization

In this section, we propose a novel robust semi-supervised CF model. MCC is utilized to construct the model for dealing with data that is contaminated by outliers. In addition, our model incorporates label information to guide the learning process. Local structure information of the data is preserved through nonnegative adaptive embedding.

A. Sparsity Induced Similarity

Graph embedding is the common way to add the local structure in machine learning methods [36], [37], [38], [39], [40]. As shown in [41], the similarity plays an important role on visual recognition. The traditional graph embedding terms [17], [20] utilize Euclidean distance and Gaussian Kernel Similarity (GKS) to measure the similarities between data samples in CF, which ignores the subspace structure. If the training samples are sufficient, the subspace structure can be discovered and effectively used for image classification [42]. However, if the training samples are insufficient such as in semi-supervised or unsupervised learning, it is impossible to obtain the precise subspace structure. In order to address this issue, Cheng et al. [43] proposed Sparsity Induced Similarity (SIS), which treats the coefficients of sparse representation as the similarities between samples. For n data samples $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, the similarity matrix is obtained by solving the following problem:

$$\begin{aligned} &\min_S \|X - XS\|_F^2 + r_\gamma(S) \\ &\text{s.t. } S_{i,i} = 0, \quad \forall i = 1, \ldots, n. \end{aligned} \tag{8}$$

Here, S is the similarity matrix, each column $\mathbf{s}_i$ of S is the sparse representation of $\mathbf{x}_i$ by X except $\mathbf{x}_i$, and $r_\gamma(S)$ is a sparsity regularization term. Because of its great performance, SIS has been used in many areas, such as visual tracking [44], object categorization [45] and image clustering [46].

B. Adaptive Embedding to Preserve Local Structure

Let $H = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_n] \in \mathbb{R}^{c\times n}$ be the low-dimensional representation matrix, namely, each column $\mathbf{h}_i$

is a low-dimensional representation of $\mathbf{x}_i$. To make the low-dimensional representation to have a similar local structure as the original data, we look for a similarity matrix S such that both $X-XS$ and $H-HS$ become small. Using SIS, one can find the matrix S to be a solution of the following problem:

$$\begin{aligned}&\min_{S}\ \lambda\|H-HS\|_F^2+\|X-XS\|_F^2+r_\gamma(S)\\&\text{s.t. } S_{i,i}=0,\quad \forall i=1,\ldots,n,\end{aligned}\tag{9}$$

where $\lambda>0$ is a parameter balancing the H-term and X-term. In the above model, we assume that H has been obtained. In the follows, we treat it as a variable and learn it simultaneously with the similarity matrix S and also basis matrix W.

C. MCC based Robust Semi-supervised CF Model

Existing CF models use squared Frobenius-norm as distance measure and are not suitable for outlier-contaminated data. To deal with outliers, we replace by MCC the squared Frobenius-norm in (4) and (9). In addition, the term $Tr\left[(H-Y)M(H-Y)^\top\right]$ is employed to utilize label information [47]. Here, Y is the $k\times n$ indicator matrix with $Y_{i,j}=1$ if $\mathbf{x}_j$ is labeled and belongs to class i and $Y_{i,j}=0$ otherwise. For example, consider 9 data samples, $\mathbf{x}_1$ and $\mathbf{x}_3$ are labeled with class 1, $\mathbf{x}_4$ and $\mathbf{x}_5$ are labeled with class 2, $\mathbf{x}_7$ and $\mathbf{x}_9$ are labeled with class 3, and the other 3 data samples are unlabeled. Then, the indicator matrix Y based on this example can be represented as follows:

$$Y=\begin{pmatrix}1&0&1&0&0&0&0&0&0\\0&0&0&1&1&0&0&0&0\\0&0&0&0&0&0&1&0&1\end{pmatrix}.\tag{10}$$

M is the diagonal matrix with $M_{i,i}=1$ if $\mathbf{x}_i$ is labeled and $M_{i,i}=0$ otherwise. Thus, this term measures the distance between representation and indicator matrices of the labeled samples, and encourages them close to each other. For example, if $\mathbf{x}$ is labeled with class 3, its representation will be pulled close to $[0,0,1]^\top$. Furthermore, this term helps to build the concepts by the labeled data samples, which will be illustrated in following experiments. Combining the two goals of (4) and (9) together and the label information, we build the following *MCC based Robust Semi-supervised CF Model*:

$$\begin{aligned}\max_{W,H,S}\ &\frac{1}{2}\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{x}_i-XW\mathbf{h}_i\|_2^2}{2\sigma_1^2}\right)+\frac{\alpha}{2}\Big\{NAE(H,S)\\&-Tr\left[(H-Y)M(H-Y)^\top\right]\Big\}-r_\beta(W)-r_\gamma(S)\\\text{s.t. }&W\geq 0,\ H\geq 0,\ S\geq 0,\\&\sum_{i=1}^{c}H_{i,j}=1,\ \forall j=1,\ldots,n,\\&S_{i,i}=0,\ \forall i=1,\ldots,n.\end{aligned}\tag{11}$$

Here,

$$\begin{aligned}&NAE(H,S)=\\&\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{h}_i-H\mathbf{s}_i\|_2^2}{2\sigma_2^2}\right)+\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{x}_i-X\mathbf{s}_i\|_2^2}{2\sigma_3^2}\right)\end{aligned}\tag{12}$$

is called the Nonnegative Adaptive Embedding (NAE) term. The constraint $\sum_{i=1}^{c}H_{i,j}=1,\ \forall j=1,\ldots,n$ is used to increase interpretability of the model and to avoid trivial solutions. The sparsity regularization term $r_\beta(W)$ is added in model (11) to improve the model's discriminability and generalization.

IV. Solving the robust semi-supervised CF model

In this section, a novel iterative method is derived to solve the problem (11). Note that the objective function of (11) is nonconcave and also lacks block-concavity. This causes great difficulty for a numerical approach to efficiently and reliably find its solution. To conquer this difficulty, we first reformulate (11) by using the Fenchel conjugate technique. Although the reformulated problem involves more variables than the original one, it has nice block structure and enables reliable numerical approaches. Then, we apply the accelerated Block Coordinate Update (BCU) [48] to solve the reformulated problem. The problem is amenable for BCU framework, and every update of BCU has an explicit solution [48], [49]. It is shown in [48] that BCU has desirable convergence behavior and very nice practical performance to solve multi-block concave problems. Existing methods for solving CF models are derived from multiplicative strategy, and they only conduct convergence analysis about nonnegative data. Different from the existing methods, the proposed algorithm and its convergence result apply for situations that can involve both nonnegative and negative values.

A. Reformulation via Fenchel Conjugate Technique

Let

$$\varphi(z)=z-z\ln(-z).\tag{13}$$

By Fenchel conjugate (see [50] for example), it is easy to verify that

$$\exp(-x)=\sup_z\left\{zx-\varphi(z)\right\}.\tag{14}$$

By setting $\frac{\partial}{\partial z}(zx-\varphi(z))=x+\ln(-z)=0$, one can find that the maximum value of (14) is reached at $z=-\exp(-x)$. Based on this finding, we define

$$O_1^{MCC}(W,H,\mathbf{p})=\frac{1}{2}\sum_{i=1}^{n}\left(p_i\frac{\|\mathbf{x}_i-XW\mathbf{h}_i\|_2^2}{2\sigma_1^2}-\varphi(p_i)\right),\tag{15a}$$

$$O_2^{MCC}(H,S,\mathbf{q})=\frac{1}{2}\sum_{i=1}^{n}\left(q_i\frac{\|H_{.i}-H\mathbf{s}_i\|_2^2}{2\sigma_2^2}-\varphi(q_i)\right),\tag{15b}$$

$$O_3^{MCC}(S,\mathbf{z})=\frac{1}{2}\sum_{i=1}^{n}\left(z_i\frac{\|\mathbf{x}_i-X\mathbf{s}_i\|_2^2}{2\sigma_3^2}-\varphi(z_i)\right),\tag{15c}$$

$$\begin{aligned}f(W,H,S,\mathbf{p},\mathbf{q},\mathbf{z})=O_1^{MCC}(W,H,\mathbf{p})+\alpha\Big\{[O_2^{MCC}(H,S,\mathbf{q})\\+O_3^{MCC}(S,\mathbf{z})-\frac{1}{2}Tr\left[(H-Y)M(H-Y)^\top\right]\Big\}.\end{aligned}\tag{15d}$$

According to the definitions of (15a)-(15d) and (14), the problem (11) is equivalent to

$$\begin{aligned}\max_{W,H,S,\mathbf{p},\mathbf{q},\mathbf{z}}\ &f(W,H,S,\mathbf{p},\mathbf{q},\mathbf{z})-r_\beta(W)-r_\gamma(S)\\\text{s.t. }&W\geq 0,\ H\geq 0,\ S\geq 0,\\&\sum_{i=1}^{c}H_{i,j}=1,\ \forall j=1,\ldots,n,\\&S_{i,i}=0,\ \forall i=1,\ldots,n,\end{aligned}\tag{16}$$

where $\mathbf{p} = (p_1, p_2, \ldots, p_n)$, $\mathbf{q} = (q_1, q_2, \ldots, q_n)$, $\mathbf{z} = (z_1, z_2, \ldots, z_n)$. For simplicity, throughout the rest of the paper, we set the sparsity regularization terms to

$$r_\beta(W) = \beta\|W\|_1, \quad r_\gamma(S) = \gamma\|S\|_1.$$

B. Iterative Method by Accelerated BCU

At each iteration of our algorithm, it updates one variable at a time with the remaining ones fixed to their latest values. Specifically, we perform the following updates:

$$\mathbf{p}^{t+1} = \underset{\mathbf{p}}{\operatorname{argmax}}\, O_1^{MCC}(W^t, H^t, \mathbf{p}), \tag{17a}$$

$$\mathbf{q}^{t+1} = \underset{\mathbf{q}}{\operatorname{argmax}}\, O_2^{MCC}(H^t, S^t, \mathbf{q}), \tag{17b}$$

$$\mathbf{z}^{t+1} = \underset{\mathbf{z}}{\operatorname{argmax}}\, O_3^{MCC}(S^t, \mathbf{z}), \tag{17c}$$

$$W^{t+1} = \underset{W\geq 0}{\operatorname{argmax}} \left\langle \nabla_W f(\hat{W}^t, H^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1}) \right. , \left. W - \hat{W}^t \right\rangle - \frac{L_W^t}{2}\|W - \hat{W}^t\|_F^2 - r_\beta(W), \tag{17d}$$

$$H^{t+1} = \underset{H\geq 0,\ \sum_{i=1}^c H_{i,j}=1}{\operatorname{argmax}} \left\langle \nabla_H f(W^{t+1}, \hat{H}^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1}), H - \hat{H}^t \right\rangle - \frac{L_H^t}{2}\|H - \hat{H}^t\|_F^2, \tag{17e}$$

$$S^{t+1} = \underset{S\geq 0, S_{i,i}=0}{\operatorname{argmax}} \left\langle \nabla_S f(W^{t+1}, H^{t+1}, \hat{S}^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1}), S - \hat{S}^t \right\rangle - \frac{L_S^t}{2}\|S - \hat{S}^t\|_F^2 - r_\gamma(S). \tag{17f}$$

Here, $\nabla_W f(\cdot)$, $\nabla_H f(\cdot)$ and $\nabla_S f(\cdot)$ are partial gradients with respect to W, H and S. L_W^t, L_H^t and L_S^t are the Lipschitz constants of $\nabla_W f(W, H^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$, $\nabla_H f(W^{t+1}, H, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$ and $\nabla_S f(W^{t+1}, H^{t+1}, S, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$ with respect to W, H and S, and

$$\begin{aligned} \hat{W}^t &= W^t + \omega_W^t(W^t - W^{t-1}), \\ \hat{H}^t &= H^t + \omega_H^t(H^t - H^{t-1}), \\ \hat{S}^t &= S^t + \omega_S^t(S^t - S^{t-1}) \end{aligned} \tag{18}$$

are extrapolated points with the weights $\omega_W^t, \omega_H^t, \omega_S^t \in [0, 1)$ which can significantly accelerate the BCU method for solving certain multi-block optimization problem [48], [49], [51].

It is noted that the updates of all the variables are treated in two different way. The variables W, H and S are updated by block proximal gradient method while the others are updated by simple block maximization, because directly maximizing the objective of (16) with respect to W, H and S can be very difficult. In the way shown in (17), each of the subproblems has a closed-form solution. In the following, we will discuss how to solve each of them explicitly.

1) Solve the subproblems of $\mathbf{p}$, $\mathbf{q}$ *and* $\mathbf{z}$*:* As mentioned at the beginning of Section IV-A, the maximum value of (14) can be obtained when $z = -\exp(-x)$. Therefore, the solutions for subproblems of (17a)-(17c) can be obtained explicitly as

$$p_i^{t+1} = -\exp\left(-\frac{\|\mathbf{x}_i - XW^t\mathbf{h}_i^t\|_2^2}{2\sigma_1^2}\right) \quad \forall i = 1, \ldots, n, \tag{19a}$$

$$q_i^{t+1} = -\exp\left(-\frac{\|\mathbf{h}_i^t - H^t\mathbf{s}_i^t\|_2^2}{2\sigma_2^2}\right) \quad \forall i = 1, \ldots, n, \tag{19b}$$

$$z_i^{t+1} = -\exp\left(-\frac{\|\mathbf{x}_i - X\mathbf{s}_i^t\|_2^2}{\sigma_3^2}\right) \quad \forall i = 1, \ldots, n. \tag{19c}$$

From (19), it is shown that the updates to dual variables will renew the weights to each samples' distance metric and improve the robustness to outliers. If one sample is outlier, the corresponding distance is large, and the weight tends to zero. In this way, the negative effects of outliers are eliminated.

2) Solve W*-subproblem:* It is noted that the W-subproblem (17d) can be reformulated as

$$\min_{W\geq 0} \frac{1}{2}\|W - A\|_F^2 + \lambda\|W\|_1, \tag{20}$$

where $A = \hat{W}^t + \frac{1}{L_W^t}\nabla_W f(\hat{W}^t, H^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$ and $\lambda = \frac{\beta}{L_W^t}$. The equivalence between (17d) and (20) is given in Appendix A. The above problem has a closed-form solution so that the W-subproblem (17d) can be solved explicitly. For convenience, the process to obtain solution for problem (20) is illustrated in Algorithm 1.

Algorithm 1 Proximal operator: $W = \text{Prox}(A, \lambda)$

```
Set W to zero matrix
for i = 1, ..., d do
    for j = 1, ..., n do
        if A_ij > λ then
            Let W_ij = A_ij − λ.
        end if
    end for
end for
```

3) Solve H*-subproblem:* The H-subproblem (17e) can be formulated as

$$\min_{H\geq 0,\ \sum_{i=1}^c H_{i,j}=1} \frac{1}{2}\|H - B\|_F^2, \tag{21}$$

where $B = \hat{H}^t + \frac{1}{L_H^t}\nabla_H f(W^{t+1}, \hat{H}^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$. The problem (21) can be decomposed into n smaller independent problems, each one involving one column of H and B in the form of

$$\min_{\mathbf{h}\geq 0,\ \sum_{i=1}^c h_i=1} \frac{1}{2}\|\mathbf{h} - \mathbf{b}\|_F^2. \tag{22}$$

This problem is a projection onto a simplex. According to [52], it can be exactly solved, so can the H-subproblem (17e). For convenience, the process to obtain the solution for problem (22) is illustrated in Algorithm 2.

Algorithm 2 Projection onto simplex: $\mathbf{h} = \text{Proj-Sim}(\mathbf{b})$ [52]

1: Sort $\mathbf{b}$ in the ascending order as $b_{(1)} \leq \ldots \leq b_{(c)}$, and set $i = c - 1$.
2: Compute $t_i = \frac{\sum_{j=i+1}^c b_{(j)} - 1}{c-i}$. If $t_i \geq b_{(i)}$ then $\hat{t} = t_i$ and go to Step 4, otherwise set $i \leftarrow i - 1$ and redo Step 2 if $i \geq 1$ or go to Step 3 if $i = 0$.
3: Set $\hat{t} = \frac{\sum_{j=1}^c b_j - 1}{c}$.
4: $\mathbf{h} = (\mathbf{b} - \hat{t})_+$ as the projection of $\mathbf{b}$ onto Δ^c.

4) Solve S*-subproblem:* In the similar way as done in Section IV-B2, S-subproblem can be reformulated as

$$\min_{S\geq 0, S_{i,i}=0} \frac{1}{2}\|S - C\|_F^2 + \eta\|S\|_1, \tag{23}$$

where $C = \hat{S}^t + \frac{1}{L_S^t}\nabla_S f(W^{t+1}, H^{t+1}, \hat{S}^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$ and $\eta = \frac{\gamma}{L_S^t}$. Because of the constraint that $S_{i,i} = 0$, $\forall i =$

$1, \ldots, n$, the above problem (23) is equivalent to the following formulation

$$\min_{S \geq 0} \frac{1}{2} \left\| S - \hat{C} \right\|_F^2 + \eta \|S\|_1, \tag{24}$$

where $\hat{C}$ equals to C except for the diagonal elements, and $\hat{C}_{i,i} = 0, \forall i = 1, \ldots, n$. The problem (24) can be solved by Algorithm 1, and thus a closed-form solution of (23) can be obtained.

C. Parameters Setting

It is noted that, to fully determine the updates of variables W, H and S in (17) requires the values of Lipschitz constants L_W^t, L_H^t, L_S^t and the weights ω_W^t, ω_H^t, ω_S^t. In the proposed algorithm, the Lipschitz constant values are set to

$$L_W^t = \|X^\top X\|_2 \|H^t P^{t+1} (H^t)^\top\|_2, \tag{25a}$$

$$L_H^t = \|(W^{t+1})^\top X^\top X W^{t+1}\|_2 \|P^{t+1}\|_2 + \alpha \|(I - S^t) Q^{t+1} (I - S^t)^\top\|_2 + \|U\|_2, \tag{25b}$$

$$L_S^t = \|(H^{t+1})^\top H^{t+1}\|_2 \|Q^{t+1}\|_2 + \|X^\top X\|_2 \|Z^{t+1}\|_2, \tag{25c}$$

where P, Q and Z are diagonal matrices with $\mathbf{p}$, $\mathbf{q}$ and $\mathbf{z}$ as the diagonal elements. The derivations about how to obtain the Lipschitz constants are illustrated in the appendix. For the extrapolation weights, they are set according to [48] as

$$\omega_W^t = \min\left(\hat{\omega}^t, \delta_\omega \sqrt{\frac{L_W^{t-1}}{L_W^t}}\right), \tag{26a}$$

$$\omega_H^t = \min\left(\hat{\omega}^t, \delta_\omega \sqrt{\frac{L_H^{t-1}}{L_H^t}}\right), \tag{26b}$$

$$\omega_S^t = \min\left(\hat{\omega}^t, \delta_\omega \sqrt{\frac{L_S^{t-1}}{L_S^t}}\right), \tag{26c}$$

where $\delta_\omega < 1$ is predetermined and $\hat{\omega}^t = (\tau_{t-1} - 1)/\tau_t$ with

$$\tau_0 = 1, \tau_t = \frac{1}{2}\left(1 + \sqrt{1 + 4\tau_{t-1}^2}\right). \tag{27}$$

Summarizing all the discussions in this subsection together, we have an iterative method toward finding a solution to (11), and its pseudo-code is illustrated in Algorithm 3, called MRSCF. Note that in Line 14, we reset the iterate if a decrease of the objective is detected. As demonstrated in [53], [54], maintaining monotonicity of the objective can significantly improve the performance of BCU.

D. Computational Complexity

We give the computational complexity per iteration of Algorithm 3. The analysis is based on general case without considering the special structure of the data matrix X. If X has certain structure, e.g., sparsity, the computational complexity can be lower. We assume the cluster number is $k < \min(d, n)$. The main cost of Algorithm 3 lies in the update of $\mathbf{p}$, $\mathbf{q}$, $\mathbf{z}$, W, H and S. The costs of updating $\mathbf{p}$, $\mathbf{q}$ and $\mathbf{z}$ by (19) are $O(ndk)$, $O(n^2k)$ and $O(n^2d)$ respectively. For updating W, H and S, the main cost is the computation of $\nabla_W f$, $\nabla_H f$ and $\nabla_S f$. For computation of the partial gradient $\nabla_W f$ in (34), we can first compute XW and $HPH^\top$ and $XPH^\top$, then $XWHPH^\top$, finally multiply $X^\top$ to $XWHPH^\top - XPH^\top$. In this way, it takes about $3ndk + nk^2$. In the same way, the computation of $\nabla_H f$ in (36) takes about $3ndk + 2n^2k$. The computation of $\nabla_S f$ in (38) takes about $3n^2d + nk^2$. Therefore, the per-iteration computational complexity is $O(ndk + n^2d)$ since $k < \min(d, n)$, and the algorithm is scalable to dimension of the data.

Algorithm 3 **M**CC based **R**obust **S**emi-supervised **C**oncept **F**actorization (MRSCF)

1: **Input**: Data matrix $X \in \mathbb{R}^{d \times n}$, indicator matrix Y, parameters α, β and γ.
2: **Output**: Clustering indicator matrix (representation matrix) H.
3: **Initialize** $W^0 \in \mathbb{R}_+^{n \times k}$, $H^0 \in \mathbb{R}_+^{k \times n}$, $S^0 \in \mathbb{R}_+^{n \times n}$, choose a positive number $\delta_\omega < 1$, and set $t = 0$.
4: **while** Not convergent **do**
5: Update $\mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1} \leftarrow$(19).
6: Compute L_W^t, L_H^t, L_S^t and $\omega_W^t, \omega_H^t, \omega_S^t$ according to (25) and (26) respectively.
7: Let $\hat{W}^t = W^t + \omega_W^t (W^t - W^{t-1})$.
8: Let $\hat{H}^t = H^t + \omega_H^t (H^t - H^{t-1})$.
9: Let $\hat{S}^k = S^t + \omega_S^t (S^t - S^{t-1})$.
10: Update $W^{t+1} \leftarrow$ solving problem (20) according to Alg. 1.
11: Update $H^{t+1} \leftarrow$ solving problem (21) according to Alg. 2.
12: Update $S^{t+1} \leftarrow$ solving problem (23) according to Alg. 1.
13: **if** $f(W^{t+1}, H^{t+1}, S^{t+1}, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{q}^{t+1}) - r_\beta(W^{t+1}) - r_\gamma(S^{t+1})) \leq f(W^t, H^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{q}^{t+1}) - r_\beta(W^t) - r_\gamma(S^t))$ **then**
14: Set $\hat{W}^t = W^t$, $\hat{H}^t = H^t$, $\hat{S}^t = S^t$ and back to **Step** 10.
15: **end if**
16: Let $t \leftarrow t + 1$.
17: **end while**

E. Convergence results

In this section, we analyze the convergence of Algorithm 3. For ease of description, we denote

$$V = (W, H, S, \mathbf{p}, \mathbf{q}, \mathbf{z})$$

and

$$F(V) = f(W, H, S, \mathbf{p}, \mathbf{q}, \mathbf{z}) - \beta\|W\|_1 - \gamma\|S\|_1.$$

We first show the boundedness of the iterates and then apply the convergence results in [48].

Lemma 1 (Boundedness of iterate sequence)**.** *Let $\{V^t = (\mathbf{p}^t, \mathbf{q}^t, \mathbf{z}^t, W^t, S^t, H^t)\}$ be the sequence generated from Algorithm 3. If β and γ are both positive, then $\{V^t\}$ is bounded.*

Proof. From (19a), it follows that $-1 \leq p_i^t < 0$, and thus $\{\mathbf{p}^t\}$ is bounded. Similarly, $\{\mathbf{q}^t\}$ and $\{\mathbf{z}^t\}$ are both bounded. In addition, from the conditions $H^t \geq 0$ and $\sum_{i=1}^c H_{i,j}^t = 1$, we have $0 \leq H_{i,j} \leq 1$. Hence $\{H^t\}$ is bounded. Furthermore, since $F(V^{t+1}) \geq F(V^t), \forall t$, both $\{W^t\}$ and $\{S^t\}$ must be bounded because otherwise $F(V^t)$ can approach to $-\infty$. Therefore, we complete the proof. □

Theorem 1 (Global iterate sequence convergence)**.** *Let $\{V^t = (\mathbf{p}^t, \mathbf{q}^t, \mathbf{z}^t, W^t, S^t, H^t)\}$ be the sequence generated from Algorithm 3. Then V^t converges to a critical point $\bar{V}$ of the problem* (16)*.*

Proof. For a set $\mathcal{X}$, denote $\iota_{\mathcal{X}}$ as the indicator function on $\mathcal{X}$. Let

$$\Phi(V) = F(V) - \iota_{\mathcal{W}}(W) - \iota_{\mathcal{H}}(H) - \iota_{\mathcal{S}}(S),$$

where

$$\mathcal{W} = \{W : W \geq 0\}, \mathcal{H} = \{H : H \geq 0, \sum_{i=1}^{c} H_{i,j} = 1, \forall j\},$$

and

$$\mathcal{S} = \{S : S \geq 0, S_{i,i} = 0, \forall i\}.$$

Then the problem (16) is equivalent to $\max_V \Phi(V)$.

Below we show the result by verifying all conditions required by Theorem 2.8 of [48]. First, note that Φ is the sum of a real analytic function and a semialgebraic function. Hence, it satisfies the Kurdyka-Lojasiewicz inequality [55], [56]. Secondly, it is easy to see $\varphi''(z) = -\frac{1}{z}$, where φ is defined in (13). From the proof of Lemma 1, we have $-1 \leq p_i^t < 0$, and thus F is strongly concave with respect to $\mathbf{p}$ with modulus 1 on the box $\times_{i=1}^{n}[-1, 0)$. Similarly, F is strongly concave with respect to $\mathbf{q}$ and $\mathbf{z}$ with modulus 1. Thirdly, the smooth part of F is Lipschitz differentiable about W, S, and H, as we derive in the appendix. Finally, from Lemma 1, we have that $\{V^t\}$ is bounded, and thus it must have a finite cluster point. Therefore, all conditions required by Theorem 2.8 of [48] are satisfied, and the result directly follows. □

F. Why MRSCF Model Works Better

Two Semi-supervised CF methods are compared to MRSCF in the experiments. One is Semi-supervised Locally Consistent Concept Factorization (SemiLCCF) method [17] and the other is Constrained Concept Factorization (CCF) method [22]. The objective function of LCCF method is

$$O_{SemiLCCF} = \frac{1}{2}\|X - XWH\|_F^2 + \frac{\lambda}{2}Tr(HLH^\top), \tag{28}$$

where L is the precomputed graph Laplacian. The label information is incorporated into the graph structure by modifying the weight matrix. In MRSCF model (11), if we let all bandwidth parameters $\sigma_1, \sigma_2, \sigma_3 \to \infty$ and $\beta, \gamma = 0$, its objective function becomes

$$\begin{aligned} O_{MRSCF} = &-\frac{1}{2}\|X - XWH\|_F^2 - \frac{\alpha}{2}\{Tr(H(I-S)(I-S)^\top H^\top) \\ &- \|X - XS\|_F^2 - Tr[(H-Y)^\top M(H-Y)^\top]\}. \end{aligned} \tag{29}$$

The matrix $(I-S)(I-S)^\top$ in the second term of (29) is used to save the local structure information. Compared with graph Laplacian in SemiLCCF model, it is not precomputed but adaptively learned from data's nonnegative sparse representation and label information, which is more discriminative than precomputed graph Laplacian [42]. Secondly, because of the Frobenius norm adopted in LCCF to measure the quality of data reconstruction, it is very sensitive to outliers in the data. Furthermore, since SemiLCCF only has nonnegativity constraints $W \geq 0, H \geq 0$, the graph regularization in SemiLCCF model is not lower-bounded. Suppose (W^*, H^*) is the obtained solution, given $\delta > 1$, it is easy to verify that $(\delta W^*, \frac{1}{\delta}H^*)$ can lead to smaller objective value. Consequently, this will lead $W^* \to \infty$ and $H^* \to 0$, which is meaningless and referred to the scale transfer problem [19]. While, in MRSCF model, the constraint, $\sum_{i=1}^{c} H_{i,j} = 1, \forall j = 1, \ldots, n$, has been added in the model and the scale transfer problem is fixed.

The objective function of CCF method is

$$O_{CCF} = \frac{1}{2}\|X - XWZA\|_F^2, \tag{30}$$

where A is a constraint matrix to incorporate the label information. If labeled samples belong to the same class, they are constrained to have the same representations, which can be easily clustered into the same class. This kind of constraint could be too strict, because the samples belonging to the same class do not necessarily have exactly the same representations. Compared with CCF model, MRSCF method utilizes a softer way to incorporate the label information. The labeled samples belonging to the same class will learn strong weight connections in S by the second and forth terms of (29). These connections will benefit the other unlabeled samples to learn the connections between them and labeled data by the third term of (29), and the label information can help to learn better concepts in the first term of (29). It is also noted that the objective function of CCF is constructed by Frobenius norm, which is sensitive to outliers in the data. Finally, compared with MRSCF method, CCF only utilizes the data's reconstruction information. It lacks the discriminative local structure information. Therefore, it can explain why MRSCF method can achieve superior performance in clustering problem in the following experiments.

V. Experiments

In this section, the experimental results are illustrated. The effectiveness of the proposed MRSCF method is evaluated through image clustering and compared with seven other related methods on four datasets. The compared methods not only include standard CF and NMF methods, but also include semi-supervised methods. They are listed as follows:

1) **CF**: Concept Factorization [6];
2) **NMF**: Nonnegative Matrix Factorization [4];
3) **SemiGNMF**: Semi-supervised Graph Regularized Nonnegative Matrix Factorization [21];
4) **SemiLCCF**: Semi-supervised Locally Consistent Concept Factorization [17];
5) **CCF**: Constrained Concept Factorization [22];
6) **LCF**: Local-Coordinate Concept Factorization [18];
7) **RDCF**: Robust and Discriminative Concept Factorization [20].

Both SemiGNMF and SemiLCCF methods incorporate the label information by modifying the weight matrix of the graph.

A. Datasets

In the experiments, four benchmark datasets are used, whose statistics are shown in Table II. Yale, WarpAR and Orl are face datasets, each instance of which demonstrates a single gray face image. All the images contain faces with different lighting

conditions, facial expressions and with/without occlusions. MNIST is a handwritten digit dataset, and each instance demonstrates a single gray handwritten digit image. The whole MNIST dataset contains $70,000$ samples. We can of course use the whole dataset. But for simplicity and to save time, we randomly select 500 samples with each digit have 50 samples from the first $10,000$ training samples to generate a small subset of MNIST dataset. To test the robustness of the proposed method, we add dummy images with the same size as original image into the datasets as outliers. Each pixel of dummy image is randomly set to 0 or 255. The examples of dummy images are show in Figure 1 and numbers of outliers for each datasets are illustrated in Table II.

Fig. 1. The examples of dummy image added in the datasets

TABLE II
THE DATASETS DETAIL

Dataset	♯ Instances(n)	♯ Features(d)	♯ Classes(C)	♯ Outliers
Yale	165	1024	15	33
WarpAR	130	1024	10	26
Orl	400	1024	40	80
MNIST	500	784	10	100

B. Experimental Settings

Some parameters need to be set and explained in advance. In the clustering experiments, in order to efficiently test the clustering performance of the proposed method, we conduct the clustering experiment many times with variations of cluster number k. The cluster number ranges from 2 to 10. For each cluster number k, we repeat 10 times to randomly select k clusters from the dataset and run the test, and we calculate the average and variance over the 10 test runs as the final scores. Since the clustering results of each method depend on the initialization, each test run consists of 10 subruns with different random initializations and the best result is reported. For all the compared methods except CCF, the cluster label can be generated from H in two ways. One is applying K-Means to the representation H, and the other is setting the cluster label of instance i as $c = \operatorname{argmax}_j H_{i,j}$. Both ways are applied, and the best performance of each compared method is illustrated. According to [20], RDCF and MRSCF methods just use the latter one to generate cluster label. For each test, 30% of the samples are labeled for semi-supervised learning and the others for testing. The local structure trade-off parameters of LCCF, GNMF, RDCF, MRSCF methods and the sparsity induced regularization parameters of MRSCF method are set by the "grid-search" strategy from set $\{0.001, 0.01, 0.1, 1, 10, 100, 500, 1000\}$. According to [20], the orthogonal preserving regularization parameter of RDCF is set to 10000. For LCF method, according to [22], the regularization parameter is set by the "grid-search" strategy from set $\{1, 2, 4, 8, 16, 32, 64\}$. For fairness, all the methods run to 100 iterations. According to [33], the kernel size σ of MCC is set as a function of the average reconstruction error, $\sigma_1 = \sqrt{\frac{\theta}{2n}\|X - XWH\|_F^2}$, $\sigma_2 = \sqrt{\frac{\theta}{2n}\|H - HS\|_F^2}$, $\sigma_3 = \sqrt{\frac{\theta}{2n}\|X - XS\|_F^2}$, where θ is a constant to control the noise. We set $\theta = 1$ throughout the paper.

Two metrics are used to evaluate the clustering performance [6]: clustering accuracy (ACC) and normalized mutual information (NMI), which are defined below. The ACC is computed by

$$ACC = \frac{\sum_{i=1}^{n} \delta(q_i, \text{map}(p_i))}{n}, \tag{31}$$

where p_i and q_i are the predicted and true labels of the ith instance, and $\delta(\cdot)$ is the indicator function where $\delta(a, b) = 1$ if $a = b$ and $\delta(a, b) = 0$ otherwise. map$(\cdot)$ is a permutation mapping function, which is realized by Kuhn-munkres algorithm [57]. NMI is defined to measure the similarity of two label vectors P and Q:

$$NMI(P, Q) = \frac{I(P, Q)}{\sqrt{H(P)H(Q)}}, \tag{32}$$

where $I(P, Q)$ is the mutual information of P and Q, and $H(P)$ and $H(Q)$ are the entropies of P and Q [58]. Higher value of ACC (NMI) indicates better clustering result.

C. Clustering Results

The clustering results of different methods are presented in Figures 2-5 and Tables III-VI. Figures 2-5 illustrate the clustering results of ACC and NMI versus the number of clusters of different methods on the Yale, Orl, WarpAR and MNIST datasets respectively. The corresponding details of Figures 2-5 and the standard deviation are provided in Tables III-VI respectively. From the figures and tables, we see that the proposed MRSCF method consistently outperforms over all the other compared methods on all four tested datasets. On the Yale dataset, compared with the second best method CCF and RDCF, the proposed MRSCF method achieves more than 20% improvement in ACC and 16% improvement in NMI on the average values. One possible reason is that MRSCF method can better utilize the label information compared with CCF method, and because of MCC, MRSCF method can better handle outlier-contaminated data. Although RDCF is also a robust CF method, it can only handle the situation that image contains noise on pixels but no outliers. Furthermore, RDCF does not utilize the label information. On the Orl dataset, compared with the second best method, MRSCF can achieve 9% improvement in ACC and 6% improvement in NMI on average value. It is also noted that the unsupervised RDCF method is even better than three semi-supervised methods. It may be because the outliers seriously effect the performance of semi-supervised methods. Therefore, it is necessary to consider the robustness of the semi-supervised learning methods. On the WarpAR dataset, compared with the second best RDCF method, the proposed MRSCF method can achieve more than 22% improvement in ACC and 12% improvement in NMI on average values. On the MNIST dataset, compared with the second best CCF method, the proposed MRSCF method can

TABLE III
CLUSTERING RESULTS ON THE YALE DATABASE

k	Accuracy(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	70.45 ± 16.79	70.00 ± 16.66	74.09 ± 16.27	73.64 ± 15.32	80.45 ± 14.94	77.73 ± 18.68	79.09 ± 17.51	**90.91 ± 8.38**
3	56.60 ± 12.14	54.85 ± 9.04	58.48 ± 12.28	55.45 ± 7.43	57.58 ± 7.42	63.64 ± 10.50	63.94 ± 11.69	**79.70 ± 7.43**
4	50.23 ± 9.98	49.77 ± 11.61	50.23 ± 9.06	47.95 ± 4.60	56.36 ± 14.37	53.86 ± 6.35	61.36 ± 11.23	**81.36 ± 7.23**
5	47.64 ± 10.02	47.45 ± 13.15	44.73 ± 7.05	43.64 ± 11.06	56.73 ± 9.95	54.36 ± 13.71	52.55 ± 12.05	**72.36 ± 7.17**
6	45.61 ± 8.72	48.33 ± 7.59	42.27 ± 7.31	41.36 ± 9.61	51.82 ± 11.61	48.03 ± 7.55	50.15 ± 8.90	**73.64 ± 7.08**
7	45.84 ± 4.76	48.05 ± 4.07	40.39 ± 4.70	47.92 ± 6.96	53.64 ± 6.42	50.52 ± 5.94	51.95 ± 6.60	**73.77 ± 4.78**
8	44.55 ± 6.60	45.57 ± 6.36	40.00 ± 5.63	43.33 ± 8.23	49.77 ± 6.62	47.50 ± 7.74	48.18 ± 7.03	**71.25 ± 6.57**
9	40.81 ± 2.31	44.14 ± 2.79	37.68 ± 5.61	38.69 ± 5.38	44.75 ± 5.11	42.83 ± 4.52	47.98 ± 6.21	**69.29 ± 5.70**
10	40.00 ± 4.32	45.00 ± 4.57	36.91 ± 4.67	40.55 ± 3.45	46.82 ± 3.57	42.73 ± 6.18	44.82 ± 4.95	**70.27 ± 4.65**
Avg.	49.08	50.35	47.20	49.50	55.32	53.47	55.56	**75.84**

k	Normalized Mutual Information(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	28.55 ± 36.59	25.44 ± 34.79	30.54 ± 34.31	28.77 ± 33.85	42.45 ± 35.00	42.02 ± 39.73	42.92 ± 42.31	**66.20 ± 29.73**
3	21.90 ± 18.18	25.20 ± 19.20	28.34 ± 20.19	25.57 ± 16.45	34.95 ± 16.01	36.55 ± 11.50	34.69 ± 18.50	**51.28 ± 14.24**
4	30.66 ± 14.56	29.60 ± 21.15	26.89 ± 14.41	26.62 ± 11.37	33.36 ± 20.22	35.32 ± 9.12	41.42 ± 16.70	**60.30 ± 15.83**
5	32.24 ± 13.29	30.65 ± 14.78	28.40 ± 11.42	28.67 ± 15.64	45.34 ± 13.07	35.64 ± 17.76	34.14 ± 15.98	**53.52 ± 9.50**
6	34.87 ± 11.95	39.57 ± 10.71	31.45 ± 10.35	31.20 ± 12.89	42.49 ± 13.91	35.18 ± 8.47	40.12 ± 13.60	**59.65 ± 8.87**
7	39.07 ± 5.38	40.67 ± 4.51	34.40 ±6.37	39.28 ± 9.25	47.38 ± 6.97	42.20 ± 5.78	45.06 ± 8.68	**61.66 ± 6.88**
8	37.97 ± 8.30	41.73 ± 7.63	36.52 ± 9.15	37.07 ± 11.37	47.24 ± 5.98	41.34 ± 8.50	42.73 ± 9.03	**59.91 ± 7.13**
9	38.55 ± 2.66	41.76 ± 3.68	34.68 ± 6.07	35.54 ± 6.04	43.12 ± 4.56	40.31 ± 5.21	45.15 ± 6.30	**59.58 ± 6.46**
10	39.03 ± 3.87	43.14 ± 5.12	34.61 ± 5.39	38.84 ± 4.02	47.00 ± 3.87	41.02 ± 6.23	43.87 ± 3.72	**61.84 ± 5.35**
Avg.	33.64	35.31	31.76	32.39	42.59	38.84	41.12	**59.33**

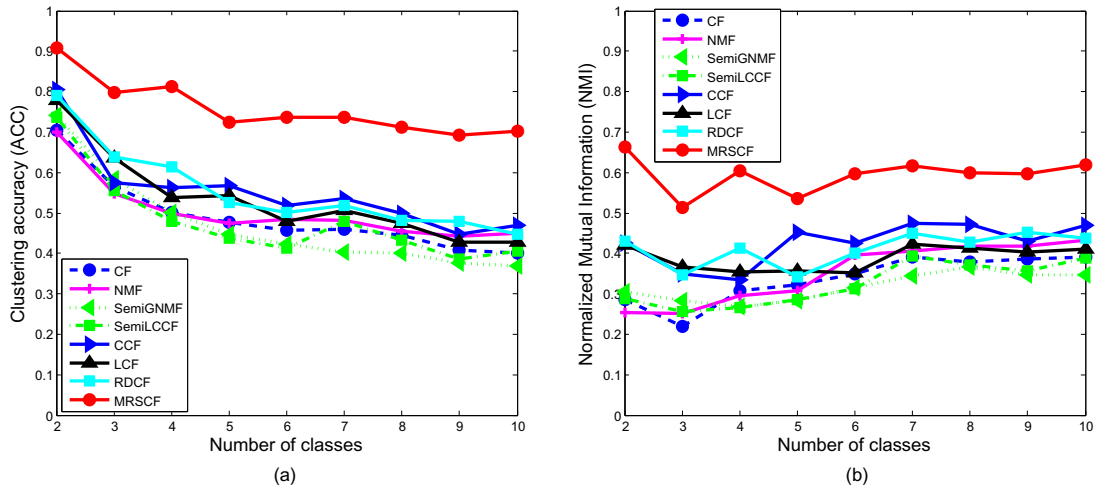

Fig. 2. Clustering performance on Yale database. (a) The clustering accuracy (ACC) versus number of clusters. (b) The normalized mutual information (NMI) versus number of clusters.

achieve almost 15% improvement in ACC and 9% in NMI on average values.

D. Clustering with Different Outlier Levels

In order to illustrate the robustness of the proposed method, the clustering experiments are conducted on Yale dataset with different outlier levels. The outlier levels are changed from 0% to 50% with increment 10%. The results are presented in Table VII and Figure 7. At each outlier level, the experiments are repeated 10 times on randomly selected 10 classes from the dataset. The average and variance over 10 runs are reported as the final scores. From the table and figures, it is noted that the performances of compared methods except MRSCF degenerate if data contain outliers, and the proposed MRSCF method can obtain the best clustering results in each outlier level. Even without outliers, MRSCF can still achieve the best scores, and much better than the CCF method. The reason may be that the proposed adaptive embedding framework can better introduce the label information. It is interesting to note that MCC metric is suitable not only for outliers contaminated data, but also for clean data. It is also noted that some scores of the compared methods are not monotonically decreasing with outlier level. For example, the CCF method gives higher accuracy at outlier level 30% than at 20%. We display the concepts of the CCF method with 20% and 30% outlier levels in Figure 6. From the figure, we see that the concepts under 30% outlier level contain more meaningless ones than those under 20% outlier level. However, the first and sixth concepts under 20% outlier level belong to the same person, and that leads to confusion in clustering task and worse performance than 30% outlier level results.

TABLE IV
CLUSTERING RESULTS ON THE ORL DATABASE

k	Accuracy(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	78.00 ± 14.70	76.00 ± 16.25	83.50 ± 16.13	84.00 ± 15.78	79.50 ± 13.31	82.00 ± 19.26	90.00 ± 16.58	**95.00 ± 5.48**
3	58.67 ± 12.75	67.33 ± 14.74	80.33 ± 16.09	81.67 ± 14.85	77.33 ± 14.97	76.00 ± 11.91	84.67 ± 14.66	**89.00 ± 7.90**
4	60.00 ± 12.40	65.25 ± 13.30	74.50 ± 12.24	73.25 ± 15.13	54.75 ± 8.47	68.25 ± 14.41	82.75 ± 13.94	**90.00 ± 5.81**
5	48.60 ± 8.53	59.40 ± 13.30	69.40 ± 12.90	65.40 ± 10.85	59.00 ± 11.91	58.20 ± 12.50	81.00 ± 11.36	**90.60 ± 6.33**
6	52.00 ± 10.56	54.17 ± 12.85	57.33 ± 12.07	59.83 ± 11.51	56.33 ± 7.52	52.50 ± 12.83	77.00 ± 9.27	**89.33 ± 3.74**
7	48.71 ± 9.42	57.57 ± 11.07	66.00 ± 11.26	66.86 ± 8.54	57.71 ± 9.98	56.86 ± 9.56	81.14 ± 9.58	**89.29 ± 2.95**
8	43.25 ± 7.65	50.25 ± 10.78	55.75 ± 9.70	61.38 ± 11.34	49.25 ± 6.78	53.38 ± 6.10	75.12 ± 9.69	**84.75 ± 5.88**
9	42.44 ± 5.98	50.22 ± 8.59	56.44 ± 11.67	56.89 ± 8.50	51.78 ± 9.64	50.67 ± 7.73	74.78 ± 5.86	**86.11 ± 4.31**
10	41.20 ± 8.73	47.80 ± 6.32	56.00 ± 7.85	54.20 ± 4.94	51.90 ± 7.08	45.20 ± 6.06	73.50 ± 7.42	**87.00 ± 2.65**
Avg.	52.54	58.66	67.00	67.05	59.73	64.34	80.00	**89.01**

k	Normalized Mutual Information(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	38.16 ± 35.74	35.56 ± 32.13	53.33 ± 38.95	52.58 ± 36.52	44.46 ± 40.19	53.74 ± 42.69	74.10 ± 38.17	**79.24 ± 20.56**
3	36.74 ± 23.06	46.93 ± 22.36	65.55 ± 25.15	64.42 ± 24.42	62.97 ± 17.39	60.47 ± 14.36	70.65 ± 21.26	**75.69 ± 15.67**
4	42.50 ± 16.95	50.59 ± 18.68	69.32 ± 15.01	64.38 ± 17.47	39.69 ± 16.14	53.17 ± 19.09	76.36 ± 18.63	**81.23 ± 10.06**
5	35.39 ± 12.09	49.88 ± 17.60	65.29 ± 12.76	58.54 ± 12.80	47.33 ± 13.84	46.67 ± 14.11	77.25 ± 11.32	**84.84 ± 9.85**
6	42.42 ± 13.04	47.86 ± 14.58	57.13 ± 14.27	58.69 ± 13.00	48.57 ± 10.94	44.48 ± 12.31	74.45 ± 10.67	**83.91 ± 5.41**
7	45.57 ± 12.83	58.09 ± 10.80	69.33 ± 8.92	67.78 ± 8.59	57.54 ± 9.90	54.42 ± 8.28	77.37 ± 7.03	**84.89 ± 3.08**
8	40.74 ± 7.62	51.09 ± 12.09	60.33 ± 8.64	62.58 ± 12.48	50.23 ± 7.58	53.19 ± 7.23	75.24 ± 8.74	**80.13 ± 6.94**
9	45.08 ± 8.41	55.70 ± 11.05	60.91 ± 11.06	61.34 ± 8.36	53.78 ± 10.44	56.04 ± 9.36	75.96 ± 5.95	**82.28 ± 5.93**
10	45.58 ± 10.72	55.59 ± 6.92	64.37 ± 8.21	60.62 ± 5.21	56.00 ± 7.87	50.19 ± 7.41	75.14 ± 5.08	**84.56 ± 2.57**
Avg.	41.35	50.14	62.84	61.21	51.17	51.37	75.16	**81.86**

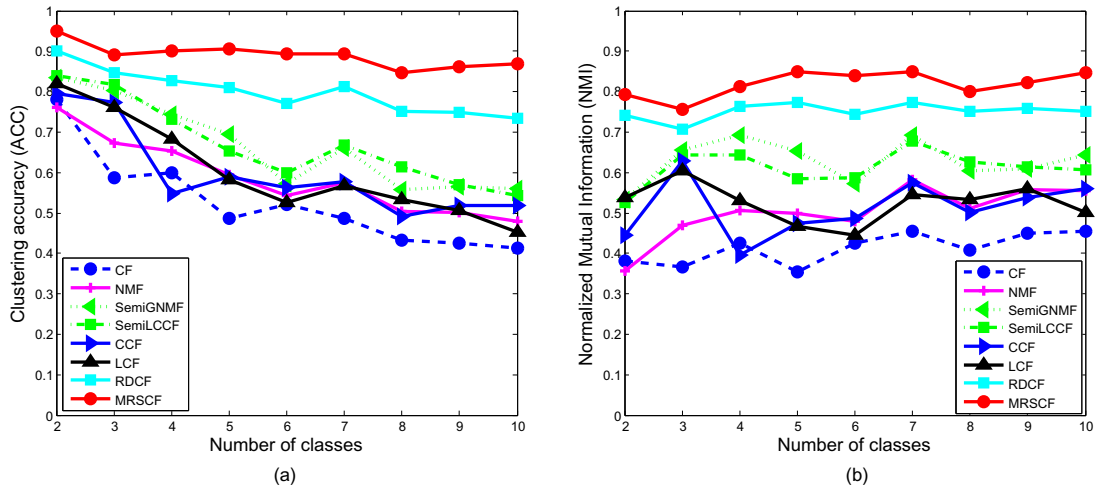

Fig. 3. Clustering performance on Orl database. (a) The clustering accuracy (ACC) versus number of clusters. (b) The normalized mutual information (NMI) versus number of clusters.

Fig. 6. 11 $(k+1)$ concepts (XW) obtained by CCF with 20% (first row) and 30% (second row) outliers in each row respectively.

E. Results with 10% Samples Labeled

To illustrate the utilization of the labeled samples, an experiment is conducted on MNIST dataset with only 10% samples labeled, and the proposed MRSCF is compared with the semi-supervised methods SemiGNMF, SemiLCCF and CCF. The results are illustrated under the situations with $k \in \{2, 4, 6, 8, 10\}$. The other parameters are set as the same as those in Section V-B. The results are reported in Table VIII. The proposed MRSCF method still gives the best clustering results in both ACC and NMI metrics.

TABLE VIII
CLUSTERING RESULTS ON THE MNIST DATABASE WITH 10% SAMPLES LABELED

k	Accuracy(%)			
	SemiGNMF	SemiLCCF	CCF	MRSCF
2	91.90 ± 12.16	93.40 ± 12.60	90.10 ± 14.01	**95.40 ± 5.92**
4	71.15 ± 12.20	70.15 ± 12.04	69.80 ± 8.19	**84.10 ± 7.47**
6	58.43 ± 11.03	55.40 ± 9.31	62.13 ± 5.93	**81.17 ± 2.93**
8	53.50 ± 5.37	48.60 ± 4.62	55.85 ± 6.41	**75.35 ± 4.13**
10	49.98 ± 2.14	46.20 ± 4.61	50.38 ± 2.42	**68.96 ± 1.95**

k	Normalized Mutual Information(%)			
	SemiGNMF	SemiLCCF	CCF	MRSCF
2	73.66 ± 31.21	78.63 ± 29.26	67.75 ± 31.04	**79.46 ± 20.51**
4	60.48 ± 11.00	54.17 ± 11.58	51.31 ± 8.74	**63.60 ± 11.07**
6	55.07 ± 7.53	46.92 ± 6.69	53.79 ± 4.99	**65.40 ± 4.25**
8	52.72 ± 2.84	42.36 ± 5.55	49.56 ± 4.55	**60.57 ± 4.34**
10	51.72 ± 1.37	43.43 ± 3.02	49.86 ± 1.35	**58.90 ± 2.38**

TABLE V
CLUSTERING RESULTS ON THE WARPAR DATABASE

k	Accuracy(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	64.23 ± 12.29	62.69 ± 17.03	72.31 ± 10.71	71.15 ± 11.05	66.54 ± 13.22	76.15 ± 12.85	72.69 ± 15.23	**84.62 ± 8.43**
3	49.23 ± 11.79	46.67 ± 10.81	52.05 ± 8.35	49.23 ± 7.68	52.82 ± 12.47	51.54 ± 10.97	58.21 ± 10.45	**74.87 ± 7.41**
4	42.88 ± 7.35	41.15 ± 5.91	45.00 ± 8.78	41.35 ± 6.62	47.69 ± 8.89	49.23 ± 8.95	47.50 ± 7.60	**71.35 ± 8.49**
5	34.62 ± 5.97	38.77 ± 4.45	40.46 ± 6.95	42.77 ± 7.40	43.08 ± 7.69	41.08 ± 7.63	41.69 ± 8.82	**65.38 ± 5.89**
6	33.85 ± 4.87	35.51 ± 6.49	38.85 ± 4.41	39.74 ± 3.72	41.79 ± 6.10	37.18 ± 5.19	39.74 ± 4.76	**67.44 ± 4.37**
7	32.53 ± 3.12	30.44 ± 3.07	35.82 ± 3.08	34.07 ± 3.68	37.91 ± 3.16	33.63 ± 3.15	37.25 ± 3.91	**60.33 ± 4.73**
8	30.29 ± 3.66	30.10 ± 3.80	32.69 ± 4.1	31.83 ± 4.09	35.77 ± 4.51	32.60 ± 2.17	35.10 ± 2.92	**60.38 ± 6.39**
9	28.63 ± 2.17	26.67 ± 2.19	29.06 ± 1.79	28.55 ± 1.92	34.79 ± 2.29	35.56 ± 2.74	32.05 ± 1.92	**57.86 ± 3.67**
10	25.38 ± 1.75	25.69 ± 3.41	26.38 ± 2.38	25.46 ± 1.21	33.46 ± 2.80	29.69 ± 2.13	31.15 ± 2.07	**53.08 ± 4.14**
Avg.	37.96	37.52	41.40	40.46	43.76	42.92	43.93	66.15

k	Normalized Mutual Information(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	14.27 ± 15.05	15.64 ± 25.97	21.58 ± 18.87	20.15 ± 19.47	16.34 ± 16.55	30.69 ± 29.63	26.04 ± 23.57	**45.91 ± 24.82**
3	17.61 ± 15.09	16.47 ± 17.60	22.18 ± 14.64	17.20 ± 12.88	23.39 ± 22.45	19.42 ± 18.22	29.54 ± 14.27	**44.33 ± 14.64**
4	17.09 ± 9.97	15.75 ± 9.42	23.64 ± 9.60	17.92 ± 10.31	23.99 ± 13.18	25.76 ± 10.04	27.02 ± 10.85	**47.93 ± 11.73**
5	15.52 ± 7.01	20.82 ± 6.42	27.47 ± 9.21	26.38 ± 9.36	25.87 ± 9.93	21.72 ± 10.79	26.58 ± 10.17	**44.97 ± 7.37**
6	18.91 ± 6.20	23.91 ± 7.36	27.45 ± 5.85	28.27 ± 6.13	29.03 ± 7.13	23.58 ± 5.66	29.05 ± 3.91	**50.86 ± 6.27**
7	23.33 ± 5.02	23.18 ± 5.23	29.21 ± 4.70	27.62 ± 4.94	29.40 ± 4.76	24.12 ± 3.77	31.79 ± 4.80	**45.52 ± 5.48**
8	23.12 ± 4.07	26.42 ± 4.46	27.43 ± 5.55	26.97 ± 4.47	29.92 ± 5.34	25.49 ± 3.00	30.73 ± 4.24	**48.65 ± 6.13**
9	25.02 ± 4.11	24.94 ± 3.46	28.04 ± 1.66	26.98 ± 2.62	31.31 ± 3.33	31.07 ± 2.81	31.84 ± 2.55	**47.89 ± 4.08**
10	23.38 ± 2.27	27.20 ± 3.56	25.45 ± 2.44	25.18 ± 2.24	32.43 ± 3.19	26.62 ± 2.45	30.88 ± 1.98	**45.57 ± 3.56**
Avg.	19.81	21.59	25.83	24.07	26.85	25.39	29.27	46.85

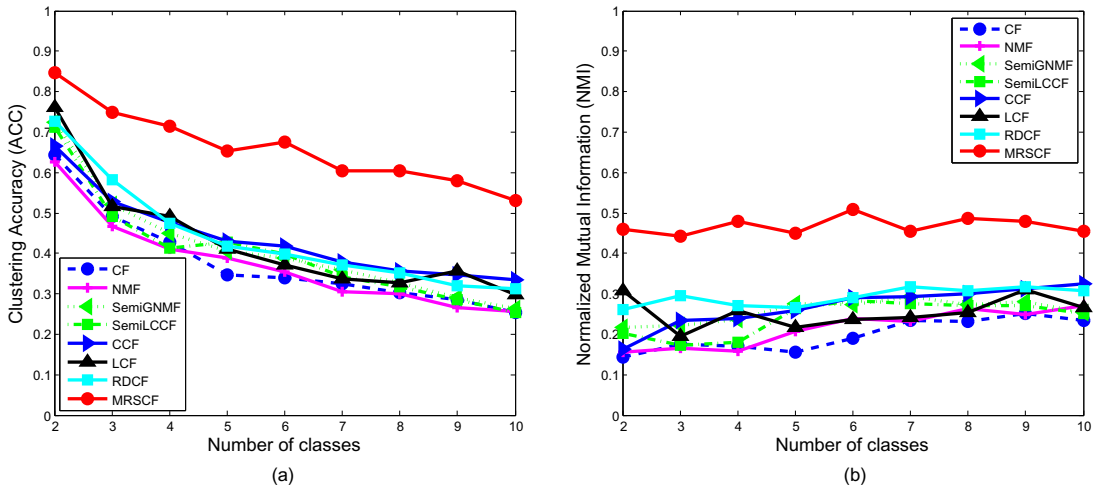

Fig. 4. Clustering performance on WarpAR database. (a) The clustering accuracy (ACC) versus number of clusters. (b) The normalized mutual information (NMI) versus number of clusters.

F. The Concepts of Different Methods

To better visualize the effectiveness of the proposed method, 10 concepts obtained from Yale dataset of different concept factorization based methods are shown in Figure 8. From the figure, it is noted that the concepts of F-norm based methods CF, LCCF, CCF, LCF and RDCF are seriously affected by outliers, while the proposed MRSCF method can obtain the clean concepts. The reason could be that MRSCF method utilizes MCC to construct the model, thus it can eliminate the negative effects of outliers. Furthermore, the concepts obtained by MRSCF are more discriminative than other compared methods. That is because MRSCF introduces nonegative adaptive embedding term and label information. Therefore, the proposed MRSCF method can obtain better clustering results than other compared methods.

G. Sensitivity of Parameters

To further study the proposed MRSCF method, the sensitivity of model parameters on the clustering performances are analyzed. We present the clustering results under different parameters on Orl dataset. Since the MRSCF method involves three parameters, we fix one and consider the effects of other two by grid search. Fig. 9 plot the ACC and NMI values given by MRSCF for different combination of two parameters. It is note that the proposed method is sensitive with parameter α but not sensitive with γ and β. Therefore, in the application, we only need to focus on the tuning of parameter α.

H. Convergence Study

To intuitively show the convergence property of the proposed MRSCF algorithm, the convergence curves of MRSCF on four datasets are illustrated in Fig. 10. Each convergence

TABLE VI
CLUSTERING RESULTS ON THE MNIST DATABASE

k	Accuracy(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	86.60 ± 15.63	84.90 ± 17.05	94.20 ± 7.12	94.40 ± 7.76	92.30 ± 8.66	90.00 ± 12.92	84.00 ± 22.27	**97.20 ± 1.78**
3	67.40 ± 13.08	64.20 ± 9.82	78.60 ± 16.68	84.20 ± 10.94	81.80 ± 10.90	77.76 ± 12.23	63.40 ± 4.88	**88.80 ± 2.06**
4	67.40 ± 10.37	66.65 ± 10.90	70.15 ± 9.00	71.95 ± 8.63	72.10 ± 8.90	64.75 ± 5.05	65.15 ± 9.77	**88.50 ± 2.70**
5	55.28 ± 6.82	54.96 ± 9.17	71.16 ± 13.25	64.36 ± 10.98	67.20 ± 13.11	59.52 ± 7.06	61.44 ± 8.94	**84.56 ± 3.71**
6	57.43 ± 6.63	57.97 ± 5.94	60.50 ± 6.90	57.50 ± 4.61	63.00 ± 5.63	52.70 ± 4.72	61.67 ± 6.97	**84.30 ± 2.79**
7	54.89 ± 6.53	52.91 ± 4.85	57.51 ± 4.90	51.37 ± 2.65	63.49 ± 5.62	53.60 ± 4.52	57.51 ± 5.79	**81.86 ± 3.07**
8	54.45 ± 5.50	53.78 ± 4.97	55.25 ± 5.98	49.90 ± 3.60	62.53 ± 4.42	58.98 ± 3.87	56.17 ± 4.63	**79.45 ± 1.75**
9	52.27 ± 2.96	50.76 ± 2.81	51.33 ± 3.43	47.67 ± 4.63	60.44 ± 3.78	50.98 ± 3.16	55.16 ± 3.66	**76.27 ± 1.55**
10	47.76 ± 1.39	47.90 ± 3.09	50.10 ± 3.08	47.02 ± 3.69	56.80 ± 3.77	47.26 ± 1.94	53.34 ± 2.28	**75.62 ± 1.69**
Avg.	60.39	59.34	65.42	63.15	68.85	61.73	61.98	**84.06**

k	Normalized Mutual Information(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
2	58.70 ± 34.05	55.97 ± 35.73	76.67 ± 21.36	77.09 ± 23.79	69.73 ± 28.13	65.04 ± 26.93	63.80 ± 41.38	**83.99 ± 9.38**
3	38.41 ± 17.37	36.48 ± 10.79	66.40 ± 16.75	66.95 ± 16.22	57.47 ± 16.25	53.98 ± 16.43	43.02 ± 9.04	**65.50 ± 4.96**
4	51.26 ± 11.52	51.48 ± 8.99	63.38 ± 8.76	57.75 ± 10.29	60.12 ± 9.68	47.61 ± 7.02	55.16 ± 10.17	**70.13 ± 5.68**
5	42.65 ± 9.26	41.61 ± 11.19	62.79 ± 12.65	53.14 ± 10.64	56.01 ± 12.90	43.71 ± 7.57	49.53 ± 7.39	**65.93 ± 6.85**
6	46.33 ± 8.27	47.83 ± 6.61	56.90 ± 5.23	48.79 ± 6.28	54.33 ± 4.92	42.54 ± 3.24	55.56 ± 4.63	**67.52 ± 4.69**
7	47.50 ± 5.45	46.71 ± 4.62	55.23 ± 4.05	45.52 ± 4.45	55.86 ± 6.38	45.68 ± 5.56	53.47 ± 4.66	**65.44 ± 5.01**
8	49.32 ± 4.17	49.05 ± 3.67	53.77 ± 4.57	44.32 ± 5.55	58.46 ± 3.31	44.55 ± 3.82	53.84 ± 4.37	**63.87 ± 2.22**
9	48.35 ± 3.24	47.06 ± 2.92	53.36 ± 4.94	43.24 ± 4.91	54.93 ± 2.93	47.50 ± 2.72	52.62 ± 3.40	**60.36 ± 1.89**
10	45.46 ± 1.96	46.12 ± 1.54	50.90 ± 1.79	44.40 ± 3.48	55.04 ± 2.54	44.66 ± 2.08	51.99 ± 2.83	**60.09 ± 2.42**
Avg.	47.55	46.92	59.93	53.43	57.99	48.36	53.22	**66.98**

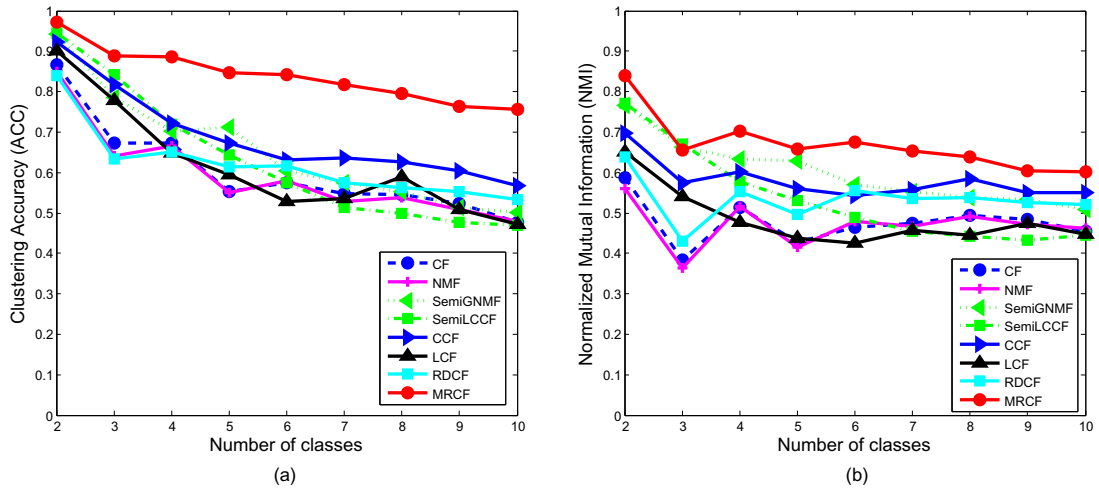

Fig. 5. Clustering performance on MNIST database. (a) The clustering accuracy (ACC) versus number of clusters. (b) The normalized mutual information (NMI) versus number of clusters.

curve shows a variation of the objective value in (11). It is clear that the objective value increases step-by-step and converges rapidly.

VI. CONCLUSION AND FUTURE WORK

Based on MCC, we have proposed a novel MCC based robust semi-supervised CF model. Because of MCC which is a local measure, this model is particularly suitable for situations where data contain large outliers. We have also given an MCC based framework to simultaneously introduce local structure information and label information. Different from the other existing CF methods, which utilize the multiplicative strategy to solve the model, an accelerated block coordinate update iterative algorithm is derived to solve the proposed model. The convergence property of the proposed algorithm has been shown. Experimental studies on outlier-contaminated datasets have shown the superiority of the proposed method over several state-of-the-art methods. Our better experimental results attribute to two main factors. The first one is the MCC metric that is used in our model. It is a local measure and can truncate large outliers. Therefore, it is more robust than global metrics such as the Frobenius norm and $\ell_{2,1}$-norm. The second factor is the adaptive and softer way, by which our method MRSCF incorporates the label information.

We find that at least three questions are worth for further investigation in the future:

1) Compared with $\ell_{2,1}$-norm, which is a global distance measure, MCC is more robust with large outliers, because it is a local measure. In this way, MCC not only can be used for CF model, but also can be explored to other machine learning methods to overcome the large outliers.

TABLE VII
CLUSTERING RESULTS ON THE YALE DATABASE WITH DIFFERENT OUTLIER LEVEL

OL	Accuracy(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
0%	37.36 ± 5.34	39.09 ± 3.18	42.09 ± 3.32	43.19 ± 4.67	47.82 ± 3.60	40.82 ± 3.99	46.36 ± 3.50	**70.36 ± 3.69**
10%	36.55 ± 3.56	39.09 ± 3.60	39.64 ± 4.60	42.00 ± 3.37	43.73 ± 4.73	40.18 ± 5.04	38.45 ± 3.84	**70.64 ± 3.23**
20%	36.18 ± 3.79	38.45 ± 4.29	39.91 ± 4.31	40.09 ± 4.03	42.73 ± 2.82	41.27 ± 5.64	38.00 ± 5.44	**71.00 ± 3.31**
30%	36.45 ± 3.29	37.36 ± 5.00	39.91 ± 2.97	41.00 ± 3.91	44.36 ± 4.36	39.18 ± 3.71	39.45 ± 4.76	**70.73 ± 3.63**
40%	37.82 ± 5.15	37.00 ± 5.84	39.00 ± 4.31	40.27 ± 4.47	42.73 ± 3.86	38.64 ± 3.46	40.91 ± 4.00	**70.91 ± 3.70**
50%	36.09 ± 5.68	37.18 ± 3.62	39.82 ± 3.22	40.82 ± 4.27	45.18 ± 5.09	37.09 ± 3.74	39.00 ± 5.51	**71.00 ± 3.67**
Avg.	36.74	38.03	40.06	41.23	44.43	39.53	40.36	**70.77**
OL	Normalized Mutual Information(%)							
	CF	NMF	SemiGNMF	SemiLCCF	CCF	LCF	RDCF	MRSCF
0%	36.48 ± 3.81	38.60 ± 3.43	39.83 ± 4.05	40.13 ± 4.39	46.67 ± 3.60	40.11 ± 5.25	45.67 ± 2.69	**62.97 ± 3.80**
10%	35.53 ± 4.95	37.85 ± 3.84	37.84 ± 4.53	39.43 ± 3.24	45.03 ± 3.98	39.28 ± 5.25	38.48 ± 4.12	**63.39 ± 3.19**
20%	33.84 ± 4.12	36.84 ± 4.59	37.99 ± 4.62	38.69 ± 3.74	42.62 ± 2.95	38.79 ± 5.56	38.77 ± 4.08	**63.82 ± 3.39**
30%	34.29 ± 3.56	36.53 ± 4.83	37.60 ± 3.29	38.90 ± 3.70	45.00 ± 2.73	38.00 ± 4.33	39.14 ± 4.23	**63.57 ± 3.52**
40%	35.77 ± 5.47	35.67 ± 5.94	36.65 ± 4.90	37.51 ± 3.76	42.46 ± 2.69	37.28 ± 3.43	40.80 ± 3.64	**63.82 ± 3.57**
50%	34.79 ± 4.90	35.87 ± 4.27	37.94 ± 3.62	38.79 ± 4.31	45.28 ± 3.28	35.63 ± 3.92	38.73 ± 4.03	**63.80 ± 3.66**
Avg.	35.12	36.89	37.98	38.41	44.51	38.18	40.26	**63.56**

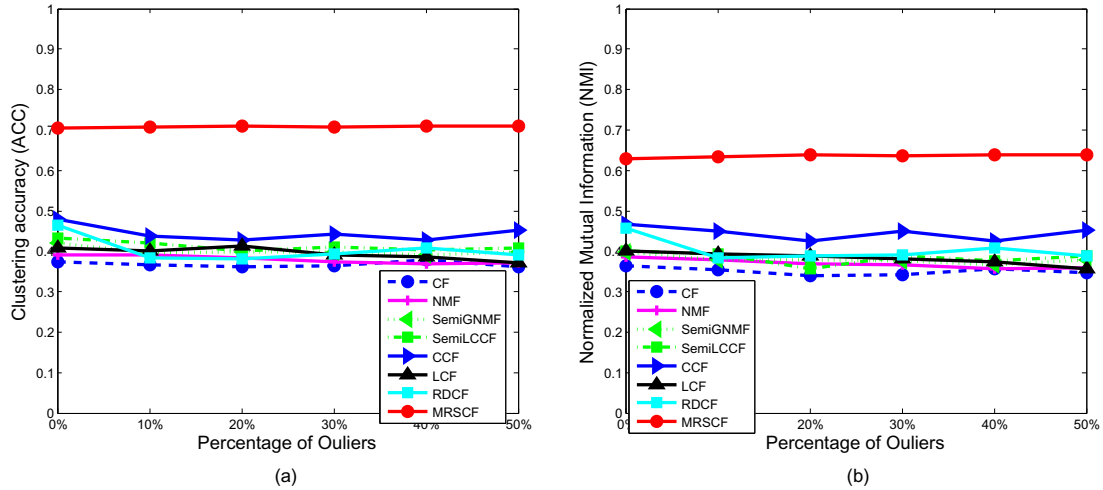

Fig. 7. Clustering performance on MNIST database. (a) The clustering accuracy (ACC) versus percentage of outliers. (b) The normalized mutual information (NMI) versus percentage of outliers.

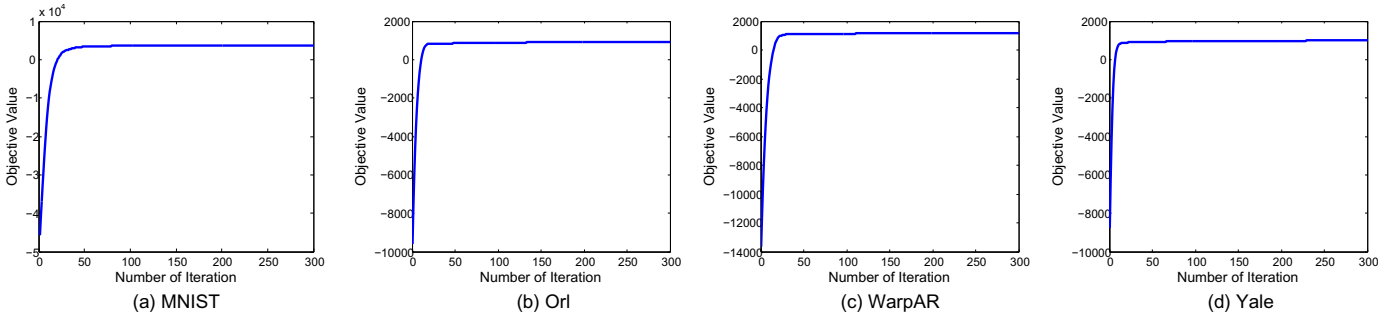

Fig. 10. The convergence curve of MRSCF method over four datasets.

2) It is interesting to consider the problem that the data not only have outliers but also have noises, and how to construct a model to eliminate both negative effects.
3) The robust framework for introducing label information proposed in this paper is suitable for concept factorization method. Whether it is also suitable for other matrix factorization based feature learning methods, is another interesting point.

APPENDIX A
THE EQUIVALENCE BETWEEN (17D) AND (20)

Let $B = \nabla_W f(\hat{W}^t, H^t, S^t, \mathbf{p}^{t+1}, \mathbf{q}^{t+1}, \mathbf{z}^{t+1})$ and $L = L_W^t$, the objective of problem (17d) can be written as

$$\begin{aligned} &Tr[(W-\hat{W}^t)^\top B] - \frac{L}{2}Tr[(W-\hat{W}^t)^\top (W-\hat{W}^t)] - \beta\|W\|_1 \\ =&Tr\left[W^\top B - (\hat{W}^t)^\top B\right] - \frac{L}{2}Tr\left[W^\top W - 2W^\top \hat{W}^t + (\hat{W}^t)^\top \hat{W}^t\right] \\ &- \beta\|W\|_1. \end{aligned} \tag{33}$$

Fig. 8. 10 concepts (XW) obtained by CF, SemiLCCF, CCF, LCF, RDCF and MRSCF in each row respectively when data is contaminated by dummy images.

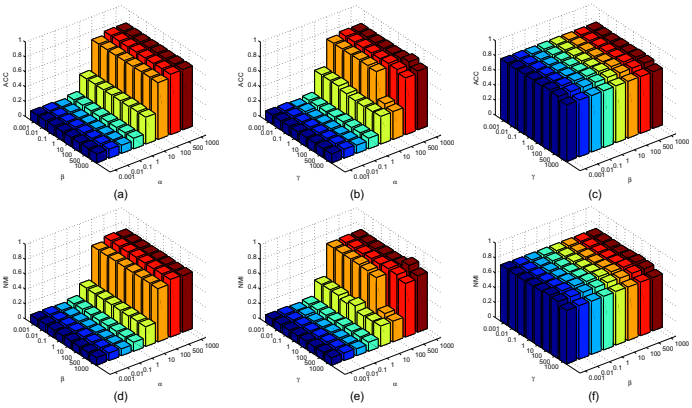

Fig. 9. The clustering performances with the variations of parameters.

Eliminating the terms in (33) that are independent of W, problem (17d) can be reformulated as follows:

$$\begin{aligned}
&\underset{W\geq 0}{\operatorname{argmax}}\ Tr\left(W^\top B-\frac{L}{2}W^\top W+LW^\top\hat{W}^t\right)-\beta\|W\|_1\\
\Leftrightarrow&\underset{W\geq 0}{\operatorname{argmax}}\ -\frac{L}{2}\left[Tr\left(W^\top W-\frac{2}{L}W^\top B+2W^\top\hat{W}^t\right)+\frac{2\beta}{L}\|W\|_1\right]\\
\Leftrightarrow&\underset{W\geq 0}{\operatorname{argmin}}\ Tr\left(W^\top W-\frac{2}{L}W^\top B+2W^\top\hat{W}^t\right)+\frac{2\beta}{L}\|W\|_1\\
\Leftrightarrow&\underset{W\geq 0}{\operatorname{argmin}}\ Tr\left(W^\top W-2W^\top(\hat{W}^t+\frac{1}{L}B)\right)+\frac{2\beta}{L}\|W\|_1.
\end{aligned}$$

Finally, the problem (17d) is equivalent to the following form:

$$\underset{W\geq 0}{\operatorname{argmin}}\ \frac{1}{2}\|W-\left(\hat{W}^t+\frac{1}{L}B\right)\|_F^2+\frac{\beta}{L}\|W\|_1.$$

APPENDIX B
LIPSCHITZ CONSTANT DERIVATION

A. The derivation of L_W^t

From the definition of f in (15d), $\nabla_W f$ can be directly computed as

$$\nabla_W f=X^\top XWHPH^\top-X^\top XPH^\top. \tag{34}$$

For any $\tilde{W}$ and $\hat{W}$, we have

$$\begin{aligned}
&\|\nabla_W f(\tilde{W})-\nabla_W f(\hat{W})\|_F\\
=&\|X^\top X\hat{W}HPH^\top-X^\top X\tilde{W}HPH^\top\|_F\\
=&\|X^\top X(\hat{W}-\tilde{W})HPH^\top\|_F\\
\leq&\|X^\top X\|_2\|\hat{W}-\tilde{W}\|_F\|HPH^\top\|_2\\
=&(\|X^\top X\|_2\|HPH^\top\|_2)\|\hat{W}-\tilde{W}\|_F.
\end{aligned}$$

Therefore, the Lipchitz constant

$$L_W^t=\|X^\top X\|_2\|H^tP^{t+1}(H^t)^\top\|_2. \tag{35}$$

B. The derivation of L_H^t

The $\nabla_H f$ can be computed as

$$\begin{aligned}
\nabla_H f=&W^\top X^\top XWHP-W^\top X^\top XP\\
&+\alpha H(I-S)Q(I-S)^\top-\alpha HM+YM.
\end{aligned} \tag{36}$$

For any $\tilde{H}$ and $\hat{H}$, we can obtain that

$$\begin{aligned}
&\|\nabla_H f(\tilde{H})-\nabla_H f(\hat{H})\|_F\\
=&\|W^\top X^\top XW\hat{H}P+\alpha\hat{H}(I-S)Q(I-S)^\top-\alpha\hat{H}M\\
&-W^\top X^\top XW\tilde{H}P-\alpha\tilde{H}(I-S)Q(I-S)^\top+\alpha\tilde{H}M\|_F\\
\leq&\|W^\top X^\top XW(\hat{H}-\tilde{H})P\|_F+\alpha\|(\hat{H}-\tilde{H})(I-S)Q(I-S)^\top\|_F\\
&+\alpha\|(\hat{H}-\tilde{H})M\|_F\\
\leq&\left[\|W^\top X^\top XW\|_2\|P\|_2+\alpha(\|(I-S)Q(I-S)^\top\|_2+\|M\|_2)\right]*\\
&\|\hat{H}-\tilde{H}\|_F.
\end{aligned}$$

Therefore, the Lipchitz constant

$$\begin{aligned}
L_H^t=&\|(W^{t+1})^\top X^\top XW^{t+1}\|_2\|P^{t+1}\|_2\\
&+\alpha\|(I-S^t)Q^{t+1}(I-S^t)^\top\|_2+\|M\|_2.
\end{aligned} \tag{37}$$

C. The derivation of L_S^t

The $\nabla_S f$ can be computed as

$$\nabla_S f=\alpha(H^\top HSQ+X^\top XSZ-H^\top HQ-X^\top XZ). \tag{38}$$

For any $\tilde{S}$ and $\hat{S}$, we have

$$\begin{aligned}
&\|\nabla_S f(\tilde{S})-\nabla_H f(\hat{S})\|_F\\
=&\alpha\|H^\top H\hat{S}Q+X^\top X\hat{S}Z-H^\top H\tilde{S}Q-X^\top X\tilde{S}Z\|_F\\
\leq&\alpha\|H^\top H(\hat{S}-\tilde{S})Q\|_F+\alpha\|X^\top X(\hat{S}-\tilde{S})Z\|_F\\
\leq&\alpha\|H^\top H\|_2\|\hat{S}-\tilde{S}\|_F\|Q\|_2+\alpha\|X^\top X\|_2\|\hat{S}-\tilde{S}\|_F\|Z\|_2\\
=&\alpha(\|H^\top H\|_2\|Q\|_2+\|X^\top X\|_2\|Z\|_2)\|\hat{S}-\tilde{S}\|_F.
\end{aligned}$$

Therefore, the Lipchitz constant is

$$L_S^t=\|(H^{t+1})^\top H^{t+1}\|_2\|Q^{t+1}\|_2+\|X^\top X\|_2\|Z^{t+1}\|_2. \tag{39}$$

REFERENCES

[1] H. Hotelling, "Analysis of a complex of statistical variables into principal components." *Journal of educational psychology*, vol. 24, no. 6, p. 417, 1933.

[2] S. Haykin, *Independent Component Analysis*. Kluwer Academic Publishers,, 1998.

[3] R. M. Gray, "Vector quantization," *Readings in Speech Recognition*, vol. 1, no. 2, pp. 75–100, 1990.

[4] D. D. Lee, "Algorithms for nonnegative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, no. 6, pp. 556–562, 2000.

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

[6] W. Xu and Y. Gong, "Document clustering by concept factorization," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004, pp. 202–209.

[7] "K-means clustering and principal component analysis," in *Intl Conf. Machine Learning*, 2004.

[8] N. K. L. And and D. L. Sheinberg, "Visual object recognition," *Annual Review of Neuroscience*, vol. 19, no. 2, p. 577, 1996.

[9] O. Chapelle, B. Schlkopf, and A. Zien, "Semi-supervised learning," *Intelligent Systems Reference Library*, vol. 49, no. 2, pp. 215–239, 2006.

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

[11] W. Ou, S. Yu, L. Gai, L. Jian, K. Zhang, and X. Gang, "Multi-view nonnegative matrix factorization by patch alignment framework with view consistency," *Neurocomputing*, vol. 204, no. C, pp. 116–124, 2016.

[12] W. Ou, L. Fei, T. Yi, S. Yu, and P. Wang, "Co-regularized multiview nonnegative matrix factorization with correlation constraint for representation learning," *Multimedia Tools and Applications*, vol. 77, no. 10, pp. 12 955–12 978, 2018.

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

[14] D. Cai, X. He, X. Wu, and J. Han, "Non-negative matrix factorization on manifold," in *Eighth IEEE International Conference on Data Mining*, 2008, pp. 63–72.

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

[16] J. P. Brunet, P. Tamayo, T. R. Golub, J. P. Mesirov, and E. S. Lander, "Metagenes and molecular pattern discovery using matrix factorization," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 12, pp. 4164–9, 2004.

[17] D. Cai, X. He, and J. Han, "Locally consistent concept factorization for document clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 6, pp. 902–913, 2011.

[18] H. Liu, Z. Yang, J. Yang, Z. Wu, and X. Li, "Local coordinate concept factorization for image representation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1071–1082, 2014.

[19] Q. Gu, C. Ding, and J. Han, "On trivial solution and scale transfer problems in graph regularized nmf," in *International Joint Conference on Artificial Intelligence*, 2011, pp. 1288–1293.

[20] Y. Guo, G. Ding, J. Zhou, and Q. Liu, "Robust and discriminative concept factorization for image representation," 2015, pp. 115–122.

[21] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[22] H. Liu, G. Yang, Z. Wu, and D. Cai, "Constrained concept factorization for image representation." *IEEE Transactions on Cybernetics*, vol. 44, no. 7, p. 1214, 2014.

[23] J. C. Principe, *Information theoretic learning: Renyi's entropy and kernel perspectives*. Springer Science & Business Media, 2010.

[24] M. Oldstone, R. Ahmed, M. J. Buchmeier, P. Blount, and A. Tishon, "On measures of entropy and information," *Proc.fourth Berkeley Symp. on Math. statist. & Prob.univ.of Calif*, vol. 1, no. 5073, pp. 547–561, 1961.

[25] W. Liu, P. P. Pokharel, and J. C. Principe, "Correntropy: Properties and applications in non-gaussian signal processing," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5286–5298, 2007.

[26] J. J. Wang, X. Wang, and X. Gao, "Non-negative matrix factorization by maximizing correntropy for cancer clustering." *Bmc Bioinformatics*, vol. 14, no. 1, p. 107, 2013.

[27] R. He, T. Tan, L. Wang, and W. S. Zheng, "l 2, 1 regularized correntropy for robust feature selection," in *Computer Vision and Pattern Recognition*, 2012, pp. 2504–2511.

[28] L. Du, X. Li, and Y. D. Shen, "Robust nonnegative matrix factorization via half-quadratic minimization," in *IEEE International Conference on Data Mining*, 2013.

[29] B. Chen and J. C. Príncipe, "Maximum correntropy estimation is a smoothed map estimation," *IEEE Signal Processing Letters*, vol. 19, no. 8, pp. 491–494, 2012.

[30] B. Chen, J. Wang, H. Zhao, N. Zheng, and J. C. Príncipe, "Convergence of a fixed-point algorithm under maximum correntropy criterion," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1723–1727, 2015.

[31] B. Chen, L. Xing, H. Zhao, N. Zheng, and J. C. Príncipe, "Generalized correntropy for robust adaptive filtering," *Transactions on Signal Processing*, vol. 64, no. 13, pp. 3376–3387, 2016.

[32] B. Chen, L. Xing, J. Liang, N. Zheng, and J. C. Principe, "Steady-state mean-square error analysis for adaptive filtering under the maximum correntropy criterion," *IEEE signal processing letters*, vol. 21, no. 7, pp. 880–884, 2014.

[33] R. He, W.-S. Zheng, and B.-G. Hu, "Maximum correntropy criterion for robust face recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1561–1576, 2011.

[34] T. Ensari, J. Chorowski, and J. M. Zurada, "Correntropy-based document clustering via nonnegative matrix factorization," in *International Conference on Artificial Neural Networks and Machine Learning*, 2012.

[35] S. R. Sain, *The Nature of Statistical Learning Theory*. Springer, 1995.

[36] S. Yan, D. Xu, B. Zhang, H. J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[37] W. Meng, L. Hao, T. Dacheng, L. Ke, and W. Xindong, "Multimodal graph-based reranking for web image search," *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4649–4661, 2012.

[38] M. Wang, X. Liu, and X. Wu, "Visual classification by -hypergraph modeling," *Knowledge and Data Engineering IEEE Transactions on*, vol. 27, no. 9, pp. 2564–2574, 2015.

[39] M. Wang, W. Fu, S. Hao, D. Tao, and X. Wu, "Scalable semi-supervised learning by efficient anchor graph regularization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1864–1877, 2016.

[40] W. Meng, W. Fu, S. Hao, H. Liu, and X. Wu, "Learning on big graph: Label inference and regularization with anchor hierarchy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 5, pp. 1101–1114, 2017.

[41] D. G. Lowe, "Similarity metric learning for a variable-kernel classifier," *Neural Computation*, vol. 7, no. 1, pp. 72–85, 1995.

[42] H. Cheng, Z. Liu, L. Yang, and X. Chen, "Sparse representation and learning in visual recognition: Theory and applications," *Signal Processing*, vol. 93, no. 6, pp. 1408–1425, 2013.

[43] H. Cheng, Z. Liu, and J. Yang, "Sparsity induced similarity measure for label propagation," in *IEEE International Conference on Computer Vision*, 2009, pp. 317–324.

[44] H. Liu and F. Sun, "Visual tracking using sparsity induced similarity," in *International Conference on Pattern Recognition*, 2010, pp. 1702–1705.

[45] H. Wang, H. Huang, and C. Ding, "Image categorization using directed graphs," in *European Conference on Computer Vision Conference on Computer Vision*, 2010, pp. 762–775.

[46] Y. Gao, A. Choudhary, and G. Hua, "A nonnegative sparsity induced similarity measure with application to cluster analysis of spam images," in *IEEE International Conference on Acoustics Speech and Signal Processing*, 2010, pp. 5594–5597.

[47] F. Dornaika and Y. E. Traboulsi, "Learning flexible graph-based semi-supervised embedding," *IEEE Trans Cybern*, vol. 46, no. 1, pp. 206–218, 2015.

[48] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1758–1789, 2013.

[49] H.-J. M. Shi, S. Tu, Y. Xu, and W. Yin, "A primer on coordinate descent algorithms," *arXiv preprint arXiv:1610.00040*, 2016.

[50] R. T. Rockafellar, *Convex analysis*. Princeton university press, 2015.

[51] Z. Peng, T. Wu, Y. Xu, M. Yan, and W. Yin, "Coordinate friendly structures, algorithms and applications." *Annals of Mathematical Sciences and Applications*, vol. 1, no. 1, pp. 57–119, 2016.

[52] Y. Chen and X. Ye, "Projection onto a simplex," *arXiv preprint arXiv:1101.6081*, 2011.

[53] Y. Xu, "Alternating proximal gradient method for sparse nonnegative tucker decomposition," *Mathematical Programming Computation*, vol. 7, no. 1, pp. 39–70, 2015.

[54] Y. Xu and W. Yin, "A globally convergent algorithm for nonconvex optimization based on block coordinate update," *Journal of Scientific Computing*, vol. 72, no. 2, pp. 700–734, 2017.

[55] S. Łojasiewicz, "Sur la géométrie semi-et sous-analytique," *Ann. Inst. Fourier*, vol. 43, no. 5, pp. 1575–1595, 1993.

[56] K. Kurdyka, "On gradients of functions definable in o-minimal structures," in *Annales de l'institut Fourier*, vol. 48, no. 3. Chartres: L'Institut, 1950-, 1998, pp. 769–784.

[57] L. Lovász and M. D. Plummer, *Matching theory*. American Mathematical Soc., 2009, vol. 367.

[58] R. M. Gray, *Entropy and information theory*. Springer Science & Business Media, 2011.