Discriminative Embedded Clustering: A Framework for Grouping High-Dimensional Data

Chenping Hou, *Member, IEEE*, Feiping Nie, Dongyun Yi, and Dacheng Tao, *Senior Member, IEEE*

***Abstract*— In many real applications of machine learning and data mining, we are often confronted with high-dimensional data. How to cluster high-dimensional data is still a challenging problem due to the curse of dimensionality. In this paper, we try to address this problem using joint dimensionality reduction and clustering. Different from traditional approaches that conduct dimensionality reduction and clustering in sequence, we propose a novel framework referred to as discriminative embedded clustering which alternates them iteratively. Within this framework, we are able not only to view several traditional approaches and reveal their intrinsic relationships, but also to be stimulated to develop a new method. We also propose an effective approach for solving the formulated nonconvex optimization problem. Comprehensive analyses, including convergence behavior, parameter determination, and computational complexity, together with the relationship to other related approaches, are also presented. Plenty of experimental results on benchmark data sets illustrate that the proposed method outperforms related state-of-the-art clustering approaches and existing joint dimensionality reduction and clustering methods.**

***Index Terms*— Clustering, dimensionality reduction, discriminative embedded clustering (DEC), high-dimensional data, subspace learning.**

I. Introduction

CLUSTERING is a fundamental and important topic in both machine learning and data mining fields. It has been widely used in many areas, ranging from science to engineering. The basic goal of clustering is to assign data of similar patterns into the same cluster and reveal the meaningful structure of the data. It is one of the most important tools for explorative data mining. In the literature, many clustering approaches have been proposed, such as K-means, spectral clustering (SC) [1], [2] and maximum margin clustering (MMC) [3]. K-means assigns points to its nearest cluster centroid. SC makes use of the spectrum of the similarity matrix to capture the low dimensional and nonlinear manifold structure of data points. MMC is margin-based and has been proposed for finding a decision boundary in low-density regions to separate data points into two different clusters. Other related deep researches have also been provided for data clustering, such as [4].

In many real applications, we are often confronted with very high-dimensional data. Taking web text data as an example, if we use a vector space model to describe each document, the dimensionality is often more than 5000 since the word vocabulary is frequently large. In addition, the vectorization of an image of 256×256 resolution is a 65 536-D vector. Due to the curse of dimensionality, manipulating high-dimensional data, such as clustering or classification [5], [6], is still a challenging problem. Many dimensions are not helpful or may even worsen the performance of the subsequent clustering algorithms. To tackle this problem, one direct way is to first employ dimensionality reduction approaches and then cluster embedding data in low-dimensional space. In the past decades, a number of dimensionality reduction approaches have been deeply investigated. Linear approaches, such as principal component analysis (PCA) and its extension in [7], linear discriminant analysis (LDA) [8], orthogonal centroid method (OCM) [9], maximum margin criterion (MMC) [10], and orthogonal least squares discriminant analysis (OLSDA) [11], have been widely studied and used in many applications. They all assume that the high-dimensional data almost lie on a low-dimensional linear subspace and use different criteria to learn different subspaces. By contrast with linear methods, nonlinear approaches assume that high-dimensional data lie on a low-dimensional nonlinear manifold. Representative methods include Isometric Mapping [12], locally linear embedding [13], Laplacian eigenmap [14], local spline embedding [15], and their variants [16]–[19]. They try to preserve some properties during the processing of dimensionality reduction. Among these approaches, PCA for dimensionality reduction and K-means for clustering are two of the mostly used strategies. We referred to them in the following sections as PCAKM.

Although we can initially reduce the dimensionality by any approach and then use clustering approaches to group high-dimensional data, the performance can also be improved since these two techniques are conducted in sequence. The procedure of dimensionality reduction is not related to the subsequent clustering techniques. Intuitively, if we consider the requirement of clustering during the process of dimensionality reduction and vice versus, the performance of clustering will be improved.

In the past years, several researchers have been dedicated to performing dimensionality reduction and clustering simultaneously. Briefly, there are two different ways: joint feature selection (or feature ranking) with clustering and joint feature learning (or feature extraction) with clustering. The first type

Manuscript received September 15, 2013; accepted July 3, 2014. This work was supported by the Australian Research Council under Project FT-130101457, Project DP-120103730, and Project LP-140100569.

C. Hou and D. Yi are with the College of Science, National University of Defense Technology, Changsha 410073, China (e-mail: hcpnudt@hotmail.com; dongyun.yi@gmail.com).

F. Nie is with the Department of Computer Science and Engineering, University of Texas, Arlington, TX 76019 USA (e-mail: feipingnie@gmail.com).

D. Tao is with the Centre for Quantum Computation and Intelligent Systems, Faculty of Engineering and Information Technology, University of Technology, Sydney, Ultimo, NSW 2007, Australia (e-mail: dacheng.tao@uts.edu.au).

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

Digital Object Identifier 10.1109/TNNLS.2014.2337335

of method, commonly called subspace clustering, maintains the original features during the processing of feature selection. In the second category, several features are combined to formulate new representations for clustering. A native approach for joint feature selection with clustering might be to search all possible subspace and use clustering validation to determinate the subspace with the best clustering performance. This is infeasible because of the subset generation problem is an non-deterministic polynomial-problem [20]. Commonly, based on the procedure for searching the optimal subspace, we can divide these methods into two categories: bottom-up subspace searching methods and top-down subspace searching methods. Typical bottom-up methods include CLIQUE [21], ENCLUS [22], MAFIA [23], CLTree [24], and DOC [25]. Typical top-down methods include PROCLUS [26], ORCLUS [27], δ-Clusters [28], and COSA [29]. These methods achieve noteworthy performance in many applications. Details are given in [20].

Recently, many researchers have been dedicated to transforming original features and joining feature learning with clustering. Ding *et al.* [30] and Ding and Li [31] have combined LDA with K-means by learning a subspace and clustering alternately. They use the K-means algorithm to generate class labels and use LDA to learn the subspace alternately. De la Torre and Kanade [32] have shown the benefits of clustering in a low-dimensional discriminative space rather than in the principal components and proposed a new clustering algorithm called discriminative cluster analysis. Ye *et al.* [33] have also investigated the problem of alternating clustering and distance metric learning iteratively by projecting data into a low-dimensional manifold, where the separability of the data is maximized. These methods all concern simultaneous subspace selection by LDA and clustering. Theoretical studies have also been provided in [34]. Niu *et al.* [35] have presented a method which reduces the dimensionality of the original data for SC. It automatically learns the relevant dimensions and SC simultaneously. Gu and Zhou [36] have presented a subspace MMC method that integrates dimensionality reduction with MMC in a joint framework. Domeniconi *et al.* [37] have introduced an algorithm that discovers clusters in subspaces spanned by different combinations of dimensions via local weightings of features. We have also used trace ratio to learn a subspace for feature selection [38] and face image clustering [39]. Other dimensionality reduction-based clustering methods, such as [40], [41], have also been presented for the simultaneous combination of dimensionality reduction and clustering.

In this paper, we introduce a novel framework, referred to as discriminative embedded clustering (DEC), for high-dimensional data clustering. It combines subspace learning and clustering in a unified framework. There are two main objective functions, the first being dimensionality reduction and the second being clustering. Several previous methods can be viewed as special cases when we set different values of a parameter of DEC. For example, PCAKM can be viewed as a special case of DEC when $\lambda \to 0$. Since the formulated problem is not joint convex with respect to two group parameters, we propose to update them in an alternate way. When we fix one group parameter, the subproblem is joint convex to the variables, hence alternating minimization can be adopted to obtain the global optimum. We also discuss convergence behavior, together with the relationship with other methods, the parameter determination problem and computational complexity. Plenty of experimental results on different kinds of data sets have been provided for illustration.

It is worth highlighting the novelty of our proposed framework.

1) We propose a unified framework for clustering high-dimensional data that combines subspace learning and clustering in a common procedure.
2) Our proposed framework provides a unified view to analyze and understand many joint subspace learning and clustering methods. Furthermore, it encourages us to develop new methods for clustering high-dimensional data.
3) Our work outperforms existing methods on six benchmark data sets that demonstrates its promising performance in real applications.

The rest of this paper is organized as follows. Section II provides notations and related works. We formulate the proposed DEC framework and provide an effective method of solving this problem in Section III. We discuss the relationship to prior works in Section IV. The performance analyses, including convergence behavior, computational complicity, and parameter determination, are presented in Section V. Section VI provides promising comparison results on various kinds of data sets, followed by the conclusion and future works in Section VII.

II. Related Work

In this section, we will introduce two kinds of representative method. The first group contains subspace learning methods which have close relationship to our proposed framework. In the second group, we take LDAKM as the representative method of previous investigations about simultaneous dimensionality reduction and clustering. Before going into the details, let us introduce the notation.

A. Notation

We try to cluster high-dimensional data in this paper. Let $\{\mathbf{x}_i \in \mathbb{R}^D | i = 1, 2, \ldots, n\}$ as the data in high-dimensional space and the associated low-dimensional representations be $\{\mathbf{y}_i \in \mathbb{R}^d | i = 1, 2, \ldots, n\}$, where D and d are dimensionalities of high- and low-dimensional spaces, respectively. Define $\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_n\} \subset \mathbb{R}^c$ as the cluster indicator of $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, respectively. Here, $\mathbf{f}_i = [f_{i1}, f_{i2}, \ldots, f_{ic}]$. $f_{ij} = 1$ if and only if $\mathbf{x}_i$ belongs to the jth cluster and $f_{ij} = 0$ otherwise. c is the predefined number of clusters.

Define $\mathbf{e} = [1, 1, \ldots, 1] \in \mathbb{R}^{1\times n}$ as a row vector of all ones and $\lambda > 0$ as a balance parameter. Denote $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{D\times n}$, $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n] \in \mathbb{R}^{d\times n}$, and $\mathbf{F} = [\mathbf{f}_1^T, \mathbf{f}_2^T, \ldots, \mathbf{f}_n^T]^T \in \mathbb{R}^{n\times c}$. Define $\mathbf{Q} \in \mathbb{R}^{D\times d}$ as the transformation matrix such that $\mathbf{Y} = \mathbf{Q}^T\mathbf{X}$. Other notations are summarized in Table I. We will explain the meaning of each term when it is first used.

TABLE I
NOTATIONS

D	Dimensionality of high dimensional space;
d	Dimensionality of embedded subspace;
n	Number of points;
c	Number of clusters;
λ	Balance parameter;
$\mathbf{e} = [1, 1, \cdots, 1] \in \mathbb{R}^{1\times n}$	The row vector of all ones;
$\mathbf{x}_i \in \mathbb{R}^D$	The i-th high dimensional data;
$\mathbf{y}_i \in \mathbb{R}^d$	Low dimensional embedding of $\mathbf{x}_i$;
$\mathbf{f}_i \in \mathbb{R}^c$	Cluster indicator of $\mathbf{x}_i$;
$\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]$	Data matrix in high dimensional space;
$\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_n]$	Data matrix in low dimensional subspace;
$\mathbf{F} = [\mathbf{f}_1^T, \cdots, \mathbf{f}_n^T]^T$	Cluster indicator matrix;
$\mathbf{Q} \in \mathbb{R}^{D\times d}$	The transformation matrix;
$\mathbf{G} \in \mathbb{R}^{d\times c}$	The cluster centroid matrix;

B. Representative Subspace Learning Methods

In this section, we will introduce a number of subspace learning methods, including PCA-, OCM-, MMC-based subspace learning, and OLSDA.

PCA is one of the most frequently used subspace learning methods. It attempts to find orthogonal linear transformation directions in which the variance of original data is maintained as much as possible. Without loss of generality, assume that $\mathbf{X}$ has zero empirical mean, i.e., $\mathbf{e}\mathbf{X}^T = \mathbf{0}$. Mathematically, we measure the variance of the embedded points by maximizing

$$\mathrm{Tr}(\mathbf{Y}\mathbf{Y}^T) = \mathrm{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{Q}). \tag{1}$$

Here, PCA assumes that the transformation is linear, $\mathbf{Y} = \mathbf{Q}^T\mathbf{X}$.

We can also view the formulation in (1) in another aspect. Recalling the definition of total variance matrix $\mathbf{S}_t$ in LDA, we know that

$$\mathbf{S}_t = \sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \tag{2}$$

where $\bar{\mathbf{x}}$ is the data mean. Since we have assumed that $\mathbf{X}$ has zero empirical mean, i.e., $\bar{\mathbf{x}} = \mathbf{0}$, the problem in (1) becomes

$$\begin{aligned}&\max\ \mathrm{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{Q})\\ &= \max\ \mathrm{Tr}\left(\mathbf{Q}^T\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T\mathbf{Q}\right)\\ &= \max\ \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}).\end{aligned} \tag{3}$$

Since we constrain $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$, the optimal solution of PCA can be obtained by the eigen-decomposition of $\mathbf{S}_t$ or $\mathbf{X}\mathbf{X}^T$.

The second method is OCM. It provides a dimensionality reducing linear transformation preserving the clustering structure in the given data. Assume that centroids are taken as representatives of each cluster and the vectors of the input space are transformed by an orthonormal basis of the space spanned by the centroids. OCM tries to search for a dimensionality reduction transformation matrix $\mathbf{Q}$ with which the cluster existing in $\mathbf{X}$ is preserved. Thus, the objective function of OCM is

$$\begin{aligned}&\max_{\mathbf{Q}}\ \sum_{i}^{c} n_i\|\bar{\mathbf{y}}_i - \bar{\mathbf{y}}\|_F^2\\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I},\quad \mathbf{Y} = \mathbf{Q}^T\mathbf{X},\end{aligned} \tag{4}$$

where $\bar{\mathbf{y}}_i$ is the centroids for data in the ith cluster, $\bar{\mathbf{y}}$ is the centroids for all data in low-dimensional space, and n_i is the number of points in the ith category. $\|\cdot\|_F$ represents the Frobenius norm of a matrix.

After some deduction, the optimization problem of OCM can be reformulated as

$$\begin{aligned}&\max\ \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_b\mathbf{Q})\\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I}.\end{aligned} \tag{5}$$

Here, $\mathbf{S}_b$ is referred to as the between scatter matrix in LDA. It is defined as

$$\mathbf{S}_b = \sum_{i=1}^{c} n_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})^T, \tag{6}$$

where $\bar{\mathbf{x}}_i$ is the mean of data in the ith cluster.

Similarly, as shown in (5), the optimal solution to OCM can be derived by eigen-decomposition of $\mathbf{S}_b$.

The third is the MMC-based subspace learning approach that tries to calculate the most discriminant vectors and to avoid the small sample size problem of LDA. Geometrically, MMC maximizes the average margin between classes. Mathematically, the objective function of MMC is

$$\begin{aligned}&\max\ \mathrm{Tr}(\mathbf{Q}^T(\mathbf{S}_b - \mathbf{S}_w)\mathbf{Q})\\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I}.\end{aligned} \tag{7}$$

Here, $\mathbf{S}_w$ is referred to as the within scatter matrix, whose definition is

$$\mathbf{S}_w = \sum_{i=1}^{c}\sum_{\mathbf{x}_j\in\mathcal{X}_i}(\mathbf{x}_j - \bar{\mathbf{x}}_i)(\mathbf{x}_j - \bar{\mathbf{x}}_i)^T, \tag{8}$$

where $\mathcal{X}_i$ contains samples from the ith cluster.

Finally, we introduce OLSDA. It is proposed from the perspective of least squares regression. To obtain great discriminative power, the data from the same class are expected to be regressed to a single vector, and the single vector for each class can be reasonably selected as the class centroid after the transformation.

With this objective, OLSDA can be formulated as

$$\begin{aligned}&\min_{\mathbf{Q}}\ \|\mathbf{Y} - \mathbf{T}\|_F^2\\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I},\quad \mathbf{Y} = \mathbf{Q}^T\mathbf{X},\end{aligned}$$

where $\mathbf{T} = [\mathbf{t}_{l_1}, \mathbf{t}_{l_2}, \ldots, \mathbf{t}_{l_n}]$ is the class centroid for all data points and $\mathbf{t}_i = \mathbf{Q}^T\bar{\mathbf{x}}_i$. Here, l_i is the class label of $\mathbf{x}_i$.

Define a weighted matrix $\mathbf{A}$ as

$$A_{ij} = \begin{cases} 1/n_i, & l_i = l_j\\ 0, & \text{otherwise,}\end{cases}$$

where n_i is defined as in (4). It can be checked that $\mathbf{A}$ is an idempotent matrix, i.e., $\mathbf{A} = \mathbf{A}^2$ and the within scatter matrix can be rewritten as $\mathbf{S}_w = \mathbf{X}(\mathbf{I} - \mathbf{A})\mathbf{X}^T$.

The objective function of OLSDA can be reformulated as

$$\begin{aligned}\|\mathbf{Y}-\mathbf{T}\|_F^2 &= \|\mathbf{Q}^T\mathbf{X} - [\mathbf{Q}^T\mathbf{x}_{l_1}, \mathbf{Q}^T\mathbf{x}_{l_2}, \ldots, \mathbf{Q}^T\mathbf{x}_{l_n}]\|_F^2 \\ &= \|\mathbf{Q}^T\mathbf{X} - \mathbf{Q}^T\mathbf{X}\mathbf{A}\|_F^2 = \mathrm{Tr}(\mathbf{Q}^T\mathbf{X}(\mathbf{I}-\mathbf{A})^2\mathbf{X}^T\mathbf{Q}) \\ &= \mathrm{Tr}(\mathbf{Q}^T\mathbf{X}(\mathbf{I}-\mathbf{A})\mathbf{X}^T\mathbf{Q}) = \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_w\mathbf{Q}).\end{aligned}$$

Then, the optimization problem of OLSDA is

$$\begin{aligned}&\min\ \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_w\mathbf{Q}) \\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I}.\end{aligned} \tag{9}$$

In summary, the above-mentioned four methods can be regarded as the variants of LDA, using different kinds of criteria and having different performances. Although these methods consider the separateness of different clusters, they can not be used for dimensionality reduction directly since we do not have labels in the clustering task.

C. LDAKM

LDAKM is the representative method of joint embedding learning and clustering. It has been widely investigated from various aspects by different researchers. It employs K-means in the embedded subspace to obtain the data clusters and then uses LDA on the original data to derive the transformation. More concretely, it first optimizes the following objective function of K-means:

$$\min\ \sum_{i=1}^{c}\sum_{\mathbf{x}_j\in\mathcal{X}_i}(\mathbf{y}_j-\mathbf{g}_i)^T(\mathbf{y}_j-\mathbf{g}_i), \tag{10}$$

where $\mathbf{g}_i$ is the ith cluster centroid in the embedded subspace.

After calculating the cluster of each point, we construct $\mathbf{S}_b$ and $\mathbf{S}_w$ as in (6) and (8) in the original high-dimensional space. LDAKM then uses LDA to compute the transformation matrix by optimizing the following objective function:

$$\max\ \mathrm{Tr}((\mathbf{Q}^T\mathbf{S}_b\mathbf{Q})(\mathbf{Q}^T\mathbf{S}_w\mathbf{Q})^{-1}). \tag{11}$$

With several iterations between K-means and LDA, LDAKM will obtain the clustering results together with the transformation matrix $\mathbf{Q}$. The problem in (10) can be solved by simply employing a traditional K-means clustering approach and the problem in (11) is tackled by general eigen-decomposition of $\mathbf{S}_b$ and $\mathbf{S}_w$.

III. Discriminative Embedded Clustering

We now introduce our algorithm formally. Considering that PCA and K-means are two of the most widely used methods in dimensionality reduction and clustering, we investigate how to unify them, which motivates our novel framework.

A. Formulation

We now introduce our algorithm formally. Considering that PCA and K-means are two most widely used methods in dimensionality reduction and clustering, we would like to investigate how to unified them together that motivates our novel framework.

As stated above, the reason why K-means and LDA can be integrated is that LDA uses the label information derived by K-means. In other words, it is difficult to combine an unsupervised dimensionality reduction approach, such as PCA, with K-means, since unsupervised methods are unable to use label information directly. To tackle this problem, we observe that if we use K-means in the low-dimensional subspace, the transformation matrix dominates the performance of K-means. Thus, we propose to share the transformation matrix between two procedures, i.e., dimensionality reduction and clustering, rather than the label information as in LDAKM. To achieve this goal, we propose a framework which unifies many methods. We also present an effective solving method to guarantee convergence.

There are two main objective functions of DEC: the first one concerns dimensionality reduction and the second one concerns the loss of a clustering algorithm. Before going into the details, let us reformulate the objective function of K-means in (10).

Recalling the basic idea of K-means, we know it tries to minimize the distance of each point to its nearest centroid. Denote c centroids as $\{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_c\} \subset \mathbb{R}^d$ and $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_c]$. The corresponding cluster indicators are $\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_n\} \subset \mathbb{R}^c$. $\mathbf{f}_i = [f_{i1}, f_{i2}, \ldots, f_{ic}]$. $f_{ij} = 1$ if and only if $\mathbf{x}_i$ is assigned to the jth cluster and $f_{ij} = 0$ otherwise. Considering the objective function of K-mean in (10), we reformulate it as

$$\begin{aligned}&\sum_{i=1}^{c}\sum_{\mathbf{x}_j\in\mathcal{X}_i}(\mathbf{y}_j-\mathbf{g}_i)^T(\mathbf{y}_j-\mathbf{g}_i) \\ &\qquad = \sum_{i=1}^{c}\sum_{j=1}^{n}(\mathbf{y}_j-\mathbf{g}_i)^T(\mathbf{y}_j-\mathbf{g}_i)\chi_{\mathcal{X}_i}(\mathbf{x}_j),\end{aligned} \tag{12}$$

where $\chi_{\mathcal{X}_i}(\mathbf{y}_j)$ is the characteristic function whose definition is

$$\chi_{\mathcal{X}_i}(\mathbf{x}_j) = \begin{cases}1, & \mathbf{x}_j \in \mathcal{X}_i \\ 0, & \mathbf{x}_j \notin \mathcal{X}_i.\end{cases} \tag{13}$$

Recalling the definition of $\mathbf{G}$ and $\mathbf{F}$, we know that

$$\sum_{i=1}^{c}(\mathbf{y}_j-\mathbf{g}_i)^T(\mathbf{y}_j-\mathbf{g}_i)\chi_{\mathcal{X}_i}(\mathbf{x}_j) = \left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right)^T\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right). \tag{14}$$

Thus, the objective function of K-means in (12) becomes

$$\begin{aligned}&\sum_{i=1}^{c}\sum_{j=1}^{n}(\mathbf{y}_j-\mathbf{g}_i)^T(\mathbf{y}_j-\mathbf{g}_i)\chi_{\mathcal{X}_i}(\mathbf{x}_j) \\ &\quad = \sum_{j=1}^{n}\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right)^T\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right) \\ &\quad = \sum_{j=1}^{n}\mathrm{Tr}\left(\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right)\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right)^T\right) \\ &\quad = \mathrm{Tr}\left(\sum_{j=1}^{n}\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right)\left(\mathbf{y}_j-\mathbf{G}\mathbf{f}_j^T\right)^T\right) \\ &\quad = \mathrm{Tr}\left(\left(\mathbf{Y}-\mathbf{G}\mathbf{F}^T\right)\left(\mathbf{Y}-\mathbf{G}\mathbf{F}^T\right)^T\right) \\ &\quad = \|\mathbf{Y}-\mathbf{G}\mathbf{F}^T\|_F^2.\end{aligned} \tag{15}$$

Considering the formulation of K-means in (15), we propose to combine it with PCA. Assuming that the objective functions of PCA in (1) and K-means in (15) share the same $\mathbf{Q}$, we induce the constraint $\mathbf{Y} = \mathbf{Q}^T\mathbf{X}$ into (15) and combine them by directly adding a balance parameter. Mathematically, the objective function of DEC is

$$\mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) - \lambda\|\mathbf{Q}^T\mathbf{X} - \mathbf{G}\mathbf{F}^T\|_F^2. \tag{16}$$

As in the previous method, we should add constraints on the optimization variables. First, as in PCA, the columns of $\mathbf{Q}$ should be orthogonal, i.e., $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$. Then, the parameter $\mathbf{F}$ should be an indicator matrix. Each row of $\mathbf{F}$ should and must have a single nonzero element 1. Formally, the formulation of DEC is

$$\begin{aligned} &\arg\max_{\mathbf{Q},\mathbf{G},\mathbf{F}} \ \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) - \lambda\|\mathbf{Q}^T\mathbf{X} - \mathbf{G}\mathbf{F}^T\|_F^2 \\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I}, \end{aligned} \tag{17}$$

where $\mathbf{F}$ is an indicator matrix.

As we can see from the above formulation, the derivation of $\mathbf{Q}$ is not only related to the first objective function of dimensionality reduction, it is also related to the objective function of clustering. In other words, when we reduce the dimensionality of high-dimensional data, we also consider the requirement of the subsequent clustering. On the contrary, since we have clustered data in the embedded space, it is directly related to the transformation matrix.

Although the formulation of DEC in (17) is simple, it is meaningful because of the following.

1) Unlike traditional approaches which share label information by combining dimensionality reduction and clustering, we propose a new way that also shares the transformation matrix. This enables joint unsupervised dimensionality reduction and clustering.
2) As we will show in Section IV, the proposed DEC model is a general framework. Other methods can be unified within this framework by the selection of a suitable parameter λ. We can also choose a suitable parameter for developing new methods which are more effective.
3) Since the optimization problem in (17) is not jointly convex with respect to $\mathbf{Q}$, $\mathbf{G}$, and $\mathbf{F}$, it is hard to find its global optimal solution. We will propose strategies for solving this problem.

B. Solution

In this section, we try to find an approximated solution to the proposed problem in (17) since it is not jointly convex with respect to all the variables. There are two groups of parameters. The first concerns dimensionality reduction, i.e., $\mathbf{Q}$, and the second concerns clustering, i.e., $\mathbf{G}$ and $\mathbf{F}$. One direct way is to optimize these two groups alternately. Nevertheless, when $\mathbf{Q}$ is fixed, it is hard to find the optimal solution to the formulated problem. Thus, in our solution, we alternate between $\mathbf{Q}$, $\mathbf{G}$, and $\mathbf{F}$.

1) Fixing $\mathbf{Q}$, $\mathbf{G}$ *and Optimizing* $\mathbf{F}$*:* As obtained from (17), when $\mathbf{Q}$ and $\mathbf{G}$ are fixed, the first term in (17) is constant and it is only necessary to minimize the second term. Since $\mathbf{G}$ is fixed and $\mathbf{F}$ is constrained to be a indicator, the optimal $\mathbf{F}$ is

$$F_{ij} = \begin{cases} 1, & j = \arg\min_k \ \|\mathbf{Q}^T\mathbf{x}_i - \mathbf{g}_k\|_F^2 \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

Certainly, when we fix $\mathbf{Q}$ and $\mathbf{G}$, DEC is equal to K-means in assigning clusters of embedded points, given the cluster center. In addition, when $\mathbf{Q}$ and $\mathbf{G}$ are fixed, the solution in (18) is the global optimization of the problem in (17), since the objective function at each point is minimized when we use (18) to derive $\mathbf{F}$.

2) Fixing $\mathbf{F}$ *and Optimizing* $\mathbf{Q}$, $\mathbf{G}$*:* When the cluster indicator $\mathbf{F}$ is fixed, the optimal $\mathbf{Q}$ and $\mathbf{G}$ to the problem in (17) can also be derived in a closed form. We derive the solution by first taking the deviation of the objective function in (17) with respect to $\mathbf{G}$. For convenience, denote

$$\mathcal{L}(\mathbf{Q},\mathbf{G}) = \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) - \lambda\|\mathbf{Q}^T\mathbf{X} - \mathbf{G}\mathbf{F}^T\|_F^2. \tag{19}$$

Then

$$\begin{aligned} \frac{\partial\mathcal{L}(\mathbf{Q},\mathbf{G})}{\partial\mathbf{G}} &= -\lambda\frac{\partial(\mathrm{Tr}(\mathbf{G}\mathbf{F}^T - \mathbf{Q}^T\mathbf{X})^T(\mathbf{G}\mathbf{F}^T - \mathbf{Q}^T\mathbf{X}))}{\partial\mathbf{G}} \\ &= -\lambda\frac{\partial(\mathrm{Tr}(\mathbf{G}\mathbf{F}^T\mathbf{F}\mathbf{G}^T) - 2\mathrm{Tr}(\mathbf{G}\mathbf{F}^T\mathbf{X}^T\mathbf{Q}))}{\partial\mathbf{G}} \\ &= -2\lambda(\mathbf{G}\mathbf{F}^T\mathbf{F} - \mathbf{Q}^T\mathbf{X}\mathbf{F}). \end{aligned} \tag{20}$$

Let the above equations equal zero, and we have

$$\mathbf{G} = \mathbf{Q}^T\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}. \tag{21}$$

Induce the derived result in (21) into (19), and we have

$$\begin{aligned} &\mathcal{L}(\mathbf{Q}) \\ &= \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) - \lambda\|\mathbf{Q}^T\mathbf{X} - \mathbf{Q}^T\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\|_F^2 \\ &= \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) - \lambda\mathrm{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{Q} - \mathbf{Q}^T\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T\mathbf{Q}) \\ &= \mathrm{Tr}[\mathbf{Q}^T(\mathbf{S}_t - \lambda\mathbf{X}\mathbf{X}^T + \lambda\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T)\mathbf{Q}]. \end{aligned} \tag{22}$$

The optimization problem in (17) becomes

$$\begin{aligned} &\arg\max_{\mathbf{Q}} \ \mathcal{L}(\mathbf{Q}) \\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q} = \mathbf{I}. \end{aligned} \tag{23}$$

Based on the results of matrix theory, the optimal solution to the problem in (23) can be derived by picking up d eigenvectors corresponding to the d largest eigenvalues of $\mathbf{S}_t - \lambda\mathbf{X}\mathbf{X}^T + \lambda\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T$. This solution is the global optimization to the problem in (23).

In summary, when one group of parameters is fixed, our derived solution is the global optimization to the problem in (17). We link it to the following proposition.

Proposition 1: When $\mathbf{Q}$ and $\mathbf{G}$ are fixed, the derived $\mathbf{F}$ in (18) is the global solution to the problem in (17). Similarly, when $\mathbf{F}$ is fixed, the derived $\mathbf{G}$ in (21) and the derived $\mathbf{Q}$ by picking up eigenvectors corresponding to the d largest eigenvalues of $\mathbf{S}_t - \lambda\mathbf{X}\mathbf{X}^T + \lambda\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T$ are also the global solutions to the problem in (17).

The proof of this proposition is direct. When $\mathbf{Q}$ and $\mathbf{G}$ are fixed, the optimization problem in (17) is equal to traditional K-means on $\mathbf{Q}^T\mathbf{X}$ with fixed centroid. Thus, the optimal solution is unique. When $\mathbf{F}$ is fixed, $\mathbf{G}$ is determined by $\mathbf{Q}$

and the optimization problem in (23) has global optimization derived by eigen-decomposition.

In addition, in employing the above alternation, we find another problem which should be highlighted here. Although the above solving strategy can guarantee convergence to a local minimum, the solution is not satisfied. As in traditional K-means, there is considerable local optimization which is dominated by initialization [42]. Taking the above updating rule into consideration, when $\mathbf{F}$ is fixed, the algorithm can adjust $\mathbf{Q}$ and $\mathbf{G}$ to fit this $\mathbf{F}$ quickly. In other words, when we need to update $\mathbf{F}$ in the next step, the optimal $\mathbf{F}$ is the same as previously. That is to say, the algorithm converges quickly and the derived optimal solution is dominated by initialization. This phenomenon also occurs in K-means. To reduce the local optimal problem, we can use the following update rule.

1) *Updating Rule 1 (Comparison):* In each step of updating $\mathbf{F}$, we randomly initialize $\mathbf{F}$ several times (10 in our experiments). If one of the corresponding objective functions $\|\mathbf{Q}^T\mathbf{X}-\mathbf{G}\mathbf{F}^T\|_F^2$ is smaller than that derived by the previous $\mathbf{F}$, we update $\mathbf{F}$ by this random initialization. Otherwise, we use (18) to update $\mathbf{F}$. Mathematically, in the ith iteration, we derive the optimal $\mathbf{F}_i^*$, $\mathbf{Q}_i^*$, and $\mathbf{G}_i^*$. In the $(i+1)$th iteration, the random initializations are $\{\mathbf{F}_{i+1}^1, \mathbf{F}_{i+1}^2, \ldots, \mathbf{F}_{i+1}^t\}$, where t is the number of random initializations. We update $\mathbf{F}$ by the following rule:

$$\mathbf{F}_{i+1}^* = \begin{cases} \mathbf{F}_{i+1}^j, & \|(\mathbf{Q}_i^*)^T\mathbf{X}-\mathbf{G}_i^*(\mathbf{F}_{i+1}^j)^T\|_F^2 \\ & < \|(\mathbf{Q}_i^*)^T\mathbf{X}-\mathbf{G}_i^*(\mathbf{F}_i^*)^T\|_F^2 \\ \mathbf{F}^*, & \text{otherwise,} \end{cases} \quad (24)$$

where the elements of $\mathbf{F}^*$ are defined as

$$F_{ij}^* = \begin{cases} 1, & j = \arg\min_k \ \|(\mathbf{Q}_i^*)^T\mathbf{x}_i - (\mathbf{g}_i^*)_k\|_F^2 \\ 0, & \text{otherwise.} \end{cases}$$

For comparison, we also list another two updating rules.

1) *Updating Rule 2 (Fixed):* In each step of updating $\mathbf{F}$, we use (18) to update $\mathbf{F}$ directly. Mathematically

$$(F_{i+1}^*)_{ij} = \begin{cases} 1, & j = \arg\min_k \ \|(\mathbf{Q}_i^*)^T\mathbf{x}_i - (\mathbf{g}_i^*)_k\|_F^2 \\ 0, & \text{otherwise.} \end{cases} \quad (25)$$

2) *Updating Rule 3 (Minimization):* In each step of updating $\mathbf{F}$, we randomly initialize $\mathbf{F}$ several times (10 in our experiments) and choose $\mathbf{F}$ corresponding to the smallest objective function $\|\mathbf{Q}^T\mathbf{X}-\mathbf{G}\mathbf{F}^T\|_F^2$. Mathematically

$$\mathbf{F}_{i+1}^* = \mathbf{F}_{i+1}^j, \quad j = \arg\min_k \ \|(\mathbf{Q}_i^*)^T\mathbf{X}-\mathbf{G}_i^*(\mathbf{F}_{i+1}^k)^T\|_F^2. \quad (26)$$

As mentioned above, if we use the fixed updating rule, DEC will converge to a local minimization quickly and the result will be dominated by initialization. If we use the minimization updating rule, it can not guarantee convergence since the objective function may decrease. By contrast, if we use comparison, the above problems can be avoided. We will present a proposition in Section V to show the convergence behavior. In the following experiments, we will use this updating rule without specification.

In summary, the procedure of DEC is listed in Table II.

TABLE II
PROCEDURE OF DEC

Input:

Data set: $\{\mathbf{x}_i | i = 1, 2, \cdots, n\}$, balance parameter λ.

Output: Transformation matrix $\mathbf{Q}$ and cluster indicator $\mathbf{F}$

1. Initialize $\mathbf{Q}$ by PCA, i.e., solving the problem in Eq. (3).
   Initialize $\mathbf{F}$ by conducting K-means on $\mathbf{Q}^T\mathbf{X}$.
2. Alternately update $\mathbf{Q}$, $\mathbf{G}$ and $\mathbf{F}$ until convergence.
   a. Update $\mathbf{F}$ by *Comparison* rule in Eq. (24).
   b. Update $\mathbf{Q}$ by picking up eigenvectors corresponding to the d largest eigenvalues of $\mathbf{S}_t - \lambda\mathbf{X}\mathbf{X}^T + \lambda\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T$.
   Update $\mathbf{G}$ by Eq. (21).

IV. DISCUSSION IN RELATION TO PRIOR WORK

In this section, we discuss the relation between DEC and PCA, OCM, MMC, and OLSDA. Before going into the details, let us first simplify the formulation in (22).

Note that

$$\mathbf{S}_t = \mathbf{X}\mathbf{X}^T$$
$$\mathbf{S}_b = \sum_{i=1}^{c} n_i(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})(\bar{\mathbf{x}}_i - \bar{\mathbf{x}})^T = \mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T, \quad (27)$$

and the equation $\mathbf{S}_t = \mathbf{S}_b + \mathbf{S}_w$, we change (22) as follows:

$$\begin{aligned} &\mathrm{Tr}[\mathbf{Q}^T(\mathbf{S}_t - \lambda\mathbf{X}\mathbf{X}^T + \lambda\mathbf{X}\mathbf{F}(\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{X}^T)\mathbf{Q}] \\ &= \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) - \lambda\mathrm{Tr}[\mathbf{Q}^T(\mathbf{S}_t - \mathbf{S}_b)\mathbf{Q}] \\ &= (1-\lambda)\mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q}) + \lambda\mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_b\mathbf{Q}) \\ &= \mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_b\mathbf{Q}) + (1-\lambda)\mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_w\mathbf{Q}). \end{aligned} \quad (28)$$

Example 1: When $\lambda \to 0$, DEC is equivalent to performing the two popular methods PCA and K-means in sequence.

Proof: As obtained from (17), when $\lambda \to 0$, we optimize two objective functions, i.e., $\mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_t\mathbf{Q})$ and $\|\mathbf{Q}^T\mathbf{X}-\mathbf{G}\mathbf{F}^T\|_F^2$ in sequence. Since the first term is the objective function of PCA and the second term is the objective function of K-means, DEC is equivalent to conducting these two methods in sequence. ■

Example 2: When $\lambda = 1$, DEC is equivalent to using OCM to reduce dimensionality and performing K-means on the reduced data alternately.

Proof: Considering the optimization problem in (23) and the equation in (29), we know that $\mathbf{Q}$ is derived by solving the problem with objective function $\mathrm{Tr}(\mathbf{Q}^T\mathbf{S}_b\mathbf{Q})$. Recalling the formulation of OCM in (5), it is clear that they are the same. Considering the updating rule of DEC in Table II, we know that when $\lambda = 1$, DEC is equivalent to alternation between OCM and K-means. ■

Example 3: When $\lambda = 2$, DEC is equivalent to using MMC to reduce dimensionality and performing K-means on the reduced data alternately.

The proof of this result is the same as that in Example 2. We would like to exclude it.

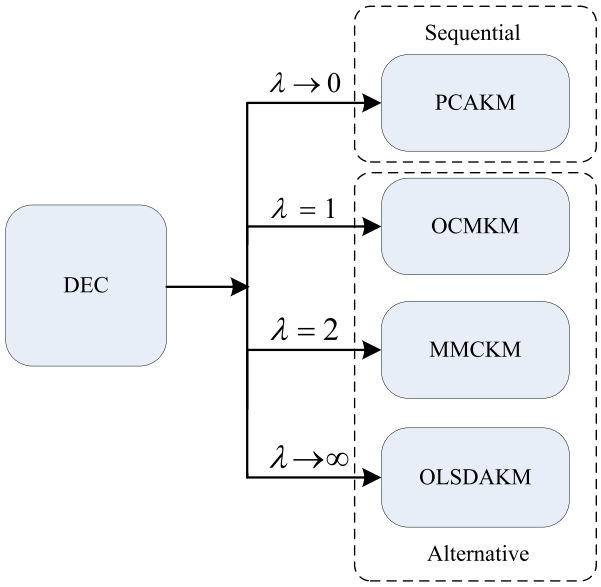


Fig. 1. Relationship of our DEC framework and other related methods.

Example 4: When $\lambda \to +\infty$, DEC is equivalent to using OLSDA to reduce dimensionality and performing K-means on the reduced data alternately.

The proof of this result is the same as previously. Note that, unlike the scenario in which $\lambda = 0$, our framework alternates when $\lambda \to +\infty$ since the updating of $\mathbf{Q}$ is also related to $\mathbf{F}$.

In summary, we can view different methods within the proposed framework DEC. Different methods correspond to different selections of parameter λ. For convenience, we call these methods PCAKM, OCMKM, MMCKM, and OLSDAKM, respectively. We can also design new methods by choosing suitable parameters within this framework. The relationships of our DEC framework with other related methods are shown in Fig. 1.

V. Performance Analysis

In this section, we will analyze the performance of DEC in three aspects, i.e., the convergence behavior, the computational complexity analysis and the parameter determination.

A. Convergence Analysis

As mentioned above, DEC is solved in an alternate way; namely, we fix one group of variables and optimize the other. As indicated in Proposition 1, we can obtain the global optimization in each alternation. The following proposition shows that our algorithm will monotonically increase the objective function of the problem in (17) in each iteration.

Proposition 2: The procedures of DEC shown in Table II will monotonically increase the objective function of the problem in (17) in each iteration.

Proof: Assume that we have derived $\mathbf{Q}$, $\mathbf{G}$, as $\mathbf{Q}_{(i)}$ and $\mathbf{G}_{(i)}$ in the ith iteration. In the $(i+1)$th iteration, we fix $\mathbf{Q}$, $\mathbf{G}$ as $\mathbf{Q}_{(i)}$, $\mathbf{G}_{(i)}$ and optimize $\mathbf{F}_{(i+1)}$ using (24). Recalling the results in Proposition 1 and the updating rule in (14), we have the following inequality:

$$\begin{aligned}\mathrm{Tr}\big(\mathbf{Q}_{(i)}^T\mathbf{S}_t\mathbf{Q}_{(i)}\big) - \lambda\big\|\mathbf{Q}_{(i)}^T\mathbf{X} - \mathbf{G}_{(i)}\mathbf{F}_{(i)}^T\big\|_F^2 \\ \le \mathrm{Tr}\big(\mathbf{Q}_{(i)}^T\mathbf{S}_t\mathbf{Q}_{(i)}\big) - \lambda\big\|\mathbf{Q}_{(i)}^T\mathbf{X} - \mathbf{G}_{(i)}\mathbf{F}_{(i+1)}^T\big\|_F^2. \quad (29)\end{aligned}$$

Similarly, when we fix $\mathbf{F}$ as $\mathbf{F}_{(i+1)}$ and optimize $\mathbf{Q}$, $\mathbf{G}$ by maximizing the objective function in (19), the following result holds:

$$\begin{aligned}\mathrm{Tr}\big(\mathbf{Q}_{(i)}^T\mathbf{S}_t\mathbf{Q}_{(i)}\big) - \lambda\big\|\mathbf{Q}_{(i)}^T\mathbf{X} - \mathbf{G}_{(i)}\mathbf{F}_{(i+1)}^T\big\|_F^2 \\ \le \mathrm{Tr}\big(\mathbf{Q}_{(i+1)}^T\mathbf{S}_t\mathbf{Q}_{(i+1)}\big) - \lambda\big\|\mathbf{Q}_{(i+1)}^T\mathbf{X} - \mathbf{G}_{(i+1)}\mathbf{F}_{(i+1)}^T\big\|_F^2. \quad (30)\end{aligned}$$

Combining the formulas in (29) and (30) will result in inequality, which indicates the increase of the objective function during iteration. ■

The following point should be highlighted. The final results of DEC are closely related to the initialization, since initialization seriously impacts the final performance [42]. We use two traditional methods, i.e., PCA and K-means, to initialize. To avoid rapid convergence to the local optimization, we also randomly initialize several times in updating $\mathbf{F}$. Experimental results show that this kind of initialization works well. Therefore, our derived solution may be close to the global optimal solution.

B. Computational Complexity Comparison

In this section, we analyze the computational complexity of different methods. Since PCAKM, OCMKM, MMCKM, and OLSDAKM can be unified within our framework, and K-means, LDAKM are closely related to DEC, we analyze their computational complexities. Since different implementations of these methods may have different computational complexities, we simply give common analyses.

The first method is K-means. It is not a convex optimization problem and its computational complexity is $O(ncD)$, where n is the number of data points, c is the number of clusters, and D is the dimensionality of the input data. The computational complexity of K-means is linear with respect to n, D, and c, respectively.

The second group of methods consists of PCAKM and LDAKM. They contain two main steps, i.e., eigen-decomposition in deriving $\mathbf{Q}$ and K-means in updating $\mathbf{F}$. Their computational complexities are $O(D^2n)$ and $O(ncd)$. Since PCAKM does not need iteration, its computational complexity is $O(D^2n + ncd)$. Assuming that the number of iterations is T, LDAKM has the computational complexity $O((D^2n+ncd)T)$. Note that, unlike traditional K-means, the K-means clustering in PCAKM and LDAKM are conducted on the data in the embedding subspace.

The third group of methods contains OCMKM, MMCKM, OLSDAKM, and DEC. They are all unified within our framework. Their main difference is the concrete form of λ. They contain two main steps, i.e., eigen-decomposition and K-means, as previously. Assuming there are T steps in total, their computational complexities are $O((D^2n+ncd)T)$.

As seen from above analyses, the most computational step of these methods is the eigen-decomposition. If the input data is of very high dimensionality, on one hand, we need

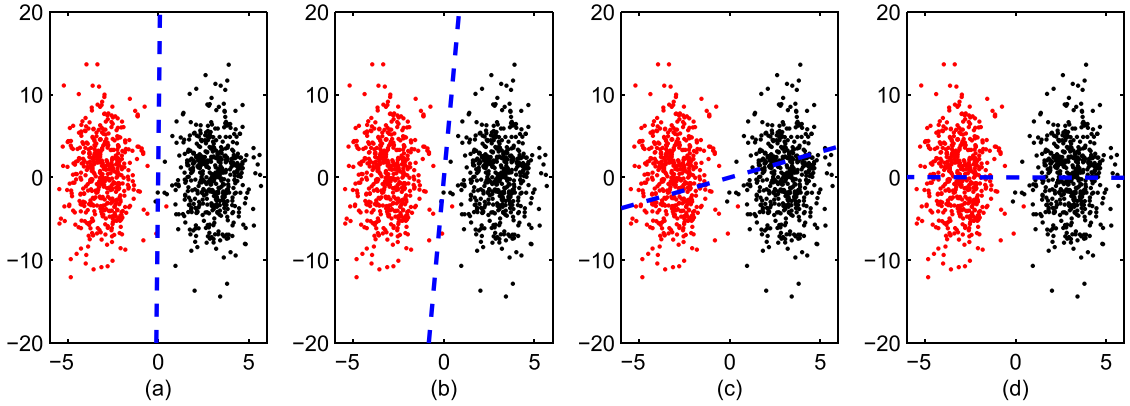


Fig. 2. Projection directions of DEC on a toy example with different numbers of iterations. (a) Initialization. (b) $T = 1$. (c) $T = 10$. (d) $T = 20$.

to use fast eigen-decomposition algorithms, such as Lanczos algorithm [43]. On the other hand, when the dimensionality is very high, we should reduce the dimensionality at first using algorithm with low computational cost to avoid the curse of dimensionality.

Commonly, the computational complexities of six methods have the following relationships. PCAKM $<$ LDAKM $=$ OCMKM $=$ MMCKM $=$ OLSDAKM $=$ DEC. Nevertheless, different implementations of the same method may have different time costs in real computing. We will provide numerical results in the next section.

C. Parameter Determination

There are two important parameters in our DEC algorithm. The first is the dimensionality of the embedding subspace d and the second is the balance parameter λ. Since parameter determination is still an open problem in the related fields, we determine parameters heuristically and empirically.

The first parameter d is used to reveal the intrinsic dimensionality of the original data. When d is large, the representation of the original data is still redundant and the curse of dimensionality still occurs. When d is too small, there will be overlap of different clusters. We determine this empirically by grid search in this paper by varying this parameter and choosing the one with the best clustering accuracy. The experiments in Section VI-C will provide numerical results with different d.

The second parameter is the balance parameter λ. Evidently, this parameter balances the effects of dimensionality reduction and clustering in subspace. The larger λ is, the more attention we pay to clustering. As shown in Fig. 1, several traditional methods can be unified within DEC by selecting suitable parameters. In our methods, we determine heuristically at two levels. We vary this parameter within a large range by large step size and determine the approximate optimal parameter. Subsequently, we vary this parameter near the approximate optimal value with small range and small length of interval. This two-level strategy reduces the number of experiments and finds a suitable λ. Experimental results show that when we find suitable λ, DEC outperforms other methods in most cases. This kind of method can help us to find the optimal parameter. See more details in Section VI-F.

VI. Experiments

There are five kinds of experiment in this section. The first group contains experimental result on a toy data. The second group mainly focuses on the evaluation of clustering accuracy. To demonstrate the convergence property, we provide objective function values with different numbers of iterations in the third group. We also show results with different reduced dimensionality by different updating rules. The forth group contains the computational time comparison results. Finally, clustering accuracies with different parameters are presented. We first introduce the data sets and evaluation method.

To show the effectiveness of joint dimensionality reduction and clustering, we have presented the projection directions of DEC with different numbers of iterations in Fig. 2.

A. Data Description and Evaluation Metric

There are six different kinds of matrix data sets, i.e., Umist,[1] Coil,[2] Isolet,[3] Corel,[4] Pollen,[5] and Orl.[6] They can be classified into three types. Umist, Coil, Pollen, and Orl are used by converting matrix representations of images into vectors. Isolet is voice data collected from 26 individuals. Corel contains images from 30 different topics over all the Corel data sets. In our implementation, we compute its characteristics from different aspects, including color histogram (nine dimensions) + edge direction histogram (18 dimensions) + wavelet (nine dimensions) as in [44]. Following preprocessing, the detailed statistical characteristics of different data sets are listed in Table III.

As mentioned above, we compare the performance of DEC with PCAKM, OCMKM, MMCKM, and OLSDAKM, since they are closely related to DEC. We also provide the results of K-means to show whether the dimensionality reduction is effective. Since LDAKM is the most famous method which combines dimensionality reduction and clustering, we also show its results.

[1] http://images.ee.umist.ac.uk/danny/database.html
[2] http://www1.cs.columbia.edu/CAVE/research/softlib/coil-20.html
[3] http://archive.ics.uci.edu/ml/datasets/ISOLET
[4] http://kdd.ics.uci.edu/databases/CorelFeatures/CorelFeatures.data.html
[5] http://ome.grc.nia.nih.gov/iicbu2008/pollen/index.html
[6] http://www.uk.research.att.com/facedatabase.html

TABLE III
CHARACTERS OF DIFFERENT DATA SETS

Data	Size (n)	Scale (D)	Clusters(c)	# of Red Dim
Umist	575	644	20	$2, 4, \cdots, 20$
Coil	1440	1024	20	$5, 6, \cdots, 14$
Isolet	1559	617	26	$5, 10, \cdots, 50$
Corel	3000	36	30	$1, 2, \cdots, 10$
Pollen	625	625	7	$2, 4, \cdots, 20$
Orl	400	1024	40	$20, 21, \cdots, 29$

The performance of different methods is evaluated by clustering accuracy. Its is defined as follows:

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

where map$(\cdot)$ is a function that maps each cluster index to a class label. It can be found by the Hungarian algorithm [45]. l_i is the true class label of $\mathbf{x}_i$. r_i is the cluster index derived from $\mathbf{f}_i$. $\delta(a, b)$ is the function that equals to 1 when a equals b and 0 otherwise. This metric discovers one-to-one relationships between clusters and the true classes. It measures the extent to which a cluster contains examples from the corresponding category. We compute it by summing the number of matches between all pair clusters. Higher clustering accuracy means better clustering performance.

B. Toy Example

In this section, we will give a toy experiments. The data set can be derived into two parts. Every part contains 500 samples. Data points of the left part are generated by MATLAB from a 2-D normal distribution with mean $[-3, 0]$ and covariance matrix $[1, 0; 0, 20]$. Data points of the right part are generated by a 2-D normal distribution with mean $[3, 0]$ and the same covariance matrix.

To show the effectiveness of DEC in combining dimensionality reduction and clustering, we show the projection directions of DEC with different numbers of iterations. The results with iterations number $T = 0$ [Fig. 2(a)], $T = 1$ [Fig. 2(b)], $T = 10$ [Fig. 2(c)], and $T = 20$ [Fig. 2(d)] are shown in Fig. 2. The blue dashed lines are the projection directions derived by DEC. In addition, the corresponding clustering accuracies of DEC are 0.5050, 0.5140, 0.8510, and 0.9980, respectively. In this situation, the mean clustering accuracy of K-means with 50 independent random initializations is 0.5984.

As seen from the results shown in Fig. 2, we have the following observations: 1) with the help of clustering, DEC improves the performance of finding suitable projection directions step-by-step and 2) with the help of dimensionality reduction, the clustering accuracy of DEC also increases steadily.

C. Clustering Accuracy Comparison

In this section, we provide the clustering results of different methods on different data sets. By repeating each method for 50 independent runs, we set the reduced dimensionality varying from 1 to 30, as shown in Table III. The balance parameter λ is determined heuristically by grid search, as mentioned in Section V-C.

With different data sets and different reduced dimensionality, the mean clustering accuracies of 50 independent runs on all data sets are shown in Fig. 3(a) (Umist data), Fig. 3(b) (Coil data), Fig. 3(c) (Isolet data), Fig. 3(d) (Corel data), Fig. 3(e) (Pollen data), and Fig. 3(f) (Orl data), respectively.

There are several observations from the performance comparisons as follows.

1) Of the different methods and different data sets, DEC performs best. It achieves the highest clustering accuracy in most cases, especially on the data sets Isolet and Pollen. This is mainly due to the fact that DEC balances the influence of dimensionality reduction and subspace learning in a suitable way.
2) With the increase of reduced dimensionality, not all methods achieve higher clustering accuracy. This is consistent with intuition, since different data sets have different intrinsic dimensionality.
3) If we choose suitable parameters in DEC for clustering, joint dimensionality reduction and clustering perform better than sequential methods. Taking the results in Fig. 3 as an example, PCAKM often performs worse than the methods in our framework, which may be because we consider the requirement of clustering in deriving low-dimensional embeddings.
4) When we represent the original data by its low-dimensional embedding, it is not always helpful for the next clustering. Taking Fig. 3(a) as an example, we achieve higher clustering accuracy when K-means is used to cluster the original data directly compared to the results achieved in using K-means to cluster the embedding derived by PCA.

D. Convergence Behavior

In this section, we will show the results of convergence behavior. As mentioned in Sections III-B and V-A, we have proposed three different kinds of rules for updating $\mathbf{F}$. We have also proved that the iteration will converge if we use the comparison rule. For illustration, we conducted experiments on three different data sets, i.e., Umist, Coil, and Isolet. As shown in the top plane of Fig. 4, the objective function values of DEC with different updating rules are provided. Note that the reduced dimensionality is $c - 1$ as in the traditional LDA approach and the balance parameter is the same as previously given in the experiments in Section VI-C. Together with these results, we show the clustering accuracies with different reduced dimensionality using different updating rules.

There are several observations from Fig. 4.

1) With the increase in the number of iterations, the comparison updating rule guarantees the convergence of our algorithm. The other two rules, i.e., fixed and minimization, also seem to be convergent. Nevertheless, their objective function values are smaller than those of comparison.

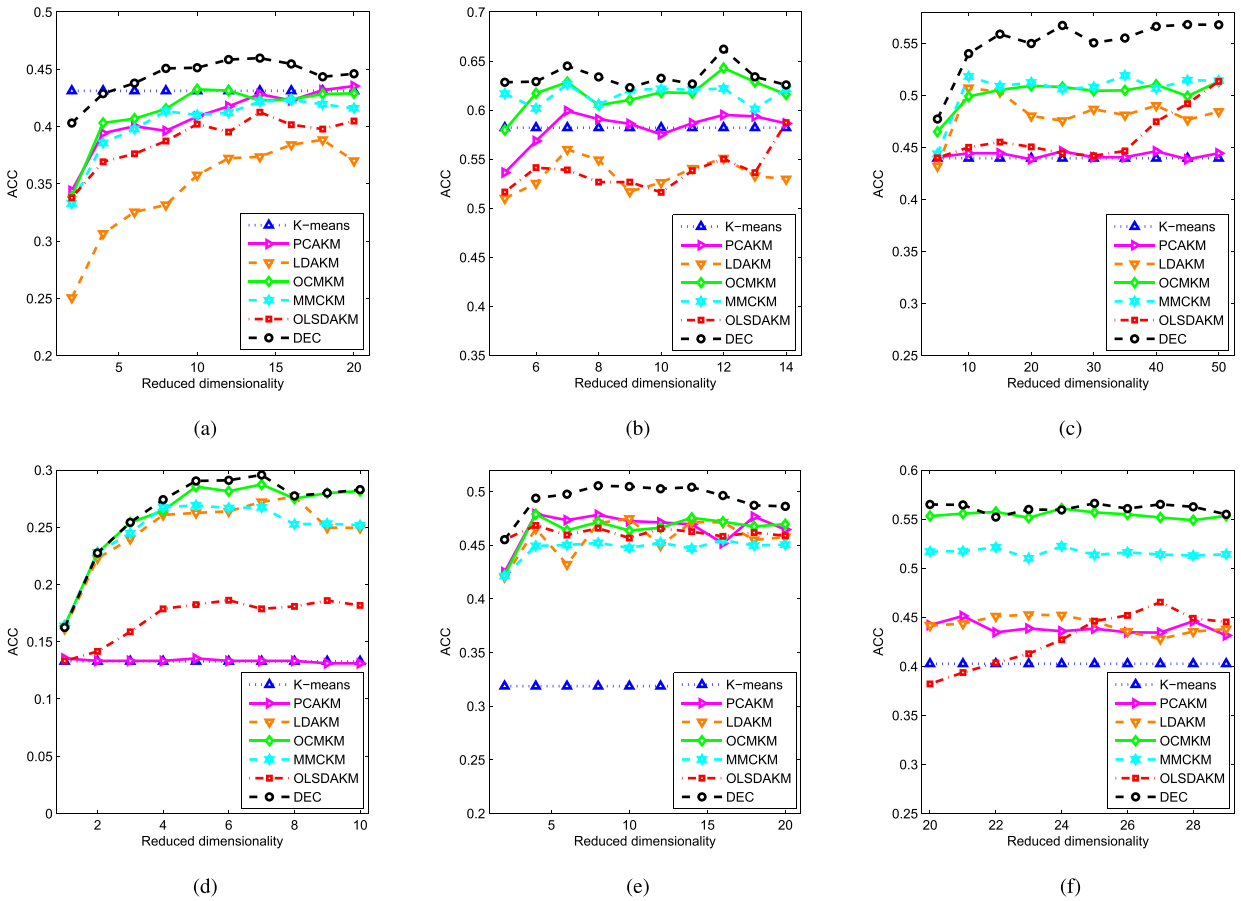


Fig. 3. Clustering accuracy of different methods on six different data sets with different numbers of reduced dimensionality. (a) Umist. (b) Coil. (c) Isolet. (d) Corel. (e) Pollen. (f) Orl.

2) If we use the comparison updating rule with different reduced dimensionality, the clustering accuracy is larger than for the other two rules. This may be because we can obtain a better local minimum using this rule. It is also consistent with initiation since it has larger objective function values than the other two rules.

E. Computational Time Comparison

Since computational efficiency is very important for real applications, we will show a number of experimental results with different data sizes and scales.

We conduct experiments on three representative data sets, Coil, Corel, and Pollen. For illustration, we compare DEC with K-means, PCAKM, LDAKM, OCMKM, MMCKM, and OLSDA. For impartiality, these methods are all implemented in their original formulation, without the use of other accelerating strategies. With each fixed number of reduced dimensionality, we randomly repeat each experiment for 50 runs. With a naive MATLAB implementation, the calculations are made on a 3.2-GHz Windows machine. The computational time of the different methods is listed in Tables IV–VI. The balance parameter λ is also heuristically determined.

We make the following observations from the results in these tables.

1) Of the different methods on the different data sets, K-means and PCAKM consume the least time. The time cost for DEC is similar to the other iterative methods, such as OCMKM, and the reason for this is that they need to iterate several times.
2) Although some methods have the same computational complexities, as mentioned in Section V-B, the time cost in real applications is different. This may be caused by the fact that different methods need different numbers of iterations to guarantee convergence, and the eigen-decomposition of different matrices has different time costs.

F. Clustering With Different Parameters

In this section, we will provide results with different λ, since it is the most important parameter in our framework.

As mentioned in Section V-C, we determine this parameter in two levels. At the top level, we determine it in a wider range with larger step length and choose a suitable interval. At the lower level, we vary λ within this interval and choose the optimal value. This strategy decreases the cost of determining a suitable parameter.

To show the influence of λ, we conduct experiments on three data sets, i.e., Umist, Coil, and Isolet. With the same setting as previously given in Section VI-C and different λ, we report the mean clustering accuracy in Fig. 5. On the top plane, the range of λ and the length of the step is large. On the

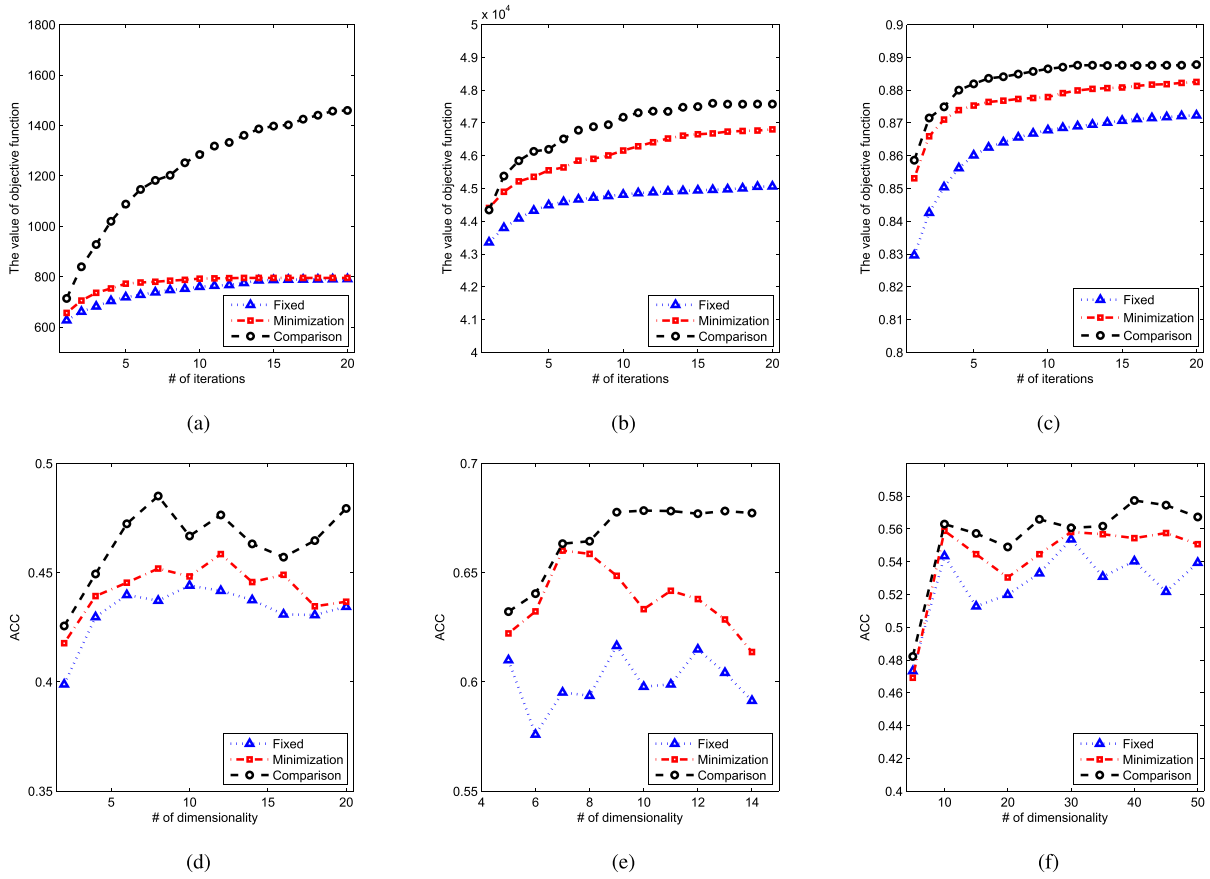


Fig. 4. Objective function and clustering accuracy of different updating rules on different data sets. The reduced dimensionality is set as $c-1$. (a) Objective values on Umist. (b) Objective values on Coil. (c) Objective values on Isolet. (d) Clustering accuracy on Umist. (e) Clustering accuracy on Coil. (f) Clustering accuracy on Isolet.

TABLE IV
COMPUTATIONAL TIME OF DIFFERENT METHODS ON COIL DATA WITH DIFFERENT REDUCED DIMENSIONALITY. (MEAN ± STD/%)

	K-means	PCAKM	LDAKM	OCMKM	MMCKM	OLSDAKM	DEC
6	1.67±0.51	0.52±0.03	63.82±0.39	61.15±1.97	61.84±2.39	190.84±4.84	62.03±1.88
8	1.62±0.53	0.59±0.05	63.92±0.36	61.81±2.48	62.51±1.69	163.38±3.39	63.92±1.55
10	1.46±0.34	0.57±0.02	64.27±0.51	62.79±2.21	64.38±1.61	146.27±2.41	65.10±1.40
12	1.53±0.57	0.56±0.02	64.41±0.46	64.08±0.99	65.49±1.34	146.94±2.42	67.17±1.33
14	1.47±0.35	0.57±0.03	64.97±0.44	66.86±2.43	67.42±2.13	166.17±2.81	69.75±1.29

TABLE V
COMPUTATIONAL TIME OF DIFFERENT METHODS ON COREL DATA WITH DIFFERENT REDUCED DIMENSIONALITY. (MEAN ± STD/%)

	K-means	PCAKM	LDAKM	OCMKM	MMCKM	OLSDAKM	DEC
2	0.47±0.09	0.33±0.11	30.06±1.81	101.27±9.05	95.13±8.90	145.69±11.88	92.55±8.49
4	0.48±0.15	0.35±0.09	31.07±1.37	103.90±8.34	99.73±8.73	148.07±9.63	95.09±9.56
6	0.49±0.12	0.39±0.12	32.76±0.96	107.24±9.96	105.24±9.42	149.86±9.56	98.02±7.47
8	0.50±0.21	0.41±0.08	33.60±1.33	108.05±9.03	108.25±7.04	150.47±10.29	101.30±8.21
10	0.55±0.11	0.43±0.10	34.18±1.59	109.33±11.57	114.69±8.07	155.10±9.65	104.07±8.82

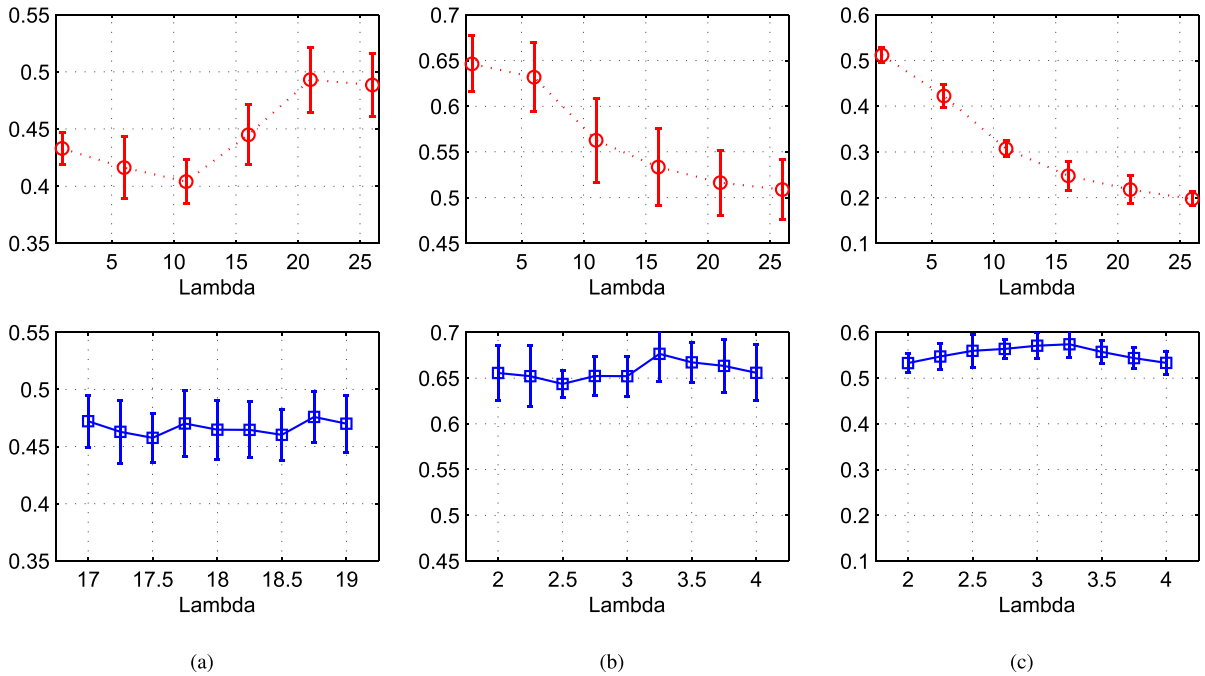
lower plane, we vary λ within a smaller range and provided the corresponding clustering accuracy.

As can be observed from the results in Fig. 5, we see that λ dominates the performance of our approach. If it is not suitably selected, performance will be degraded. The efficiency of DEC is also demonstrated, since DEC includes many well-known methods as special cases. We can also see from the bottom plane that when λ varies within a small range, performance does not change drastically. This is helpful for determining this parameter.

TABLE VI

COMPUTATIONAL TIME OF DIFFERENT METHODS ON POLLEN DATA WITH DIFFERENT REDUCED DIMENSIONALITY. (MEAN ± STD/%)

	K-means	PCAKM	LDAKM	OCMKM	MMCKM	OLSDAKM	DEC
4	0.26±0.08	0.95±0.03	5.47±0.15	29.85±0.51	28.74±0.76	27.51±0.38	26.55±0.70
8	0.27±0.07	0.95±0.04	5.50±0.06	30.03±0.53	29.05±0.35	27.96±0.39	26.61±0.53
12	0.22±0.06	0.96±0.03	5.56±0.04	29.86±0.51	29.17±0.49	27.92±0.38	27.85±0.47
16	0.25±0.05	0.95±0.04	5.57±0.06	30.05±0.54	29.24±0.48	27.75±0.32	28.07±0.45
20	0.27±0.11	0.95±0.02	5.53±0.06	30.03±0.52	29.41±0.56	27.68±0.48	28.18±0.53



(a) (b) (c)

Fig. 5. Clustering accuracy of proposed method on different data sets with different parameter λ. The reduced dimensionality is set as $c-1$. (a) Top: clustering accuracy on Umist when $\lambda \in \{1, 6, 11, \ldots, 26\}$. Bottom: clustering accuracy on Umist when $\lambda \in \{17, 17.25, 17.5, \ldots, 19\}$. (b) Top: clustering accuracy on Coil when $\lambda \in \{1, 6, 11, \ldots, 26\}$. Bottom: clustering accuracy on Coil when $\lambda \in \{2, 2.25, 2.5, \ldots, 4\}$. (c) Top: clustering accuracy on Isolet when $\lambda \in \{1, 6, 11, \ldots, 26\}$. Bottom: clustering accuracy on Isolet when $\lambda \in \{2, 2.25, 2.5, \ldots, 4\}$.

VII. CONCLUSION

In this paper, we have proposed an efficient framework, DEC, for clustering high-dimensional data. In contrast to traditional approaches, we join dimensionality reduction and clustering into a unified framework. Several traditional methods are viewed within this framework and are considered as special cases of our model. The convergence behavior, computational complexity, and parameter determination problem have also been analyzed. A significant number of experimental results have been proposed to show the efficiency of DEC. Further research will include the extension of DEC to supervised cases.

REFERENCES

[1] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[2] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2001, pp. 849–856.

[3] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum margin clustering," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2004.

[4] F. Nie, Z. Zeng, I. W. Tsang, D. Xu, and C. Zhang, "Spectral embedded clustering: A framework for in-sample and out-of-sample spectral clustering," *IEEE Trans. Neural Netw.*, vol. 22, no. 11, pp. 1796–1808, Nov. 2011.

[5] C. Hou, F. Nie, D. Yi, and Y. Wu, "Efficient image classification via multiple rank regression," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 340–352, Jan. 2013.

[6] C. Hou, F. Nie, C. Zhang, D. Yi, and Y. Wu, "Multiple rank multi-linear SVM for matrix data classification," *Pattern Recognit.*, vol. 47, no. 1, pp. 454–469, 2014.

[7] J. Li and D. Tao, "Simple exponential family PCA," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 3, pp. 485–497, Mar. 2013.

[8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2000.

[9] H. Park, M. Jeon, and J. B. Rosen, "Lower dimensional representation of text data based on centroids and least squares," *BIT Numer. Math.*, vol. 43, no. 2, pp. 427–448, 2003.

[10] X. R. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 157–165, Jan. 2006.

[11] F. Nie, S. Xiang, Y. Liu, C. Hou, and C. Zhang, "Orthogonal vs. uncorrelated least squares discriminant analysis for feature extraction," *Pattern Recognit. Lett.*, vol. 33, no. 5, pp. 485–491, 2012.

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

[13] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, Dec. 2000.
[14] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.
[15] S. Xiang, F. Nie, C. Zhang, and C. Zhang, "Nonlinear dimensionality reduction with local spline embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1285–1298, Sep. 2009.
[16] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.
[17] C. Hou, C. Zhang, Y. Wu, and F. Nie, "Multiple view semi-supervised dimensionality reduction," *Pattern Recognit.*, vol. 43, no. 3, pp. 720–730, 2010.
[18] J. Chen, Z. Ma, and Y. Liu, "Local coordinates alignment with global preservation for dimensionality reduction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 1, pp. 106–117, Jan. 2013.
[19] Y. Pang, Z. Ji, P. Jing, and X. Li, "Ranking graph embedding for learning to rerank," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 8, pp. 1292–1303, Aug. 2013.
[20] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: A review," *ACM SIGKDD Explorations Newslett.*, vol. 6, no. 1, pp. 90–105, 2004.
[21] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 1998, pp. 94–105.
[22] C.-H. Cheng, A. W. Fu, and Y. Zhang, "Entropy-based subspace clustering for mining numerical data," in *Proc. 5th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 1999, pp. 84–93.
[23] S. Goil, H. Nagesh, and A. Choudhary, "MAFIA: Efficient and scalable subspace clustering for very large data sets," Dept. Electr. Comput. Eng., Northwestern Univ., Evanston, IL, USA, Tech. Rep. CPDC-TR-9906-010, 1999.
[24] B. Liu, Y. Xia, and P. S. Yu, "Clustering through decision tree construction," in *Proc. 9th Int. Conf. Inform. Knowl. Manage.*, Nov. 2000, pp. 20–29.
[25] C. M. Procopiuc, M. Jones, P. K. Agarwal, and T. M. Murali, "A Monte Carlo algorithm for fast projective clustering," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2002, pp. 418–427.
[26] C. C. Aggarwal, C. Procopiuc, J. L. Wolf, P. S. Yu, and J. S. Park, "Fast algorithms for projected clustering," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 1999, pp. 61–72.
[27] C. C. Aggarwal and P. S. Yu, "Finding generalized projected clusters in high dimensional spaces," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2000, pp. 70–81.
[28] J. Yang, W. Wang, H. Wang, and P. Yu, "δ-clusters: Capturing subspace correlation in a large data set," in *Proc. 18th Int. Conf. Data Eng.*, 2002, pp. 517–528.
[29] J. H. Friedman and J. J. Meulman, "Clustering objects on subsets of attributes," *J. Roy. Statist. Soc.*, vol. 66, no. 4, pp. 815–849, 2004.
[30] C. Ding, X. He, H. Zha, and H. D. Simon, "Adaptive dimension reduction for clustering high dimensional data," in *Proc. ICDM*, 2002, pp. 147–154.
[31] C. Ding and T. Li, "Adaptive dimension reduction using discriminant analysis and K-means clustering," in *Proc. ICML*, 2007, pp. 521–528.
[32] F. De La Torre and T. Kanade, "Discriminative cluster analysis," in *Proc. ICML*, 2006, pp. 241–248.
[33] J. Ye, Z. Zhao, and H. Liu, "Adaptive distance metric learning for clustering," in *Proc. IEEE CVPR*, Jun. 2007, pp. 1–7.
[34] J. Ye, Z. Zhao, and M. Wu, "Discriminative K-means for clustering," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2007.
[35] D. Niu, J. G. Dy, and M. I. Jordan, "Dimensionality reduction for spectral clustering," in *Proc. Int. Conf. Artif. Intell. Statist.*, vol. 15. 2011, pp. 552–560.
[36] Q. Gu and J. Zhou, "Subspace maximum margin clustering," in *Proc. CIKM*, Nov. 2009, pp. 1337–1346.
[37] C. Domeniconi, D. Papadopoulos, D. Gunopulos, and S. Ma, "Subspace clustering of high dimensional data," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, Apr. 2004, pp. 517–521.
[38] D. Wang, F. Nie, and H. Huang, "Unsupervised feature selection via unified trace ratio formulation and K-means clustering (track)," in *Proc. Eur. Conf. Mach. Learn. Principles Pract. Knowl. Discovery Databases (ECML PKDD)*, Nancy, France, 2014.
[39] C. Hou, F. Nie, C. Zhang, and Y. Wu, "Learning a subspace for face image clustering via trace ratio criterion," *Opt. Eng.*, vol. 48, no. 6, p. 060501, 2009.
[40] T. Li, S. Ma, and M. Ogihara, "Document clustering via adaptive subspace iteration," in *Proc. 27th Annu. Int. ACM SIGIR Conf. Res. Develop. Inform. Retr.*, Jul. 2004, pp. 218–225.
[41] R. W. Sembiring, S. Sembiring, and J. M. Zain, "An efficient dimensional reduction method for data clustering," *Bull. Math.*, vol. 4, no. 1, pp. 43–58, 2012.
[42] F. Nie, D. Xu, and X. Li, "Initialization independent clustering with actively self-training method," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 1, pp. 17–27, Feb. 2012.
[43] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*. Manchester, U.K.: Manchester Univ. Press, 1992.
[44] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu, "Semi-supervised SVM batch mode active learning for image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–7.
[45] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. New York, NY, USA: Dover, Jul. 1998.

Chenping Hou (M'12) received the B.S. and Ph.D. degrees in applied mathematics from the National University of Defense Technology, Changsha, China, in 2004 and 2009, respectively.

He is currently an Associate Professor with the College of Science, National University of Defense Technology, Changsha. His current research interests include pattern recognition, machine learning, data mining, and computer vision.

Dr. Hou is a member of the Association for Computing Machinery.

Feiping Nie received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2009.

He is currently a Research Assistant Professor with the University of Texas at Arlington, Arlington, TX, USA. His current research interests include machine learning and its application fields, such as pattern recognition, data mining, computer vision, image processing, and information retrieval.

Dongyun Yi received the B.S. degree from Nankai University, Tianjin, China, and the M.S. and Ph.D. degrees from the National University of Defense Technology, Changsha, China.

He was a Visiting Researcher with the University of Warwick, Coventry, U.K., in 2008. He is a Professor with the College of Science, National University of Defense Technology. His current research interests include statistics, systems science, and data mining.

Dacheng Tao (M'07–SM'12) is a Professor of Computer Science with the Centre for Quantum Computation and Intelligent Systems and the Faculty of Engineering and Information Technology, University of Technology, Sydney, NSW, Australia. He mainly applies statistics and mathematics for data analysis problems in data mining, computer vision, machine learning, multimedia, and video surveillance. He has authored and co-authored more than 100 scientific articles at top venues, including the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the Conference on Neural Information Processing Systems, the International Conference on Machine Learning, the International Conference on Artificial Intelligence and Statistics, the IEEE International Conference on Data Mining series (ICDM), the IEEE International Conference on Computer Vision and Pattern Recognition, the International Conference on Computer Vision, the European Conference on Computer Vision, *ACM Transactions on Knowledge Discovery from Data*, ACM Multimedia conference, and ACM Conference on Knowledge Discovery and Data Mining.

Prof. Tao was a recipient of the Best Theory/Algorithm Paper Runner Up Award in the IEEE ICDM'07 and the Best Student Paper Award in the IEEE ICDM'13.