Image Clustering Using Local Discriminant Models and Global Integration

Article *in* IEEE Transactions on Image Processing · October 2010

DOI: 10.1109/TIP.2010.2049235 · Source: PubMed

Yi Yang
Carnegie Mellon University
412 PUBLICATIONS **24,475** CITATIONS
SEE PROFILE

Dong Xu
Tongji University
191 PUBLICATIONS **19,835** CITATIONS
SEE PROFILE

Feiping Nie
University of Texas at Arlington
435 PUBLICATIONS **19,187** CITATIONS
SEE PROFILE

Yueting Zhuang
Zhejiang University
386 PUBLICATIONS **8,750** CITATIONS
SEE PROFILE

Image Clustering using Local Discriminant Models and Global Integration

Yi Yang[1], Dong Xu[2], Feiping Nie[2], Shuicheng Yan[3], YueTing Zhuang[1]

[1]College of Computer Science, Zhejiang University, Hangzhou, China, 310027

[2]School of Computer Engineering, Nanyang Technological University, Singapore, 639798

[3]Department of Electrical and Computer Engineering, National University of Singapore, Singapore, 117576

Abstract

In this paper, we propose a new image clustering algorithm, referred to as Clustering using Local Discriminant Models and Global Integration (LDMGI). To deal with the data points sampled from a nonlinear manifold, for each data point, we construct a local clique comprising this data point and its neighboring data points. Inspired by the Fisher criterion, we use a local discriminant model for each local clique to evaluate the clustering performance of samples within the local clique. To obtain the clustering result, we further propose a unified objective function to globally integrate the local models of all the local cliques. With the unified objective function, spectral relaxation and spectral rotation are used to obtain the binary cluster indicator matrix for all the samples. We show that LDMGI shares a similar objective function with the Spectral Clustering (SC) algorithms, *e.g.*, Normalized Cut (NCut). In contrast to NCut in which the Laplacian matrix is directly calculated based on a Gaussian function, a new Laplacian matrix is learnt in LDMGI by exploiting both manifold structure and local discriminant information. We also prove that K-means and Discriminative K-means (DisKmeans) are both special cases of LDMGI. Extensive experiments on six benchmark image datasets demonstrate the effectiveness of LDMGI. We observe in the experiments that LDMGI is more robust to algorithmic parameter, when compared with NCut. Thus LDMGI is more appealing for the real image clustering applications in which the ground truth is generally not available for tuning algorithmic parameters.

Index Terms

Image Clustering, Local Discriminant Model, K-means Clustering, Spectral Clustering

I. Introduction

In the past decades, a large family of clustering methods were proposed to partition data points into clusters such that the data points within the same group are similar to each other, while the data points in different groups are dissimilar. Many clustering methods have been successfully used for image clustering to better organize, represent and browse images as well as to improve the performances of related applications, such as Content Based Image Retrieval (CBIR), image annotation, and image indexing [6].

Many works [4] [8] [11] [19] [44] have demonstrated that the performances of CBIR systems can be improved by adopting image clustering methods. For example, to improve the performance of CBIR, images are grouped into clusters in [12] by using the classical clustering algorithm K-means as a pre-processing technique. Clustering algorithms were successfully applied in automatic image annotation systems [16] [20] [30] as well. In [16], the personal album images were first clustered and then a web-based annotation method was developed to obtain the conceptual labels for image clusters, followed by a graph-based semi-supervised learning method to propagate the conceptual labels to the whole photo album. In [17] and [32], the image clustering algorithms were utilized to extract a small subset of representative images for representing the diverse views of the landmarks. It has also been reported that image clustering is helpful for efficient indexing of high-dimensional images [40].

K-means has been frequently used for image clustering because of its simplicity. The traditional K-means clustering algorithm iteratively assigns each data point to the cluster with the closest centroid and updates the centroid of the samples in each cluster. However, the performance of K-means may drop significantly when the dimensionality of the image feature vector is high [5]. To address this problem, a straightforward way is to project the high-dimensional data into a low-dimensional subspace through subspace learning algorithms, *e.g.,* Principal Components Analysis (PCA) and then perform K-means in the lower-dimensional subspace. In recent years, some researchers have shown that Linear Discriminant Analysis (LDA) [5] [36] [42] is a better choice than PCA for data clustering because discriminant information is utilized in LDA. Those researchers suggested simultaneously perform clustering and subspace learning by integrating K-means and LDA into a joint framework [5] [36] [42]. The initial works [5] [36] iteratively employed K-means to generate the cluster labels for LDA and exploited LDA to select the most discriminative subspace for K-means clustering. More recently, Ye *et al.* proposed a new discriminative K-means (DisKmeans) algorithm to simplify the iterative procedures as a trace maximization problem [42], which can be directly solved. It has been experimentally demonstrated that K-means with LDA outperforms traditional K-means and K-means with PCA for data clustering, because of the utilization of the discriminant information [5] [36] [42]. However, manifold information is not considered in K-means and DisKmeans. Consequently, K-means and DisKmeans can not effectively cluster the data points sampled from nonlinear manifolds.

Beside discriminant information, the manifold structure of data is another crucial property to be considered in data clustering. The well-known spectral clustering (SC) algorithm normalized cut (NCut) [29] and its extension k-way NCut [43] have achieved promising clustering performances in image segmentation and many other applications, in which the similarities among neighboring points are considered to construct a graph for data clustering. However, the Gaussian function is directly adopted to compute a Laplacian matrix in NCut for data clustering, which may be sensitive to the bandwidth parameter in many applications, as reported in [37]. Recently, Local Learning based Clustering (LLC) [38] was proposed to learn a Laplacian matrix for data clustering, which is based on the assumption that the cluster label of each data point can be well predicted by its neighboring data points with a kernel rigid regression function. LLC can be regarded as an SC algorithm [38] as well. While the manifold information is utilized in NCut and LLC, the discriminant information is not sufficiently exploited.

To utilize both manifold information and discriminant information, we propose a new clustering method, namely Clustering using Local Discriminant Models and Global Integration (LDMGI), for image clustering. Similarly to [28] [29] [38] [46], for each data point, we consider a local clique comprising this data point as well as its neighboring data points. We construct a local discriminant model for each local clique and evaluate the clustering performance of samples in the clique with a variant of the Fisher criterion (referred to as the local discriminant score), which is defined as the ratio of the between-cluster scatter to the total scatter. A larger local discriminant score indicates that the samples in the local clique from different clusters are better separated. To globally integrate the local models, we further propose a unified objective function to maximize the sum of local discriminant scores from all the local cliques. We observe that LDMGI shares an objective function similar to other spectral clustering algorithms (*e.g.*, NCut). Eigenvalue decomposition method is then used to obtain the relaxed continuous valued solution of the scaled cluster assignment matrix, which is further discretized to get the binary cluster assignment matrix for all the samples. When compared with NCut which directly calculates the Laplacian matrix with the Gaussian function, LDMGI learns a new Laplacian matrix by using both manifold structure and local discriminant information, making it more robust for data clustering.

We theoretically prove that K-means and DisKmeans are both special cases of LDMGI. When compared with K-means and DisKmeans, LDMGI can effectively deal with the data points sampled from a nonlinear manifold. It is worth mentioning that K-means clustering and the spectral clustering algorithms are traditionally regarded as two different types of clustering techniques. However, an interesting observation in this paper is that K-means, DisKmeans and the spectral clustering algorithms are connected via LDMGI. That is K-means and DisKmeans are special cases of LDMGI, while LDMGI is a spectral clustering algorithm. While the connection between K-means and NCut has been discovered in [7] in terms of a kernel view, this work presents another new perspective to understand and explain the relationships between K-means (or DisKmeans) and spectral clustering algorithms.

Extensive experiments on six benchmark image datasets (*i.e.*, one object image dataset COIL-20, one digital number image dataset USPS, one human gait image dataset USF HumanID and three face image datasets UMIST, YALE-B and MSRA) demonstrate the effectiveness of LDMGI. When compared with NCut which is sensitive to the bandwidth parameter of the Gaussian function, LDMGI is robust to the algorithmic parameters. The ground truth is generally not available for tuning the parameters of clustering algorithms. Thus, LDMGI is more suitable for real clustering applications.

It is worthwhile to highlight the following aspects of this paper:

- We propose a new clustering algorithm LDMGI, which utilizes both manifold information and discriminant information for data clustering.
- We theoretically prove that K-means and DisKmeans are both special cases of LDMGI. We also show that LDMGI is a type of spectral clustering algorithm. Thus our work provides a new perspective for discovering and understanding the relationships between K-means (or DisKmeans) and other spectral clustering algorithms (*e.g.*, NCut).
- Our work outperforms several existing data clustering methods on 12 publicly available image datasets. LDMGI is also more robust to the parameter, when compared with NCut.

The rest of this paper is organized as follows. In Section II, we introduce the details of the proposed LDMGI algorithm. The connections between LDMGI and other existing clustering algorithms are discussed in Section III. Experimental results are reported in Section IV. Finally, we conclude this paper in Section V.

II. Clustering using Local Discriminant Models and Global Integration

A. Local Discriminant Models

Let us define $\mathcal{X} = \{x_1, x_2, ..., x_n\}$ as the image data set to be clustered, where $x_i \in \mathbb{R}^m (1 \le i \le n)$ is the i-th data point and n is the total number of images. Clustering is to partition $\mathcal{X}$ into c clusters $\{C_j\}_{j=1}^c$. Denote the cluster assignment matrix by $Y = [y_1, y_2, ..., y_n]^T \in \{0,1\}^{n \times c}$, where $y_i \in \{0,1\}^{c \times 1} (1 \le i \le n)$ is the cluster assignment vector for the image x_i. The j-th element of y_i is 1 if $x_i \in C_j$, and 0 otherwise. We also denote $X = [x_1, x_2, ..., x_n]$ as the data matrix. For any d, we define $\mathbf{1}_d \in \mathbb{R}^d$ as a column vector with all its elements equal to 1, $H_d = I - \frac{1}{d}\mathbf{1}_d\mathbf{1}_d^T \in \mathbb{R}^{d \times d}$ as a matrix for centering the data by subtracting the mean of the data. It follows that $H_d = H_d^T = H_d H_d$. Following [43], instead of directly solving the cluster assignment matrix Y, we first define the scaled cluster assignment matrix G as follows:

$$G = [g_1, g_2, ..., g_n]^T = Y(Y^TY)^{-1/2}, \tag{1}$$

in which g_i is the scaled cluster assignment vector for image x_i. It is obvious that the j-th column of G is given by:

$$G_j = [\underbrace{0, ..., 0}_{\sum_{i=1}^{j-1} n_i}, \underbrace{1, ...1}_{n_j}, \underbrace{0, ..., 0}_{\sum_{i=j+1}^{c} n_i}]^T / \sqrt{n_j}, \tag{2}$$

where n_j is the number of samples in the j-th cluster C_j. Since Y^TY is a diagonal matrix, it is easy to verify that:

$$G^TG = (Y^TY)^{-1/2}Y^TY(Y^TY)^{-1/2} = I. \tag{3}$$

As shown in [9], the total scatter matrix S_t and the between-cluster scatter matrix S_b can be defined as:

$$\begin{aligned} S_t &= \sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T = \tilde{X}\tilde{X}^T & (4) \\ S_b &= \sum\nolimits_{i=1}^{c} n_i(\mu_i - \mu)(\mu_i - \mu)^T = \tilde{X}GG^T\tilde{X}^T, & (5) \end{aligned}$$

where μ_i is the mean of samples in the i-th cluster C_i, μ is the mean of all samples and $\tilde{X} = XH_n$ is the data matrix after subtracting the mean μ. Intuitively, to better cluster the data, the distance between data from different clusters should be as large as possible while the distance between data from the same cluster should be as small as possible. Inspired by the Fisher criterion [9], the optimal scaled cluster assignment matrix G^* can be obtained by maximizing the following objective function:

$$\begin{aligned} G^* &= \arg\max_G Tr[(S_t + \lambda I)^{-1}S_b] \\ &= \arg\max_G Tr\left[(\tilde{X}\tilde{X}^T + \lambda I)^{-1}\tilde{X}GG^T\tilde{X}^T\right] \\ &= \arg\max_G Tr\left[G^T\tilde{X}^T(\tilde{X}\tilde{X}^T + \lambda I)^{-1}\tilde{X}G\right], \end{aligned} \tag{6}$$

where the identity matrix I is added to avoid the singular value problem and $\lambda > 0$ is a regularization parameter. Considering that $G^T G = I$, we can also minimize the following objective function to obtain G^*:

$$G^* = \arg\min_{G} Tr\left[G^T G - G^T \tilde{X}^T (\tilde{X}\tilde{X}^T + \lambda I)^{-1} \tilde{X} G\right]. \tag{7}$$

Note that both the total scatter matrix S_t and the between-cluster scatter matrix S_b in Eq. (6) are defined according to Euclidean distances among the data. Thus, it may be not easy to partition the data sampled from the nonlinear manifold into the correct clusters with such a criterion. Previous manifold learning algorithms [1] [28] [46] and clustering algorithms [29] [38] [47] have demonstrated that it is beneficial to exploit local geometric properties for recovering the intrinsic manifold structure of data. Considering local structure of a manifold to be approximately linear [28], for each data point x_i, we consider a local clique $\mathcal{N}_k(x_i)$ comprising k data points including x_i and its $k-1$ nearest neighbors, and employ a local linear discriminant model to evaluate the clustering results for the data points in $\mathcal{N}_k(x_i)$. The overall clustering results can then be obtained by globally optimizing the local discriminant models of all the local cliques.

Let $F_i = \{i_0, i_1, i_2, ..., i_{k-1}\}$ be the index set of the samples in $\mathcal{N}_k(x_i)$, where we set $i_0 = i$. We define $X_i = [x_{i_0}, x_{i_1}, \ldots, x_{i_{k-1}}]$ as the local data matrix comprising all the data points in $\mathcal{N}_k(x_i)$. Given the scaled cluster assignment matrix G, we define $G_{(i)} = [g_{i_0}, g_{i_1}, \ldots, g_{i_{k-1}}]^T \in \mathbb{R}^{k \times c}$ as the local scaled cluster assignment matrix for the i-th clique, in which g_{i_j} is the scaled cluster assignment vector of $x_{i_j} \in \mathcal{N}_k(x_i)$ $(j = 0, \ldots, k-1)$. More specifically, $G_{(i)}$ is selected from the scaled cluster assignment matrix G, namely:

$$G_{(i)} = S_i^T G, \tag{8}$$

where $S_i \in \mathbb{B}^{n \times k}$ is the selection matrix with its element $(S_i)_{pq} = 1$, if $p = F_i\{q\}$; $(S_i)_{pq} = 0$, otherwise. Let $m_{i_p}|_{p=1}^{c}$ be the number of samples in the local clique $\mathcal{N}_k(x_i)$ which are from the p-th cluster. Without loss of generality, we assume that the samples in the local clique are reorganized such that the samples from the same cluster are put together. The j-th column of $G_{(i)}$ can be written as:

$$G_{(i)}^{j} = [\ \underbrace{0, ..., 0}_{\sum_{p=1}^{j-1} m_{i_p}}, \underbrace{1, ...1}_{m_{i_j}}, \underbrace{0, ..., 0}_{\sum_{p=j+1}^{c} m_{i_p}}\ ]^T / \sqrt{n_j}. \tag{9}$$

Thus we have:

$$G_{(i)} G_{(i)}^T = [G_{(i)}^1, G_{(i)}^2, ..., G_{(i)}^c][G_{(i)}^1, G_{(i)}^2, ..., G_{(i)}^c]^T = \begin{bmatrix} K_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & K_c \end{bmatrix}, \tag{10}$$

where $K_p \in R^{m_{i_p} \times m_{i_p}}$ is a square matrix with all its elements equal to $1/n_p$ $(1 \leq p \leq c)$. We define the local discriminant score for the i-th clique as follows:

$$\arg\max_{G_{(i)}} Tr\left[G_{(i)}^T \tilde{X_i}^T (\tilde{X}_i \tilde{X}_i^T + \lambda I)^{-1} \tilde{X}_i G_{(i)}\right], \tag{11}$$

where $\tilde{X}_i = X_i H_k$. As shown in Eq. (10), $G_{(i)} G_{(i)}^T$ is a block diagonal matrix. Similar to Eq. (5), $\tilde{X}_i G_{(i)} G_{(i)}^T \tilde{X}_i^T$ still measures the inter-cluster separability of samples within the local clique, and a detailed justification is provided in the Appendix. Thus, a larger local discriminant score indicates that the samples in the local clique from different clusters are better separated. Considering the number of samples in each local clique to be small, we impose a regularization term $Tr\left(G_{(i)}^T H_k G_{(i)}\right)$ to control the capacity of the local discriminant model. Similar to Eq. (7), we have the following objective function for the i-th local clique:

$$\arg\min_{G_{(i)}} Tr\left\{G_{(i)}^T H_k G_{(i)} - G_{(i)}^T \tilde{X}_i{}^T (\tilde{X}_i\tilde{X}_i^T + \lambda I)^{-1}\tilde{X}_i G_{(i)}\right\}. \tag{12}$$

In the following, we show that the objective function in Eq. (12) with the regularization term can be rewritten to Eq. (13) by using *lemma*-1 and *Theorem*-1, which can be more efficiently solved when compared with Eq. (11).

lemma 1: For any matrix A, we have $A(A^TA+\lambda I)^{-1} = (AA^T+\lambda I)^{-1}A$.

Proof: It is obvious that $(AA^T+\lambda I)A = A(A^TA+\lambda I)$. Then, we have:

$$\begin{aligned} &A(A^TA+\lambda I)^{-1} = (AA^T+\lambda I)^{-1}(AA^T+\lambda I)A(A^TA+\lambda I)^{-1} \\ &= (AA^T+\lambda I)^{-1}A(A^TA+\lambda I)(A^TA+\lambda I)^{-1} = (AA^T+\lambda I)^{-1}A. \end{aligned}$$

■

Theorem 1: The optimization problem of Eq.(12) is equivalent to the following optimization problem:

$$\arg\min_{G_{(i)}} Tr\left\{G_{(i)}^T \left[H_k(\tilde{X}_i^T\tilde{X}_i + \lambda I)^{-1}H_k\right] G_{(i)}\right\}. \tag{13}$$

Proof: According to *lemma*-1 and $\tilde{X}_i H_k = \tilde{X}_i$, we have:

$$\begin{aligned} &G_{(i)}^T[\tilde{X}_i^T(\tilde{X}_i\tilde{X}_i^T+\lambda I)^{-1}\tilde{X}_i]G_{(i)} \\ &= G_{(i)}^T[H_k\tilde{X}_i^T(\tilde{X}_i\tilde{X}_i^T+\lambda I)^{-1}\tilde{X}_iH_k]G_{(i)} \\ &= G_{(i)}^T[H_k(\tilde{X}_i^T\tilde{X}_i+\lambda I)^{-1}\tilde{X}_i^T\tilde{X}_iH_k]G_{(i)} \\ &= G_{(i)}^T[H_k(\tilde{X}_i^T\tilde{X}_i+\lambda I)^{-1}(\tilde{X}_i^T\tilde{X}_i+\lambda I-\lambda I)H_k]G_{(i)} \\ &= G_{(i)}^T[H_k-\lambda H_k(\tilde{X}_i^T\tilde{X}_i+\lambda I)^{-1}H_k]G_{(i)}. \end{aligned}$$

Therefore,the optimization problem of Eq. (12) is equivalent to the problem of Eq. (13). ■

Since k is the number of samples in each local clique, it is usually much smaller than the dimension of data points. $\tilde{X}_i^T\tilde{X}_i$ is of size $k \times k$, so it is much faster to calculate the inverse of $\tilde{X}_i^T\tilde{X}_i + \lambda I$ in Eq. (13) in comparison to the inverse of $\tilde{X}_i\tilde{X}_i^T + \lambda I$ in Eq. (12). Additionally, with Eq. (13), the proposed clustering algorithm LDMGI can be easily extended to its kernel version, which will be studied in the future. The objective function of Eq. (13) can be also rewritten as:

$$\arg\min_{G_{(i)}} Tr\left[G_{(i)}^T L_i G_{(i)}\right], \tag{14}$$

$$\text{where} \qquad L_i = H_k(\tilde{X}_i^T\tilde{X}_i + \lambda I)^{-1}H_k. \tag{15}$$

B. Global Integration

To obtain the optimal G, we expect that the data points are well separated in all the local cliques. Thus, we propose the following objective function to globally integrate the local discriminant models by summing Eq. (14) over all the local cliques, which can be formulated as:

$$\arg\min_{G} \sum\nolimits_{i=1}^{n} Tr\left[G_{(i)}^{T} L_i G_{(i)}\right]. \tag{16}$$

Note that the local scaled cluster assignment matrix $G_{(i)}$ is selected from the scaled assignment matrix G, *i.e.*, $G_{(i)} = S_i^T G$. Then, the objective function in Eq. (16) can be rewritten as:

$$\begin{aligned} &\arg\min_{G} \sum\nolimits_{i=1}^{n} Tr\left[G^T S_i L_i S_i^T G\right] \\ &= \arg\min_{G} Tr\left[G^T \left(\sum\nolimits_{i=1}^{n} S_i L_i S_i^T\right) G\right]. \end{aligned} \tag{17}$$

Let us define:

$$L = \sum\nolimits_{i=1}^{n} S_i L_i S_i^T = [S_1, S_2, ..., S_n] \begin{bmatrix} L_1 & & \\ & ... & \\ & & L_n \end{bmatrix} [S_1, S_2, ..., S_n]^T. \tag{18}$$

The objective function of LDMGI becomes:

$$\arg\min_{G} Tr\left(G^T L G\right). \tag{19}$$

As shown in Eq. (1) and Eq. (3), we have two constraints for G, which are G is a scaled cluster assignment matrix, i.e. $G = Y(Y^TY)^{-1/2}$, and $G^TG = I$. Then the objective function of LDMGI can be written as:

$$\arg\min_{G} Tr(G^T L G) \quad \text{s.t.} \quad G^TG = I \quad \text{and} \quad G = Y(Y^TY)^{-1/2}. \tag{20}$$

In the following *Theorem*-2, we prove that the matrix L defined in Eq. (18) is a Laplacian matrix. Therefore, the objective function of LDMGI in Eq. (20) has a similar form to NCut [29]. Ncut directly adopts the Gaussian function to compute a Laplacian matrix for clustering, which is sensitive to the bandwidth parameter of the Gaussian function [37]. In contrast, we learn a new Laplacian matrix by employing discriminant information and manifold structure, which is more robust for data clustering.

lemma 2: Given a positive semi-definite matrix C, DCD^T is a positive semi-definite matrix for any matrix D.

Proof: C is a positive semi-definite matrix, so we can write C as $C = M^TM$. Then, we have $DCD^T = DM^TMD^T = (MD^T)^T(MD^T)$. Therefore, DCD^T is a positive semi-definite matrix. ■

Theorem 2: The matrix L defined in Eq. (18) is a Laplacian matrix.

Proof: $L_i = H_k(\tilde{X}_i^T \tilde{X}_i + \lambda I)^{-1} H_k$. It is obvious $(\tilde{X}_i^T \tilde{X}_i + \lambda I)^{-1}$ is a positive semi-definite matrix. According to *lemma*-2, L_i is a positive semi-definite matrix. In addition, we have $H\mathbf{1}_k = \left(I - \frac{1}{k}\mathbf{1}_k\mathbf{1}_k^T\right)\mathbf{1}_k = \mathbf{0}$. Then, it can be shown that $L_i\mathbf{1}_k = \mathbf{0}$. Therefore, we conclude that L_i is a Laplacian matrix.

Let us define $A = \begin{bmatrix} L_1 & & \\ & ... & \\ & & L_n \end{bmatrix}$ and $S = [S_1, S_2, ..., S_n]$, then $L = SAS^T$. L_i is a positive semi-definite matrix, so $q_i^T L_i q_i \geqslant 0$, $\forall\, q_i \in \mathbb{R}^k$. Let us denote $z = nk$. We have $q^T A q = \sum_i q_i^T L_i q_i \geqslant 0$, $\forall q = [q_1^T, q_2^T, ..., q_n^T]^T \in$

$\mathbb{R}^z$. Thus, A is a positive semi-definite matrix. According to *lemma*-2, $L = SAS^T$ is a positive semi-definite matrix. In addition, we have $A\mathbf{1}_z = \mathbf{0}$ and $S^T\mathbf{1}_n = \mathbf{1}_z$. Then, $L\mathbf{1}_n = SAS^T\mathbf{1}_n = SA\mathbf{1}_z = \mathbf{0}$. Thus, we can conclude that L is a Laplacian matrix. ■

C. Relaxation and Discretization

As shown in [29], the objective function in Eq. (20) is NP hard, because the scaled cluster assignment matrix G is constrained as $G = Y(Y^TY)^{-1/2}$. Following [29], we first remove this constraint and relax G to the continuous-valued domain. That is, the objective function is simplified as $\min\limits_{G^TG=I} Tr\left(G^TLG\right)$. Then eigenvalue decomposition method can be used to solve the optimization problem:

$$Lu_i = \lambda_i u_i, \tag{21}$$

where $\lambda_0 < \lambda_1 <, ..., < \lambda_n$ are the eigenvalues and $u_i|_{i=1}^n$ are the corresponding eigenvectors. It is obvious that there is a trivial solution with $\lambda_0 = 0$ and $u_0 = \mathbf{1}_n$. After removing this trivial solution, we have the optimal solution G^* of the objective function in Eq. (17):

$$G^* = [u_1, u_2, \ldots, u_{c-1}, u_c]. \tag{22}$$

To obtain the cluster assignment matrix $Y \in \mathbb{B}^{n\times c}$, we then discretize G^* using spectral rotation [43]. For any orthogonal matrix (referred to as a rotation matrix) $R \in \mathbb{R}^{c\times c}$, it is easy to show that $\hat{G}^* = G^*R$ is the solution of the optimization problem $\min\limits_{G^TG=I} Tr\left(G^TLG\right)$ as well. Thus the global optimal solution G^* is not unique. We therefore compute the optimal binary valued cluster assignment matrix Y which is closest to G^*R. Following [43], we define a mapping function for G^* as:

$$Y^* = f^{-1}(G^*) = Diag(G^*G^{*T})^{-1/2}G^*, \tag{23}$$

where $Diag(G^*G^{*T})$ is the diagonal matrix with the same size and the same diagonal elements of G^*G^{*T}. In [43], it has been proven that $f^{-1}(G^*R) = Y^*R$ and Y^*R is the optimal solution to the optimization problem in 20. Then, the optimal binary valued solution Y and the rotation matrix R can be simultaneously obtained by solving the following optimization problem:

$$\begin{aligned} &\arg \min_{Y\in\mathbb{B}^{n\times c}, R\in\mathbb{R}^{c\times c}} \|Y - Y^*R\|^2 \\ &\text{s.t.} \quad Y\mathbf{1}_c = \mathbf{1}_n, R^TR = I. \end{aligned} \tag{24}$$

We iteratively optimize Y and R until convergence. For more details, please refer to [43]. We summarize our proposed LDMGI algorithm in the following Procedure-1.

1) For each data point x_i, construct its local clique $\mathcal{N}_k(x_i), i = 1, \ldots, n$;
2) Compute $L_i|_{i=1}^n$ according to Eq. (15);
3) Compute L using Eq. (18);

4) Solve the eigenvalue decomposition problem in Eq. (21) to obtain the optimal $G^* = [u_1, u_2, ..., u_c]$. The trivial solution $u_0 = \mathbf{1}_n$ corresponding to the eigenvalue $\lambda_0 = 0$ is removed;
5) Compute Y^* according to Eq. (23) and solve the optimization problem shown in Eq. (24) using spectral rotation [43].

Procedure-1: The LDMGI Clustering Algorithm.

D. *Complexity Analysis*

In LDMGI, we first need to compute the matrix L_i in Eq. (15), and the time complexity is $O(k^3)$. Then the total complexity to calculate n matrices $L_i|_{i=1}^n$ is $O(nk^3)$. Note that the elements of S_i are binary values (*i.e.*, 0 or 1), it is therefore unnecessary to conduct the "multiplication" operation to calculate $S_i L_i S_i^T$ in Eq. (18). Ignoring the complexity of "addition" operations in Eq. (18), the time complexity to calculate L is still $O(nk^3)$, which is linear with respect to the number of samples n. The complexity of the eigenvalue decomposition for the matrix $L \in \mathbb{R}^{n\times n}$ is $O(n^3)$. Since the total number of samples n is much larger than k which specifies the number of nearest neighbors, the total complexity of LDMGI is $O(n^3)$.

III. Connections with previous clustering algorithms

The most well-known and widely-used clustering algorithms are K-means clustering and spectral clustering. In this section, we discuss the connections between our LDMGI and some representative clustering techniques of K-means and spectral clustering.

A. *Connection with K-means*

K-means is a simple and frequently used clustering algorithm. The K-means clustering algorithm partitions n data points into c clusters $\{C_j\}_{j=1}^c$ by minimizing the following objective function:

$$\arg\min_{C_j|_{j=1}^c} \sum_{j=1}^{c} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2, \tag{25}$$

where μ_j is the mean of samples in the j-th cluster C_j. The traditional K-means clustering algorithm uses an EM-like approach to iteratively assign each data point to the cluster with the closest centroid and update the centroid of the samples in each cluster. In [45], it is proven that the objective function of K-means can be rewritten as:

$$\arg\max_{G^T G = I} Tr(G^T \tilde{X}^T \tilde{X} G), \tag{26}$$

where $\tilde{X} = XH_n$ and G is the scaled cluster assignment matrix. The connection between K-means and LDMGI is discovered as follows.

Proposition 1: K-means is a special case of the proposed LDMGI clustering algorithm, when $k = n$ and $\lambda \to \infty$.

Proof: When $k = n$, we have $G_{(i)} = G$ and $\tilde{X}_i = \tilde{X}$. First, we prove $Tr(G^T H_n G) = c - 1$:

$$\begin{aligned}
& Tr\left(G^T H_n G\right) \\
&= Tr\left[G^T(I - \frac{1}{n}\mathbf{1_n 1_n^T})G\right] \\
&= Tr(G^T G) - \frac{1}{n}Tr(\mathbf{1}_n^T G G^T \mathbf{1}_n) \\
&= c - \frac{1}{n}Tr\left[\mathbf{1}_n^T Y(Y^T Y)^{-1} Y^T \mathbf{1}_n\right] \\
&= c - \frac{1}{n}Tr\left\{\left[\begin{array}{ccc} n_1, & \dots & ,n_c \end{array}\right]\left[\begin{array}{ccc} \frac{1}{n_1} & & \\ & \dots & \\ & & \frac{1}{n_c} \end{array}\right]\left[\begin{array}{c} n_1 \\ \dots \\ n_c \end{array}\right]\right\} \\
&= c - 1.
\end{aligned}$$

Therefore, the objective function in Eq. (12) is equivalent to the objective function in Eq. (11), when $k = n$. When $\lambda \rightarrow \infty$, the objective function in Eq. (11) can be rewritten as:

$$\begin{aligned}
& \arg \max_{G^T G = I} \sum\nolimits_{i=1}^{n} Tr\left[G^T \tilde{X}^T(\tilde{X}\tilde{X}^T + \lambda I)^{-1}\tilde{X}G\right] \\
&= \arg \max_{G^T G = I} Tr(G^T \tilde{X}^T \tilde{X} G).
\end{aligned}$$

Thus, K-means is a special case of LDMGI, when $k = n$ and $\lambda \rightarrow \infty$. ■

B. Connection with discriminative K-means (DisKmeans)

Recently, some researchers suggested to simultaneously perform K-means clustering and LDA by optimizing the following objective function [5] [36] [42]:

$$\begin{aligned}
& \arg \max_{W,G} Tr\left[\left(W^T(S_t + \gamma I)W\right)^{-1} W^T S_b W\right] \\
&= \arg \max_{W,G} Tr\left[\left(W^T(\tilde{X}\tilde{X}^T + \gamma I)W\right)^{-1} W^T \tilde{X} G G^T \tilde{X}^T W\right],
\end{aligned} \tag{27}$$

where W is the projection matrix, and G is the scaled cluster assignment matrix. The initial works [5] [36] iteratively optimize G and W by employing K-means to generate the cluster labels for LDA and using LDA to select the most discriminative subspace for K-means clustering. In [42], Ye *et al.* have proven that the objective function shown in Eq. (27) can be simplified to the following trace maximization problem:

$$\arg \max_{G^T G = I} Tr\left\{G^T\left[I - (I + \frac{1}{\gamma}\tilde{X}^T\tilde{X})^{-1}\right]G\right\}. \tag{28}$$

Instead of using the iterative procedure, Ye *et al.* proposed the DisKmeans clustering algorithm in which only G needs to be solved [5]. The connection between DisKmeans and LDMGI is as follows.

Proposition 2: DisKmeans [42] is a special case of LDMGI, when $k = n$.

Proof: Again, when $k = n$, we have $G_{(i)} = G$ and $\tilde{X}_i = \tilde{X}$. As shown in the proof of *Proposition*-1, we have $Tr(G^T H_n G) = c - 1$. Therefore, when $k = n - 1$, the objective function of LDMGI can be rewritten as:

$$\begin{aligned} &\arg \max_{G^T G=I} Tr \left[G^T \tilde{X}^T (\tilde{X}\tilde{X}^T + \lambda I)^{-1} \tilde{X} G \right] \\ =& \arg \max_{G^T G=I} Tr \left[G^T \tilde{X}^T (I + \frac{1}{\gamma} \tilde{X}\tilde{X}^T)^{-1} \tilde{X} G \right] \\ =& \arg \max_{G^T G=I} Tr \left\{ G^T \left[(I + \frac{1}{\gamma} \tilde{X}^T \tilde{X})^{-1} \tilde{X}^T \tilde{X} \right] G \right\} \\ =& \arg \max_{G^T G=I} Tr \left\{ G^T \left[\gamma (I + \frac{1}{\gamma} \tilde{X}^T \tilde{X})^{-1} (\frac{1}{\gamma} \tilde{X}^T \tilde{X} + I - I) \right] G \right\} \\ =& \arg \max_{G^T G=I} Tr \left\{ G^T \left[I - (I + \frac{1}{\gamma} \tilde{X}^T \tilde{X})^{-1} \right] G \right\}. \end{aligned} \tag{29}$$

In the first step, we set $\lambda = \gamma$, and *lemma*-1 is used in the second step. Therefore, DisKmeans [42] is a special case of LDMGI. ■

Both DisKmeans and LDMGI make use of discriminant information for clustering. However, manifold information is not utilized in DisKmeans. Thus, DisKmeans cannot effectively cope with data points sampled from a nonlinear manifold. In contrast, LDMGI employs both discriminant information and manifold structure, and it is more robust for data clustering.

C. Connection with spectral clustering

In recent years, several spectral clustering algorithms have been proposed [23] [29] [38]. These algorithms can be generally formulated as the following optimization problem:

$$\arg \min_{G^T G=I} Tr(G^T \hat{L} G), \tag{30}$$

where $\hat{L}$ is a Laplacian matrix. Different spectral clustering algorithms use different Laplacian matrices and the Laplacian matrix plays a key role in spectral clustering.

In the well-known NCut [29] and its extension k-way NCut [43], the Laplacian matrix is defined as:

$$\tilde{L} = I - D^{-1/2} A D^{-1/2}, \tag{31}$$

where D is a diagonal matrix with its diagonal elements defined as $D_{ii} = \sum_{j=1}^{n} A_{ij}$ for $i = 1, ..., n$ and the graph affinity matrix A is computed by using Gaussian function:

$$A_{ij} = \begin{cases} \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right) & x_i \text{ and } x_j \text{ are k-nearest neighbors;} \\ 0 & \text{otherwise,} \end{cases} \tag{32}$$

where σ is the bandwidth parameter.

More recently, Wu and Schölkopf proposed a local learning approach for data clustering (referred to as LLC here) [38]. Denote $\mathcal{N}_i = \{x_{i_1}, x_{i_2}, ..., x_{i_k}\}$ as the k-nearest neighbor set of x_i excluding x_i and $G = [g^1, g^2, ..., g^c] \in \mathbb{R}^{n \times c}$ as the scaled cluster assignment matrix, in which $g^l = [g_1^l, ..., g_n^l]^T \in \mathbb{R}^n$ $(1 \le l \le c)$ is the l-th column of G. LLC is based on the assumption that the cluster label g_i^l of each data point x_i $(1 \le i \le n)$ can be well predicted by a local model $o^l(x_i) = \sum_{x_j \in \mathcal{N}_i} \beta_{ij}^l K(x_i, x_j)$, where K is a kernel matrix, β_{ij}^l are the linear combination coefficients.

LLC adopted a two-step approach to learn a Laplacian matrix for data clustering. In the first step of LLC, a local model is trained for each data point x_i using the following kernel ridge regression function:

$$\arg\min_{\beta_i^l \in \mathbb{R}^k} \eta(\beta_i^l)^T \mathrm{K}_i \beta_i^l + \left\| \mathrm{K}_i \beta_i^l - g_{(i)}^l \right\|^2, \tag{33}$$

where $\eta > 0$ is a regularization parameter, $\beta_i^l = [\beta_{ii_1}^l, ...\beta_{ii_k}^l]^T$, $g_{(i)}^l = [g_{i_1}^l, ...g_{i_k}^l]^T$, and $\mathrm{K}_i \in \mathbb{R}^{k \times k}$ denotes the sub-kernel matrix, *i.e.* $\mathrm{K}_i = [K(x_u, x_v)]$ for $x_u, x_v \in \mathcal{N}_i$. In the second step, each locally trained model is used to predict the cluster labels of one data point only. Then the clustering results can be obtained by minimizing the sum of prediction errors of all the local models:

$$\sum\nolimits_{l=1}^{c} \sum\nolimits_{i=1}^{n} (g_i^l - o^l(x_i))^2. \tag{34}$$

Let us define $\alpha_i^T = k_i^T(\mathrm{K}_i + \eta I)^{-1}$ where $k_i = [K(x_i, x_{i_1}), ..., K(x_i, x_{i_k})]^T$ and $A = [a_{ij}] \in \mathbb{R}^{N \times N}$ which is constructed as follows: $\forall x_i, x_j$, if $x_j \in \mathcal{N}_i$, then a_{ij} is equal to the corresponding element of α_i; otherwise, a_{ij} equals zero. It has been proven in [38] that the optimization problem shown in Eq. (34) is equivalent to the following optimization function:

$$\arg\min_{G^T G = I} Tr(G^T L_l G), \tag{35}$$

where L_l is a Laplacian matrix which is defined as $L_l = (I - A)^T (I - A)$. With Eq. (35), it is obvious that LLC has a form similar to Eq. (30), thus it is a spectral clustering method [38]. Considering that the matrix L defined in Eq. (18) is also a Laplacian matrix (See *Theorem 2*), we have the following proposition.

Proposition 3: LDMGI is a spectral clustering algorithm, which learns a Laplacian matrix for data clustering by explicitly employing discriminant information and manifold structure.

Discussions: NCut, LLC and our LDMGI are all spectral clustering algorithms. They are different because they use different Laplacian matrices for data clustering. Compared with Ncut, LDMGI learns a Laplacian matrix for data clustering while Ncut directly adopts a Gaussian function to compute the Laplacian matrix for data clustering. As reported in [37], a major disadvantage of using a Gaussian function to compute a Laplacian matrix is that it is sensitive to the bandwidth parameter σ. In addition, the discriminant information is not utilized in NCut. On the other hand, LDMGI learns a Laplacian matrix for clustering by using discriminant information and manifold structure, making it more effective and robust for image clustering.

While a Laplacian matrix is also learnt in [38], LLC adopted a *two-step* approach without having a unified objective function. In contrast, LDMGI has a unified objective function (*i.e.*, Eq. (17)), and the optimization problem can be solved within *a single step*. Moreover, LDMGI explicitly utilizes the discriminant information by using a variant Fisher criterion, and our experiment demonstrates that LDMGI outperforms LLC for image clustering.

Traditionally, K-means clustering and spectral clustering are regarded as different families of clustering algorithms. While the connection between K-means and NCut has been discussed in [7] in terms of a kernel view, this work presents another new perspective to understand and explain the relationships between K-means (or DisKmeans) and

TABLE I

DATABASES DESCRIPTION.

Dataset	Sample size	Feature dimension	Number of classes
COIL-20	1440	1024	20
USPS	9298	256	10
MNIST-T	5000	784	10
MNIST-S	6996	784	10
USF HumanID	2190	704	122
UMIST	575	644	20
YALE-B	2414	1024	38
MSRA	1031	1024	12
JAFFE	213	676	10
Pointing04	2790	1120	15
MPEG7	600	200	30
HumanEva	5000	48	5

spectral clustering algorithms via LDMGI. That is, K-means and DisKmeans are both special cases of LDMGI, and LDMGI is a type of spectral clustering algorithm as well.

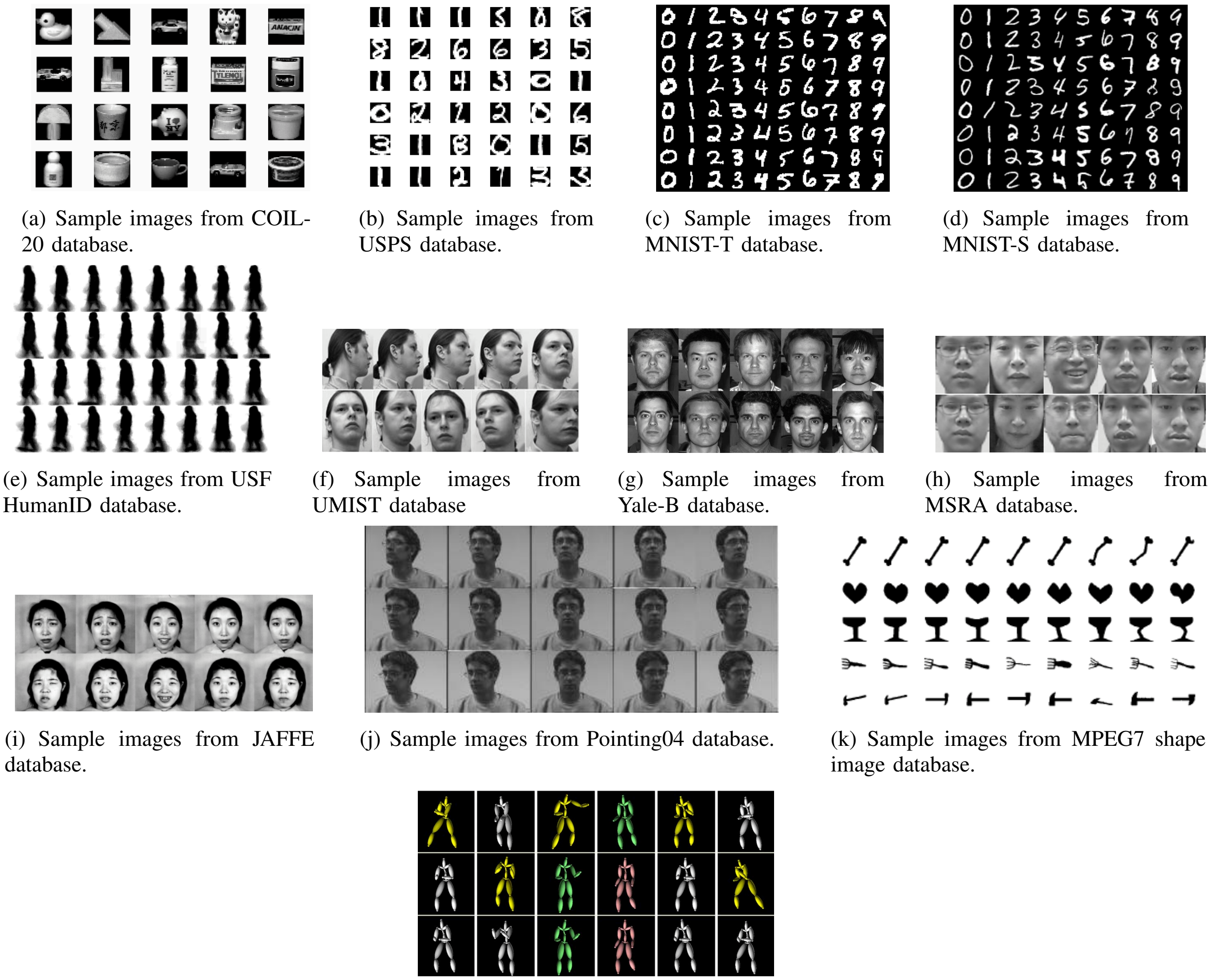
IV. EXPERIMENTS

We compare the proposed LDMGI with K-means, DisKmeans [42] and two spectral clustering algorithms NCut [29] and LLC [38]. For LDMGI, NCut and LLC, spectral rotation [43] is used to obtain the clustering results. For LLC, both a Gaussian kernel (denoted as LLC-G) and a linear kernel (denoted as LLC-L) are used as the kernel function $\mathrm{K}(x_i, x_j)$ in Eq. (33).

A. Image Dataset

In our experiments, over 30,000 samples from 12 benchmark databases were used to test the image clustering performance. These databases include one object image database COIL-20 [22], three handwritten digit image databases which are USPS and two sub-databases (referred to as MNIST-T and MNIST-S) of MNIST [18], one human gait image database [27], five face image databases, i.e., YALE-B [10], UMIST [14], MSRA [15], JAFFE [21] and Pointing04 [13], one shape image database MPEG7 [3] and one 3D human pose database HumanEva [31].

The COIL-20 [22] database has 1440 images of 20 objects, with each object containing 72 images captured from different views. The images are resized to 32×32. The USPS database contains 9298 gray-scale handwritten digit images scanned from envelopes by the U.S. Postal Service. The image size is 16×16. The MNIST database [18] has a training set of 60,000 examples, and a test set of 10,000 examples with the image size as 28×28. Considering it is computationally expensive to run all the clustering algorithms on such a large scale database, two sub-databases are chosen from the MNIST database. Following [39], we use the first part of the test set of the MNIST database (referred to as MNIST-T) which is cleaner than the second part [18]. MNIST-T consists of 5000 images of handwritten numbers with each digital number having 500 images. In the second sub-database (referred to as MNIST-S), for each class we randomly sample one handwritten digit image per ten images. Thus, we totally have 6996 images of handwritten numbers in MNIST-S. The USF HumanID database constructed by Sarkar *et*



(a) Sample images from COIL-20 database.

(b) Sample images from USPS database.

(c) Sample images from MNIST-T database.

(d) Sample images from MNIST-S database.

(e) Sample images from USF HumanID database.

(f) Sample images from UMIST database

(g) Sample images from Yale-B database.

(h) Sample images from MSRA database.

(i) Sample images from JAFFE database.

(j) Sample images from Pointing04 database.

(k) Sample images from MPEG7 shape image database.

(l) Sample data from HumanEva 3D pose database. Different color indicate different poses.

Fig. 1. Sample data from 12 databases used in the experiment.

al. [27] consists of people walking in elliptical paths in front of the camera. There are up to 32 sequences for each person, and the full data set consists of 1870 sequences from 122 individuals. Similarly as in [27], for each sequence, the binary silhouette images within one gait cycle are averaged to obtain the gray-level average silhouette image. We further resize each average silhouette image to 32×22 and then represent it as a 704-dimensional feature vector. 2190 images from the first five Probe sets are used in this experiment. The UMIST face image database [14] has 575 multi-view face images of 20 people, covering a broad range of poses from profile to frontal views. The images are resized to 28×23 pixels. The YALE-B database [10] contains 2414 near frontal images from 38 persons under different illuminations. Each image is resized to 32×32 and then represented by a 1024-dimensional feature vector. The MSRA face images are collected by Microsoft Research Asia (MSRA) [15]. It contains 12 subjects, captured in two different sessions with different illuminations and backgrounds. The images are resized

TABLE II

Performance Comparison (ACC %) of K-means, DisKmeans [42], NCut [43], LLC (LLC-L and LLC-G) [38] and LDMGI. In this table, we report the best ACC corresponding to the best objective value from multiple random initializations and parameters.

	K-means	DisKmeans	NCut	LLC-L	LLC-G	LDMGI
COIL-20	67.9	68.1	73.6	68.9	72.0	**82.4**
USPS	67.4	71.5	78.7	70.1	71.6	**81.5**
MNIST-T	56.6	50.4	68.2	66.1	68.5	**72.8**
MNIST-S	53.2	51.8	66.2	71.0	71.2	**77.9**
USF HumanID	36.9	40.8	42.9	42.6	40.3	**43.5**
UMIST	42.9	45.3	60.8	61.3	62.4	**69.1**
YALE-B	12.3	43.9	47.7	48.9	49.0	**55.1**
MSRA	82.3	84.2	96.6	95.8	95.2	**98.1**
JAFFE	84.0	78.4	91.1	89.7	90.6	**93.9**
Pointing04	35.8	39.9	73.5	76.1	75.0	**77.9**
MPEG7	57.0	64.8	69.2	66.7	69.3	**71.8**
HumanEva	50.9	64.0	89.7	80.3	82.5	**93.5**

to 32×32. All of the 1031 frontal face images in the first session are used in this experiment. The Japanese Female Facial Expression (JAFFE) database consists of 213 images of different facial expressions conducted by 10 Japanese female models [21]. The images are resized to 26×26. The Pointing04 face database consists of 15 sets of images [13]. Each set contains two series of 93 images of the same person. In our experiment, each face is resized to 40×28, and then represented by a 1120-dimensional normalized feature vector. To evaluate the shape image clustering performance, the first 600 images from MPEG7 shape image database [3] [35] were first aligned by employing Hungarian method and the pair-wise distances among images were computed based on shape context algorithm [2]. After that, Multi-Dimensional Scaling (MDS) is used to obtain a 200 dimensional normalized feature vector for each shape image. In total, there are 30 types of shapes in this database. HumanEva database contains five types of poses (Box, Gestures, Jog, Walk and Throw-Catch) performed by different subjects [31]. In the experiment, each pose data is represented by a 48-dimensional feature vector, which is from 16 markers in the skeleton model with each marker associated with three coordinates. The data from all the poses are redundant, because the motion capture process is repetitive and the frame rate is high. Following [24], we randomly sample 5000 pose data of one person in this experiment. Sample data from these datasets are shown in Figure 1, and Table I summarizes the detailed information in terms of the total number of data, the total number of cluster and the feature dimension.

B. Evaluation metrics

The performance is evaluated by comparing the cluster labels from the clustering algorithms and the ground truth labels. We use Clustering Accuracy (ACC) and Normalized Mutual Information (NMI) for performance evaluation in this work.

Clustering Accuracy (ACC): For the i-th image, let us denote q_i as the clustering result from the clustering

TABLE III

Performance comparison (NMI %) of K-means, DisKmeans [42], NCut [43], LLC (LLC-L and LLC-G) [38] and LDMGI. In this table, we report the best NMI corresponding to the best objective value from multiple random initializations and parameters.

	K-means	DisKmeans	NCut	LLC-L	LLC-G	LDMGI
COIL-20	77.3	77.5	85.0	85.1	86.0	**91.1**
USPS	61.2	68.4	83.6	77.6	78.2	**86.5**
MNIST-T	50.3	48.4	68.4	61.3	64.2	**69.2**
MNIST-S	49.5	46.8	67.3	68.6	66.8	**73.9**
USF HumanID	68.9	72.6	75.2	74.8	72.6	**76.1**
UMIST	64.5	66.1	79.2	78.2	78.6	**86.6**
YALE-B	18.3	55.2	67.1	65.1	64.9	**71.1**
MSRA	82.8	85.4	95.6	95.1	94.6	**97.2**
JAFFE	88.1	80.7	91.8	89.5	91.3	**93.6**
Pointing04	42.8	43.4	79.9	81.3	80.8	**84.7**
MPEG7	73.9	76.4	78.7	77.8	79.2	**81.2**
HumanEva	43.1	58.2	82.3	65.4	74.9	**87.7**

TABLE IV

Performance comparison (Mean ACC ± Standard Deviation %) of K-means, DisKmeans [42], NCut [43], LLC (LLC-L and LLC-G) [38] and LDMGI. The experiments are independently repeated for twenty times. The results shown in boldface are significantly better than the others, with a significance level of 0.05.

	K-means	DisKmeans	NCut	LLC-L	LLC-G	LDMGI
COIL-20	59.2 ± 3.8	53.5 ± 6.1	68.3 ± 5.3	63.1 ± 3.3	67.5 ± 5.3	**75.3** ± 4.9
USPS	64.9 ± 3.6	69.4 ± 3.7	73.4 ± 6.3	66.6 ± 3.1	70.1 ± 3.9	**80.5** ± 5.6
MNIST-T	55.3 ± 2.6	48.8 ± 3.4	66.2 ± 3.4	66.1 ± 2.3	64.7 ± 3.6	**71.5** ± 3.5
MNIST-S	53.0 ± 4.1	48.6 ± 5.6	64.5 ± 2.0	66.7 ± 3.4	68.2 ± 4.5	**76.3** ± 3.4
USF HumanID	35.9 ± 1.0	39.7 ± 1.1	**42.6** ± 0.4	**42.3** ± 0.3	40.1 ± 0.2	**42.9** ± 0.5
UMIST	42.7 ± 2.5	43.0 ± 3.2	60.1 ± 0.7	59.6 ± 1.7	59.4 ± 3.2	**67.0** ± 1.7
YALE-B	11.3 ± 0.7	41.1 ± 3.1	46.2 ± 1.5	48.5 ± 0.6	46.7 ± 0.6	**55.0** ± 1.2
MSRA	81.7 ± 2.7	85.1 ± 2.2	95.2 ± 2.6	94.6 ± 2.7	91.4 ± 3.3	**95.9** ± 2.2
JAFFE	71.3 ± 10.7	75.6 ± 9.4	83.9 ± 6.5	82.8 ± 6.8	79.7 ± 5.6	**90.4** ± 6.0
Pointing04	35.2 ± 1.5	36.3 ± 4.2	70.6 ± 3.2	72.9 ± 3.3	70.2 ± 2.9	**77.2** ± 1.9
MPEG7	56.5 ± 4.1	58.4 ± 3.9	66.9 ± 1.8	64.0 ± 2.2	65.9 ± 2.0	**68.0** ± 3.2
HumanEva	55.5 ± 7.0	59.1 ± 6.9	79.3 ± 9.7	70.6 ± 7.6	81.5 ± 4.4	**89.2** ± 6.1

algorithm and p_i as the ground truth label. The ACC is defined as:

$$ACC = \frac{\sum_{i=1}^{n} \delta(p_i, map(q_i))}{n} \tag{36}$$

where n is the total number of images, $\delta(x, y) = 1$, if $x = y$, and $\delta(x, y) = 0$, otherwise, $map(q_i)$ is the optimal mapping function that permutes clustering labels to match the ground truth labels. The optimal mapping can be obtained by using the Kuhn-Munkres algorithm [25]. A larger ACC indicates a better performance.

Normalized Mutual Information (NMI): Normalized Mutual Information (NMI) is another widely used measure for evaluating the clustering results. For two arbitrary variables *P* and *Q*, NMI is defined as follows [33]:

$$NMI(P, Q) = \frac{I(P, Q)}{\sqrt{H(P)H(Q)}}, \tag{37}$$

where $I(P, Q)$ is the mutual information between P and Q and $H(P)$ and $H(Q)$ denote the entropies of P and

TABLE V

Performance comparison (Mean NMI ± Standard Deviation %) of K-means, DisKmeans [42], NCut [43], LLC (LLC-L and LLC-G) [38] and LDMGI. The experiments are independently repeated for twenty times. The results shown in boldface are significantly better than the others, with a significance level of 0.05.

	K-means	DisKmeans	NCut	LLC-L	LLC-G	LDMGI
COIL-20	74.5 ± 2.1	70.5 ± 3.2	82.3 ± 2.4	83.5 ± 0.8	85.3 ± 0.5	**90.0** ± 0.7
USPS	60.6 ± 1.2	67.6 ± 1.6	82.4 ± 1.9	76.4 ± 1.5	76.8 ± 1.5	**86.1** ± 2.3
MNIST-T	49.3 ± 1.0	45.7 ± 2.0	67.4 ± 1.4	61.2 ± 1.3	62.4 ± 1.6	**68.9** ± 1.5
MNIST-S	48.5 ± 2.0	44.8 ± 4.0	66.6 ± 1.0	67.9 ± 1.4	65.1 ± 2.8	**73.9** ± 1.5
USF HumanID	67.8 ± 0.6	70.5 ± 0.7	**75.1** ± 0.1	**74.6** ± 0.1	72.5 ± 0.1	**75.9** ± 0.2
UMIST	63.5 ± 2.5	63.6 ± 2.6	79.1 ± 0.6	76.2 ± 1.1	76.2 ± 1.3	**84.5** ± 1.0
YALE-B	17.8 ± 0.1	52.0 ± 3.1	66.6 ± 0.6	64.6 ± 0.5	65.8 ± 0.1	**70.8** ± 0.5
MSRA	81.8 ± 1.7	86.0 ± 1.5	95.3 ± 0.7	95.8 ± 0.3	94.6 ± 0.4	**96.5** ± 0.6
JAFFE	79.2 ± 6.7	79.7 ± 6.2	90.6 ± 1.0	88.7 ± 0.7	89.6 ± 0.6	**92.6** ± 1.4
Pointing04	41.9 ± 1.0	40.0 ± 4.4	79.2 ± 1.2	79.6 ± 1.7	78.6 ± 1.5	**84.2** ± 0.1
MPEG7	72.7 ± 1.9	73.7 ± 2.0	78.5 ± 0.3	77.1 ± 0.6	78.0 ± 0.7	**80.3** ± 0.4
HumanEva	44.6 ± 5.1	49.7 ± 6.5	75.5 ± 5.8	57.9 ± 6.0	74.1 ± 3.6	**84.6** ± 4.3

Q respectively [33]. It is obvious that $NMI(P,Q)$ equals 1 if P is identical with Q, and it becomes 0 if P is independent from Q. Let t_l be the number of samples in the cluster $\mathcal{C}_l$ $(1 \leq l \leq c)$ obtained by using the clustering algorithms and $\tilde{t}_h$ be the number of samples in the h-th groundtruth class $(1 \leq h \leq c)$. Given the clustering results, NMI is defined as [33]:

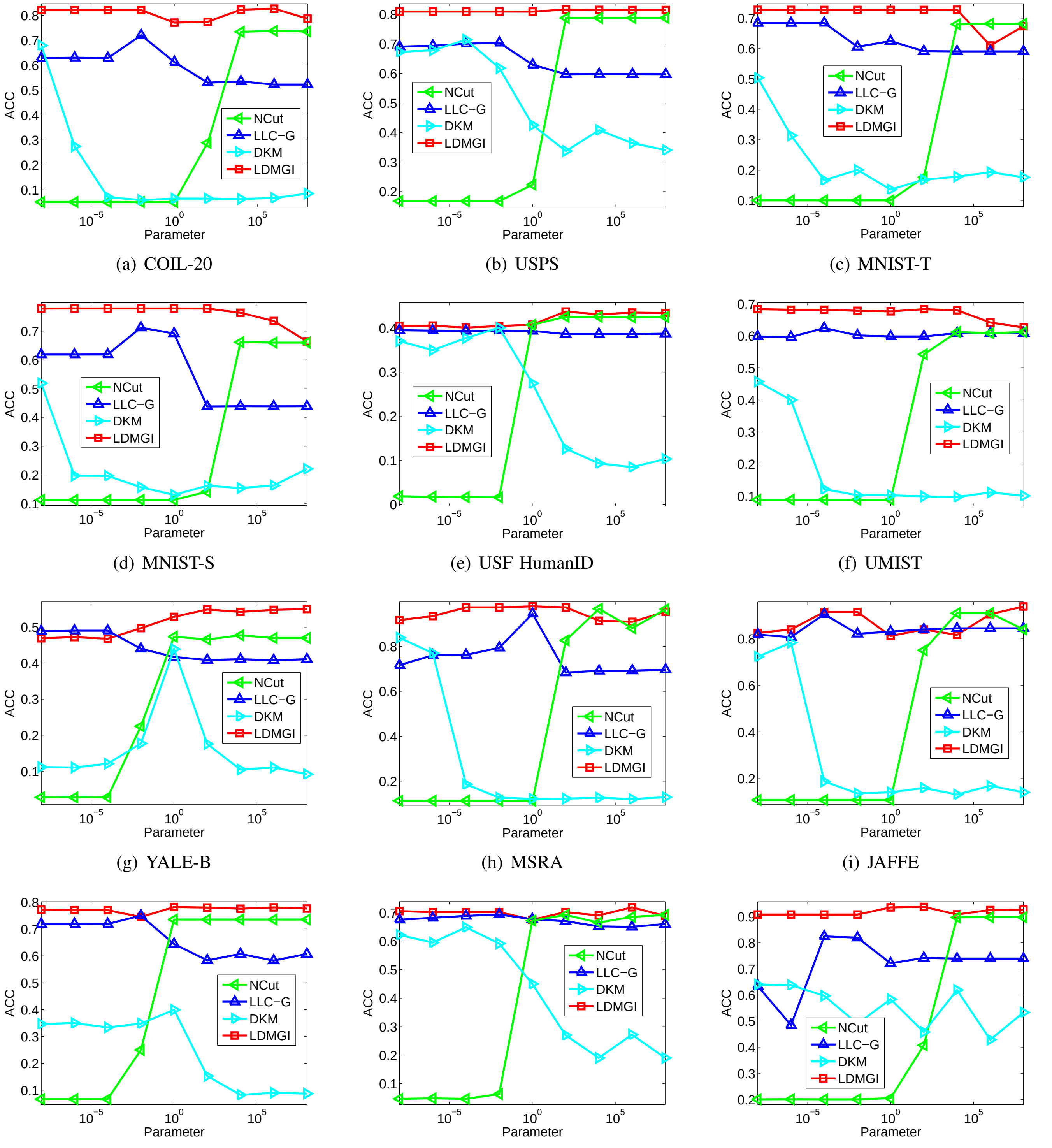
$$NMI = \frac{\sum_{l=1}^{c}\sum_{h=1}^{c} t_{l,h}\log(\frac{n\cdot t_{l,h}}{t_l\tilde{t}_h})}{\sqrt{\left(\sum_{l=1}^{c} t_l \log\frac{t_l}{n}\right)\left(\sum_{h=1}^{c}\tilde{t}_h \log\frac{\tilde{t}_h}{n}\right)}}, \tag{38}$$

where $t_{l,h}$ is the number of samples that are in the intersection between the cluster $\mathcal{C}_l$ and the h-th ground truth class. Again, a larger NMI indicates a better clustering result.

C. Performance Comparison

Several parameters need to be set beforehand for these clustering techniques. The total number of clusters c is provided for all the clustering algorithms. For LDMGI, NCut and LLC, we also need to determine the parameter k which specifies the number of nearest neighbors. Generally speaking, k should be set as a small number to preserve the local manifold structure. In this work, we observe that the performances of three algorithms LDMGI, NCut and LLC generally achieve the best performance on all image datasets, when $k = 5$. To fairly compare their performances, we fixe k as 5 for all the three algorithms on all the image databases. We need to determine the optimal parameter γ in Eq. (28) for DisKmeans, η in Eq. (33) for LLC, σ in Eq. (32) for NCut, and λ in Eq. (15) for our LDMGI. For fair comparison, we set these parameters as $\{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{0}, 10^{2}, 10^{4}, 10^{6}, 10^{8}\}$ and then report the best clustering results from the optimal parameters. Different parameters may be used for different databases (See Fig. 2 and Fig. 3).

The results of all clustering algorithms depend on the initialization [43]. As suggested in [41], for all the clustering algorithms, we independently repeat the experiments for 20 times with random initializations to reduce



(a) COIL-20

(b) USPS

(c) MNIST-T

(d) MNIST-S

(e) USF HumanID

(f) UMIST

(g) YALE-B

(h) MSRA

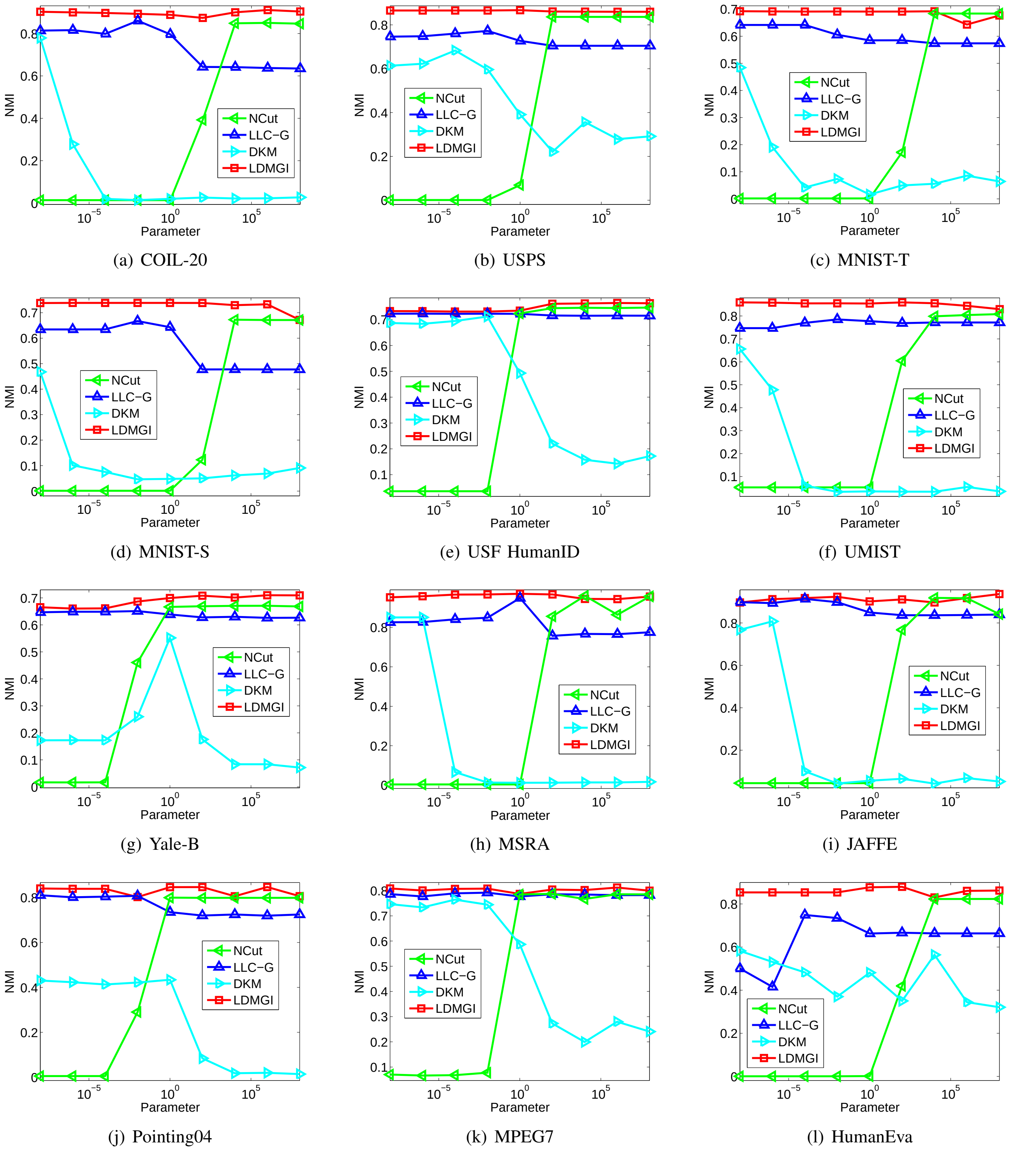
(i) JAFFE

(j) Pointing04

(k) MPEG7

(l) HumanEva

Fig. 2. Performance variations (ACC) of NCut [43], LLC-G [38], DisKmeans [42] and our LDMGI with different parameters. DisKmeans is denoted by DKM in the figure for better illustration.



(a) COIL-20

(b) USPS

(c) MNIST-T

(d) MNIST-S

(e) USF HumanID

(f) UMIST

(g) Yale-B

(h) MSRA

(i) JAFFE

(j) Pointing04

(k) MPEG7

(l) HumanEva

Fig. 3. Performance variations (NMI) of NCut [43], LLC-G [38], DisKmeans [42] and our LDMGI with different parameters. DisKmeans is denoted by DKM in the figure for better illustration.

the statistical variation. For clustering algorithms DisKmeans, LLC, NCut and LDMGI, we also need to search for the best parameters in the set $\{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{0}, 10^{2}, 10^{4}, 10^{6}, 10^{8}\}$. Thus, on each database 20 rounds of clustering are performed for K-means, and 180 rounds of clustering are performed for other clustering algorithms. In Tables II and III, we report the best results from these 20 or 180 rounds of clustering corresponding to the best objective values in terms of ACC and NMI respectively. For NCut, LLC-G, DisKmeans and our LDMGI, we also find that the best results for each parameter corresponding to the best objective values from twenty rounds of random initializations. To compare the parameter sensitivity of different algorithms, we plot the performance variations of these algorithms with different parameters in Fig. 2 (*resp.* Fig. 3) in terms of ACC (*resp.* NMI). For each clustering algorithm, we also calculate the the mean ACC (*resp.* NMI) from twenty rounds of random initializations for each parameter and then we additionally report the best mean ACC (*resp.* NMI) together with the standard deviation corresponding to the optimal parameter in Table IV (*resp.* Table V). Significance test is conducted with the significance level of 0.05 and the results are reported in able IV and Table V. From these results, we have the following observations:

1) When comparing two global clustering approaches (*i.e.*, Kmeans and DisKmeans [42]), we find that DisKmeans is generally better than K-means, which demonstrates that the utilization of discriminant information is useful to improve the image clustering performance. We also observe that ACC (*resp.* NMI) on Yale-B database is significantly improved from 12.3 (*resp.* 18.3) to 43.9 (*resp.* 55.2) by using DisKmeans instead of Kmeans. Our observation is also consistent with the prior work [5] [36] [42], which indicates that it is helpful to utilize discriminative information for data clustering and subspace learning.

2) Three local clustering approaches (*i.e.*, NCut [43], LLC (LLC-L and LLC-G) [38] and our LDMGI) are generally better than K-means and DisKmeans. It demonstrates that NCut, LLC and LDMGI can effectively make use of the essential manifold structure of the images which is not considered in the global clustering methods K-means and DisKmeans. Note that the local manifold structure has been experimentally shown to be useful in dimension reduction [15] and data clustering [38]. This observation confirms that local data structural information is crucial for image clustering.

3) As shown in Tables II and III, NCut outperforms LLC-G (or LLC-L) on some databases and NCut is worse than LLC-G (or LLC-L) on other databases. In other words, there is no consistent winner between NCut and LLC-G (or LLC-L) on all the databases. We also observe that LLC-G is generally better than LLC-L, possibly owing to the utilization of nonlinear kernel function in the local regression model.

4) From Tables II and III, we observe that our method LDMGI achieves the best results over all the 12 datasets in terms of ACC and NMI. As shown in Tables IV and V, LDMGI also outperforms K-means, DisKmeans, NCut [43], LLC (LLC-L and LLC-G) in terms of mean ACC and NMI over all the 12 databases. When comparing LDMGI with the second best result from NCut on COIL-20 database, the mean ACC (*resp.* NMI) is significantly improved from 68.3% to 75.3% (*resp.* from 82.3% to 90.0%). Judged by t-test (with a significance level of 0.05), LDMGI is significantly better than K-means, DisKmeans, NCut, LLC (LLC-L and LLC-G) in 11 out of 12 databases.

These results demonstrate that LDMGI can effectively utilize local manifold structure as well as the discriminant information for image clustering.

5) Except for the simplest clustering algorithm K-means, one parameter needs to be set beforehand for other clustering algorithms (*i.e.*, γ in Eq. (28) for DisKmeans, η in Eq. (33) for LLC, σ in Eq. (32) for NCut, and λ in Eq. (15) for our LDMGI). From Fig. 2 and Fig. 3, we can clearly observe that DisKmeans and NCut are sensitive to their parameters. When comparing NCut with LLC-G, we observe that NCut is more sensitive to the bandwidth parameter of the Gaussian function. Our observation that NCut is sensitive to the bandwidth parameter is consistent with the prior work [37].

6) From Figs. 2 and 3, we also observe that LDMGI is more robust to the algorithmic parameter λ , when compared with NCut [29], LLC-G [38] and DisKmeans [42]. A possible reason is that our LDMGI learns a Laplacian matrix for data clustering during which both local data structural information and discriminant information are used. In real applications, ground truth is generally not available for tuning the parameters of clustering algorithms. Thus, LDMGI is more suitable for real image clustering applications because LDMGI is not only accurate but also stable to the algorithmic parameter.

7) For all the clustering algorithms, we also observe that the ACCs in Table II (*resp.* NMIs in Table III) corresponding to the best objective values are generally higher than the mean ACCs in Table IV (*resp.* the mean NMIs in Table V). Since it is still a non-trivial task to obtain the globally optimal solutions for the clustering algorithms, it is reasonable to choose the clustering results from the optimal initialization corresponding to the best objective values in the practical clustering applications.

Discussions about Features: Pose, illumination and expression are three major challenges in face recognition. Therefore, the clustering performances on UMINST database [14] (with strong pose variation) and Yale-B database [10] (with significant illumination variation) using the low-level gray-scale intensity features are relatively poor, when compared with the results on other databases. The explanation is the extracted gray-scale intensity features are not discriminant enough to effectively group face images into different subjects in these cases. Similarly, it has been reported that human identification performance on USF HumanID database [27] is also poor using the gray-scale intensity features extracted from average silhouette images when the images are captured from different types of surfaces (e.g., grass and concrete) [27]. One can employ more discriminant features to improve the clustering performance.

Here, we take UMIST face database as an example to test the clustering performance using Gabor feature [39]. We extracted 40 Gabor features with five different scales and eight different orientations [39], so each image is then represented as a 25760-dimensional feature vector. We additionally take MPEG7 shape image database as another example to test the clustering performance by directly using the gray-scale intensity values as features. Specifically, the size of each shape image is 60×100, thus we represent each image as a 6000-dimensional feature vector. Again, we independently perform all the clustering algorithms for 20 times with random initializations. We report the clustering results corresponding to the best objective values in Table VI and report the mean ACC and NMI

TABLE VI

PERFORMANCE COMPARISON (ACC AND NMI %) OF K-MEANS, DISKMEANS [42], NCUT [43], LLC (LLC-L AND LLC-G) [38] AND LDMGI. IN THIS TABLE, WE REPORT THE BEST ACC AND NMI CORRESPONDING TO THE BEST OBJECTIVE VALUE FROM MULTIPLE RANDOM INITIALIZATIONS AND PARAMETERS.

	K-means	DisKmeans	NCut	LLC-L	LLC-G	LDMGI
UMIST-Gabor (ACC)	41.7	44.9	76.4	68.5	73.6	**84.0**
UMIST-Gabor (NMI)	61.2	63.2	87.1	77.7	84.4	**90.4**
MPEG7-Gray (ACC)	55.5	58.2	57.2	57.3	53.6	**61.2**
MPEG7-Gray (NMI)	68.7	68.9	69.7	69.8	65.9	**72.6**

TABLE VII

PERFORMANCE COMPARISON (MEAN ACC AND NMI ± STANDARD DEVIATION %) OF K-MEANS, DISKMEANS [42], NCUT [43], LLC (LLC-L AND LLC-G) [38] AND LDMGI. THE EXPERIMENTS ARE INDEPENDENTLY REPEATED FOR TWENTY TIMES. THE RESULTS SHOWN IN BOLDFACE ARE SIGNIFICANTLY BETTER THAN THE OTHERS, WITH A SIGNIFICANCE LEVEL OF 0.05.

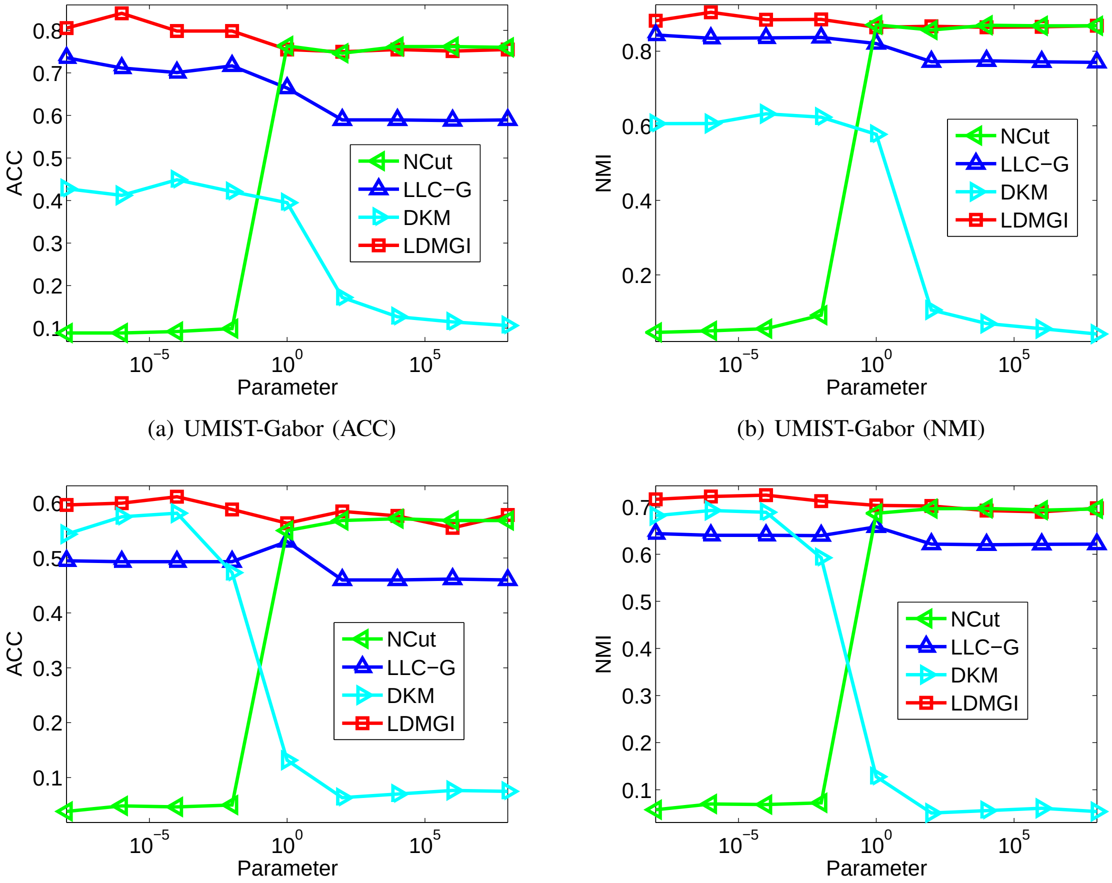
	K-means	DisKmeans	NCut	LLC-L	LLC-G	LDMGI
UMIST-Gabor (ACC)	41.2 ± 2.7	41.6 ± 2.4	73.3 ± 1.9	68.2 ± 1.9	72.1 ± 2.7	**78.7** ± 3.1
UMIST-Gabor (NMI)	57.9 ± 2.6	58.4 ± 2.4	85.6 ± 0.8	77.6 ± 1.0	84.1 ± 0.6	**87.5** ± 1.4
MPEG7-Gray (ACC)	49.7 ± 4.2	51.3 ± 3.6	56.5 ± 1.3	55.9 ± 1.2	56.5 ± 1.3	**58.1** ± 1.9
MPEG7-Gray (NMI)	64.5 ± 2.8	65.6 ± 2.1	69.5 ± 0.2	69.4 ± 0.5	69.5 ± 0.2	**71.6** ± 0.9

together with the standard deviation in Table VII. The performance variations of NCut, LLC-G, DisKmeans and our LDMGI with different parameters are shown in Fig 4. In Table VI, Table VII and Fig 4, we refer the results on UMIST and MPEG7 shape image databases as UMIST-Gabor and MPEG7-Gray in order to distinguish the new results and the previous results shown in Tables II, III, IV and V, and Figs. 2 and 3.

From these results, we observe that the clustering performances of most of the algorithms on UMIST database are improved by using Gabor features, when compared with the gray-scale intensity features. Similarly, we also observe the clustering performances of most of the methods degrade on MPEG7 shape image database by using the gray-scale intensity features, when compared with the features obtained by using MDS algorithm based on the pairwise distances calculated with shape context algorithm [2]. Therefore, feature is another important factor for image clustering and it is crucial to use discriminant feature in the real image clustering applications. However, the design of discriminant features is data-dependent and thus it is out of the scope of this piece of work, which will be investigated in the future. It is worth mentioning again that our method LDMGI still achieves the best results using alternative features and it is also more stable to the algorithmic parameter λ. Moreover, it is also significantly better than other methods (See Table VII). Again, it demonstrates that LDMGI can effectively utilize both manifold structure and the discriminant information for image clustering.

V. CONCLUSIONS

Observing that discriminant information and manifold information are both crucial for image clustering, we have proposed a new clustering algorithm, referred to as Clustering with Local Discriminant Models and Global Integrations (LDMGI). For each data point, we consider a local clique comprising this data point and its neighboring data points. A local discriminant model is used to measure the clustering performance of samples in each local



(a) UMIST-Gabor (ACC)

(b) UMIST-Gabor (NMI)

(c) MPEG7-Gray (ACC)

(d) MPEG7-Gray (NMI)

Fig. 4. ACC and NMI variations of NCut [43], LLC-G [38], DisKmeans [42] and our LDMGI with different parameters. DisKmeans is denoted by DKM in the figure for better illustration.

clique. To globally integrate the local models from all the cliques, we further propose a unified objective function. We observe that LDMGI shares a similar objective function as SC algorithms (*e.g.*, NCut). Thus, the optimal cluster assignment matrix can be obtained by employing spectral relaxation and spectral rotation. Moreover, the connections between LDMGI and other existing clustering algorithms are theoretically analyzed. Our experiments demonstrate that LDMGI outperforms other existing clustering methods and LDMGI is also more robust to algorithmic parameter, when compared with NCut. In the future, we will investigate how to extract more discriminant image features to further improve the clustering performance as well as study how to automatically determine the optimal parameters for LDMGI.

Appendix

In this Appendix, we provide detailed justification for our local discriminant model. In practice, we prove that $Tr(\tilde{X}_i G_{(i)} G_{(i)}^T \tilde{X}_i^T)$ in Eq. (11) can be used to measure the inter-cluster separability of samples in the local clique. Let us define a graph for the i-th clique with each node corresponding to one data point $x_u \in \mathcal{N}_k(x_i)$. We also denote the graph similarity matrix $A \in \mathbb{R}^{k\times k}$ as:

$$A = \begin{bmatrix} B_1 \\ \vdots \\ B_c \end{bmatrix} - G_{(i)}G_{(i)}^T = \begin{bmatrix} B_1 \\ \vdots \\ B_c \end{bmatrix} - \begin{bmatrix} K_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & K_c \end{bmatrix}, \tag{39}$$

where $K_p \in R^{m_{i_p}\times m_{i_p}}$ is the square matrix defined in Eq. (10) with all its elements equal to $1/n_p$ $(1 \le p \le c)$, and $B_p \in \mathbb{R}^{m_{i_p}\times k}$ is a matrix with all its elements equal to $\frac{m_{i_p}}{kn_p}$ $(1 \le p \le c)$.

According to the definition of A in Eq. (39), we observe that $A_{uv} > 0$ if $x_u \in \mathcal{N}_k(x_i)$ and $x_v \in \mathcal{N}_k(x_i)$ are from different clusters, and $A_{uv} < 0$ if x_u and x_v are from the same cluster. Following [1], for the samples $x_u \in \mathcal{N}_k(x_i)$ and $x_v \in \mathcal{N}_k(x_i)$, we use the following objective function to measure the inter-cluster separability:

$$\arg\max_{G_i} \sum\nolimits_{x_u,x_v} \|x_u - x_v\|^2 A_{uv}. \tag{40}$$

The above objective function has a two-fold explanation [1]. If the samples x_u and x_v are from the same cluster, (*i.e.*, $A_{uv} < 0$), the distance between x_u and x_v should be smaller to maximize the objective function. Likewise, if the samples x_u and x_v are from different clusters (*i.e.*, $A_{uv} > 0$), the distance between x_u and x_v should be larger to maximize the objective function. Thus, if we use Eq. (39) to define the graph similarity matrix A, the objective function in Eq. (40) measures the inter-cluster separability.

Let us define the corresponding Laplacian matrix of A as $L_A = D - A$, where D is the corresponding diagonal matrix with its element as $D_{uu} = \sum_v A_{uv}$. For any u, we have:

$$D_{uu} = \sum\nolimits_{v=1}^{k} A_{uv} = \frac{m_{i_p}}{kn_p} \times k - \frac{1}{n_p} \times m_{i_p} = 0. \tag{41}$$

Thus, L_A can be rewritten as:

$$L_A = D - A = -\left(\begin{bmatrix} B_1 \\ \vdots \\ B_c \end{bmatrix} - G_{(i)}G_{(i)}^T \right) = G_{(i)}G_{(i)}^T - \begin{bmatrix} b_1 \\ \vdots \\ b_c \end{bmatrix} \mathbf{1}_k^T, \tag{42}$$

where $b_p \in \mathbb{R}^{m_{i_p}}$ is a vector with all its elements equal to $\frac{m_{i_p}}{kn_p}$ $(1 \le p \le c)$. For any Laplacian matrix L_A, we have $L_A = H_k L_A H_k$. Considering that $\mathbf{1}_k^T H_k = 0$, we arrive at:

$$L_A = H_k L_A H_k = H_k \left(G_{(i)}G_{(i)}^T - \begin{bmatrix} b_1 \\ \vdots \\ b_c \end{bmatrix} \mathbf{1}_k^T \right) H_k = H_k G_{(i)} G_{(i)}^T H_k. \tag{43}$$

Finally, according to [1], the objective function in Eq. (40) can be rewritten as:

$$\begin{aligned} &\sum\nolimits_{x_u,x_v} \|x_u - x_v\|^2 A_{uv} \\ &= Tr(X_i(D-A)X_i^T) \\ &= Tr(X_i L_A X_i^T) \\ &= Tr(X_i H_k L_A H_k X_i^T) \\ &= Tr(X_i H_k G_{(i)} G_{(i)}^T H_k X_i^T) \\ &= Tr(\tilde{X}_i G_{(i)} G_{(i)}^T \tilde{X}_i^T). \end{aligned} \tag{44}$$

Thus, we conclude that $Tr(\tilde{X}_i G_{(i)} G_{(i)}^T \tilde{X}_i^T)$ can be used to measure the inter-cluster separability of samples in the local clique.

References

[1] M. Belkin and P. Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6):1373–1396, 2002.

[2] S. Belongie, J. Malik, and J. Puzicha (April 2002). Shape Matching and Object Recognition Using Shape Contexts. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (24):509C521, 2002.

[3] M. Bicego and V. Murino Similarity-based classification of sequences using hidden Markov models, In *Pattern Recognition*, 37(12): pages 92281-2291, 2004.

[4] Y. Chen, J. Z. Wang, and R. Krovetz. Clue: cluster-based retrieval of images by unsupervised learning. *IEEE Transactions on Image Processing*, 14(8):1187–1201, 2005.

[5] H. Chris, Q. Ding, and T. Li. Adaptive dimension reduction using discriminant analysis and k-means clustering. In *International Conference on Machine Learning*, pages 521–528, 2007.

[6] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 40(2), 2008.

[7] I.S. Dhillon, Y. Guan and B. Kulis. Kernel kmeans, Spectral Clustering and Normalized Cuts. In *Knowledge Discovery and Data Mining Conference*, pages 551-556, 2004.

[8] Y. Du and J. Z. Wang. A scalable integrated region-based image retrieval system. In *International Conference on Image Processing* , pages 22–25, 2001.

[9] K. Fukunaga. *Introduction to statistical pattern recognition (2nd ed.).* Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[10] A. Georghiades, P. Belhumeur, and D. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):643–660, 2001.

[11] J. Goldberger, S. Gordon, and H. Greenspan. Unsupervised image-set clustering using an information theoretic framework. *IEEE Transactions on Image Processing*, 15(2):449–458, 2006.

[12] S. Gordon, H. Greenspan, and J. Goldberger. Applying the information bottleneck principle to unsupervised clustering of discrete and continuous image representations. In *IEEE International Conference on Computer Vision*, pages 370–377, 2003.

[13] N. Gourier, D. Hall and J. Crowley, Estimating face orientation from robust detection of salient facial features. In *ICPR Workshop on Visual Observation of Deictic Gestures*, 2004.

[14] D. Graham and N. Allinson. Web site of umist multi-view face database: http://images.ee.umist.ac.uk/danny/database.html, image engineering and neural computing lab, umist, uk.

[15] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang. Face Recognition Using Laplacianfaces. *IEEE Transactions on Patten Analysis and Machine Intelligence*. 27(3): 328–340 2005.

[16] J. Jia, N. Yu, and X.-S. Hua. Annotating personal albums via web mining. In *ACM Multimedia*, pages 459–468, 2008.

[17] L. S. Kennedy and M. Naaman. Generating diverse and representative image search results for landmarks. In *World Wide Web Conference*, pages 297–306, 2008.

[18] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition." Proceedings of the IEEE, 86(11):2278-2324, 1998

[19] J. Li. Two-scale image retrieval with significant meta-information feedback. In *ACM Multimedia*, pages 499–502, 2005.

[20] J. Li and J. Z. Wang. Real-time computerized annotation of pictures. In *ACM Multimedia*, pages 911–920, 2006.

[21] Michael J. Lyons, Julien Budynek, and Shigeru Akamatsu Automatic Classification of Single Facial Images *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21 (12): 1357-1362, 1999.

[22] S. Nene, S. Nayar, and H. Murase. Columbia object image library (coil-20). *Technical Report CUCS-005-96*, 1996.

[23] A.Y. Ng , M.I. Jordan and Y. Weiss. On Spectral Clustering: Analysis and an algorithm. In *Neural Information Processing Systems*, pages 849–856, 2001.

[24] Huazhong Ning, Wei Xu, Yihong Gong, Thomas Huang. Discriminative Learning of VisualWords for 3D Human Pose Estimation. In *CVPR* 2008.

[25] C. H. Papadimitriou and K. Steiglitz. Combinatorial Optimization: Algorithms and Complexity. Dover, New York, 1998.

[26] F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. *IEEE Workshop on Applications of Computer Vision*, 1994.

[27] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer. The humanid gait challenge problem: Data sets, performance, and analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2):162–177, 2005.

[28] L. K. Saul and S. T. Roweis. Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4:119–155, 2003.

[29] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analalysis Machine Intelligence*, 22(8):888–905, 2000.

[30] R. Shi, W. Jin, and T.-S. Chua. A novel approach to auto image annotation based on pairwise constrained clustering and semi-naïve bayesian model. In *Multimedia Modeling*, pages 322–327, 2005.

[31] L. Sigal and M. J. Black. HumanEva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown University, Department of Computer Science, 2006.

[32] I. Simon, N. Snavely, and S. M. Seitz. Scene summarization for online image collections. In *IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[33] A. Strehl and J. Ghosh. Cluster ensembles C a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research,* 3:583–617, 2002.

[34] Tang JKT, Leung H, Komura T, Shum HPH. Emulating human perception of motion similarity. In *Computer Animation and Virtual Worlds*, 19(3-4): 211-221, 2008.

[35] N. Thakoor, J. Gao, S. Jung Hidden Markov Model-Based Weighted Likelihood Discriminant for 2-D Shape Classification in *IEEE Transactions on Image Processing*, 16(11):2707 - 2719, 2007

[36] F. D. Torre and T. Kanade. Discriminative cluster analysis. In *International Conference on Machine Learning*, pages 241–248, 2006.

[37] F. Wang and C. Zhang. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering*, 20(1):55–67, 2008.

[38] M. Wu and B. Schölkopf. A local learning approach for clustering. In *Neural Information Processing Systems* , pages 1529–1536, 2006.

[39] D. Xu, S. Yan, L. Zhang, S. Lin, H. Zhang and Thomas Huang, Reconstruction and Recognition of Tensor-based Objects with Concurrent Subspaces Analysis. In *IEEE Trans. on Circuits Systems for Video Technology*, pp. 36-47, 2008.

[40] H. Xu, D. Yu, D. Xu, and A. Zhang. Ss-clustertree: a subspace clustering based indexing algorithm over high-dimensional image features. In *International Conference on Image and Video Retrieval*, pages 95–104, 2008.

[41] R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, 16(3): 645–678, 2005.

[42] J. Ye, Z. Zhao, and M. Wu. Discriminative k-means for clustering. In *Neural Information Processing Systems* , 2008.

[43] S. X. Yu and J. Shi. Multiclass spectral clustering. In *IEEE International Conference on Computer Vision*, pages 313–319, 2003.

[44] J. Yuan, J. Li, and B. Zhang. Learning concepts from large scale imbalanced data sets using support cluster machines. In *ACM Multimedia*, pages 441–450, 2006.

[45] H. Zha, X. He, C. H. Q. Ding, M. Gu, and H. D. Simon. Spectral relaxation for k-means clustering. In *Neural Information Processing Systems* , pages 1057–1064, 2001.

[46] Z. Zhang and H. Zha. Principal Manifolds and Nonlinear Dimensionality Reduction via Tangent Space Alignment *SIAM Journal Scientific Computing*, 26(1): 313–338, 2004.

[47] X. Zheng, D. Cai, X. He, W.-Y. Ma, and X. Lin. Locality preserving clustering for image database. In *ACM Multimedia*, pages 885–891, 2004.