Per-Cluster Ensemble Kernel Learning for Multi-Modal Image Clustering With Group-Dependent Feature Selection

Jeng-Tsung Tsai, Yen-Yu Lin, *Member, IEEE*, and Hong-Yuan Mark Liao, *Fellow, IEEE*

***Abstract*—In this paper, we present a clustering approach, MK-SOM, that carries out cluster-dependent feature selection, and partitions images with multiple feature representations into clusters. This work is motivated by the observations that human visual systems (HVS) can receive various kinds of visual cues for interpreting the world. Images identified by HVS as the same category are typically coherent to each other in certain crucial visual cues, but the crucial cues vary from category to category. To account for this observation and bridge the semantic gap, the proposed MK-SOM integrates *multiple kernel learning* (MKL) into the training process of *self-organizing map* (SOM), and associates each cluster with a learnable, ensemble kernel. Hence, it can leverage information captured by various image descriptors, and discoveries the cluster-specific characteristics via learning the per-cluster ensemble kernels. Through the optimization iterations, cluster structures are gradually revealed via the features specified by the learned ensemble kernels, while the quality of these ensemble kernels is progressively improved owing to the coherent clusters by enforcing SOM. Besides, MK-SOM allows the introduction of side information to improve performance, and it hence provides a new perspective of applying MKL to address both unsupervised and semi-supervised clustering tasks. Our approach is comprehensively evaluated in the two applications. The superior and promising results manifest its effectiveness.**

***Index Terms*—Cluster-dependent feature selection, clustering, image grouping, multiple kernel learning (MKL), object categorization.**

I. Introduction

MULTIMEDIA data clustering aims at partitioning data into a set of groups (clusters) so that data residing in the same cluster are coherent and similar to each other. It helps not only to select a few representative data samples for the whole dataset but also to identify the common properties of data within each cluster. As a fundamental component of multimedia content analysis, clustering is essential to widespread applications, such as multimedia mining, summarization, retrieval, and understanding. A key ingredient of designing successful clustering algorithms for multimedia content analysis is how to effectively bridge the semantic gap between the low-level data representations and high-level semantic concepts. We aim at addressing this issue in the work. Also, unless further specified, we focus on image data. Nevertheless, the proposed approach is developed in a general way so that it can be also applied to other kinds of multimedia data, such as music, videos, or webpages.

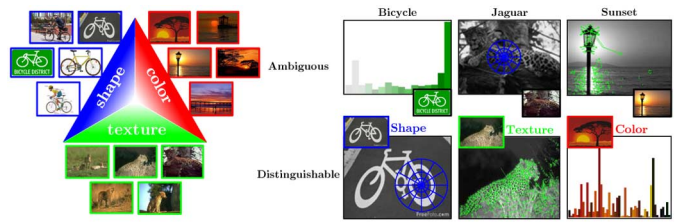

Fig. 1. Images from three different categories: `sunset`, `bicycle` and `jaguar`. Color features are discriminative for separating images in category `sunset` from the rest, while shape and texture features are discriminative for category `bicycle` and `jaguar`, respectively. It shows the importance of using multiple features and indicates that the optimal features for clustering are often cluster-dependent.

Although the development of image descriptors has gained significant progress, the general conclusion is still that there is no a single descriptor to well characterize the whole dataset in most increasingly complex applications of clustering. We thus focus on developing a clustering approach that allows the data to be characterized by multiple descriptors. The generalization is helpful for reducing the gap between the data similarities and their semantic concepts.

This motivation is illustrated by considering images in Fig. 1. The *human visual system* (HVS) can partition those images without ambiguity into three categories, `bicycle`, `sunset` and `jaguar`, because HVS can perceive and reason those images with various visual cues. However, for an automatic object/image categorization system, color related features are required to separate the images in category `sunset` from the others. Analogously, shape and texture based features are respectively needed to identify categories `bicycle` and `jaguar`. This example not only illustrates the importance of using multiple features but also points out that the optimal features for clustering are often vary from cluster to cluster. Our approach aims to boost clustering performance, and is developed by

Manuscript received September 02, 2013; revised January 27, 2014; accepted September 17, 2014. Date of publication September 23, 2014; date of current version November 13, 2014. This work was supported in part by Ministry of Science and Technology (MOST) under Grant 103-2221-E-001-026-MY2 and by Institute for Information Industry (III) under Grant 103-EC-17-A-24-1170. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. K. Selcuk Candan.

J.-T. Tsai is with the Department of Computer Science, University of Southern California, Los Angeles, CA 90089 USA (e-mail: jengtsut@usc.edu).

Y.-Y. Lin is with the Research Center for Information Technology Innovation, Academia Sinica, Taipei 115, Taiwan (e-mail: yylin@citi.sinica.edu.tw).

H.-Y. M. Liao is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan (e-mail: liao@iis.sinica.edu.tw).

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/TMM.2014.2359769

exploiting the two observations, *the use of multiple descriptors* and *cluster-dependent feature selection*.

The key idea of our approach is to integrate *multiple kernel learning* (MKL) [1]–[4] into the training process of *self-organizing map* (SOM) [5] in a *cluster-specific* manner. By cluster-specific manner, it means that we associate each cluster with a *learnable*, *ensemble* kernel. The rationale behind this idea is justified as follows. First, the recent advances in image descriptors result in diverse feature representations of data, such as vectors [6], *bag-of-features*[7], *pyramids*[8], and tensors [9]. We adopt kernel matrices to serve as the unified feature presentation for data captured by various descriptors. In this way, complementary information captured by heterogeneous descriptors can be fused in the domain of kernel matrices. Second, to account for that the optimal features for clustering are cluster-dependent, each ensemble kernel is derived to select features for best interpreting data in the corresponding clusters. Namely, data belonging to the cluster are coherent to each other and distinguishable from the rest. Third, we adapt MKL to address unsupervised clustering by incorporating it into SOM. Namely, the optimization of MKL, including kernel selection and model derivation, is performed with respect to the objective function of SOM.

Restricted by the unsupervised nature in clustering, performing cluster-dependent feature selection suffers from a *cause-and-effect* problem: learning the cluster-specific ensemble kernels needs clustering results; clustering results are derived according to the learned cluster-specific kernels. We deal with the problem by integrating the two parts into a joint optimization problem where cluster-specific ensemble kernels as well as clustering results are derived in an alternate, iterative manner, and an iterative procedure with gradient descent as the optimization technique is used in SOM. Following this framework, the proposed ideas are useful and applicable especially when a clustering technique features such cause-and-effect property, and it allows us to derive cluster-specific ensemble kernels and clustering results in a progressive way. Besides, SOM has been shown to be a useful optimization framework for clustering complex data [10]. It has been applied to a widespread applications. Generalizing SOM has the potential for advancing the applications. These mentioned above are the main reasons that we carry out cluster-dependent feature selection based on SOM.

Specifically, we formulate the training of MKL and the clustering procedure of SOM into a joint optimization problem. Note that this philosophy on the surface is similar to many MKL approaches like our previous work [11], which also involves learning an ensemble kernel, but does not support cluster-dependent feature selection. The key idea here is to tightly couple the two stages instead of casting MKL as a preprocessing step. We enable them to be mutually beneficial through the optimization iterations, where cluster structures are gradually revealed via the features specified by the corresponding ensemble kernels, while the qualities of these kernels are progressively improved owing to the more coherent clusters by enforcing SOM. Therefore, they can work together, and lead to better clustering results. We term our approach *MK-SOM (Multi-kernel Self-organizing Map)* to identify its two key components, MKL and SOM. MK-SOM can distinguish itself with the following three main contributions.

First, MK-SOM generalizes SOM to deal with data in multiple feature representations, and to select features cross different feature spaces in a cluster-specific manner. On the other hand, MK-SOM provides a new perspective of applying multiple kernel learning, which typically addresses supervised applications, to both unsupervised and semi-supervised clustering. Besides, MK-SOM interpretably partitions data in the sense that each learned ensemble kernel concretely specifies the discriminant features for the corresponding cluster. This property is precious in data analysis, descriptor design, complementary discovery, etc.

Second, MK-SOM allows the introduction of side information, i.e., *must-links* and *cannot-links*, to enhance the performance of clustering. This flexibility is important especially when the data to be partitioned become complex. The additional side information is helpful in tackling the difficulties of clustering caused by the unsupervised nature. To this end, we employ the *softmax activation function* to give the differentiable surrogate of the formulation in MK-SOM. Compared with our prior work [12], [13], it can be verified that the resulting formulation can be much more efficiently optimized by only applying gradient decent methods.

Last but not least, MK-SOM is comprehensively evaluated in two applications and compared with a set of existing clustering algorithms. The superior and promising results manifest the effectiveness of MK-SOM in accuracy, convergency speed, transparent feature selection, and the exploitation of side information.

II. Related Work

The development of clustering methods has been leading to an extensive literature. In the section, our survey emphasizes the key concepts relevant to the proposed approach.

A. Unsupervised Clustering

Clustering stems from non-labeled data analysis, since it is a core technique to uncover the underlying structure of data without any prior knowledge. According to [10], [14], [15], most unsupervised clustering methods can be divided into two categories, i.e., *hierarchical clustering* and *partitional clustering*. Methods of the former complete clustering by hierarchically leveraging the linkage relationships among data, such as single linkage, complete linkage and graph degree linkage [14], while those of the latter category determine the partitions by optimizing a certain objective function of clusters and data, such as within-cluster sum of squared error. The proposed framework belongs to the latter category.

Methods, such as k*-means*, *affinity propagation*[16] and *mean shift*[17], have been applied to a broad range of image clustering tasks. Similar to attributing the artificial neurons as clusters on the SOM in the proposed framework, there also exists other biologically inspired algorithms [15], [18] which make use of artificial characters for finding the clustering results. Though these approaches are designed with theoretic merit, their performance critically depends on the feature representation of data.

B. Clustering with Feature Selection

Features adopted to characterize data are closely related to clustering tasks, since good features facilitate the discovery

of data clusters. Therefore, a branch of research efforts has been made to couple data clustering with feature selection. For instance, methods in [19]–[21] impose the Gaussian mixture models on data distributions, while methods in [19], [22], [23] conduct discriminant analysis for subspace selection. In these methods, feature selection and data clustering are considered to be mutually beneficial, and the two steps are typically performed alternately.

Feature selection can also be done cluster-wise. Methods in [24], [25] learn a distance function for each cluster via re-weighting feature dimensions. Grozavu *et al.* [26] integrate feature selection into SOM by weighting the data samples and the distance functions to yield better prototype vector quantization. However, the aforementioned methods assume that data are linearly separable, and are only applicable to data in a single vector space. The restrictions may reduce the overall effectiveness when the data of interest can be more precisely characterized by considering multiple descriptors and diverse forms. This restriction becomes even more evident for image data clustering, because many powerful descriptors are developed in various forms, e.g., *bag-of-features* [7], [27], *pyramids*[8], [28], matrices or high order tensor [9], [29].

C. Multi-View Clustering

Multi-view clustering, e.g., [30]–[40], becomes an emerging branch of clustering methods due to its ability to take different views of data into account simultaneously. It also allows the flexibility in view definition. For example, different views correspond to different reference points in the data [39], or to diverse features that characterize the data [33], [34], [37], [40]. The seminal technique of *cluster ensembles* by Strehl and Ghosh [30] provides a useful mechanism for combining multiple clustering results. The ensemble partitioning is optimized such that it shares as much information with each of the elementary ones as possible. However, the obtained partitioning is optimized in a global fashion, neglecting the fact the optimal features are often cluster-dependent. Besides, many methods of multi-view clustering are carried out with high computational complexity and do not scale well.

D. Semi-Supervised Clustering

It is possible to deal with the unsupervised nature of clustering by introducing a small amount of labeled data into the procedure so that the quality of clustering results can be considerably boosted, especially in complex tasks. The partially labeled data can activate discriminant learning. The additional information can be utilized in various ways to enhance clustering, such as modifying the similarity matrix [41] and deriving a new distance metric [42]–[44].

E. Multiple Kernel Learning

MKL refers to learning a kernel machine with multiple kernel functions or kernel matrices. Recent advances in MKL, e.g., [1]–[4], have shown that learning with multiple kernels often increases the accuracy. In these MKL algorithms, an *ensemble kernel*, a convex combination of the input base kernels, is derived to fuse the information carried by the base kernels. Built upon this powerful foundation, there are a number of works, such as [37], [45]–[47], which achieve considerable improvement in the tasks of data clustering, where learning a single ensemble kernel to characterize the whole dataset is mainly focused on. However, the underlying structure of data is often more than complex. Instead, our approach allows learning cluster-dependent kernels to characterize data in a finer fashion, which distinguishes this work from other MKL-based clustering algorithms.

Specifically, our approach integrates MKL into the clustering procedure in the manner that each cluster is associated with a learnable ensemble kernel. Our approach combines information captured by various data descriptors in the domain of kernel matrices, and achieves cluster-specific feature selection via learning the corresponding ensemble kernels. Furthermore, we generalize our prior work [12] to exploit pairwise constraints to lead to better clustering results. To this end, the *softmax activation function* is introduced to give the differentiable surrogate of the formulation in our approach. It follows that the optimization can be effectively and efficiently achieved by simply applying gradient decent methods.

III. Preliminary

In this section, the notations used in this paper and the clustering problem to be tackled are first specified. Then, we give a brief review of the SOM, upon which our approach is developed.

A. Problem Statement

Given a dataset $D = \{\mathbf{x}_i\}_{i=1}^N$, we partition D into C clusters, i.e., $D = \bigcup_{c=1}^C \mathcal{C}_c$, $\mathcal{C}_c \neq \emptyset, \forall c$, and $\mathcal{C}_c \bigcap \mathcal{C}'_c = \emptyset, \forall c \neq c'$, with the aim that samples belonging to the same cluster are similar to each other, while those in different clusters are dissimilar. In this work, the clusters are characterized by the corresponding *prototypes* $W = \{\mathbf{w}_c\}_{c=1}^C$. Like most partition-based clustering methods, e.g., k-means, we focus on minimizing the *sum of squared error* (SSE)

$$E(D) = \sum_{i=1}^{N} \min_c \|\mathbf{x}_i - \mathbf{w}_c\|^2 \tag{1}$$

where each sample $\mathbf{x}_i$ is assigned to the nearest cluster.

In increasingly complex tasks of data or image clustering, it is difficult to find a universal descriptor to well characterize the whole dataset. We consider employing M kinds of data descriptors to represent each sample $\mathbf{x}_i \in D$. That is, $\mathbf{x}_i = \{\mathbf{x}_{i,m} \in \mathcal{X}_m\}_{m=1}^M$. Each descriptor is associated with a distance function $d_m : \mathcal{X}_m \times \mathcal{X}_m \to \mathbb{R}$ to measure the dissimilarity between data under this descriptor. Different descriptors may result in diverse forms of feature representations for image data, such as vectors [6], bags of features [48], or pyramids [8]. To avoid the difficulties caused by working with these varieties, we represent data under each descriptor by a kernel matrix. It totally leads to M kernel matrices $\{K_m \in \mathbb{R}^{N\times N}\}_{m=1}^M$ as well as the corresponding kernel functions $\{k_m : \mathcal{X}_m \times \mathcal{X}_m \to \mathbb{R}\}_{m=1}^M$:

$$K_m(i,j) = k_m(\mathbf{x}_{i,m}, \mathbf{x}_{j,m}) = \exp\left(\frac{-d_m^2(\mathbf{x}_{i,m}, \mathbf{x}_{j,m})}{\sigma_m^2}\right) \tag{2}$$

where σ_m is a positive constant. As suggested in [49], σ_m is set as the average distance among data under the mth descriptor, unless further specified.

Our approach is developed with the requirement that data are accessed by referencing only the M kernels defined in (2). One advantage of so doing is that it uses these kernels as the unified information bottleneck, and enjoys the convenience of working with arbitrary descriptors and distance measures. The other advantage is that nonlinear data clustering can be activated, and various features are fused in the domain of kernel matrices. Specifically, we integrate multiple kernel learning into the clustering procedure, and associate each cluster with an ensemble kernel, a convex combination of the M base kernels in (2). It carries out *cluster-dependent feature selection* in the sense that each cluster is allowed to select the most plausible feature combination by optimizing the associated ensemble kernel.

The performance of clustering is often restricted by the unsupervised nature. To tackle this issue, our approach also supports the use of a few *must-links* and *cannot-links*, each of which specifies either a pair of data samples must reside in the same cluster or not. These links provide the additional constraints to guide the clustering process.

B. Self-Organizing Map (SOM)

The SOM [5] can work with the objective function given in (1) by associating a neuron with each cluster prototype c and parameterizing it by $\mathbf{w}_c$. The SOM is typically optimized by gradient descent in an iterative manner. At iteration t, all data in D are sequentially picked as input to update the neurons. With input $\mathbf{x}_i$, the nearest neuron, the so-called *winner neuron*, is first determined by Euclidean distance. Suppose the winner neuron is the cth neuron. It is updated by moving closer to $\mathbf{x}_i$ with offset

$$\Delta \mathbf{w}_c = \eta^t \mathcal{N}_c^t (\mathbf{x}_i - \mathbf{w}_c) \tag{3}$$

where η is the *learning rate* and $\mathcal{N}_c^t = \exp(\frac{-\|\mathbf{x}_i - \mathbf{w}_c\|^2}{R^t})$ is the *neighborhood kernel* centered on the cth neuron.

The SOM adopts a *coarse-to-fine* strategy to better optimize the neurons. Both the learning rate η^t and the hyperparameter R in the neighborhood kernel decrease monotonically along optimization procedure, and are updated iteratively by

$$\eta^{t+1} \leftarrow \mu \eta^t \quad \text{and} \quad R^{t+1} \leftarrow \nu R^t \tag{4}$$

where μ and ν are two positive constants. We set μ and ν as 0.85 and 1 in all the experiments, respectively. The iterative procedure is repeated until convergence or the number of maximum iterations is reached.

The kernelized SOM (or *kernel SOM*) [50]–[52] has been developed and applied to nonlinear data clustering. We further generalize SOM to work with multiple kernels. The proposed approach integrates the cluster-specific multiple kernel learning into the clustering procedure, and cast them as energy minimization problem. Through the optimization iterations, discriminant features for each cluster are gradually derived and selected, while the cluster structure is revealed by these features.

IV. MK-SOM Framework

The proposed MK-SOM framework is described in the section. First, we show how MK-SOM generalizes kernel SOM to perform cluster-dependent feature selection via multiple kernel learning. Subsequently, pairwise constraints are introduced into MK-SOM to boost the clustering performances. Finally, we detail the optimization of MK-SOM.

A. Learning SOM with Multiple Kernels

To leverage the rich information captured by various data descriptors, we extend the SOM for coping with cluster-dependent feature selection by multiple kernel learning.

To begin with, we consider kernel SOM. Let $\phi : \mathcal{X} \mapsto \mathcal{F}$ denote the feature map induced by a kernel K. It transforms data from input domain X to Reproduced Kernel Hilbert Space (RKHS) $\mathcal{F}$, i.e.,

$$\mathbf{x}_i \mapsto \phi(\mathbf{x}_i), \quad \text{for } i = 1, 2, \ldots, N. \tag{5}$$

The objective function of kernel SOM is then expressed by

$$E_{\mathrm{KSOM}} = \sum_{i=1}^{N} \min_c \|\phi(\mathbf{x}_i) - \mathbf{w}_c\|^2. \tag{6}$$

It can be proved by contradiction that the optimal $\mathbf{w}_c$ in (6) must lie in the span of the mapped data, i.e.,

$$\mathbf{w}_c = \sum_{n=1}^{N} \alpha_{c,n} \phi(\mathbf{x}_n) \tag{7}$$

where $\{\alpha_{c,n}\}_{n=1}^N$ are the sample coefficients of neuron $\mathbf{w}_c$. It follows that objective function (6) can be further expanded as

$$\begin{aligned} & E_{\mathrm{KSOM}}(D) \\ &= \sum_{i=1}^{N} \min_c \left\| \phi(\mathbf{x}_i) - \sum_{n=1}^{N} \alpha_{c,n} \phi(\mathbf{x}_n) \right\|^2 && (8) \\ &= \sum_{i=1}^{N} \min_c \left[\phi^\top p(\mathbf{x}_i) \phi(\mathbf{x}_i) 2 \sum_{n=1}^{N} \alpha_{c,n} \phi^\top p(\mathbf{x}_i) \phi(\mathbf{x}_n) \right. \\ &\quad \left. + \sum_{n=1}^{N} \sum_{n'=1}^{N} \alpha_{c,n} \alpha_{c,n'} \phi^\top p(\mathbf{x}_n) \phi(\mathbf{x}_{n'}) \right] && (9) \\ &= \sum_{i=1}^{N} \min_c \left[K(i,i) - 2 \boldsymbol{\alpha}_c^\top p K(:,i) + \boldsymbol{\alpha}_c^\top p K \boldsymbol{\alpha}_c \right] && (10) \end{aligned}$$

where $\boldsymbol{\alpha}_c = [\alpha_{c,1} \alpha_{c,2} \cdots \alpha_{c,N}]^\top \in \mathbb{R}^N$ and $K(:,i) \in \mathbb{R}^N$ is the ith column of kernel matrix K.

Like most MKL approaches, such as [1]–[4], [11], we treat an ensemble kernel as a convex combination of the M base kernels. In this work, we propose to associate each cluster with a learnable ensemble kernel. The ensemble kernel associated to the cth cluster is

$$K^{(c)} = \sum_{m=1}^{M} \beta_{c,m} K_m, \tag{11}$$

where $\{K_m\}_{m=1}^M$ are the base kernels and $\{\beta_{c,m}\}_{m=1}^M$ are the learnable kernel weights. It follows that the discriminant features for each cluster can be selected across different descriptors

and merged to compose the ensemble kernel. Generalized from (10), the objective function of MK-SOM is defined as follows:

$$E_{\mathrm{MK-SOM}}(D) = \sum_{i=1}^{N} \min_{c} [K^{(c)}(i,i) - 2\boldsymbol{\alpha}_c^\top K^{(c)}(:,i) + \boldsymbol{\alpha}_c^\top K^{(c)} \boldsymbol{\alpha}_c]. \quad (12)$$

The resulting constrained optimization problem of MK-SOM becomes

$$\min_{\{\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c\}_{c=1}^{C}} E_{\mathrm{MK-SOM}}(D) \quad (13)$$

$$\text{subject to} \quad \sum_{m=1}^{M} \beta_{c,m} = 1, \quad \text{for } c = 1, \ldots, C \quad (14)$$

$$\beta_{c,m} \geq 0, \ \forall \beta_{c,m}. \quad (15)$$

In constrained optimization problem (13), we optimize the neurons $\{\mathbf{w}_c\}_{c=1}^{C}$, where $\mathbf{w}_c$ is parameterized by a *sample coefficient vector* $\boldsymbol{\alpha}_c = [\alpha_{c,1} \cdots \alpha_{c,N}]^\top \in \mathbb{R}^N$ and a *kernel weight vector* $\boldsymbol{\beta}_c = [\beta_{c,1} \cdots \beta_{c,M}]^\top \in \mathbb{R}^M$. For each cluster c, $\boldsymbol{\alpha}_c$ and $\boldsymbol{\beta}_c$ jointly specify how the ensemble kernel and the cluster boundary are constructed. Hence, cluster-dependent feature selection across heterogeneous descriptors is allowed in the formulation (13). For the sake of clearness, the optimization of (13) is described in Section IV-C.

B. Incorporating Pairwise Constraints in SOM

Suppose we are given a set of pairwise constraints, each of which is either a must-link or a cannot-link. These constraints are precious, since they can activate supervised learning in unsupervised clustering tasks. Specifically, we propose an *associate-separate* model to deal with each pairwise constraint. It first associates each of the two samples in a link to the most plausible neuron, and then separate the other neurons from the sample.

Must-links: Suppose that a set of must-links, $S = \{(\mathbf{x}_i, \mathbf{x}_j)\}$, is given. For each must-link $(\mathbf{x}_i, \mathbf{x}_j)$, the steps of association and separation are depicted in the following.

On Association: Since samples $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same cluster, we conduct the cluster association by searching shared cluster $\mathcal{C}_{\pi_i}$ that is averagely closest to the two samples, i.e.,

$$\mathcal{C}_{\pi_i} = \arg\min_{\pi_i} \|\phi(\mathbf{x}_i) - \mathbf{w}_{\pi_i}\|^2 + \|\phi(\mathbf{x}_j) - \mathbf{w}_{\pi_i}\|^2. \quad (16)$$

For the ease of expression, we simply let $\mathcal{C}_{\pi_j} = \mathcal{C}_{\pi_i}$.

On Separation. Note that $\mathcal{C}_{\pi_i}$ in (16) may not be the nearest cluster to either $\mathbf{x}_i$ or $\mathbf{x}_j$. Thus, discriminant learning can be applied to separate the rest clusters from the two samples. Specifically, the following energy function is considered:

$$\tilde{J}(\mathbf{x}_k, \mathcal{C}_{\pi_k}) = \|\phi(\mathbf{x}_k) - \mathbf{w}_{\pi_k}\|^2 - \min_{p \neq \pi_k} \|\phi(\mathbf{x}_k) - \mathbf{w}_p\|^2 \quad (17)$$

where $k \in \{i, j\}$. It is clear that $\tilde{J}(\mathbf{x}_k, \mathcal{C}_{\pi_k}) \leq 0$ ensures $\mathbf{x}_k$ residing in $\mathcal{C}_{\pi_k}$. However, the $\min$ operation makes $\tilde{J}(\mathbf{x}_k, \mathcal{C}_{\pi_k})$ non-differentiable, and is difficult to be incorporated into SOM, which is optimized by gradient descent methods. To address this issue, we introduce the *softmax activation function* (or the log-sum-exp trick). It gives the differentiable surrogate of $\min$ operation by

$$\min_{p \neq \pi_k} \|\phi(\mathbf{x}_k) - \mathbf{w}_p\|^2 \approx -\frac{1}{\gamma} \log \left(\sum_{p \neq \pi_k} \exp(-\gamma \|\phi(\mathbf{x}_k) - \mathbf{w}_p\|^2) \right) \quad (18)$$

where the *smoothness parameter* γ is a positive constant, and is used to control the degree of precision in approximation. We empirically set $\gamma = 2^8$, which gives sufficiently good approximation. By substituting (18) into (17), we have

$$\tilde{J}(\mathbf{x}_k, \mathcal{C}_{\pi_k}) \approx J(\mathbf{x}_k, \mathcal{C}_{\pi_k}) = \|\phi(\mathbf{x}_k) - \mathbf{w}_{\pi_k}\|^2 + \frac{1}{\gamma} \log \left(\sum_{p \neq \pi_k} \exp(-\gamma \|\phi(\mathbf{x}_k) - \mathbf{w}_p\|^2) \right). \quad (19)$$

The must-link $(\mathbf{x}_i, \mathbf{x}_j)$ eventually induces a pair of the log-loss functions

$$L_{\mathrm{m-lnk}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k \in \{i,j\}} \log\left(1 + \exp(J(\mathbf{x}_k, \mathcal{C}_{\pi_k}))\right). \quad (20)$$

Cannot-Links: The associate-separate model for a set of *cannot-links*, $S' = \{(\mathbf{x}_i, \mathbf{x}_j)\}$, is detailed in the following.

On Association. We first conduct the cluster association for a cannot-link $(\mathbf{x}_i, \mathbf{x}_j)$. As specified by the cannot-link, we associate $\mathbf{x}_i$ and $\mathbf{x}_j$ to two different clusters by

$$(\mathcal{C}_{\pi_i}, \mathcal{C}_{\pi_j}) = \arg\min_{\pi_i \neq \pi_j} \|\phi(\mathbf{x}_i) - \mathbf{w}_{\pi_i}\|^2 + \|\phi(\mathbf{x}_j) - \mathbf{w}_{\pi_j}\|^2. \quad (21)$$

On Separation. Like the separation stage for dealing with must-links, discriminant learning is activated such that each sample in the link can reside in the associated cluster and be far apart from the rest clusters. Similar to (20), the loss function for a cannot-link $(\mathbf{x}_i, \mathbf{x}_j)$ is defined as

$$L_{\mathrm{c-lnk}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k \in \{i,j\}} \log(1 + \exp(J(\mathbf{x}_k, \mathcal{C}_{\pi_k}))). \quad (22)$$

Note that two distinct clusters are associated to samples in a cannot-link, i.e., $\mathcal{C}_{\pi_i} \neq \mathcal{C}_{\pi_j}$, while a common cluster is associated to samples in a must-link.

With the designed loss functions for must-links (20) and cannot-links (22), the objective function pertaining to pairwise constraints is given as follows:

$$E_{\mathrm{LNK}}(S, S') = \sum_{(\mathbf{x}_i, \mathbf{x}_j) \in S} L_{\mathrm{m}-lnk}(\mathbf{x}_i, \mathbf{x}_j) + \sum_{(\mathbf{x}_i, \mathbf{x}_j) \in S'} L_{\mathrm{c}-lnk}(\mathbf{x}_i, \mathbf{x}_j). \quad (23)$$

By incorporating (23) into (13), the objective function of the proposed approach becomes

$$\min_{\{\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c\}_{c=1}^{C}} E_{\mathrm{MK-SOM}}(D) + \lambda E_{\mathrm{LNK}}(S, S') \quad (24)$$

$$\text{subject to} \quad \sum_{m=1}^{M} \beta_{c,m} = 1, \ \text{for } c = 1, \ldots, C \quad (25)$$

$$\beta_{c,m} \geq 0,\ \forall \beta_{c,m} \tag{26}$$

where λ is a positive constant.

C. Optimization Procedure

There are two cluster-dependent variables to be learned in (24), including the sample coefficient vectors $\{\boldsymbol{\alpha}_c\}_{c=1}^{C}$ and the kernel weight vectors $\{\boldsymbol{\beta}_c\}_{c=1}^{C}$. Our approach inheriting from SOM adopts an iterative optimization procedure. At each iteration, these variables are sequentially updated with respect to each data sample and each pairwise constraint. The iterative procedure is repeated until convergence. Namely, the objective function in (24) can not be further reduced. Owing to the complication of (24), it is difficult to simultaneously solve $\{\boldsymbol{\alpha}_c\}$ and $\{\boldsymbol{\beta}_c\}$. We hence optimize one of the two variables by fixing the other, and then switch their roles.

To start the iterative optimization procedure, $\{\boldsymbol{\alpha}_c\}$ are randomly drawn from the uniform distribution on the interval $[-1/N, 1/N]$, while $\{\boldsymbol{\beta}_c\}$ are averagely distributed in initialization such that the constraints (25) and (26) hold. We describe how the two sets of variables are updated for each sample and each pairwise constraint in the following.

Algorithm 1: *The Training Procedure of MK-SOM*

Input: Dataset $D = \{\mathbf{x}_i\}_{i=1}^{N}$ in the form of M kernel matrices $\{K_m \in \mathbb{R}^{N\times N}\}_{m=1}^{M}$ (cf. (2)); Must-link set $S = \{(\mathbf{x}_i, \mathbf{x}_j)\}$; Cannot-link set $S' = \{(\mathbf{x}_i, \mathbf{x}_j)\}$; Number of clusters C;

Output: Sample coefficient vectors $\{\boldsymbol{\alpha}_c\}_{c=1}^{C}$; Kernel weight vectors $\{\boldsymbol{\beta}_c\}_{c=1}^{C}$;

Initialization: $\{\alpha_{c,n}\}_{c=1,n=1}^{C,N}$ are randomly drawn on the interval $[-1/N, 1/N]$; $\{\beta_{c,m}\}_{c=1,m=1}^{C,M}$ are set as $1/M$;

while *not converged* **do**

for *each data sample* $\mathbf{x}_i$ **do**

1. Determine the winner neuron c by (27);
2. Update $\boldsymbol{\alpha}_c$ by steepest gradient descent (cf. (29));
3. Update $\boldsymbol{\beta}_c$ by reduced gradient descent (cf. (33));

for *each must-link* $(\mathbf{x}_i, \mathbf{x}_j)$ **do**

1. Associate samples $\mathbf{x}_i$ and $\mathbf{x}_j$ with a common cluster $\mathcal{C}_{\pi_i}$ by (16);
2. Update $\{\boldsymbol{\alpha}_c\}_{c=1}^{M}$ and $\{\boldsymbol{\beta}_c\}_{c=1}^{M}$ with gradient in (35) and (37), respectively;

for *each cannot-link* $(\mathbf{x}_i, \mathbf{x}_j)$ **do**

1. Associate samples $\mathbf{x}_i$ and $\mathbf{x}_j$ with clusters $\mathcal{C}_{\pi_i}$ and $\mathcal{C}_{\pi_j}$ respectively by (16);
2. Update $\{\boldsymbol{\alpha}_c\}_{c=1}^{M}$ and $\{\boldsymbol{\beta}_c\}_{c=1}^{M}$ with gradient in (35) and (37), respectively;

Check convergency;

Assign each data sample to the winner neuron to complete clustering;

Variable Update w.r.t. Data Samples: The parameters $\{\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c\}_{c=1}^{C}$ of the C neurons are updated with respect to incoming data samples in the stage. For each sample $\mathbf{x}_i \in D$, it can be observed that only $E_{\mathrm{MK-SOM}}(D)$ in the objective function (24) is relevant to a single sample $\mathbf{x}_i$. Hence, we first find the winner neuron c by

$$c = \arg\min_{c} \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2 \tag{27}$$

$$= \arg\min_{c} K^{(c)}(i,i) + 2\boldsymbol{\alpha}_c^{\top} pK^{(c)}(:,i) + \boldsymbol{\alpha}_c^{\top} pK^{(c)}\boldsymbol{\alpha}_c \tag{28}$$

where $K^{(c)}$, defined in (11), is the ensemble kernel associated with the cth neuron. Then, the parameters $\boldsymbol{\alpha}_c$ and $\boldsymbol{\beta}_c$ in the constrained optimization problem (24) are updated w.r.t. $\mathbf{x}_i$ as follows:

On Updating $\boldsymbol{\alpha}_c$: By fixing $\boldsymbol{\beta}_c$, the steepest gradient descent is applied to seek the element-wise update of $\boldsymbol{\alpha}_c = [\alpha_{c,1}\alpha_{c,2}\cdots\alpha_{c,N}]^{\top}$ by

$$\alpha_{c,n} \leftarrow \alpha_{c,n} - \eta \frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \alpha_{c,n}} \tag{29}$$

where

$$\frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \alpha_{c,n}} = -2[K^{(c)}(n,i) - \boldsymbol{\alpha}_c^{\top} K^{(c)}(:,n)]. \tag{30}$$

On Updating $\boldsymbol{\beta}_c$: By fixing $\boldsymbol{\alpha}_c$, we update $\boldsymbol{\beta}_c$ with respect to $\mathbf{x}_i$. The additional constraints in (25) and (26) cause that $\boldsymbol{\beta}_c$ can no longer be optimized by steepest gradient descent. We overcome this problem by employing the *reduced gradient descent*[4], which is developed to cope with constraints in the procedure of gradient descent.

Like other gradient descent methods, the partial derivatives of $\boldsymbol{\beta}_c = [\beta_{c,1}\cdots\beta_{c,M}]^{\top}$ in reduced gradient descent are computed by

$$\frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \beta_{c,m}} = K_m(i,i) - 2\boldsymbol{\alpha}_c^{\top} K_m(:,i) + \boldsymbol{\alpha}_c^{\top} K_m \boldsymbol{\alpha}_c. \tag{31}$$

The reduced gradient descent updates the optimization variables in a *relative* manner such that the equality constraints in (25) can be satisfied. Suppose $\beta_{c,\mu}$ is the largest element in $\boldsymbol{\beta}_c$. The *reduced gradient* $\mathbf{r}_c = [r_{c,1}\cdots r_{c,M}]^{\top}$ is calculated by

$$r_{c,m} = \begin{cases} \sum_{v\neq u} \left(\frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \beta_{c,v}} - \frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \beta_{c,\mu}}\right), & \text{if } m = \mu \\ -\frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \beta_{c,m}} + \frac{\partial \|\mathbf{w}_c - \phi(\mathbf{x}_i)\|^2}{\partial \beta_{c,\mu}}, & \text{otherwise.} \end{cases} \tag{32}$$

With the reduced gradient $\mathbf{r}_c$, $\boldsymbol{\beta}_c$ is updated by

$$\boldsymbol{\beta}_c \leftarrow \boldsymbol{\beta}_c + \tau \mathbf{r}_c. \tag{33}$$

The equality constraints in (25) still hold after update because of $\sum_{m=1}^{M} r_{c,m} = 0$. On the other hand, the key to the satisfaction of the inequality constraints in (26) is the step size $\tau \in [0, \tau_{max}]$, found by using line search, with the maximum step size

$$\tau_{max} = \min_{\{m | r_{c,m} < 0\}} -\frac{\beta_{c,m}}{r_{c,m}}. \tag{34}$$

Variable Update w.r.t. Pairwise Constraints: We alternately update $\{\boldsymbol{\alpha}_c\}$ and $\{\boldsymbol{\beta}_c\}$ for the incoming pairwise constraints

by minimizing $E_{\mathrm{LNK}}(S, S')$ in (24). It should be noted that there is a major difference between the updates with respect to a data sample and a pairwise constraint. For a pairwise constraint, either a must-link or a cannot-link, we perform discriminant learning over all the neurons, instead of the winner neuron. Thus, $\{\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c\}_{c=1}^{C}$ will be updated for each link.

Consider a pairwise constraint $(\mathbf{x}_i, \mathbf{x}_j) \in S \bigcup S'$. It can be observed in (20) and (22) that all the linkages share an identical form of the loss function, i.e., $L(\mathbf{x}_i) = \log(1 + \exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i})))$. It suffices to derive the partial derivatives for sample $\mathbf{x}_i$, since those for $\mathbf{x}_j$ can be analogously inferred.

On Updating $\{\boldsymbol{\alpha}_c\}$. The partial derivatives of $L(\mathbf{x}_i)$ of variables $\{\boldsymbol{\alpha}_c\}_{c=1}^{C}$ can be derived as follows:

$$\frac{\partial L(\mathbf{x}_i)}{\partial \boldsymbol{\alpha}_c} = \begin{cases} \frac{\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}{1+\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))} \times 2(K^{(c)}\boldsymbol{\alpha}_c - K^{(c)}(:, i)), \\ \qquad \text{if} \quad \mathrm{c} = \pi_{\mathrm{i}}, \\ \frac{\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}{1+\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))} \times -2(K^{(c)}\boldsymbol{\alpha}_c - K^{(c)}(:, i)) \\ \quad \times \frac{\exp(-\gamma V(i, c))}{\sum_{p \neq \pi_i} \exp(-\gamma V(i, p))}, \quad \text{otherwise.} \end{cases} \tag{35}$$

where

$$\begin{aligned} V(i, p) &= \|\phi(\mathbf{x}_i) - \mathbf{w}_p\|^2 \\ &= K^{(p)}(i, i) - 2\boldsymbol{\alpha}_p^{\top} K^{(p)}(:, i) + \boldsymbol{\alpha}_p^{\top} K^{(p)} \boldsymbol{\alpha}_p. \end{aligned} \tag{36}$$

Although the partial derivatives in (35) looks complicated, they have intuitive justification. The first term $\frac{\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}{1+\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}$ controls the *magnitude*. Larger $J(\mathbf{x}_i, \mathcal{C}_{\pi_i})$ in (19) implies that $\mathbf{x}_i$ tends not to belong to the associated cluster $\mathcal{C}_{\pi_i}$. Thus update with larger magnitude is required. The second term $(K^{(c)}\boldsymbol{\alpha}_c - K^{(c)}(:, i))$ specifies the *direction*. The associated cluster $\mathcal{C}_{\pi_i}$ is updated toward sample $\mathbf{x}_i$, while the rest clusters are kept far away. The third term $\frac{\exp(-\gamma V(i,c))}{\sum_{p \neq \pi_i} \exp(-\gamma V(i,p))}$ presents only for clusters that are not associated with $\mathbf{x}_i$. It can be treated as the *normalized weight*. As shown in (36), the clusters closer to $\mathbf{x}_i$ are given larger weights. With the partial derivative given in (35), $\{\boldsymbol{\alpha}_c\}_{c=1}^{C}$ are then updated.

On Updating $\{\boldsymbol{\beta}_c\}$: The partial derivatives of $L(\mathbf{x}_i)$ of variables $\{\boldsymbol{\beta}_c = [\beta_{c,1} \cdots \beta_{c,M}]\}_{c=1}^{C}$ can be similarly derived as follows:

$$\frac{\partial L(\mathbf{x}_i)}{\partial \beta_{c,m}} = \begin{cases} \frac{\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}{1+\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}(K_m(i, i) - 2\boldsymbol{\alpha}_c^{\top} K_m(:, i) + \boldsymbol{\alpha}_c^{\top} K_m \boldsymbol{\alpha}_c), \\ \qquad \text{if} \quad \mathrm{c} = \pi_{\mathrm{i}}, \\ \frac{\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))}{1+\exp(J(\mathbf{x}_i, \mathcal{C}_{\pi_i}))} - (K_m(i, i) - 2\boldsymbol{\alpha}_c^{\top} K_m(:, i) + \boldsymbol{\alpha}_c^{\top} K_m \boldsymbol{\alpha}_c) \\ \quad \times \frac{\exp(-\gamma V(i,c))}{\sum_{p \neq \pi_i} \exp(-\gamma V(i,p))}, \quad \text{otherwise.} \end{cases} \tag{37}$$

The gradient derived in (37) has similar justification to that in (35). We omit here owing to the similarity.

We summarize the training procedure of the proposed MK-SOM in Algorithm 1. After initialization, optimization variables $\{\boldsymbol{\alpha}_c, \boldsymbol{\beta}_c\}_{c=1}^{C}$ are sequentially updated with respect to each data sample and each pairwise constraint. Since the methods of gradient descent are employed in the optimization procedure, the objective function (24) decreases monotonically. That is, the iterative optimization procedure must converge. We terminate optimization once the value of the objective function can not be reduced further. The task of data clustering is then completed by assigning data samples to the corresponding winner neurons.

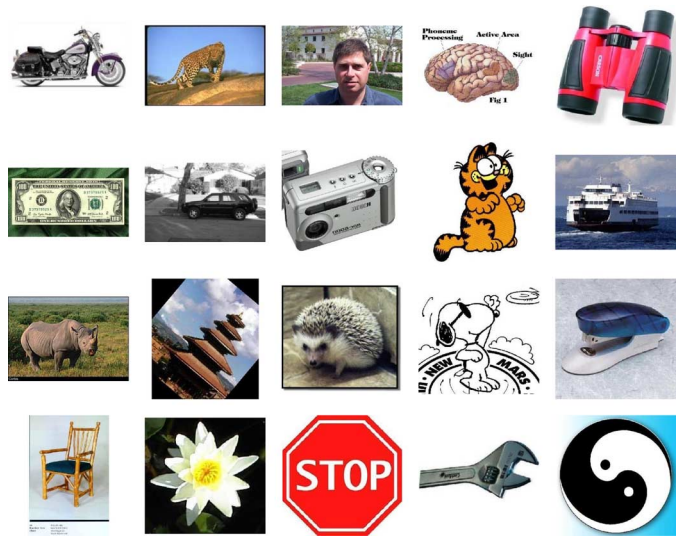

Fig. 2. Example objects selected from the 20 categories used in the experiments.

V. Experimental Results

In this section, we evaluate the performance of MK-SOM by applying it to two clustering tasks, including visual object categorization and face image grouping. In visual object categorization, the issues of initialization, convergence, and heterogeneous feature fusion in MK-SOM are the focal points. In face image grouping, we in addition demonstrate the effectiveness and advantages of MK-SOM for cluster-dependent feature selection. As the data used in the two tasks are more challenging, we analyze the effect of introducing pairwise constraints in MK-SOM.

For the configuration of SOM, because we aim to derive an ensemble kernel for each cluster, and each ensemble kernel is associated with a neuron, we consider a special case of SOM where the number of neurons equals to that of clusters. However, our approach (summarized in Algorithm 1) can be directly generalized to a fix map space, say 10×10 neurons. For the performance evaluation, two criteria are adopted: *clustering accuracy* (ACC) [22] and *normalized mutual information* (NMI) [30]. The output domains of them are both $[0, 1]$. The larger the values, the better the clustering results are. In all the experiments, we set the number of clusters, i.e., C, to the number of classes in ground truth.

A. Visual Object Categorization

In the second experiment, our approach is applied to object categorization. The Caltech-101 dataset, which was collected by Fei-Fei *et al.* [53], is used as the test bed. Clustering in this dataset is very challenging due to the large and diverse intra-class variations. Following the setting in [54], we select the same 20 object categories from Caltech-101. We randomly

TABLE I
THE PERFORMANCES IN FORM OF [ACC (%)/NMI] OF DIFFERENT CLUSTERING METHODS FOR OBJECT CATEGORIZATION

Single kernel	k-means	AP	SC	Ours
GB	52.8 / 0.578	50.7 / 0.580	69.7 / **0.726**	**72.2** / **0.726**
FH	34.0 / 0.390	45.5 / 0.503	**54.8** / **0.578**	53.4 / 0.571
GIST	36.3 / 0.430	49.2 / 0.515	54.8 / 0.592	**57.8** / **0.596**
PHOG	44.0 / 0.448	42.7 / 0.454	**60.2** / **0.600**	55.7 / 0.574
SS	46.5 / 0.526	55.5 / 0.572	63.8 / **0.655**	**63.9** / 0.654
AvgKernel	52.0 / 0.542	60.0 / 0.622	70.3 / 0.704	**73.8** / **0.732**

All kernels	CE + k-means	CE + AP	CE + SC	CoReg
	55.3 / 0.554	52.8 / 0.545	71.5 / 0.705	66.2 / 0.695
	BCE + k-means	BCE + AP	BCE + SC	MKL-DR
	65.7 / 0.475	69.0 / 0.494	73.8 / 0.652	64.2 / 0.720
				Ours
				81.1 / **0.822**

pick 30 images from each category to form a set of 600 images. Fig. 2 shows the 20 selected categories.

Descriptors and Base Kernels: Five different image descriptors are used for object feature extraction. They respectively result in the following five kernels.

- **GB:** For an object image, we randomly sample 400 edge pixels and characterize them using geometric blur descriptor [27]. We construct the dissimilarity-based kernel based on the distance function suggested in equation (2) of the work by Zhang *et al.* [48].
- **FH:** Mutch and Lowe [6] have proposed a set of features which emulate the mechanism of human visual system. These biologically inspired features are adopted to establish an RBF kernel.
- **GIST:** Images are resized to 128×128 pixels prior to applying the GIST descriptor [55]. Then, an RBF kernel is constructed.
- **PHOG:** The PHOG descriptor [28] is employed to capture image features. Together with the χ^2 distance, the kernel is built.
- **SS:** The self-similarity descriptor [56] is considered over an evenly sampled grid of each image, and then we use k-means clustering to generate visual words from the resulting local features of all images. Subsequently, the kernel is obtained by matching spatial pyramids as introduced in [8].

Baselines: We compare our approach, MK-SOM, with several powerful clustering algorithms. MK-SOM can work with one kernel and multiple kernels. For comparison in the cases where a single kernel is considered, k-means, *spectral clustering* (SC) [57], [58], and *affinity propagation* (AP) [16] are used. Specifically, we use the kernelized variant of k-means, i.e., kernel k-means. Thus, SC, kernel k-means and our approach take a kernel matrix as input. AP detects clusters by considering similarities among data. We set the pairwise similarities as the corresponding elements of the kernel matrix. For comparison in the cases where multiple kernels are jointly considered, *cluster ensembles* (CE) [30] and *Bayesian cluster ensemble* are adopted. Both CE and BCE aim at merging a number of clustering results to yield a better one. In addition, a multi-view clustering algorithm [37], called *co-regularized multi-view spectral clustering* (CoReg for short), is included in the experiments for comparison. It looks for clusterings that are consistent across data views. We adopted centroid based CoReg owing to its good performance. Furthermore, we also observe how MK-SOM could be distinguished from traditional MKL-based clustering method without cluster-dependent feature selection. We compare it with MKL-DR [11], where MKL-LPP is used that projects data in forms of kernels into a low-dimensional space based on the objective function of LPP. Clustering is then achieved by applying k-means to the projected data. In the experiments, the target numbers of clusters in all the clustering methods are set as the number of clusters in ground truth for fair comparison. Note that the number of clusters in AP is determined by the *preference value*, which is tuned with a bisection method so that the number of the yielded clusters is the same as that in ground truth.

Quantitative Results: In this experiment, we compare our approach with the state-of-the-art clustering algorithms for object categorization, and report the clustering performance in Table I. When each of the five kernels is individually used, it can be observed that the kernels critically determine the performance. Shape is a discriminant characteristic for object description. The kernel **GB** is developed to capture the shape features of objects, and gives the best performance. The clustering results by our approach are consistently better or equivalent to those by k-means, AP, and SC.

As for the cases where multiple kernels are considered jointly, our approach achieves remarkable improvement of $8.9\%(= 81.1\% - 72.2\%)$ in ACC and $0.096(= 0.822 - 0.726)$ in NMI over the best result obtained with the kernel **GB**. On the other hand, CE and BCE fuse multiple clustering results in a global manner without exploring cluster-specific properties. The quantitative results show that our method can make the most of multiple kernels, and outperforms CE, BCE, CoReg and MKL-DR.

Initialization and Convergency: Initialization and Convergency are two important issues for an alternate optimization procedure. We here test our method with different initializations, and check whether it converges and the performance variation with those initializations. Different initializations are generated in two ways. First, we give different sample coefficient vectors $\{\boldsymbol{\alpha}_c\}$ in initialization in the first way. In the other way, we set the winner neuron of each data sample at the first iteration according to the outcome of k-means, AP, and SC. They are detailed as follows:

Coefficients: As depicted in Algorithm 1, our method begins with the randomly generated sample coefficient vectors $\{\boldsymbol{\alpha}_c\}$

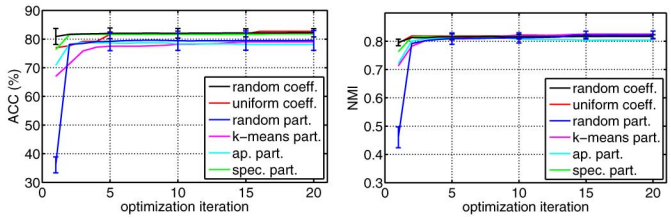

Fig. 3. With different initializations, the clustering accuracy (**left**) and normalized mutual information (right) of our approach along the iterative optimization on object categorization.

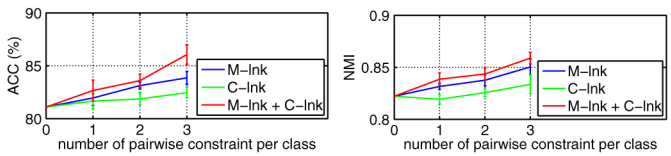

Fig. 4. Performance of MK-SOM with different numbers of pairwise constraints in object categorization.

on the interval $[-1/N, 1/N]$. We denote this setting as *random-coeff.*, and it is run five times to compute the average performance. In addition, we also have *uniform coeff.*, in which each element in $\{\alpha_{c,i}\}$ is assigned to $1/N$ in initialization.

Partitions: In addition to different initial coefficients, we also examine whether initial *data partitions* are crucial in the proposed approach because this is the issue in many partitional clustering methods. To this end, we consider several initial data partitions, including the clustering results by k-means (*k-means part.*), AP (*ap part.*), SC (*spec. part.*), and five random data partitions (*random part.*). Our approach works with each partition by setting the winner neurons of data samples as those indicated by the partition at the first iteration. The winner neuron is found in the original manner since the second iteration. Note that the sample coefficients in the cases are set uniformly, i.e., $1/N$, to fairly investigate the effects caused by initial partitions.

The clustering performances through the iterative optimization procedure are plotted in Fig. 3. It can be observed that the proposed optimization algorithm is quite *efficient* and *robust*: It converges within a few, about five, iterations and yields similar performance with diverse initializations, especially when NMI is the performance measure.

Clustering with Pairwise Constraints: We randomly generate k musk-links and k cannot-links for each object category, and adopt three experiment settings, in which must-links, cannot-links, and both of them are used respectively. We evaluate the performance by setting $k = 1 \sim 3$ respectively. Each setting is repeated five times with different randomly sampled links, and the average performance is computed. Fig. 4 summarizes the performance in terms of the mean and the standard deviation. Apparently, a few pairwise constraints benefit our approach. It suggests that our formulation can effectively exploit the information carried by the pairwise constraints, and leads to remarkable improvement. It is also worth to note that must-links are more informative than cannot-links, since the candidates of cannot-links are abundant. Nevertheless, cannot-links are helpful to discover inter-cluster variations. Therefore, using must-links as well as cannot-links results in larger improvement.

Fig. 5. Four kinds of intraclass variations caused by: (a) different lighting conditions; (b) in-plane rotations; (c) partial occlusions; and (d) out-of-plane rotations.

Fig. 6. Images obtained by applying the delighting algorithm [60] to the five images in Fig. 5(a). Clearly, variations caused by different lighting conditions are alleviated.

B. Face Image Grouping

In the last experiment, we apply MK-SOM to face image grouping, where the cluster-dependent feature selection is the focal point. The CMU PIE dataset [59], which consists of face images of 68 subjects, is adopted in the experiment. We divide them into 4 disjoint groups of equal size. Each group includes face images of 17 subjects and is attributed as a certain kind of variations. An overview can be found in Fig. 5. Specifically, for each subject in the first group, we consider only the images of the frontal pose (C27) taken in varying lighting conditions (those under the directory "lights"). For subjects in the second and third groups, the images with near frontal poses (C05, C07, C09, C27, and C29) under the directory "expression" are used. While each image from the second group is rotated by a randomly sampled angle within $[-45°, 45°]$, each from the third group is instead occluded by a nonface patch whose area is about 10% of the face region. Finally, for subjects in the fourth group, the images with out-of-plane rotations are selected under the directory "expression" and with the poses (C05, C11, C27, C29, and C37). All images are cropped and resized to 51×51 pixels. We term the four groups by the corresponding variations, i.e., *lighting*, *rotation*, *occlusion*, and *profile*, respectively.

Descriptors and Base Kernels: Compared to the clustering task performed on the Caltech-101 dataset, it becomes more challenging on the CMU PIE dataset, since different types of intra-class variations are involved. We hence need distinctive descriptors to handle the unfavorable variations. Although designing more powerful image descriptors gain significant progress in the field of computer vision, there is still no a universal descriptor to overcome all the four types of variations. Thus, multiple descriptors are required, and each of them addresses one or a few types of variations. Group-dependent, or even sub-dependent, feature selection in this dataset is

TABLE II
CLUSTERING PERFORMANCE IN FORM OF [ACC (%)/NMI] ON CMU PIE DATASET

method	kernel(s)	dataset (number of classes)				
		All (68)	Lighting (17)	Rotation (17)	Occlusion (17)	Profile (17)
Ours	DeLight	44.5 / 0.643	**100.0 / 1.000**	23.5 / 0.401	27.8 / 0.501	26.7 / 0.463
	LBP	**53.8 / 0.714**	82.4 / 0.886	**46.7 / 0.604**	54.5 / 0.661	**31.8 / 0.527**
	RsLTS	46.7 / 0.697	63.1 / 0.744	31.0 / 0.468	**67.5 / 0.772**	25.1 / 0.514
	RsL2	36.5 / 0.613	73.3 / 0.822	25.1 / 0.407	20.4 / 0.315	27.1 / 0.489
	AvgKernel	51.3 / 0.708	89.0 / 0.911	28.2 / 0.448	61.6 / 0.675	26.3 / 0.520
Ours + CE	All	52.3 / 0.717	96.9 / 0.967	29.0 / 0.478	52.5 / 0.677	30.6 / 0.529
Ours + BCE	All	43.8 / 0.651	83.1 / 0.925	27.1 / 0.431	51.0 / 0.613	14.1 / 0.419
CoReg	All	43.0 / 0.682	73.7 / 0.871	24.3 / 0.423	45.9 / 0.586	28.2 / 0.485
MKL-DR	All	38.1 / 0.680	50.2 / 0.789	30.2 / 0.461	45.1 / 0.564	27.1 / 0.482
Ours	All	**62.5 / 0.783**	**100.0 / 1.000**	**50.6 / 0.584**	**61.6 / 0.690**	**38.0 / 0.593**

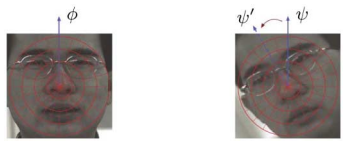

Fig. 7. Each image is divided into 96 regions. The distance between the two images is obtained when circularly shifting causes ψ' to be the new starting radial axis.

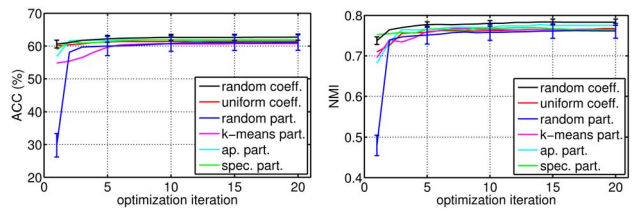

Fig. 8. With different initializations, the clustering accuracy (left) and normalized mutual information (right) of our approach along the iterative optimization on face image grouping.

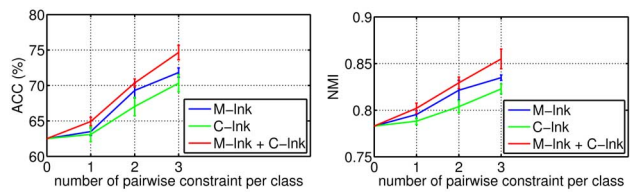

Fig. 9. Performance of MK-SOM with different numbers of pairwise constraints in face image grouping.

particularly important, since the optimal descriptor varies from group to group. With the dataset, we design and adopt a set of visual features, and establish the following four kernels.

- **DeLight:** The data representation is yielded based on the delighting algorithm [60], and the corresponding distance function is set as $1 - \cos\theta$, where θ is the angle between a data pair. Some delighting results are given in Fig. 6. It can be seen that variations caused by different lighting conditions or illuminations are significantly alleviated under the representation.
- **LBP:** As illustrated in Fig. 7, we divide each face image into $96 = 24 \times 4$ regions, and use a rotation-invariant *local binary pattern* (LBP) operator [61] (with operator setting $LBP^{riu2}_{8,1}$) to detect 10 distinct binary patterns. Thus, an image can be represented by a 960-dimensional vector, where each dimension records the number of occurrences that a specific pattern is detected in the corresponding region. To achieve rotation invariant, the distance between two such vectors, say $\mathbf{x}_i$ and $\mathbf{x}_j$, is the minimal one among the 24 values computed from the distance function $1 - \mathrm{sum}(\min(\mathbf{x}_i, \mathbf{x}_j))/\mathrm{sum}(\max(\mathbf{x}_i, \mathbf{x}_j))$ by circularly shifting the staring radial axis for $\mathbf{x}_j$. Clearly, the base kernel is constructed to cope with variations caused by rotations.
- **RsL2:** Each sample is represented by its pixel intensities in raster scan order. The Euclidean (L_2) distance is adopted.
- **RsLTS:** This base kernel is similar to RsL2, except that the distance function here is based on the least trimmed squares (LTS) with 20% outliers allowed. It aims to take account of the partial occlusions in face images.

Quantitative Results: We first evaluate MK-SOM by applying it to CMU PIE dataset using each of the four base kernels and the *AvgKernel* individually, and report the results in Table II. The obtained performance in ACC ranges from 36.5% to 53.8%. To gain insight into the quantitative results, we further compute the performance for the four groups. Note that no additional clustering is performed. We just compute the clustering performance with respect to data in each of the four groups separately. Apparently, each of the four kernels in general can lead to satisfactory performance regarding a specific type of intra-class variations. For instance, the kernel **DeLight** can perfectly handle the varied lighting conditions in the lighting group, while LBP and RsLTS yield acceptable outcomes respectively in the Rotation and Occlusion groups. Yet none of them is effective for dealing with the whole dataset. The results reveal that these kernels are complementary in the sense that they are respectively discriminant for different subsets of data. However, combining these kernels globally may not be a good strategy, since the kernels are corrupt for data with intra-class variations that they are not designed to deal with. It can be seen that the *AvgKernel* neither outperforms LBP in the whole dataset nor gets superior results to those of other kernels in the corresponding groups.

As for the cases where all the four kernels are simultaneously taken into account, it can be observed in Table II that the proposed approach achieves remarkable improvement, i.e., 8.7% in ACC and 0.069 in NMI, over the best result obtained from using a single kernel (**LBP**). Furthermore, it significantly

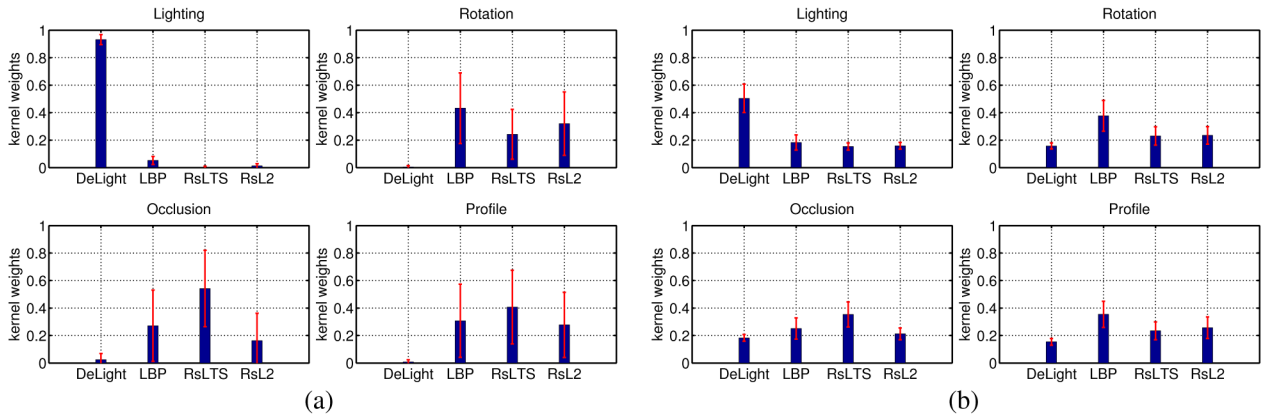

Fig. 10. Average learned kernel weights over subjects of each group. (a) Without pairwise constraints. (b) With pairwise constraints.

outperforms CE and BCE, which merge the four clustering results derived by applying our approach to the four base kernels respectively. It highlights the importance of cluster-dependent feature selection for the dataset. Unlike CE and BCE that carry out information fusion in a global fashion, our approach derives a learnable ensemble kernel to reveal the local structure of each cluster. CoReg is one of the state-of-the-art multi-view clustering algorithms, and also leverages multiple kernel learning in the learning procedure. MKL-DR takes multiple kernel learning as a preprocessing step where the most discriminative features are selected for being adopted in clustering. One main difference among CoReg, MKL-DR, and our approach is that the former two approaches focus on adaptively learning a single ensemble kernel to characterize the whole dataset, while ours aims to perform cluster-specific kernel learning to facilitate the uniqueness of different categories of data. Hence, substantial performance gain is achieved by our approach.

For an in-depth analysis of MK-SOM in cluster-dependent feature selection, we explore the learned kernel weights $\{\boldsymbol{\beta}_c\}$ of clusters (i.e., subjects in the application) in each group. This can be accomplished by the bipartite matching between the yielded clusters and the subjects in ground truth, like the method for computing ACC [22]. In Fig. 10(a), the learned kernel weights of subjects in each of the four groups are plotted in terms of the mean and the standard deviation. It can be seen that kernel **Delight** dominates in the lighting group. Kernels LBP and RsLTS contribute the most in the rotation and the occlusion groups respectively. Although the kernel weights are not precisely in proportion to the individual performance of the base kernels, discriminant kernels are appropriately selected by our approach to overcome the unfavorable variations of each group.

Running Time: The running time of our approach and other baselines that consider multiple features is reported in Table III. All the methods in the table are implemented in MATLAB, except CE is implemented in part in C. The running time is measured on a modern PC with Intel Core i73.4 GHz CPU. Since most of the methods start with random initialization, we run each method ten times, and report the average running time. Both CE and BCE are fast algorithms for fusing individual clusterings. Their running time is mostly dependent on the methods used for compiling individual clusterings. It is worth mentioning that a step of high complexity, i.e., solving an eigenvalue problem, is needed in SC, but SC is still more efficient than k-means owing to the sizes of the two datasets. Our approach is not as efficient as the comparative baselines. Most computation cost is induced for carrying out cluster-dependent feature selection, since we adopt an iterative and alternate optimization procedure to tackle the problem caused by the mutual dependence between cluster-specific ensemble kernels and clustering results. It takes our approach a few minutes to cluster the two medium-scale datasets, i.e., $(N, C) = (600, 20)$ in the object dataset and $(N, C) = (1020, 68)$ in the face dataset, where N and C are the numbers of data and clusters, respectively.

TABLE III
RUNNING TIME (SEC) OF VARIOUS METHODS ON THE DATASETS OF OBJECT CATEGORIZATION AND FACE IMAGE GROUPING

method	dataset	
	object	face
k-means + CE	20.6	137.1
AP + CE	211.4	572.9
SC + CE	**1.1**	**4.9**
k-means + BCE	21.2	138.4
AP + BCE	211.9	573.5
SC + BCE	2.3	6.8
CoReg	3.4	26.4
MKL-DR	4.8	13.3
Ours	105.3	446.4

Initialization and Convergency: We check whether our method converges in the application to face image grouping. Similar to the setups used in Fig. 3, the performance of our method along the iterative procedure of optimization is shown in Fig. 8. It can be observed that our method is stable, since it converges to similar NMI and ACC with diverse initializations.

Clustering with Pairwise Constraints: We assess MK-SOM with the introduction of pairwise constraints for face image grouping, and give the obtained performance with different numbers of pairwise constraints in Fig. 9. Similar to object categorization, pairwise constraints also boost the performance in face image grouping. The degree of improvement is even higher, since the CMU PIE dataset is much more complex: more clusters and various types of intra-cluster variations. We show the learned kernel weights with the aid of pairwise constraints in Fig. 10(b). Compared to those in Fig. 10(a), the learned kernel weights with pairwise constraints are more

closely related to their contributions to the clustering tasks for the corresponding groups. Furthermore, the standard deviations of the kernel weights are considerably reduced. It indicates that these constraints facilitate the desirable stability in optimization.

VI. Conclusion

We have presented an approach to clustering data in multiple feature representations. Its main feature lies in the ability of efficient cluster-dependent feature selection, which is motivated by the observation that the optimal data descriptors often vary from cluster to cluster. With the idea of associating each cluster with a learnable ensemble kernel, we integrate multiple kernel learning into the clustering procedure, and cast it as a joint optimization problem. Besides, it is shown that the problem can be solved efficiently by gradient descent with the introduction of softmax activation function. The cluster-specific structure is then gradually revealed by the learned ensemble kernels through the optimization iterations. The proposed approach is comprehensively evaluated in two clustering tasks as well as the associated datasets, including visual object categorization and face image grouping. The promising results in accuracy and convergency speed manifest the effectiveness of our approach.

The introduced formulation of MK-SOM provides a new way of extending the MKL framework to work in a cluster-specific manner. Besides, it generalizes MKL to address not only unsupervised but also semi-supervised learning tasks. These aspects of generalization introduce a new frontier in applying MKL to solving increasingly complex clustering tasks. This property is precious especially for unsupervised and semi-supervised multimedia content analysis, since MKL has been a wisely used way for handling the varieties of multi-modal data in multimedia research.

References

[1] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the smo algorithm," in *Proc. Int. Conf. Mach. Learning*, 2004.

[2] G. Lanckriet, N. Cristianini, P. Bartlett, L. Ghaoui, and M. Jordan, "Learning the kernel matrix with semidefinite programming," *J. Mach. Learning Res.*, vol. 5, pp. 27–72, 2004.

[3] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf, "Large scale multiple kernel learning," *J. Mach. Learning Res.*, vol. 7, pp. 1531–1565, 2006.

[4] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learning Res.*, vol. 9, pp. 2491–2521, 2008.

[5] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cybern.*, vol. 43, no. 1, pp. 59–69, 1982.

[6] J. Mutch and D. Lowe, "Multiclass object recognition with sparse, localized features," in *Proc. Conf. Comput. Vision Pattern Recogn.*, 2006.

[7] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[8] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. Conf. Comput. Vision and Pattern Recogn.*, 2006.

[9] J. Ye, R. Janardan, and Q. Li, "Two-dimensional linear discriminant analysis," in *Adv. Neural Inf. Process. Syst.*, 2004.

[10] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE Trans. Neural Netw.*, vol. 11, no. 3, pp. 586–600, May 2000.

[11] Y.-Y. Lin, T.-L. Liu, and C.-S. Fuh, "Multiple kernel learning for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 6, pp. 1147–1160, Jun. 2011.

[12] K.-C. Huang, Y.-Y. Lin, and J.-Z. Cheng, "Cluster-dependent feature selection by multiple kernel self-organizing map," in *Proc. Int. Conf. Pattern Recogn.*, 2012.

[13] Y.-Y. Lin, T.-L. Liu, and C.-S. Fuh, "Clustering complex data with group-dependent feature selection," in *Proc. Eur. Conf. Comput. Vision*, 2010.

[14] W. Zhang, X. Wang, D. Zhao, and X. Tang, "Graph degree linkage: Agglomerative clustering on directed graph," in *Proc. Eur. Conf. Comput. Vision*, 2012.

[15] E. Hancer, C. Ozturk, and D. Karaboga, "Artificial bee colony based image clustering method," in *IEEE World Congr. on Computat. Intell.*, 2012.

[16] B. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[17] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[18] M. Omran and A. P. Engelbrecht, "Particle swarm optimization method for image clustering," *Int. J. Pattern Recogn. Artif. Intell.*, vol. 19, no. 3, pp. 297–321, 2005.

[19] V. Roth and T. Lange, "Feature selection in clustering problems," in *Adv. Neural Inf. Process. Syst.*, 2003.

[20] M. Law, M. Figueiredo, and A. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1154–1166, Sep. 2004.

[21] J. Goldberger, S. Gordon, and H. Greenspan, "Unsupervised image-set clustering using an information theoretic framework," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 449–458, Feb. 2006.

[22] J. Ye, Z. Zhao, and M. Wu, "Discriminative k-means for clustering advances in neural information processing systems," in *Adv. Neural Inf. Process. Syst.*, 2007.

[23] Z. Deng, K.-S. Choi, F.-L. Chung, and S. Wang, "Enhanced soft subspace clustering integrating within-cluster and between-cluster information," *Pattern Recogn.*, vol. 43, no. 3, pp. 767–781, 2010.

[24] H. Cheng, K. Hua, and K. Vu, "Constrained locally weighted clustering," in *Int. Conf. Very Large Databases*, 2008.

[25] C. Domeniconi and M. Al-razgan, "Weighted cluster ensembles: Methods and analysis," *ACM Trans. Knowledge Discovery from Data*, vol. 2, no. 4, 2009.

[26] N. Grozavu, Y. Bennani, and M. Lebbah, "Cluster-dependent feature selection through a weighted learning paradigm," *Adv. Knowledge Discovery Manage. Studies in Computat. l Intell.*, vol. 292, pp. 133–147, 2010.

[27] A. Berg and J. Malik, "Geometric blur for template matching," in *Proc. Conf. Comput. Vision Pattern Recogn.*, 2001.

[28] A. Bosch, A. Zisserman, and X. M. noz, "Image classification using random forests and ferns," in *Proc. IEEE 11th Int. Conf. Comput. Vision (ICCV)*, 2007, pp. 1–8.

[29] S. Yan, D. Xu, B. Zhang, H. 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.

[30] A. Strehl and J. Ghosh, "Cluster ensembles a knowledge reuse framework for combining multiple partitions," *J. Mach. Learning Res.*, vol. 3, pp. 583–617, 2002.

[31] S. Bickel and T. Scheffer, "Multi-view clustering," in *Proc. IEEE Int. Conf. Data Mining*, 2004, pp. 19–26.

[32] D. Zhou and C. Burges, "Spectral clustering and transductive learning with multiple views," in *Proc. Int. Conf. Mach. Learning*, 2007.

[33] X. Wu, C.-W. Ngo, and A. G. Hauptmann, "Multimodal news story clustering with pairwise visual near-duplicate constraint," *IEEE Trans. Multimedia*, vol. 10, no. 2, pp. 188–199, Feb. 2008.

[34] T. Li, M. Ogihara, W. Peng, B. Shao, and S. Zhu, "Music clustering with features from different information sources," *IEEE Trans. Multimedia*, vol. 11, no. 3, pp. 477–485, Apr. 2009.

[35] H. Mirzaei, "A novel multi-view agglomerative clustering algorithm based on ensemble of partitions on different views," in *Proc. Int. Conf. Pattern Recogn.*, 2010, pp. 1007–1010.

[36] G. Tzortzis and A. C. Likas, "Multiple view clustering using a weighted combination of exemplar-based mixture models," *IEEE Trans. Neural Netw.*, vol. 21, no. 12, pp. 1925–1938, 2010.

[37] A. Kumar, P. Rai, and H. Daumé, III, "Co-regularized multi-view spectral clustering," in *Adv. Neural Inf. Process. Syst.*, 2011.

[38] J. Li, B. Shao, T. Li, and M. Ogihara, "Hierarchical co-clustering: A new way to organize the music data," *IEEE Trans. Multimedia*, vol. 14, no. 2, pp. 471–481, 2012.

[39] D. Nguyen, L. Chen, and C.-K. Chan, "Clustering with multiviewpoint-based similarity measure," *IEEE Trans. Knowledge Data Eng.*, vol. 24, no. 6, pp. 988–1001, 2012.

[40] X. Chen, X. Xu, J. Huang, and Y. Ye, "TW-k-means: Automated two-level variable weighting clustering algorithm for multiview data," *IEEE Trans. Knowledge Data Eng.*, vol. 25, no. 4, pp. 932–944, Apr. 2013.
[41] P. He, X. Xu, and L. Chen, "Constrained clustering with local constraint propagation," in *Proc. Eur. Conf. Comput. Vision*, 2012, pp. 223–232.
[42] E. Xing, A. Ng, M. Jordan, and S. Russell, "Distance metric learning with application to clustering with side-information," in *Adv. Neural Inf. Process. Syst.*, 2002.
[43] A. Mignon and F. Jurie, "PCCA: A new approach for distance learning from sparse pairwise constraints," in *Proc. Conf. Comput. Vision and Pattern Recogn.*, 2012, pp. 2666–2672.
[44] X. Chen, Z. Tong, H. Liu, and D. Cai, "Metric learning with two-dimensional smoothness for visual analysis," in *Proc. Conf. Comput. Vision and Pattern Recogn.*, 2012, pp. 2533–2538.
[45] B. Zhao, J. Kwok, and C. Zhang, "Multiple kernel clustering," in *Proc. SIAM Int. Conf. Data Mining*, 2009, pp. 638–649.
[46] H.-Y. Huang, Y.-Y. Chuang, and C.-S. Chen, "Multiple kernel fuzzy clustering," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 1, pp. 120–134, Feb. 2012.
[47] M. Olteanu, N. V. Vialaneix, and C. C. Ayrolles, "Multiple kernel self-organizing maps," in *Eur. Symp. on Artif. Neural Netw., Computat. Intell. Mach. Learning*, 2013, pp. 83–88.
[48] H. Zhang, A. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition," in *Proc. Conf. Comput. Vision and Pattern Recogn.*, 2006.
[49] B. Schölkopf and A. Smola, *Learning with Kernels*. Cambridge, MA, USA: MIT Press, 2002.
[50] K. Lau, H. Yin, and S. Hubbard, "Kernel self-organising maps for classification," *Neurocomputing*, vol. 69, no. 16, pp. 2033–2040, 2006.
[51] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta, "A survey of kernel and spectral methods for clustering," *Pattern Recogn.*, vol. 41, no. 1, pp. 176–190, 2008.
[52] B. Romain, J. Bertrand, R. Fabrice, and V. Nathalie, "Batch kernel SOM and related Laplacian methods for social network analysis," *Neurocomputing*, vol. 71, no. 7, pp. 1257–1273, 2008.
[53] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," in *Proc. Conf. Comput. Vision and Pattern Recogn.*, 2004, p. 178.
[54] D. Dueck and B. Frey, "Non-metric affinity propagation for unsupervised image categorization," in *Proc. IEEE 11th Int. Conf. Comput. Vision*, 2007, pp. 1–8.
[55] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vision*, vol. 42, no. 3, pp. 145–175, 2001.
[56] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *Proc. Conf. Comput. Vision and Pattern Recogn.*, 2007, pp. 1–8.
[57] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Adv. Neural Inf. Process. Syst.*, 2001, pp. 849–856.
[58] Y. Yang, D. Xu, F. Nie, S. Yan, and Y. Zhuang, "Image clustering using local discriminant models and global integration," *IEEE Trans. Image Process.*, vol. 19, no. 10, pp. 2761–2773, Oct. 2010.
[59] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, 2003.
[60] R. Gross and V. Brajovic, "An image preprocessing algorithm for illumination invariant face recognition," in *Proc. Int. Conf. Audio and Video-Based Biometric Person Authentication*, 2003, pp. 10–18.
[61] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

Jeng-Tsung Tsai received the B.S. degree in electrical engineering from Yuan Ze University, Taiwan in 2011, and is currently working toward the M.S. degree from the Department of Computer Science, University of Southern California, Los Angeles, CA, USA.

His research interests include computer vision, multimedia information systems, and machine learning.

Yen-Yu Lin (M'11) received the B.S. degree in information management and the M.S. and Ph.D. degrees in computer science and information engineering from National Taiwan University, in 2001, 2003, and 2010, respectively.

He is currently an Assistant Research Fellow with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. His current research interests include computer vision, pattern recognition, and machine learning.

Hong-Yuan Mark Liao (S'87–M'89–SM'01–F'13) received the Ph.D degree in electrical engineering from Northwestern University in 1990.

In July 1991, he joined the Institute of Information Science, Academia Sinica, Taipei, Taiwan, and is currently a Distinguished Research Fellow. He has worked in the fields of multimedia signal processing, image processing, computer vision, pattern recognition, video forensics, and multimedia protection for more than 25 years. During 2009–2011, he was the Division Chair of the computer science and information engineering division II, National Science Council of Taiwan. He is jointly appointed as a Professor of the Computer Science and Information Engineering Department of National Chiao-Tung University and the Department of Electrical Engineering and Computer Science of National Cheng Kung University. During 2009–2012, he was jointly appointed as the Multimedia Information Chair Professor of National Chung Hsing University. Since August 2010, he has been appointed as an Adjunct Chair Professor of Chung Yuan Christian University. Since August 2014, he has been appointed as an Honorary Chair Professor of National Sun Yat-sen University.

Dr. Liao received the Young Investigators' Award from Academia Sinica in 1998; the Distinguished Research Award from the National Science Council of Taiwan in 2003, 2010, and 2013; the National Invention Award of Taiwan in 2004; the Distinguished Scholar Research Project Award from National Science Council of Taiwan in 2008; and the Academia Sinica Investigator Award in 2010. His professional activities include: Co-Chair, 2004 International Conference on Multimedia and Exposition (ICME); Technical Co-chair, 2007 ICME; General Co-Chair, 17th International Conference on Multimedia Modeling; President, Image Processing and Pattern Recognition Society of Taiwan (2006–2008); Editorial Board Member, *IEEE Signal Processing Magazine* (2010–2013); Associate Editor, IEEE TRANSACTIONS ON IMAGE PROCESSING (2009–2013); IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY (2009–2012); and IEEE TRANSACTIONS ON MULTIMEDIA (1998–2001). Currently, he also serves as IEEE Signal Processing Society Region 10 Director (Asia–Pacific Region).