Collaborative Multi-feature Fusion for Transductive Spectral Learning

Hongxing Wang, *Student Member, IEEE,* and Junsong Yuan, *Senior Member, IEEE,*

***Abstract*—Much existing work of multi-feature learning relies on the agreement among different feature types to improve the clustering or classification performance. However, as different feature types could have different data characteristics, such a forced agreement among different feature types may not bring a satisfactory result. We propose a novel transductive learning approach that considers multiple feature types simultaneously to improve the classification performance. Instead of forcing different feature types to agree with each other, we perform spectral clustering in different feature types separately. Each data sample is then described by a co-occurrence of feature patterns among different feature types, and we apply these feature co-occurrence representations to perform transductive learning, such that data samples of similar feature co-occurrence pattern will share the same label. As the spectral clustering results in different feature types and the formed co-occurrence patterns influence each other under the transductive learning formulation, an iterative optimization approach is proposed to decouple these factors. Different from co-training that need to iteratively update individual feature type, our method allows all feature types to collaborate simultaneously. It can naturally handle multiple feature types together and is less sensitive to noisy feature types. The experimental results on synthetic, object and action recognition datasets all validate the advantages of our method compared to state-of-the-arts.**

***Index Terms*—multi-feature fusion; feature co-occurrence pattern; spectral clustering; transductive learning;**

I. Introduction

In many pattern classification problems, the target data, *e.g*., an image, can be naturally represented using different types (modalities) of features, *e.g*., color, shape, and texture features. Instead of using a single feature modality, a suitable integration of multiple complementary features can result in a better clustering or classification result. Much previous work has studied how to leverage multiple feature types to improve the classification performance, such as co-training [1], [2], canonical correlation analysis [3], and multiple kernel learning [4], [5], [6]. Despite previous successes, most existing multi-feature learning approaches rely on the agreement among different feature types to improve the performance: the decision of a data sample is preferred to be consistent across different feature types. However, as different feature types may have different data characteristics and distributions, a forced agreement among different feature types may not bring a satisfactory result.

To handle the different data characteristics among multiple feature types, we propose to respect the data distribution and allow different feature types to have its own clustering results. This can faithfully reflect the data characteristics in different feature types, *e.g*., color feature space can be categorized into a number of typical colors, while texture feature space categorized into a different number of texture patterns. To integrate the clustering results from different feature types, we represent each data sample by a co-occurrence of feature patterns, *e.g*., a composition of typical color and texture patterns. Unlike much previous work on co-occurrence pattern discovery [7] in spatial domain, *e.g*., [8], [9] and [10], we aim to capture co-occurrence patterns across multiple feature modalities. Such a treatment has two advantages. First, instead of forcing different feature types to agree with each other, we compose multiple feature types to reveal the compositional pattern across different feature types, thus it can naturally combine multiple features. Comparing with a direct concatenation of multiple types of features, the feature co-occurrence patterns encode the latent compositional structure among multiple feature types, thus have a better representation power. Moreover, as it allows different clustering results in different feature types, the feature co-occurrence patterns can be more flexible. Second, relying on the new feature co-occurrence representations of the data samples, we can measure the similarity between data samples of multiple features, such that data samples of similar feature co-occurrence pattern will share the same label. Our new feature co-occurrence representation does not need to optimize individual feature type iteratively like in co-training, thus is less sensitive to noisy feature types.

We study the collaborative multi-feature fusion in a transductive learning framework, where the labeled data samples can transfer the labels to the unlabeled data. To enable transductive spectral learning, we formulate a new objective function with three objectives, namely the good quality of spectral clustering in individual feature types, the label smoothness of data samples in terms of their feature co-occurrence representations, and the fitness to the labels provided by the training data. The optimization of this objective function is complicated as the spectral clustering results in different feature types and the formed co-occurrence patterns influence each other under the transductive learning formulation. We thus propose an iterative optimization approach that can decouple these factors. During the iterations, the clustering results of individual feature types and the smoothness of the labeling of data samples will help each other, leading to a better transductive learning. To evaluate our method, we conduct experiments on a synthetic dataset, as well as object and action recognition datasets. The comparison with related methods such as [11],[12] and [13] show promising results

The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: hwang8@e.ntu.edu.sg; jsyuan@ntu.edu.sg).

that our proposed method can well handle the different data characteristics of multiple feature types and is robust to noisy feature types.

We explain our proposed transduction spectral learning using multi-feature fusion in Fig. 1. There are four data classes represented by two feature modalities, *i.e.*, texture and color. The texture modality forms two texture patterns, chessboard and brick; while the color modality forms two color patterns, green and blue. All data samples belong to one of the four compositional patterns: green brick (Hexagon), blue chessboard (Triangle), green chessboard (Square), and blue brick (Circle). Clearly, the four data classes cannot be distinguished in either the texture or the color feature space alone. For example, the two classes Square and Triangle share the same texture attribute, but different in color, while the Hexagon and Square classes share the same color but different in texture. However, each class can be easily distinguished by a co-occurrence of the texture and color pattern, *e.g.*, the Hexagon class composes "brick" texture and "green" color. As a result, the unlabeled data samples of the same co-occurrence feature pattern can be labeled as the same class as the labeled data sample.

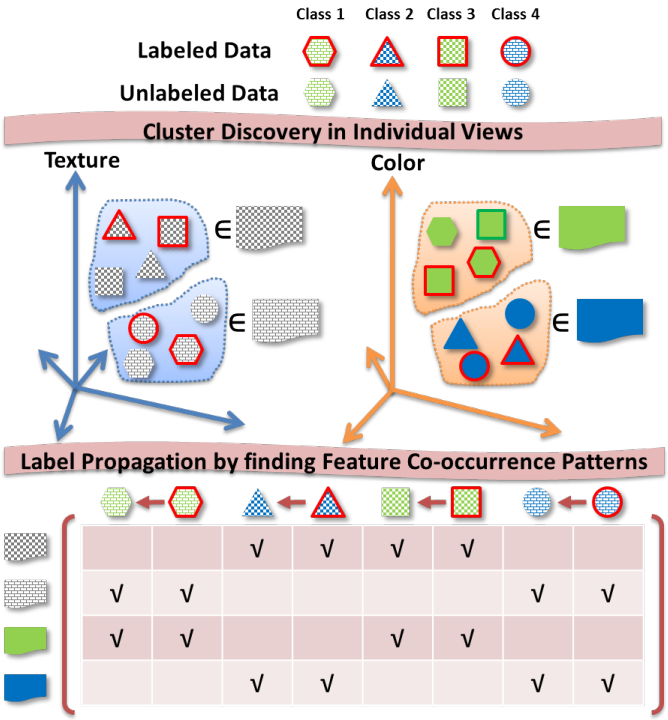


Fig. 1. Label propagation of unlabeled data by the discovery of the co-occurrence patterns among different types of clusters. See text for details and best seen in color.

II. Related Work

We review and compare our work to previous work on multi-feature learning and graph based trasductive learning.

Multi-feature learning. In terms of multi-feature learning, some existing work enforce the agreement among different feature types. For example, the method in [14] minimizes the disagreement of classifiers between two feature modalities. Similarly, the co-training methods train two classifiers separately from different feature types and make both classifiers agree on the labeling of the unlabeled data [1], [2]. The way of Canonical Correlation Analysis (CCA) is to extract shared features from multiple feature types [3], [15], [16]. Learning an ensemble kernel form different feature types is adopted in [4], [5], [6]. More strategies include multiview stochastic neighbor embedding [17], joint nonnegative matrix factorization [18], consensus pattern embedding [19], metric fusion [20], [21] and graph-based feature combination [13], [22], [23], [24], [25], [26], [27], [28]. For further discussion, we refer readers to the comprehensive surveys in [29]

Despite these previous advances, there is limited work that address the disagreement problem of different feature types in multi-feature learning. A conditional entropy criterion is introduced to detect modality disagreement caused by modality corruption or noise in [30]. However, even without the influence of modality corruption and noise, samples in individual feature types still need not to belong to the same class. The recent work include context-aware clustering in [31] and [32], hierarchical sparse coding [33], and latent subspace Markov network in [34] that incorporates the individual feature structures of multiple feature types for pattern clustering or classification. Especially in [31], the authors use the co-occurrences of feature clusters in different feature types to represent data samples. Because of this manipulation, the individual feature spaces from different feature types can have different data distributions. Although [31] can also handle multi-feature fusion, it targets at unsupervised clustering only, and its extension to transductive learning is non-trivial.

Graph transduction. The effectiveness of graph transduction in semi-supervised classification has been proven in previous work [35], [36]. In the setting of graph transductive learning, data class labels can be propagated from labeled data to unlabeled data through undirected graph [37], [38], [39], [12] or direct graph [13], [40]. Besides single-label data, some methods of graph transduction can also handle multi-label data [41], [42]. Moreover, the propagation is not confined to single graph. There are also methods proposed to deal with multiple graphs, *e.g.*, multiple feature graphs [13] and sample-class graphs [43].

Among the numerous methods, the random walk approach to transductive learning with multiple views (RWMV) in [13] is closely related to our graph-based multi-feature transduction approach. Meanwhile, our problem formulation is an extension of multi-feature graph transductive learning via alternating minimization (GTAM) in [11]. Thus both RWMV and GTAM are compared with our method in the experiments. In addition, we also compare graph transduction game (GTG) [12], which is a recent work of tranductive learning.

III. Proposed Method

We study the collaborative multi-feature fusion in a transductive learning framework, where the labeled data samples can transfer the labels to the unlabeled data. Consider a collection of partially labeled multi-class dataset $\mathcal{X} = (\mathcal{X}_l, \mathcal{X}_u)$. The labeled inputs $\mathcal{X}_l = \{x_i\}_{i=1}^{l}$ are associated with known labels

$\mathcal{Y}_l = \{y_i\}_{i=1}^{l}$, where $y_i \in \mathcal{L} = \{1, 2, \cdots, M\}$. The unlabeled data $\mathcal{X}_u = \{x_i\}_{i=l+1}^{N}$ are with missing labels $\mathcal{Y}_u = \{y_i\}_{i=l+1}^{N}$, where $y_i \in \mathcal{L}$ and the task is to infer $\mathcal{Y}_u$. A binary matrix $\mathbf{Y} \in \{1, 0\}^{N \times M}$ encodes the label information of $\mathcal{X}$, where $\mathbf{Y}_{ij} = 1$ if x_i has a label $y_i = j$ and $\mathbf{Y}_{ij} = 0$ otherwise. We set $\mathbf{Y}_{ij} = 0$ initially for unlabeled data $y_i \in \mathcal{Y}_u$. We assume each $x_i \in \mathcal{X}$ is represented as K types/modalities of features as $\{\mathbf{f}_i^{(k)}\}_{k=1}^{K}$, where $\mathbf{f}_i^{(k)} \in \mathbb{R}^{d_k}$. To enable multi-feature collaboration in label propagation, we propose our methods in the following.

A. Spectral Embedding of Multi-feature Data

To handle the different data characteristics among multiple feature types, we propose to respect the data distribution and allow different feature types to have its own clustering results. As spectral embedding can effectively capture the data clustering structure [44], we leverage it to study the data distribution in each feature type.

At first, each feature type $\{\mathcal{F}^{(k)}\} = \{\mathbf{f}_i^{(k)}\}_{i=1}^{N}$ of $\mathcal{X}$ defines an undirected graph $G_k = (\mathcal{X}, \mathcal{E}, \mathbf{W}_k)$ in which the set of vertices is $\mathcal{X}$ and the set of edges connecting pairs of vertices is $\mathcal{E} = \{e_{ij}\}$. Each edge e_{ij} is assigned a weight $w_{ij}^{(k)} = \kappa(x_i, x_j)$ to represent the similarity between x_i and x_j. The matrix $\mathbf{W}_k = (w_{ij}^{(k)}) \in \mathbb{R}^{N \times N}$ denote the similarity or kernel matrix of $\mathcal{X}$ in this feature type. Following spectral clustering, we use the following function to compute the graph similarities:

$$w_{ij} = \exp\left\{-\frac{\mathrm{dist}^2\left(\mathbf{f}_i^{(k)}, \mathbf{f}_j^{(k)}\right)}{2\sigma^2}\right\}, \tag{1}$$

where $\mathrm{dist}\left(\mathbf{f}_i^{(k)}, \mathbf{f}_j^{(k)}\right)$ denotes the distance between a pair of features; σ is the bandwidth parameter to control how fast the similarity decreases. By summing the weights of edges being connected to x_i, we can obtain the degree of this vertex $d_i^{(k)} = \sum_{j=1}^{N} w_{ij}^{(k)}$. Let $\mathbf{D}_k \in \mathbb{R}^{N \times N}$ be the vertex degree matrix by placing $\{d_i^{(k)}\}_{i=1}^{N}$ on the diagonal. Then we can write the graph Laplacian $\mathbf{\Delta}_k \in \mathbb{R}^{N \times N}$ as

$$\mathbf{\Delta}_k = \mathbf{D}_k - \mathbf{W}_k \tag{2}$$

and the normalized graph Laplacian $\mathbf{L}_k \in \mathbb{R}^{N \times N}$ as $\mathbf{L}_k = \mathbf{D}_k^{-1/2}\mathbf{\Delta}_k\mathbf{D}_k^{-1/2} = \mathbf{I}_N - \mathbf{D}_k^{-1/2}\mathbf{W}_k\mathbf{D}_k^{-1/2}$, where $\mathbf{I}_N$ is an identify matrix of order N.

After the above preprocessing to each feature type, we perform spectral clustering to group the feature points of both labeled and unlabeled data into clusters. Assume there are M_k clusters in the k_{th} feature type. The spectral clustering on this feature type is to minimize the spectral embedding cost [45]:

$$\Omega_{\mathrm{type}}\left(\mathbf{R}_k\right) = tr\left(\mathbf{R}_k^T\mathbf{L}_k\mathbf{R}_k\right), \tag{3}$$

subject to $\mathbf{R}_k^T\mathbf{R}_k = \mathbf{I}_{M_k}$, where $tr(\cdot)$ denotes the matrix trace; $\mathbf{R}_k \in \mathbb{R}^{N \times M_k}$ is the real-valued cluster indicators of the M_k clusters [44]; $\mathbf{I}_{M_k}$ is an identify matrix of order M_k. By using the Rayleigh-Ritz theorem [46], we can obtain the solution of $\mathbf{R}_k$, which consists of the first M_k eigenvectors corresponding to the M_k smallest eigenvalues of $\mathbf{L}_k$, *i.e.*, $\mathbf{r}_i^{(k)}, i = 1, 2, \cdots, M_k$, denoting as:

$$\mathbf{R}_k = \left[\mathbf{r}_1^{(k)}, \mathbf{r}_2^{(k)}, \cdots, \mathbf{r}_{M_k}^{(k)}\right] \triangleq eig\left(\mathbf{L}_k, M_k\right). \tag{4}$$

By using Eq. 4, we can independently perform spectral embedding in different feature types. In other words, we do not have to force the clustering in different feature spaces to agree with each other.

B. Building Feature Co-occurrence Patterns and Multi-feature Similarity Graph

We have obtained K label indicator matrices $\{\mathbf{R}_k\}_{k=1}^{K}$ obtained from the K types of features by Eq. 4 in the above section. To integrate them, we build a matrix $\mathbf{T}_v \in \mathbb{R}^{\sum_{k=1}^{K} M_k \times N}$ as:

$$\mathbf{T}_v = [\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_K]^T. \tag{5}$$

The n_{th} column of $\mathbf{T}_v$ is the multi-feature representation of x_n, which conveys the complementary information across multiple types of feature clusters without forcing clustering agreement among different feature types. Additionally, $\mathbf{T}_v$ stores soft feature co-occurrence patterns since $\{\mathbf{R}_k\}_{k=1}^{K}$ are soft cluster indicators of multiple feature types. Comparing to hard clustering indicators used in [31], the spectral soft relaxation can more effectively capture the feature clustering structures of individual feature types, and tolerate noisy features [44].

With the multi-feature representations of the samples in $\mathcal{X}$, *i.e.*, the feature co-occurrence patterns $\mathbf{T}_v$ in Eq. 5, we introduce the multi-feature similarity graph $G_v = (\mathcal{X}, \mathcal{E}, \mathbf{W}_v)$ based on T_v. By Laplacian embedding, the resulting soft cluster indicators $\{\mathbf{R}_k\}_{k=1}^{K}$ can be considered to obey linear similarities [23], so are the concatenation of them, *i.e.*, $\mathbf{T}_v$. Therefore we define the similarity matrix $\mathbf{W}_v \in \mathbb{R}^{N \times N}$ as a linear kernel:

$$\mathbf{W}_v = \mathbf{T}_v^T\mathbf{T}_v = \sum_{k=1}^{K} \mathbf{R}_k\mathbf{R}_k^T. \tag{6}$$

Regarding the weighting coefficients, $\mathbf{W}_v$ can be considered as an average of the linear kernels of the soft cluster indicators in multiple feature types. Therefore it will be less sensitive to poor individual feature types. What needs to be noted is that although the entries of the matrix $\mathbf{W}_v$ are not necessary all non-negative, $\mathbf{W}_v$ is semi-positive. One can also add $\mathbf{W}_v$ with a rank-1 matrix whose entries are all equal to the minimum negative entry of $\mathbf{W}_v$ to make sure each entry of $\mathbf{W}_v$ non-negative. We will omit this manipulation in the following statement and derivation as it does not affect the solution (Section III-D) to the problem Eq. 9.

According to the similarity matrix, we can obtain the degree matrix $\mathbf{D}_v \in \mathbb{R}^{N \times N}$ by

$$\mathbf{D}_v = diag\left(\mathbf{W}_v\mathbf{1}\right), \tag{7}$$

where $\mathbf{1} \in \mathbb{R}^{\mathbf{N}}$ is an all one vector. We define the normalized Laplacian as:

$$\mathbf{L}_v = \mathbf{I}_N - \mathbf{D}_v^{-1/2}\mathbf{W}_v\mathbf{D}_v^{-1/2}. \tag{8}$$

With $\mathbf{L}_v$, we encode the smoothness of the multi-feature similarity graph. It will help us to assign the same label to data samples of similar feature co-occurrence patterns.

C. Multi-feature Fusion with Transductive Learning

After we construct the multi-feature similarity graph G_v by the co-occurrence patterns based on the feature clusters of multiple feature types, it is still a non-trivial task to build a smooth connection between the feature clustering structures of multiple feature types and the label predictions of unlabeled data. In order to address the problem, we introduce a soft class label matrix $\mathbf{R}_v \in \mathbb{R}^{N\times M}$ to assist the transition. Different from the hard class labels $\mathbf{Y} \in \{0,1\}^{N\times M}$, $\mathbf{R}_v$ is a relaxed real matrix. All taken into account, we propose to minimize the spectral clustering costs of individual feature types, the labeling smoothness regularization of unlabeled data samples, and the fitting penalty of hard class labels $\mathbf{Y}$ and soft class labels $\mathbf{R}_v$ together in the following objective function:

$$\begin{aligned}
&\Omega\left(\{\mathbf{R}_i\}_{i=1}^K, \mathbf{R}_v, \mathbf{Y}\right) \\
&= \sum_{i=1}^K \Omega_{\text{type}}\left(\mathbf{R}_i\right) + \alpha\Omega_{\text{smooth}}\left(\mathbf{R}_v, \{\mathbf{R}_j\}_{j=1}^K\right) \\
&\qquad + \beta\Omega_{\text{fit}}\left(\mathbf{R}_v, \mathbf{Y}\right) \qquad (9)\\
&= \sum_{i=1}^K tr\left(\mathbf{R}_i^T\mathbf{L}_i\mathbf{R}_i\right) + \alpha tr\left(\mathbf{R}_v^T\mathbf{L}_v\mathbf{R}_v\right) \\
&\qquad + \beta tr\left\{\left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right)^T\left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right)\right\},
\end{aligned}$$

subject to $\mathbf{R}_k^T\mathbf{R}_k = \mathbf{I}_{M_k},\ \forall\ k = 1, 2, \cdots, K$; $\mathbf{R}_v \in \mathbb{R}^{N\times M}$; $\mathbf{Y} \in \{1,0\}^{N\times M}$ and $\sum_{j=1}^M \mathbf{Y}_{ij} = 1$ with balance parameters α and β. In our objective, $\mathbf{R}_k^T\mathbf{R}_k = \mathbf{I}_{M_k}$ is the requirement of unique embedding; $\sum_{j=1}^M \mathbf{Y}_{ij} = 1$ is to make a unique label assignment for each vertex; and $\mathbf{S} \in \mathbb{R}^{N\times N}$ is a normalized term to weaken the influence of noisy labels and balance class biases. Similar to [11], the diagonal elements of $\mathbf{S}$ are filled by the class-normalized node degrees: $\mathbf{s} = \sum_{j=1}^M \frac{\mathbf{Y}_{\cdot j}\odot\mathbf{D}_v\mathbf{1}}{\mathbf{Y}_{\cdot j}^T\mathbf{D}_v\mathbf{1}}$, where $\odot$ denotes Hadamard product; $\mathbf{Y}_{\cdot j}$ denotes the j_{th} column of $\mathbf{Y}$; $\mathbf{1} \in \mathbb{R}^{\mathbf{N}}$ is an all one vector.

More specifically, as discussed in Section III-A, the spectral clustering objective of multiple feature types $\sum_{i=1}^K \Omega_{\text{type}}\left(\mathbf{R}_i\right)$ is to reveal the data distributions in multiple feature types without forcing clustering agreement. In addition, to allow the soft class labels $\mathbf{R}_v$ for $\mathcal{X}$ to be consistent on closely connected vertices in the multi-feature similarity graph G_v, we regularize our objective with the following smoothing function:

$$\Omega_{\text{smooth}}\left(\mathbf{R}_v, \{\mathbf{R}_j\}_{j=1}^K\right) = tr\left(\mathbf{R}_v^T\mathbf{L}_v\mathbf{R}_v\right), \qquad (10)$$

where $\mathbf{L}_v$ is defined by Eq. 8 which is related to $\{\mathbf{R}_j\}_{j=1}^K$. Furthermore, to prevent overfitting, it should allow occasional disagreement between the soft class labels $\mathbf{R}_v$ and the hard class labels $\mathbf{Y}$ on the dataset $\mathcal{X}$. Thus, we minimize the fitting penalty:

$$\Omega_{\text{fit}}\left(\mathbf{R}_v, \mathbf{Y}\right) = tr\left\{\left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right)^T\left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right)\right\}. \qquad (11)$$

Regarding our objective of Eq. 9, it is worth noting that the three terms of this function are correlated among each other. We thus cannot minimize Ω by minimizing the three terms separately. Moreover, the binary integer constraint on $\mathbf{Y}$ also challenges the optimization. We will in Section III-D show how to decouple the dependencies among them and propose our algorithm to solve this optimization function.

Algorithm 1 COLLABORATIVE MULTI-FEATURE FUSION FOR TRANSDUCTIVE SPECTRAL LEARNING

Input: labeled data $\{\mathcal{X}_l, \mathcal{Y}_l\}$; unlabeled data $\mathcal{X}_u$; K types of features $\{\mathcal{F}^{(k)}\}_{k=1}^K$; cluster numbers of individual feature types $\{M_k\}_{k=1}^K$; class number M; parameters α and β

Output: labels on unlabeled data $\mathcal{Y}_u$

Initialization: initial label matrix $\mathbf{Y}$; normalized graph Laplacians of individual feature types $\mathbf{L}'_k \leftarrow \mathbf{L}_k, k = 1, 2\cdots K$
repeat
  / / Spectral embedding
  $\mathbf{R}_k \leftarrow eig\left(\mathbf{L}'_k, M_k\right), k = 1, 2\cdots K$ (Eq. 4)
  / / Generate feature co-occurrence patterns
  $\mathbf{T}_v = \left[\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_K\right]^T$ (Eq. 5)
  / / Build multi-feature similarity graph Laplacian
  $\mathbf{W}_v \leftarrow \mathbf{T}_v^T\mathbf{T}_v$ (Eq. 6)
  $\mathbf{L}_v \leftarrow \mathbf{I}_N - \mathbf{D}_v^{-1/2}\mathbf{W}_v\mathbf{D}_v^{-1/2}$ (Eq. 8)
  / / Compute gradient *w.r.t.* class-normalized labels
  $\nabla_{(\mathbf{SY})}\Omega \leftarrow 2\left[\alpha\mathbf{P}\mathbf{L}_v\mathbf{P} + \beta(\mathbf{P} - \mathbf{I}_N)^2\right]\mathbf{S}\mathbf{Y}$ (Eq. 15)
  / / Reset unlabeled data
  $\mathcal{X}'_u \leftarrow \mathcal{X}_u$
  / / Gradient search for unlabeled data labeling
  repeat
    $\left(\tilde{i}, \tilde{j}\right) \leftarrow \arg\min_{(i,j):\ x_i\in\mathcal{X}_u, j\in\{1.2,\cdots,M\}} \nabla_{(\mathbf{SY})}\Omega$
    $\mathbf{Y}_{\tilde{i},\tilde{j}} \leftarrow 1$
    $y_{\tilde{i}} \leftarrow \tilde{j}$
  until $\mathcal{X}'_u \leftarrow \mathcal{X}'_u \backslash \tilde{x}_i = \emptyset$
  / / Update soft class labels of unlabeled data
  $\mathbf{R}_v \leftarrow \mathbf{P}\mathbf{S}\mathbf{Y}$ (Eq. 13)
  / / Regularize graph Laplacians for each feature types
  $\mathbf{L}'_k \leftarrow \mathbf{L}_k - \alpha\sum_{k=1}^K \mathbf{D}_v^{-\frac{1}{2}}\mathbf{R}_v\mathbf{R}_v^T\mathbf{D}_v^{-\frac{1}{2}}, k = 1, 2\cdots K$ (Eq. 18)
until Ω is not decreasing

D. Optimization: Collaboration between Clustering and Classification

In this section we decouple the dependencies among the terms of Eq. 9 to solve the objective function. More specifically, we fix the soft feature clustering results $\{\mathbf{R}_k\}_{k=1}^K$ in individual feature types to optimize Ω over the class labeling results with soft class labels $\mathbf{R}_v$ and hard class labels $\mathbf{Y}$ together. And similarly, we fix the class labeling results with soft class labels $\mathbf{R}_v$ and hard class labels $\mathbf{Y}$ simultaneously to optimize Ω over the soft feature clustering results $\{\mathbf{R}_k\}_{k=1}^K$ in individual feature types. In the class labeling update step, we solve $\mathbf{R}_v$ by an analytical form, and then optimize Ω over $\mathbf{Y}$ using a gradient based greedy search approach. In the feature clustering update step, we optimize Ω over $\mathbf{R}_k, k = 1, 2, \cdots, K$ separately.

The closed form of $\mathbf{R}_v$. Since Ω is quadratic w.r.t. $\mathbf{R}_v$, similar to [11], we are allowed to zero the partial derivative to obtain the analytical solution of $\mathbf{R}_v$ w.r.t. $\mathbf{Y}$ and $\{\mathbf{R}_k\}_{k=1}^K$. We then have:

$$\frac{\partial \Omega}{\partial \mathbf{R}_v} = \alpha \mathbf{L}_v \mathbf{R}_v + \beta \left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right) = 0, \tag{12}$$

which implies

$$\mathbf{R}_v = \left(\frac{\alpha}{\beta}\mathbf{L}_v + \mathbf{I}_N\right)^{-1} \mathbf{S}\mathbf{Y} = \mathbf{P}\mathbf{S}\mathbf{Y}, \tag{13}$$

where $\mathbf{P} = \left(\frac{\alpha}{\beta}\mathbf{L}_v + \mathbf{I}_N\right)^{-1}$, which is related to $\{\mathbf{R}_k\}_{k=1}^K$ according to Eq 8.

The soft class labels $\mathbf{R}_v$ make the transition smooth from feature clustering results of multiple feature types $\{\mathbf{R}_k\}_{k=1}^K$ to the prediction of hard class labels $\mathbf{Y}$ for the dataset $\mathcal{X}$. Then we can substitute the analytical solution of $\mathbf{R}_v$ in Eq. 13 to Eq. 9, and optimize Ω over $\mathbf{Y}$.

Optimize Ω over $\mathbf{Y}$. Given $\{\mathbf{R}_k\}_{k=1}^K$, we use the gradient based greedy search approach [11] to optimize the binary integer optimization. It it worth noting that searching along the gradient of hard class labels $\mathbf{Y}$ and class-normalized labels $\mathbf{S}\mathbf{Y}$ is in fact equivalent. Therefore,

$$\mathbf{Y}^{\text{update}}\left(\{\mathbf{R}_k\}_{k=1}^K\right) = \arg\min_{\mathbf{Y}} \nabla_{\mathbf{Y}}\Omega = \arg\min_{\mathbf{Y}} \nabla_{(\mathbf{S}\mathbf{Y})}\Omega, \tag{14}$$

where the gradient of Ω over $\mathbf{S}\mathbf{Y}$ is:

$$\nabla_{(\mathbf{S}\mathbf{Y})}\Omega = 2\left[\alpha \mathbf{P}\mathbf{L}_v\mathbf{P} + \beta(\mathbf{P} - \mathbf{I}_N)^2\right]\mathbf{S}\mathbf{Y}. \tag{15}$$

Eq. 14 shows how to leverage the feature clustering structures in multiple types of features $\{R_k\}_{k=1}^K$ and the labeled data to predict the labels of unlabeled data.

Optimize Ω over $\mathbf{R}_k$, $\forall\ k = 1, 2, \cdots, K$. We propose to update data clustering results by data class labeling results, which have not been studied before to the best of our knowledge. To this end, we fix $\{\mathbf{R}_i\}_{i\neq k}$, $\mathbf{R}_v$ and $\mathbf{Y}$, and obtain an equivalent minimization function J to minimize Ω (Eq. 9), where[1]

$$\begin{aligned} &J\left(\mathbf{R}_k, \mathbf{R}_v, \mathbf{Y}, \{\mathbf{R}_i\}_{i\neq k}\right) \\ &= \sum_{i=1}^{K} tr\left\{\mathbf{R}_i^T\left(\mathbf{L}_i - \alpha \mathbf{D}_v^{-\frac{1}{2}}\mathbf{R}_v\mathbf{R}_v^T\mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_i\right\}, \end{aligned} \tag{16}$$

subject to $\mathbf{R}_k^T\mathbf{R}_k = \mathbf{I}_{M_k}$. However, the partial derivative of $\mathbf{D}_v$ w.r.t. $\mathbf{R}_k$ is intractable since there is a diagonalization operation in Eq. 7. We therefore use the values of $\{\mathbf{R}_i\}_{i=1}^K$ at the previous iteration to estimate $\mathbf{D}_v$ and treat it as a constant matrix. Then the optimization turns out to minimize the following objective:

$$\begin{aligned} &\Omega_{\text{type}}^{\text{new}}\left(\mathbf{R}_k, \mathbf{Y}, \{\mathbf{R}_j\}_{i\neq k}\right) \\ &= tr\left\{\mathbf{R}_k^T\left(\mathbf{L}_k - \alpha \mathbf{D}_v^{-\frac{1}{2}}\mathbf{R}_v\mathbf{R}_v^T\mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_k\right\}, \end{aligned} \tag{17}$$

subject to $\mathbf{R}_k^T\mathbf{R}_k = \mathbf{I}_{M_k}$. It becomes a spectral clustering with a regularized graph Laplacian:

$$\mathbf{L}_k^{\text{new}} = \mathbf{L}_k - \alpha \sum_{k=1}^{K} \mathbf{D}_v^{-\frac{1}{2}}\mathbf{R}_v\mathbf{R}_v^T\mathbf{D}_v^{-\frac{1}{2}}. \tag{18}$$

[1]The detailed derivation is shown in Appendix.

By using the Rayleigh-Ritz theorem [46], we can update $\mathbf{R}_k$ as the first M_k eigenvectors corresponding to the M_k smallest eigenvalues of $\mathbf{L}_k^{\text{new}}$:

$$\mathbf{R}_k^{\text{update}}\left(\mathbf{R}_k, \mathbf{Y}, \{\mathbf{R}_j\}_{j\neq k}\right) = eig\left(\mathbf{L}_k^{\text{new}}, M_k\right). \tag{19}$$

Eq. 19 shows how to tune the feature clustering result of each feature type $\mathbf{R}_k, \forall k = 1, 2, \cdots, K$ by learning from the known data class labels and the feature clustering results of the other feature types. It is worth noting that, at the beginning, our method does not require the clustering agreement among different feature types. However, by further optimizing the objective, individual feature types will be regularized by known data class labels, and each individual feature type will be influenced by other feature types. In fact, the regularized graph Laplacian (Eq. 18) in each feature type has become a multi-feature Laplacian representation. Such multi-feature Laplacian representations should gradually agree with each other.

We also notice that the adjustment of $\mathbf{R}_k$ is related to its value in the previous iteration. This leads to a gradual change of $\mathbf{R}_k$. Strictly, because of this manipulation, it is difficult to establish theoretical analysis on the algorithm convergence. Nevertheless in our observation our method usually converges in few steps. We show our complete solution in Algorithm 1.

IV. Experiments

A. Experiment Setting

In the experiments, the regularized parameters are both set to 1. Specifically, in our algorithm, we set $\alpha = 1$, and $\beta = 1$ as we observe they are not very sensitive. The observation on our extension of GTAM is consistent with GTAM in [11], which is also robust to the parameter setting. For a fair comparison, we set $C = 1$ in RWMV [13], and set $\mu = 1$ in GTAM [11]. As suggested in [13], the graph combination parameters in RWMV is set equally, *i.e.*, $\alpha_i = 1/M, i = 1, 2, \cdots, K$. Besides, we use $\sigma = 0.3$ and Euclidean distance measure to build graph similarities for the simulation data (Section IV-B). In the real datasets, the bandwidth parameter σ equals to the median of the pairwise distances. We measure $\text{dist}^2(\cdot,\cdot)$ as χ^2 distance in the Oxford 17-Category Flower Dataset as provided in [47], [48] (Section IV-D). Euclidean distance measure is used in the UCI Handwritten Digit Dataset (Section IV-C), Human Body Motion Dataset (Section IV-E) and UC Merced Land Use Dataset (Section IV-F). Moreover, in each real dataset experiment, we randomly pick labeled samples and run 10 rounds for performance evaluation.

B. Synthetic Data

We synthesize a toy dataset with two types of features in Fig. 2. Each type of features is described by a 2-dimensional feature space. The dataset has four classes labeled by *"1"*, *"2"*, *"3"* and *"4"*, respectively. The labeled data are highlighted using different colors. Each class has 200 samples. Feature type #*1* has two clusters: Above moon and Below moon. Feature type #*2* also has two clusters: Left moon and Right moon.

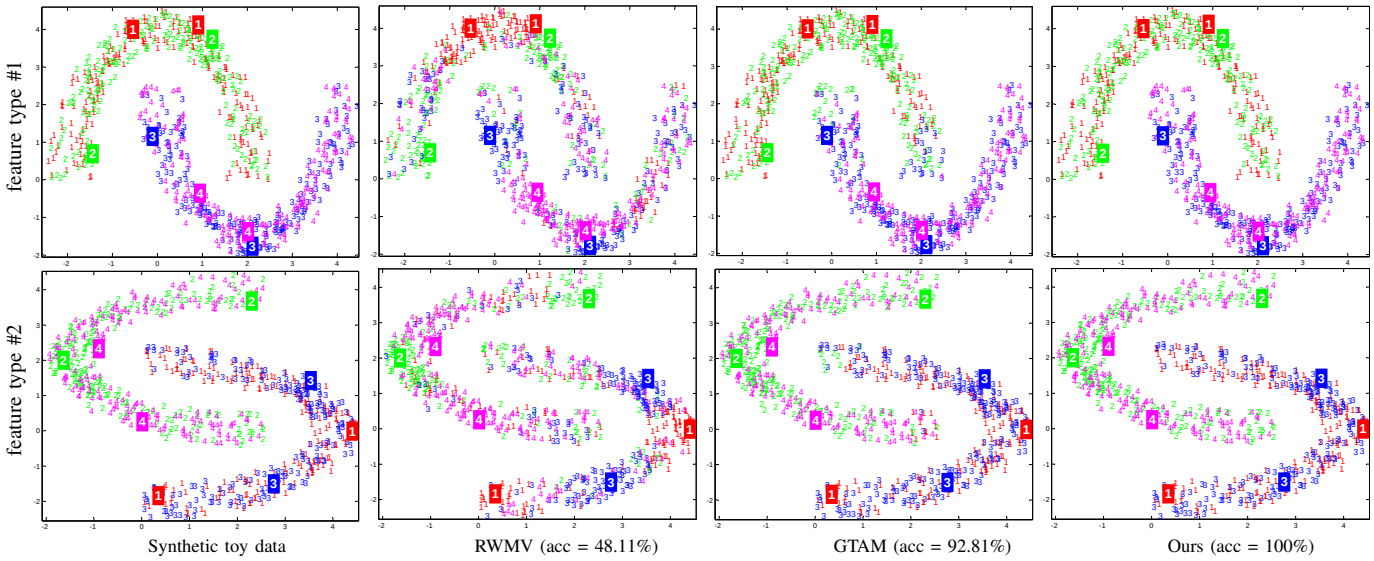


Fig. 2. Classification on synthetic toy data with two feature types. Different markers, *i.e.*, "1", "2", "3" and "4", indicate four different classes. Shading markers highlight the labeled data. The first column shows the synthetic toy data. The last three columns show the classifying results of RWMV [13], GTAM [11] and our proposed approach. Best seen in color.

It is worth noting that the feature clusters are mixed across different classes. In feature type #*1*, both classes #*1* and #*2* share cluster *A*; and both classes #*3* and #*4* share cluster *B*. In feature type #*2*, both classes #*2* and #*4* share cluster *L*; and both classes #*1* and #*3* share cluster *R*. Therefore it is infeasible to classify the data by using a single feature type. In addition, a direct concatenation of features from multiple feature types will diminish the differences among samples, thus cannot distinguish all samples from different classes. For example, by using the GTAM [11], the concatenated features obtain 92.81% accuracy, but cannot disambiguate among several samples. In terms of general multi-feature fusion approaches, *e.g.*, RWMV [13], the requirement that the data categorization results in individual feature types should agree with each other does not hold, *e.g.*, the toy data. Hence the accuracy of RWMV just reaches 48.11%.

In contrast, by utilizing the feature co-occurrence patterns among multiple feature types, our approach can learn a favourable clustering, and the accuracy is 100%. Specifically, class #*1* exhibits the co-occurrence of cluster *A* in feature type #*1* and cluster *R* in feature type #*2*; class #*2* exhibits the co-occurrence of cluster *A* in feature type #*1* and cluster *L* in feature type #*2*; class #*3* exhibits the co-occurrence of cluster *B* in feature type #*1* and cluster *R* in feature type #*2*; and class #*4* exhibits the co-occurrence of cluster *B* in feature type #*1* and cluster *L* in feature type #*2*.

C. UCI Handwritten Digit Dataset

To evaluate how multiple feature types influence handwritten digit recognition, we test the multi-feature digit dataset [49] from the UCI Machine Learning Repository [50]. It consists of features of handwritten numerals ('0'–'9') extracted from a collection of Dutch utility maps. There are 200 samples in each class. So the data set has a total of 2,000 samples. These digits are represented by six types of features: (1) 76-dimensional Fourier coefficients of the character shapes (fou); (2) 64-dimensional Karhunen-Loeve coefficients (kar); (3) 240-dimensional pixel averages in 2×3 windows (pix); (4) 216-dimensional profile correlations (fac); (5) 47-dimensional Zernike moments (zer); and (6) 6-dimensional morphological features (mor). All features can concatenate to generate the 649-dimensional features. As the source image dataset is not available [50], we show the sampled images by the 240-dimensional pixel features in Fig. 3.

In this experiment, the first 50 samples from each digit class are labeled for transductive learning. The classification results on the remaining 1500 unlabeled samples are used for evaluation. For each class, we randomly pick labeled data from the 50 labeled candidates and vary the size from 2 to 20. The accuracy comparison results are shown in Fig. 4, including our approach, GTAM [11] (on the best single feature type, the worst single feature type and the concatenations of all feature types) and RWMV [13] (on all feature types).

The various performances of individual feature types show there is a substantial disagreement among feature types in this dataset. The concatenation of all the six feature types performs better than the worst single feature but worse than the

Fig. 3. Visualization of some sampled images of UCI Handwritten Digits. Each row shows 30 images from the same class of digits.

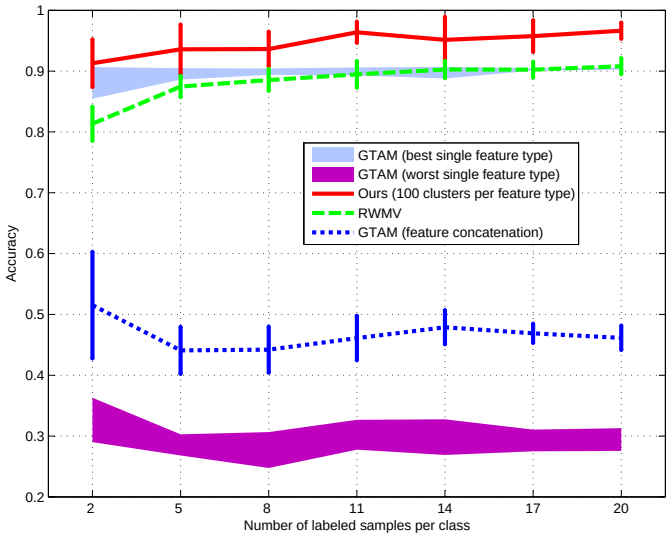


Fig. 4. Performance comparison on UCI handwritten digits.

#cluster per feature type	Accuracy	#cluster per feature type	Accuracy
5	0.870 ± 0.012	50	$\mathbf{0.970 \pm 0.001}$
10	0.925 ± 0.002	100	0.966 ± 0.013
20	0.958 ± 0.001	200	0.936 ± 0.035

TABLE I
PERFORMANCE OF OUR APPROACH ON UCI HANDWRITTEN DIGITS UNDER DIFFERENT CLUSTER NUMBERS PER FEATURE TYPE. THE SIZE OF LABELED DATA IS 20.

best single feature when using GTAM. This also shows that feature concatenation can be easily affected by the bad feature types, thus not the best choice for multi-feature transductive learning. By a linear combination of similarity matrices of the six feature types [13], the performance of RWMV can be close to that of GTAM on the best single feature type, but is still affected by the poor feature types. The best performance is achieved by our approach, which benefits from learning the feature co-occurrence patterns. In Fig. 4, we show the results of our approach with 100 clusters per feature type. On the one hand, we do not force individual feature types to have the same clustering structure, thus the feature co-occurrence patterns faithfully reflect the data distribution characteristics. On the other hand, as discussed in Section III-C, the feature co-occurrence patterns are less sensitive to poor feature types when performing graph transduction. Therefore, our approach achieves a noticeable performance improvement by combining all the individual feature types, despite some poor feature types and the disagreement among different feature types.

We also study the impact of the cluster number in each feature type. The performance comparison is shown in Table I, in which the number of clusters per feature type varies from 5 to 200, with the size of labeled samples per class equal to 20. With the increase of cluster number per feature type, the accuracy increases first then decreases. This is because either under-clustering or over-clustering will discourage the investigation of data distributions in multiple feature types. Despite that, there still exists a large number of effective over-clustering which can produce informative feature clusters, boosting the performance of graph transduction. For example, when the cluster number per feature type is between 10 to 200, the labeling accuracies of unlabeled data all reach more than 90%.

D. Oxford Flower Dataset

Our approach can also combine different visual features for object recognition. The Oxford Flower Dataset is used for experiment, which is composed of 17 flower categories, including *Buttercup, Coltsfoot, Daffodil, Daisy, Dandelion, Fritillary, Iris, Pansy, Sunflower, Windflower, Snowdrop, LilyValley, Bluebell, Crocus, Tigerlily, Tulip, Cowslip*. Each category is with 80 images. We show 5 representative flowers for each class in Fig. 5. In the experiment, we use seven pairwise distance matrices provided by the dataset. These matrices are precomputed respectively from seven types of image appearance features [47], [48]. Using these pairwise distances, we compute the similarities between pairs of features according to Eq. 1.

We label the first 30 samples per class and use them for transductive learning. The classification performance on the remaining 850 unlabeled samples is used for evaluation. We compare our approach with GTAM [11] (on the best single feature type, the worst single feature type) and RWMV [13]

Fig. 5. Sample images from Oxford 17-Category Flower Dataset. Five images are shown for each category. Each category contains instances of pose variations, scale changes, illumination variations, large intra-class variations and self-occlusion.

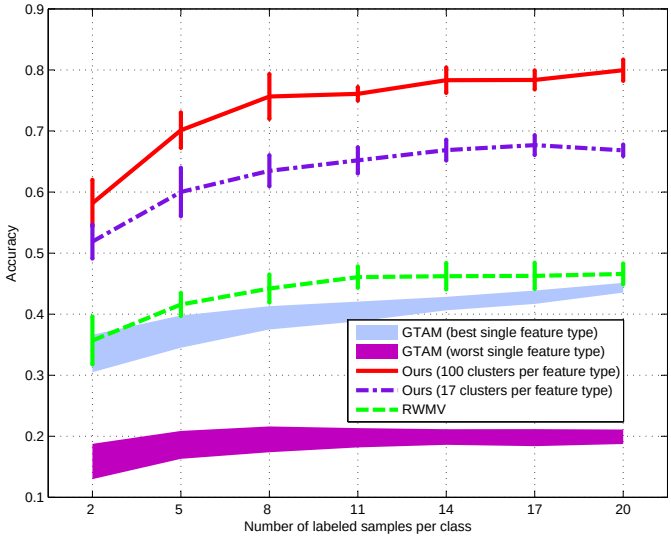
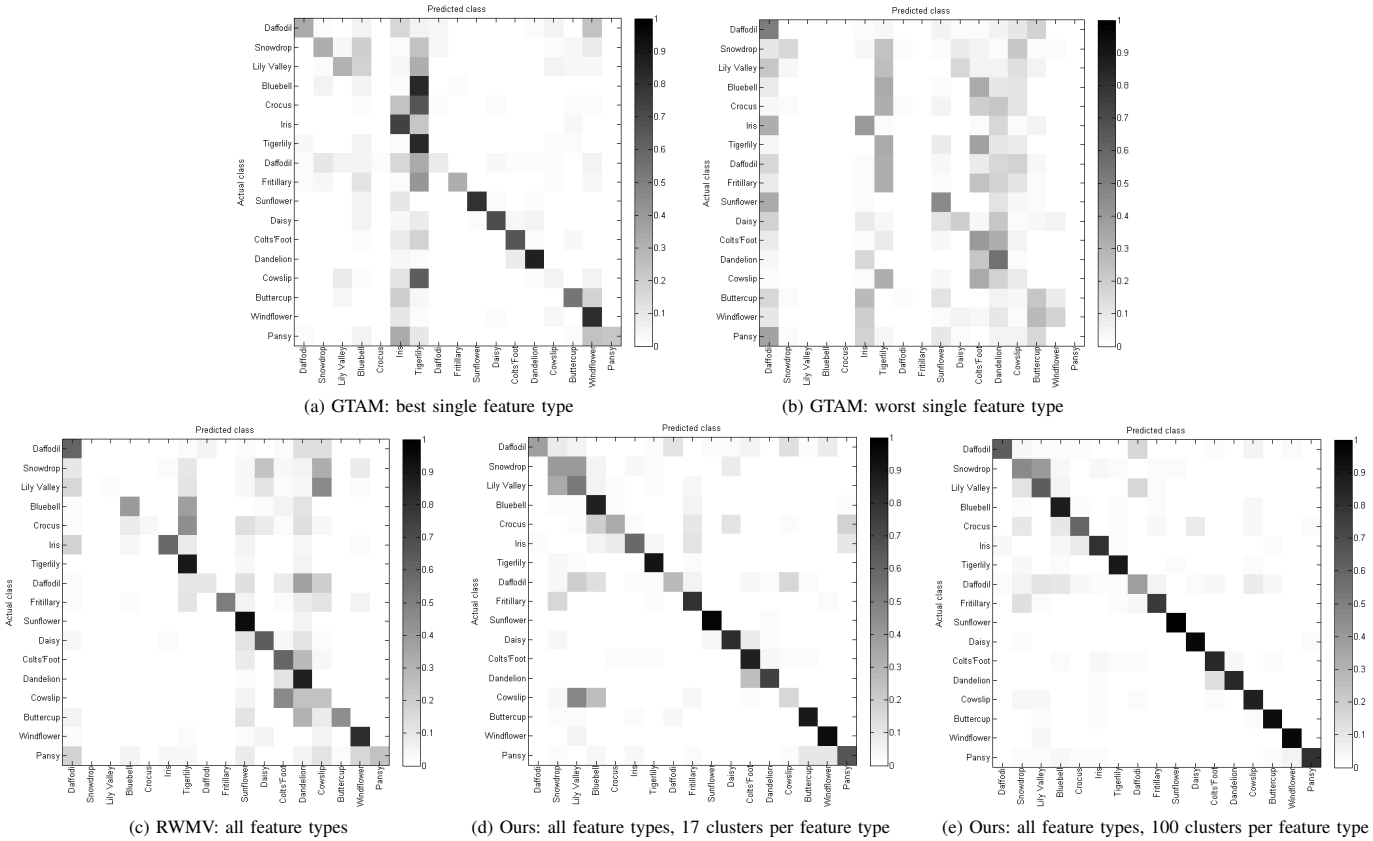


Fig. 6. Performance comparison on Oxford 17-category flowers.



(a) GTAM: best single feature type

(b) GTAM: worst single feature type

(c) RWMV: all feature types

(d) Ours: all feature types, 17 clusters per feature type

(e) Ours: all feature types, 100 clusters per feature type

Fig. 7. Confusion matrix comparison on Oxford 17-category flowers.

(on all feature types) w.r.t. mean value and standard deviation of classification accuracies in Fig. 6. For each class, we randomly pick labeled data from the 30 labeled candidates and vary the size from 2 to 20. In Fig. 7, we show the confusion matrices of compared methods when there are 20 labeled data samples for each class. Because we do not have the original features, we do not compare the results of feature concatenation.

As shown in Fig. 6, the individual types of features all show poor performances. Moreover, the best and worst single feature types confuse in different flower classes (Fig. 7 (a),(b)), resulting in a large performance gap. Therefore there are serious disagreements among different feature types. In this case, the effectiveness of the linear combination of similarity matrices is limited to reduce the classification confusion caused by different feature types. By comparing Fig. 7 (c) and Fig. 7 (a),(b), we can see that the confusion matrix generated by RWMV is only a slight smooth over different feature types. Hence RWMV only brings a little gain compared with the best single feature type (Fig. 6). In contrast, the confusion matrices in Fig. 7 (d) and (e) show that our approach can adequately alleviate classification confusion either using 17 clusters or 100 clusters per feature type. The performances consequently show significant improvements over GTAM on individual types of features and RWMV on all feature types. As mentioned in Section IV-C, because of better exploring the feature clustering structures of individual feature types, our method using 100 clusters per feature type performs better than that of using 17 clusters per feature type.

E. Human Body Motion Dataset

In video data, appearance and motion features complement each other for body motion description and recognition. Therefore, in this section, we combine such two feature types for video recognition. We experiment on the recent Body Motion Dataset, which is included in UCF101 [51] and contains 1910 videos in total, with 16 categories of human body motion actions: *Baby Crawling, Blowing Candles, Body Weight Squats, Handstand Pushups, Handstand Walking, Jumping Jack, Lunges, Pull Ups, Push Ups, Rock Climbing Indoor, Rope Climbing, Swing, Tai Chi, Trampoline Jumping,*

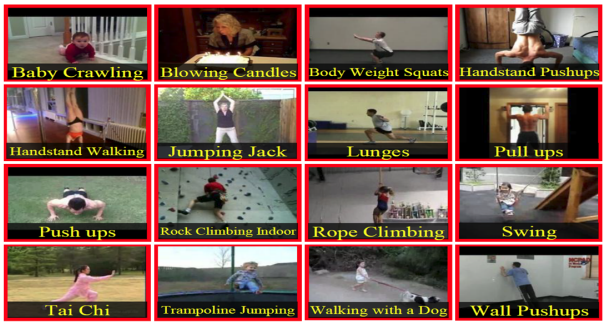


Fig. 8. Sample videos from Human Body Motion Dataset. One sample from each category is shown above.

# labeled per class	GTAM with HOG	GTAM with MBH	GTAM with feature concat	RWMV with all feature types	Ours 16 clusters per feature type	Ours 50 clusters per feature type	Ours 100 clusters per feature type
20	0.088 ± 0.004	0.140 ± 0.007	0.104 ± 0.007	0.078 ± 0.007	0.340 ± 0.042	0.464 ± 0.040	$\mathbf{0.511 \pm 0.026}$
17	0.087 ± 0.003	0.135 ± 0.008	0.101 ± 0.008	0.080 ± 0.011	0.332 ± 0.040	0.465 ± 0.032	$\mathbf{0.509 \pm 0.022}$
14	0.088 ± 0.004	0.133 ± 0.013	0.103 ± 0.009	0.082 ± 0.012	0.320 ± 0.029	0.439 ± 0.046	$\mathbf{0.488 \pm 0.031}$
11	0.090 ± 0.004	0.135 ± 0.013	0.107 ± 0.007	0.097 ± 0.024	0.301 ± 0.039	0.416 ± 0.050	$\mathbf{0.474 \pm 0.025}$
8	0.089 ± 0.008	0.132 ± 0.014	0.102 ± 0.012	0.101 ± 0.030	0.261 ± 0.036	0.381 ± 0.039	$\mathbf{0.424 \pm 0.028}$
5	0.081 ± 0.012	0.118 ± 0.019	0.099 ± 0.019	0.089 ± 0.037	0.234 ± 0.034	0.353 ± 0.026	$\mathbf{0.395 \pm 0.037}$
2	0.081 ± 0.012	0.132 ± 0.029	0.103 ± 0.023	0.075 ± 0.019	0.197 ± 0.047	0.302 ± 0.038	$\mathbf{0.317 \pm 0.034}$

TABLE II
PERFORMANCE COMPARISON ON HUMAN BODY MOTION VIDEOS.

Walking with a Dog, Wall Pushups. For each category, one sample action is shown in Fig. 8. Each video is represented as dense appearance trajectories based on Histogram of Oriented Gradients (HOG) and dense motion trajectories based on Motion Boundary Histograms (MBH) [52].

We label the first 50 samples per class for transductive learning. For each class, we randomly pick the labeled data from the 50 candidates and vary the size from 2 to 20. The classification performance on the remaining 1110 unlabeled samples are used for evaluation. Again, we compare our approach with GTAM [11] (on individual feature types and feature concatenation) and RWMV [13] (on all feature types) in Table II.

Comparing the first two columns of Table II, we can see that motion features perform better than appearance features in human body motion classification. The 3^{rd} and 4^{th} columns show that the approaches of GTAM on feature concatenation and RWMV that uses all feature types usually perform better than GTAM on the poorer feature type, but still cannot compete against GTAM on the better feature type. Therefore they are not suitable to handle appearance and motion feature fusion. In contrast, our approach using 16 clusters per feature type (as shown in the 5^{th} column) improves GTAM on the best single feature type. To further investigate clustering structures of individual feature types sufficiently, we over-cluster individual types of features and obtain 50 or 100 clusters per feature type. The results are shown in the last two columns of Table II. This process brings a significantly improved performance in all labeled data sizes, which further verifies the effectiveness of our approach in fusing appearance and motion features.

Fig. 9. Sample images from UC Merced 21-Category Land Use Dataset. Five samples from each category are shown above.

F. UC Merced Land Use Dataset

To further evaluate our method, we conduct scene recognition experiment on UC Merced Land Use Dataset [53] and compare one more recent method [12] except for GTAM and RWMV. This dataset contains 21 classes of aerial orthoimagery: *agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks, and tennis courts*. Each class has 100 images with resolution 256×256. We show 5 sample images for each class in Fig. 9. For each image, we extract SIFT features over the 16×16 patches with spacing of 6 pixels. By applying the locality-constrained linear coding (LLC) [54] on all SIFT features extracted from this dataset, and running spatial pyramid max pooling on images with 1×1, 2×2, and 4×4 sub-regions, we generate 3 scales of image representations with dimensionalities of 1×1024, $2 \times 2 \times 1024$, and $4 \times 4 \times 1024$ as three feature types. The image representations with different scales result in different types of features.

We select the first 40 samples per class as the labeled data pool and vary the number (from 2 to 20) of labeled samples from the pool. The classification performance on the remaining 1260 unlabeled samples is reported for evaluation. Besides GTAM [11] and RWMV [13], we also compare with graph transduction game (GTG) [12] in Table III. For GTAM or GTG, we separately perform it on each single feature type or feature concatenation, and report the best performance it obtains. For RWMV and our method, we report the results of muti-feature fusion. As can be seen from the 1^{st} to the 4^{th} columns, GTG generally outperforms GTAM, RWMV, and performs better than our method with 21 clusters per feature type. However, by appropriately increasing the number of clusters per feature type, the classification performance of our method can be considerably enhanced as shown in the last two columns of Table III. The results further justify the benefit of our method and the effectiveness of collaboration between clustering and classification. Overall, the performance gain depends on the spectral clustering results of using individual features, as well as the complementary among the multiple features.

V. CONCLUSION

The different data characteristics and distributions among multiple feature types challenge many existing multi-feature

# labeled per class	GTAM [11]	GTG [12]	RWMV [13]	Ours 21 clusters per feature type	Ours 50 clusters per feature type	Ours 100 clusters per feature type
20	0.334 ± 0.018	0.379 ± 0.012	0.304 ± 0.010	0.357 ± 0.020	0.485 ± 0.023	$\mathbf{0.554 \pm 0.023}$
17	0.331 ± 0.019	0.373 ± 0.018	0.298 ± 0.016	0.337 ± 0.028	0.484 ± 0.020	$\mathbf{0.527 \pm 0.023}$
14	0.340 ± 0.017	0.380 ± 0.018	0.293 ± 0.017	0.325 ± 0.029	0.458 ± 0.028	$\mathbf{0.511 \pm 0.035}$
11	0.334 ± 0.026	0.371 ± 0.020	0.290 ± 0.017	0.315 ± 0.028	0.452 ± 0.018	$\mathbf{0.488 \pm 0.025}$
8	0.333 ± 0.031	0.368 ± 0.022	0.291 ± 0.026	0.293 ± 0.026	0.409 ± 0.039	$\mathbf{0.463 \pm 0.037}$
5	0.320 ± 0.022	0.350 ± 0.018	0.276 ± 0.021	0.274 ± 0.027	0.372 ± 0.044	$\mathbf{0.400 \pm 0.036}$
2	0.310 ± 0.038	0.314 ± 0.021	0.243 ± 0.034	0.270 ± 0.043	0.314 ± 0.031	$\mathbf{0.343 \pm 0.067}$

TABLE III
PERFORMANCE COMPARISON ON UC MERCED LAND USE IMAGES.

learning methods. Instead of iteratively updating individual feature type and forcing different feature types to agree with each other, we allow each feature type to perform data clustering by its own and then represent each data sample by a co-occurrence of feature patterns across different feature types. Relying on these feature co-occurrence representations of the data samples, we propose a transductive spectral learning approach, such that the data samples of similar feature co-occurrence pattern will share the same label. To transfer the labels from the labeled data to unlabeled data under our transductive learning formulation, we develop an algorithm that can iteratively refine the spectral clustering results of individual feature types and the labeling results of unlabeled data. The experiments on both synthetic and real-world image/video datasets highlight the advantages of the proposed method to handle multi-feature fusion in transducitve learning.

APPENDIX

Given $\{\mathbf{R}_i\}_{i\neq k}$, $\mathbf{R}_v$ and $\mathbf{Y}$, we show how to obtain Eq. 16 from Ω_{smooth} (in Eq. 9). By using the linearity and cyclicity property of the matrix trace, we have:

$$\begin{aligned}
&\Omega_{\text{smooth}}\left(\mathbf{R}_v, \{\mathbf{R}_j\}_{j=1}^K\right) = tr\left(\mathbf{R}_v^T \mathbf{L}_v \mathbf{R}_v\right) \\
&= tr\left\{\mathbf{R}_v^T\left(\mathbf{I}_N - \mathbf{D}_v^{-\frac{1}{2}} \sum_{i=1}^K \mathbf{R}_i \mathbf{R}_i^T \mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_v\right\} \\
&= tr\left\{\mathbf{R}_v^T \mathbf{R}_v\right\} - tr\left\{\mathbf{R}_v^T\left(\sum_{j=1}^K \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_j \mathbf{R}_j^T \mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_v\right\} \\
&= tr\left\{\mathbf{R}_v^T \mathbf{R}_v\right\} - tr\left(\sum_{j=1}^K \mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_j \mathbf{R}_j^T \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v\right) \\
&= tr\left\{\mathbf{R}_v^T \mathbf{R}_v\right\} - \sum_{j=1}^K tr\left(\mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_j \mathbf{R}_j^T \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v\right) \\
&= tr\left\{\mathbf{R}_v^T \mathbf{R}_v\right\} - \sum_{j=1}^K tr\left(\mathbf{R}_j^T \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v \mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_j\right) \\
&= tr\left\{\mathbf{R}_v^T \mathbf{R}_v\right\} - \sum_{j=1}^K tr\left\{\mathbf{R}_j^T\left(\mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v \mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_j\right\}.
\end{aligned} \tag{20}$$

Substituting Eq. 20 into Eq. 9, and combining the constant terms, we obtain $\Omega = J + C$, where $C = \alpha tr\left\{\mathbf{R}_v^T \mathbf{R}_v\right\} + \beta tr\left\{\left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right)^T\left(\mathbf{R}_v - \mathbf{S}\mathbf{Y}\right)\right\}$ is unchanged since $\mathbf{R}_v$ and $\mathbf{Y}$ are fixed. We therefore only need to minimize J, where

$$\begin{aligned}
&J\left(\mathbf{R}_k, \mathbf{R}_v, \mathbf{Y}, \{\mathbf{R}_i\}_{i\neq k}\right) \\
&= \sum_{i=1}^K \Omega_{\text{type}}\left(\mathbf{R}_i\right) - \alpha \sum_{j=1}^K tr\left\{\mathbf{R}_j^T\left(\mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v \mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_j\right\} \\
&= \sum_{i=1}^K tr\left(\mathbf{R}_i^T \mathbf{L}_i \mathbf{R}_k\right) - \alpha \sum_{i=1}^K tr\left\{\mathbf{R}_i^T\left(\mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v \mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_i\right\} \\
&= \sum_{i=1}^K tr\left\{\mathbf{R}_i^T\left(\mathbf{L}_i - \alpha \mathbf{D}_v^{-\frac{1}{2}} \mathbf{R}_v \mathbf{R}_v^T \mathbf{D}_v^{-\frac{1}{2}}\right)\mathbf{R}_i\right\},
\end{aligned} \tag{21}$$

subject to $\mathbf{R}_k^T \mathbf{R}_k = \mathbf{I}_{M_k}$.

ACKNOWLEDGMENT

This work is supported in part by Nanyang Assistant Professorship SUG M4080134.

REFERENCES

[1] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. ACM Conf. Comp. Learn. Theory*, 1998, pp. 92–100.
[2] S. Yu, B. Krishnapuram, R. Rosales, and R. Rao, "Bayesian co-training," *Journal of Mach. Learn. Research*, vol. 12, pp. 2649–2680, 2011.
[3] M. Blaschko and C. Lampert, "Correlational spectral clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.
[4] L. Cao, J. Luo, F. Liang, and T. Huang, "Heterogeneous feature machines for visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 1095–1102.
[5] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 221–228.
[6] Y. Yeh, T. Lin, Y. Chung, and Y. Wang, "A novel multiple kernel learning framework for heterogeneous feature fusion and variable selection," *IEEE Trans. Multimedia*, vol. 14, no. 3, pp. 563–574, 2012.
[7] H. Wang, G. Zhao, and J. Yuan, "Visual pattern discovery in image and video data: a brief survey," *WIREs Data Min. Knowl. Discovery*, vol. 4, no. 1, pp. 24–37, 2014.
[8] J. Yuan, Y. Wu, and M. Yang, "Discovery of collocation patterns: from visual words to visual phrases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
[9] ——, "From frequent itemsets to semantically meaningful visual patterns," in *Proc. ACM Conf. Knowl. Discovery and Data Min.*, 2007, pp. 864–873.
[10] J. Yuan and Y. Wu, "Mining visual collocation patterns via self-supervised subspace learning," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 1–13, 2012.
[11] J. Wang, T. Jebara, and S. Chang, "Graph transduction via alternating minimization," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 1144–1151.
[12] A. Erdem and M. Pelillo, "Graph transduction as a noncooperative game," *Neural Comput.*, vol. 24, no. 3, pp. 700–723, 2012.
[13] D. Zhou and C. J. C. Burges, "Spectral clustering and transductive learning with multiple views," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 1159–1166.
[14] V. Sa, "Learning classification with unlabeled data," in *Proc. Adv. Neural Inf. Process. Syst.*, 1993, pp. 112–119.

[15] O. Yakhnenko and V. Honavar, "Multiple label prediction for image annotation with multiple kernel correlation models," in *Comput. Vis. and Pattern Recognit. Workshops*, 2009, pp. 8–15.

[16] S. Hwang and K. Grauman, "Learning the relative importance of objects from tagged images for retrieval and cross-modal search," *Int. J. Comput. Vis.*, vol. 100, no. 2, pp. 134–153, 2012.

[17] B. Xie, Y. Mu, D. Tao, and K. Huang, "m-SNE: Multiview stochastic neighbor embedding," *IEEE Trans. Syst., Man, Cybern. B*, vol. 41, no. 4, pp. 1088–1096, 2011.

[18] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-view clustering via joint nonnegative matrix factorization," in *Proc. SIAM Int. Conf. Data Min.*, 2013.

[19] B. Long, P. Yu, and Z. M. Zhang, "A general model for multiple view unsupervised learning," in *Proc. SIAM Int. Conf. Data Min.*, 2008, pp. 822–833.

[20] B. Wang, J. Jiang, W. Wang, Z.-H. Zhou, and Z. Tu, "Unsupervised metric fusion by cross diffusion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2997–3004.

[21] Y. Wang, X. Lin, and Q. Zhang, "Towards metric fusion on multi-view data: a cross-view based graph random walk approach," in *Proc. ACM Conf. Inf. and Knowl. Manage.*, 2013, pp. 805–810.

[22] T. Xia, D. Tao, T. Mei, and Y. Zhang, "Multiview spectral embedding," *IEEE Trans. Syst., Man, Cybern. B*, vol. 40, no. 6, pp. 1438–1446, 2010.

[23] A. Kumar, P. Rai, and H. Daumé III, "Co-regularized multi-view spectral clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 1413–1421.

[24] A. Kumar and H. Daumé III, "A co-training approach for multi-view spectral clustering," in *Proc. Int. Conf. Mach. Learn.*, 2011.

[25] X. Cai, F. Nie, H. Huang, and F. Kamangar, "Heterogeneous image feature integration via multi-modal spectral clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1977–1984.

[26] J. Yu, D. Liu, D. Tao, and H. S. Seah, "On combining multiple features for cartoon character retrieval and clip synthesis," *IEEE Trans. Syst., Man, Cybern. B*, vol. 42, no. 5, pp. 1413–1427, 2012.

[27] X. Wang, B. Qian, J. Ye, and I. Davidson, "Multi-objective multi-view spectral clustering via pareto optimization," in *Proc. SIAM Int. Conf. Data Min.*, 2013.

[28] H. Wang, C. Weng, and J. Yuan, "Multi-feature spectral clustering with minimax optimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014.

[29] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv preprint arXiv:1304.5634*, 2013.

[30] C. Christoudias, R. Urtasun, and T. Darrell, "Multi-view learning in the presence of view disagreement," in *Proc. Uncertain. Artif. Intell.*, 2008.

[31] H. Wang, J. Yuan, and Y. Tan, "Combining feature context and spatial context for image pattern discovery," in *Proc. Int. Conf. on Data Min.*, 2011, pp. 764–773.

[32] H. Wang, J. Yuan, and Y. Wu, "Context-aware discovery of visual co-occurrence patterns," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1805–1819, 2014.

[33] C. Weng, H. Wang, and J. Yuan, "Hierarchical sparse coding based on spatial pooling and multi-feature fusion," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2013, pp. 1–6.

[34] N. Chen, J. Zhu, F. Sun, and E. Xing, "Large-margin predictive latent subspace learning for multi-view data analysis." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 12, pp. 2365–2378, 2012.

[35] X. Zhu and A. Goldberg, "Introduction to semi-supervised learning," *Synth. Lect. Artif. Intell. Mach. Learn.*, vol. 3, no. 1, pp. 1–130, 2009.

[36] W. Liu, J. Wang, and S.-F. Chang, "Robust and scalable graph-based semisupervised learning," *Proceedings of the IEEE*, 2012.

[37] A. Blum and S. Chawla, "Learning from labeled and unlabeled data using graph mincuts," in *Proc. Int. Conf. Mach. Learn.*, 2001.

[38] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proc. Int. Conf. Mach. Learn.*, vol. 20, no. 2, 2003, pp. 912–919.

[39] D. Zhou, O. Bousquet, and J. Weston, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003.

[40] J. De, X. Zhang, and L. Cheng, "Transduction on directed graphs via absorbing random walks," *arXiv preprint arXiv:1402.4566*, 2014.

[41] X. Kong, M. K. Ng, and Z.-H. Zhou, "Transductive multilabel learning via label set propagation," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 704–719, 2013.

[42] B. Wang, Z. Tu, and J. K. Tsotsos, "Dynamic label propagation for semi-supervised multi-class multi-label classification," in *Proc. of IEEE Int. Conf. Comput. Vis.*, 2013, pp. 425–432.

[43] T. Iwata and K. Duh, "Bidirectional semi-supervised learning with graphs," in *Mach. Learn. Knowl. Discovery in Databases*, 2012, pp. 293–306.

[44] U. Von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[45] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2, 2001, pp. 849–856.

[46] H. Lütkepohl, *Handbook of matrices*. John Wiley & Sons, 1996.

[47] M. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 2, 2006, pp. 1447–1454.

[48] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. Indian Conf. Comput. Vis., Graphics Image Process.*, 2008.

[49] M. P. W. van Breukelen, D. M. J. Tax, and J. E. den Hartog, "Handwritten digit recognition by combined classifiers," *Kybernetika*, vol. 34, pp. 381–386, 1998.

[50] K. Bache and M. Lichman, "UCI mach. learn. repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[51] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[52] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 3169–3176.

[53] Y. Yang and S. Newsam, "Spatial pyramid co-occurrence for image classification," in *Proc. of IEEE Int. Conf. Comput. Vis.*, 2011.

[54] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010.

Hongxing Wang (S'11) received the B.S. degree in information and computing science, and the M.S. degree in operational research and cybernetics in 2007 and 2010, respectively, all from Chongqing University, Chongqing, China.

He is currently pursuing the Ph.D. degree at Nanyang Technological University, Singapore. His current research interests include computer vision, image and video analysis, and pattern recognition.

Junsong Yuan (SM14 M'08) is currently a Nanyang Assistant Professor and program director of video analytics at School of EEE, Nanyang Technological University, Singapore. He received Ph.D. from Northwestern University, USA, and M.Eng. from National University of Singapore. Before that, he graduated from Special Class for the Gifted Young of Huazhong University of Science and Technology, China. His research interests include computer vision, video analytics, large-scale visual search and mining, human computer interaction etc. He has published over 100 technical papers, and filed three US patents and two provisional US patents.

He serves as area chair for IEEE Winter Conf. on Computer Vision (WACV'14), IEEE Conf. on Multimedia Expo (ICME'14), Asian Conf. on Computer Vision (ACCV'14), organizing chair for ACCV'14, and co-chairs workshops at IEEE Conf. Computer Vision and Pattern Recognition (CVPR'12'13), IEEE Conf. on Computer Vision (ICCV'13), and SIGGRAPH Asia14. He serves as Guest Editor for International Journal of Computer vision (IJCV), Associate Editor for The Visual Computer journal (TVC) and Journal of Multimedia. He received Nanyang Assistant Professorship and Tan Chin Tuan Exchange Fellowship from Nanyang Technological University, Outstanding EECS Ph.D. Thesis award from Northwestern University, Best Doctoral Spotlight Award from CVPR'09, and National Outstanding Student from Ministry of Education, P.R.China. He gives tutorials at IEEE ICIP'13, FG'13, ICME'12, SIGGRAPH VRCAI'12, and PCM'12.