Multi-feature Spectral Clustering with Minimax Optimization

Hongxing Wang, Chaoqun Weng, and Junsong Yuan
School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore, 639798
{hwang8, weng0018}@e.ntu.edu.sg, jsyuan@ntu.edu.sg

Abstract

In this paper, we propose a novel formulation for multi-feature clustering using minimax optimization. To find a consensus clustering result that is agreeable to all feature modalities, our objective is to find a universal feature embedding, which not only fits each individual feature modality well, but also unifies different feature modalities by minimizing their pairwise disagreements. The loss function consists of both (1) unary embedding cost for each modality, and (2) pairwise disagreement cost for each pair of modalities, with weighting parameters automatically selected to maximize the loss. By performing minimax optimization, we can minimize the loss for the worst case with maximum disagreements, thus can better reconcile different feature modalities. To solve the minimax optimization, an iterative solution is proposed to update the universal embedding, individual embedding, and fusion weights, separately. Our minimax optimization has only one global parameter. The superior results on various multi-feature clustering tasks validate the effectiveness of our approach when compared with the state-of-the-art methods.

1. Introduction

In visual recognition, the image or video data can be naturally characterized by multiple types (modalities) of features to describe different aspects of visual characteristics, such as color, texture, or motion. Instead of using a single feature type to perform image or video categorization, it is of great interests to combine multiple complementary feature modalities to improve the clustering or classification result [12, 28, 31].

Such a multi-feature fusion, however, is challenging due to the possible incompatibility of heterogeneous features. For example, a simple concatenation of them does not guarantee good performance [6]. To perform better multi-feature clustering, some previous work chooses to exploit different feature types separately followed by combining the results through a weighted fusion. However, it is difficult to determine the fusion coefficients for different feature modalities. Kernel fusion k-means [39] and affinity aggregation spectral clustering [12] are recent additions to this family. Alternatively, some approaches aim to seek a feature representation or clustering result that can fit in different feature types. Different fitting criteria can bring a variety of consensus methods, such as correlational spectral clustering [4] and common convex representation [11]. Some recent methods propose to enforce the clustering results of different feature types to agree with each other by mutual regularization of each pair of feature modalities. For example, Kumar *et al*. [15] perform pairwise regularization to push pairs of Laplacian embeddings of different feature types close to each other, as well as centroid regularization to push all Laplacian embeddings of different feature types towards a consensus embedding for multi-feature clustering. However, both the pairwise and centroid regularization still need to specify the weights to reflect the confidence of each feature modality, which is difficult to select without prior knowledge.

We propose a novel minimax formulation to reach a consensus clustering, without requiring to specify the weighting parameter to fuse the multiple feature modalities. Our objective of consensus clustering is to find a universal feature embedding, which not only fits each feature modality well, but also unifies different modalities by minimizing the pairwise disagreement between any two of them. As a result, two types of loss need to be minimized: (1) the unary embedding cost terms for each feature modality, and (2) the pairwise disagreement cost terms for each pair of the feature modalities. The unary embedding cost is measured by the Laplacian embedding at each feature modality. While for the pairwise disagreement cost, instead of measuring the consistency of their data distribution, we project the Laplacian embedding from each feature type to a Regularized Data-Cluster Similarity Matrix using the universal feature embedding, and compute the pairwise Frobenius distance through pairs of regularized data-cluster similarity matrices. Such a measure is more robust to noises.

Our minimax formulation has the following advantages:

- It has only one global parameter, while all fusing weights can be automatically determined via minimax optimization.
- It reaches a harmonic consensus by weighting the cost terms differently during minimax optimization, such that the disagreements among different feature modalities can be effectively reconciled.

Our minimax optimization can be nicely solved via iteratively optimizing (1) the universal feature embedding, (2) individual Laplacian embeddings, and (3) the fusing weights. We test our multi-feature clustering method on four different datasets to categorize images or videos. The superior performances compared with the state of the arts validate that our method can well fuse heterogeneous feature modalities for multi-feature clustering.

2. Related Work

To perform multi-feature clustering, some work exploits different feature types separately and then combine the results with specific weights. The traditional way is to build a probabilistic model for each feature type, and then estimate a mixture of them [3, 27]. Alternatively, relying on kernel combination, one can use each feature type to compute a similarity kernel matrix for a weighted sum [16, 39]. In such a case, similarity matrices are weighted and combined for graph structure fusion [41, 36, 12]. In these methods, weighting scheme is critical to the effectiveness of multi-feature fusion.

Some other work choose to seek a consensus solution that meets different feature types as much as possible. One direct strategy is to pursue a partition consensus from multiple clustering results of different feature types [18]. Another type of consensus analysis is to look for a shared feature representation, such as canonical correlation analysis (CCA) [4, 7], general sparse coding [37], convex multi-view subspace learning [11], Pareto embedding [35], common non-negative matrix factorization (NMF) [1], and structured feature selection [30]. Other consensus methods include multipartite spectral graph partition to minimize disagreements among multiple views [8] and multi-feature low-rank affinity pursuit for spectral clustering [10].

Recently, mutual regularization has shown its effectiveness in multi-feature clustering. The idea is to enforce clustering results of different feature types to agree with each other, which is widely applied to k-means [40, 32, 33], NMF [17], topic model [13], and spectral clustering [15, 14, 6]. Among these methods, pairwise regularization is a representative strategy. However, it generally outputs different solutions from multiple feature types such that a late fusion step is required. Therefore, some approaches apply centroid regularization that regularizes each view towards a consensus solution, *e.g.*, [40, 15, 6]. However, both the pairwise and centroid regularization still need to specify the weights to reflect the confidence of each regularization cost, which is difficult to select without prior knowledge.

Our method falls into the category of mutual regularization for multi-feature spectral clustering. Unlike pervious work, it can automatically determine the weights among different regularization costs using only one hyper parameter, and enable pairwise regularization to reach a consensus solution. The work directly related to our approach are compared in this paper, which include pairwise/centroid co-regularized spectral clustering (PRSC/CRSC) [15] and multi-modal spectral clustering (MMSC) [6]. Besides the above approaches, we also provide a comparison with the recent work: affinity aggregation spectral clustering with optimized weights (AASC) [12].

3. Our Method

3.1. Laplacian Embedding

Spectral embedding of data Laplacian matrix is widely used to disclose data clustering structure [25, 20]. Given N data samples $\mathcal{X} = \{x_i\}_{i=1}^N$ and the corresponding feature descriptors $\mathcal{F} = \{\mathbf{f}_i\}_{i=1}^N$ of a specific feature type, one can compute the similarity between any sample pair using Gaussian kernel:

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

where $dist(\mathbf{f}_i, \mathbf{f}_j)$ denotes the distance between a pair of feature descriptors; σ is the bandwidth parameter. All pairwise similarities compose the similarity matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$, which is further transformed to the $N \times N$ normalized Laplacian matrix:

$$\mathcal{L} = \mathbf{D}^{-\frac{1}{2}}\left(\mathbf{I} - \mathbf{W}\right)\mathbf{D}^{-\frac{1}{2}}, \tag{2}$$

where $\mathbf{D} \in \mathbb{R}^{N\times N}$ is the diagonal degree matrix, whose diagonal entries are given by $d_{ii} = \sum_{j=1}^N w_{ij}$. Spectral embedding is to optimize the following problem [25, 20]:

$$\begin{aligned} \underset{\mathbf{U}\in\mathbb{R}^{N\times K}}{\text{minimize}} \quad & \mathcal{Q} = \operatorname{tr}\left(\mathbf{U}^{\mathrm{T}}\mathcal{L}\mathbf{U}\right) \\ \text{subject to} \quad & \mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}. \end{aligned} \tag{3}$$

The Rayleigh-Ritz theorem [19] states that the solution of $\mathbf{U}$ consists of the eigenvectors corresponding to the first K smallest eigenvalues (the first K smallest eigenvectors) of $\mathcal{L}$. In spectral clustering, the entries of $\mathbf{U}$ measure the similarities between data samples and clusters, as they indicate how likely data samples belong to specific clusters. A simple k-means clustering on rows of $\mathbf{U}$ can transform the real valued cluster similarities into discrete cluster indicators [25, 20].

3.2. Regularized Data-Cluster Similarity Matrix

For each feature type, we follow [15] to obtain the *Data-Data Similarity Matrix* by inner product:

$$\mathbf{S}\left(\mathbf{U}\right) = \mathbf{U}\mathbf{U}^{\mathrm{T}}. \tag{4}$$

Let $\mathbf{V} \in \mathbb{R}^{N\times K}$ be the final cluster indicator matrix agreed among multiple feature types. We define the *Regularized Data-Cluster Similarity Matrix* as the projection of $\mathbf{S}$ onto $\mathbf{V}$:

$$\mathbf{P}_{\mathbf{V}}\left(\mathbf{U}\right) = \mathbf{S}\left(\mathbf{U}\right)\mathbf{V} = \mathbf{U}\mathbf{U}^{\mathrm{T}}\mathbf{V}. \tag{5}$$

Compared to the original data-cluster similarity matrix $\mathbf{U}$, the regularized data-cluster similarity matrix $\mathbf{P}_{\mathbf{V}}\left(\mathbf{U}\right)$ measures the data-cluster similarity of each data sample with the final clustering solution $\mathbf{V}$. In the following, we will relax the final clustering solution $\mathbf{V}$ to be a real-valued *universal feature embedding with* orthonormal constraints: $\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I}$. As a result, Eq. 5 enables self projection to be invariant:

$$\mathbf{P}_{\mathbf{V}}\left(\mathbf{V}\right) = \mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{V}. \tag{6}$$

3.3. Towards Agreement among Regularized Data-Cluster Similarity Matrices

Suppose we have M different types of features in total. Our motivation is to encourage the regularized data-cluster similarity matrices to be similar between any two feature types, *e.g.*, type i and type j. Therefore, we propose to minimize the following disagreement measure:

$$\mathcal{D}_{\mathbf{V}}\left(\mathbf{U}_i, \mathbf{U}_j\right) = \left\|\mathbf{P}_{\mathbf{V}}\left(\mathbf{U}_i\right) - \mathbf{P}_{\mathbf{V}}\left(\mathbf{U}_j\right)\right\|_{\mathrm{F}}^2. \tag{7}$$

Instead of forcing pairwise data-data similarity matrices to agree between two feature types in [15], we relax the constraint to data-cluster similarity matrices for noise suppression. Besides that, we propose an additional requirement that the two feature embeddings $\mathbf{U}_i$ and $\mathbf{U}_j$ in Eq. 7 should accommodate the universal feature embedding $\mathbf{V}$. Thus $\mathcal{D}_{\mathbf{V}}\left(\mathbf{U}_i, \mathbf{V}\right)$ and $\mathcal{D}_{\mathbf{V}}\left(\mathbf{U}_j, \mathbf{V}\right)$ should also be minimized. We thus further extend Eq. 7 into Eq. 8 to measure the disagreement among $\mathbf{U}_i$, $\mathbf{U}_j$ and $\mathbf{V}$:

$$\begin{aligned}\mathcal{Q}_{ij} &= \frac{1}{2}\left\{\mathcal{D}_{\mathbf{V}}\left(\mathbf{U}_i, \mathbf{U}_j\right) + \mathcal{D}_{\mathbf{V}}\left(\mathbf{U}_i, \mathbf{V}\right) + \mathcal{D}_{\mathbf{V}}\left(\mathbf{U}_j, \mathbf{V}\right)\right\} \\ &= \mathrm{tr}\left\{\mathbf{V}^{\mathrm{T}}\left[\mathbf{I} - \mathrm{sym}\left(\mathbf{U}_i\mathbf{U}_i^{\mathrm{T}}\mathbf{U}_j\mathbf{U}_j^{\mathrm{T}}\right)\right]\mathbf{V}\right\},\end{aligned} \tag{8}$$

where $\mathrm{sym}\left(\mathbf{A}\right) = \left(\mathbf{A} + \mathbf{A}^{\mathrm{T}}\right)/2$ for any square matrix $\mathbf{A}$. To derive Eq. 8, we use the trace expansion of the Frobenius norm, as well as the linearity and cyclicity properties of matrix trace. Now Let

$$\mathcal{L}_{ij} = \mathbf{I} - \mathrm{sym}\left(\mathbf{U}_i\mathbf{U}_i^{\mathrm{T}}\mathbf{U}_j\mathbf{U}_j^{\mathrm{T}}\right), \tag{9}$$

then Eq. 8 becomes

$$\mathcal{Q}_{ij} = \mathrm{tr}\left(\mathbf{V}^{\mathrm{T}}\mathcal{L}_{ij}\mathbf{V}\right). \tag{10}$$

In addition, according to Eq. 3, we also need to minimize the unary cost of spectral embedding in each feature type for $1 \leq i \leq M$:

$$\mathcal{Q}_{ii} = \mathrm{tr}\left(\mathbf{U}_i^{\mathrm{T}}\mathcal{L}_i\mathbf{U}_i\right), \tag{11}$$

where $\mathcal{L}_i$ denotes the normalized Laplacian matrix of a specific feature type; $\mathbf{U}_i$ corresponds to Laplacian embedding.

Therefore, $\forall 1 \leq i \leq j \leq M$, we need to minimize both the pairwise disagreement cost defined by Eq. 10, as well as the unary spectral embedding cost defined by Eq. 11: $\sum_{j=i}^{M}\sum_{i=1}^{M}\mathcal{Q}_{ij}$. However, as the pairwise costs $\{\mathcal{Q}_{ij}\}_{i<j}$ and the unary costs $\{\mathcal{Q}_{ii}\}$ have different properties, they cannot be simply fused using the same weight. Moreover, even for the same type of costs, assigning equal weights may not be the optimal choice either, as a poor feature modality or two opposing feature modalities may introduce a larger cost of embedding or disagreement. Instead of assigning equal weights, we prefer to assign a larger penalty weight to $\mathcal{Q}_{ij}$ of higher cost, which enables us to concentrate more on minimizing $\mathcal{Q}_{ij}$ of higher cost, such that not only the overall cost can be reduced, but also the consensus can be reached by suppressing high values of individual cost $\mathcal{Q}_{ij}$. To achieve our goals, we propose the following optimization problem:

$$\begin{aligned}\min_{\{\mathbf{U}_m\}_{m=1}^{M},\,\mathbf{V}} \quad \max_{\{\alpha_{ij}\}_{j\geq i}^{M}} \quad & \sum_{j=i}^{M}\sum_{i=1}^{M}\alpha_{ij}^{\gamma}\mathcal{Q}_{ij} \\ \text{subject to} \quad & \alpha_{ij} \in \mathbb{R}^{+}, \sum_{j=i}^{M}\sum_{i=1}^{M}\alpha_{ij} = 1, \\ & \mathbf{U}_m \in \mathbb{R}^{N\times K}, \mathbf{U}_m^{\mathrm{T}}\mathbf{U}_m = \mathbf{I}, \\ & \mathbf{V} \in \mathbb{R}^{N\times K}, \mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I},\end{aligned} \tag{12}$$

where $\gamma \in [0, 1)$ is a parameter to control the distribution of weights α_{ij}^{γ}. When $\gamma = 0$, it is a special case with equal weights.

This optimization in Eq. 12 aims to achieve multi-feature fusion via minimizing the maximum weighted disagreement costs. On the one hand, maximizing the overall cost w.r.t. weight variables will highlight $\mathcal{Q}_{ij}$ of high costs, *i.e.*, large disagreement or high embedding cost. On the other hand, minimizing the overall cost w.r.t. embeddings can further reduce the highlighted costs. Moreover, it is worth noting that our objective function has only one parameter γ. Instead of manually selecting weights α_{ij} for $\mathcal{Q}_{ij}$, our objective function can optimize the fusion weights too.

3.4. Optimization

It is infeasible to simultaneously achieve the optimal values of all variables in Eq. 12, because they depend on each other. However, the objective function $\Omega = \sum_{j=i}^{M}\sum_{i=1}^{M}\alpha_{ij}^{\gamma}\mathcal{Q}_{ij}$ is concave w.r.t. each of $\{\alpha_{ij}\}_{j\geq i}^{M}$, and

Algorithm 1 MULTI-FEATURE SPECTRAL CLUSTERING WITH MINIMAX OPTIMIZATION

```
Input: data $\{\mathcal{X}_i\}_{i=1}^N$; $M$ types of features $\{\mathcal{F}^{(m)}\}_{m=1}^M$; number of
    clusters $K$; parameter $\gamma$
Output: data clustering assignment indexes $\mathcal{Y} \in \mathbb{R}^N$
    // Initialization
 1: $\alpha_{ij} \leftarrow 1 \Big/ \sum_{q=p}^{M} \sum_{p=1}^{M} 1, \forall 1 \le i, j \le M$
 2: for $i \in [1, M]$ do
 3:     $\mathcal{L}_{\text{reg},i} \leftarrow \mathcal{L}_i$                                    (Eq. 2)
 4:     $\mathbf{U}_i \leftarrow$ first $K$ smallest eigenvectors of $\mathcal{L}_{\text{reg},i}$
 5: end for
    // Main loop
 6: repeat
 7:     $\mathcal{L}_{ij} \leftarrow \mathbf{I} - \text{sym}\left(\mathbf{U}_i \mathbf{U}_i^{\text{T}} \mathbf{U}_j \mathbf{U}_j^{\text{T}}\right), \forall 1 \le i < j \le M$    (Eq. 9)
 8:     $\mathcal{L}_{\mathbf{V}} \leftarrow \sum_{j=i+1}^{M} \sum_{i=1}^{M} \alpha_{ij}^{\gamma} \mathcal{L}_{ij}$                  (Eq. 14)
 9:     $\mathbf{V} \leftarrow$ first $K$ smallest eigenvectors of $\mathcal{L}_{\mathbf{V}}$
10:     $\alpha_{ij}^{\gamma} \leftarrow \mathcal{Q}_{ij}^{\frac{\gamma}{1-\gamma}} \Big/ \left( \sum_{q=p}^{M} \sum_{p=1}^{M} \mathcal{Q}_{pq}^{\frac{1}{1-\gamma}} \right)^{\gamma}, \forall 1 \le i, j \le M$
                                                                  (Eq. 18)
11:     for $i \in [1, M]$ do
12:         $\mathcal{L}_{\text{reg},i} = \alpha_{ii}^{\gamma} \mathcal{L}_i - \sum_{j \ne i} \alpha_{ij}^{\gamma} \text{sym}\left(\mathbf{U}_j \mathbf{U}_j^{\text{T}} \mathbf{V} \mathbf{V}^{\text{T}}\right)$    (Eq. 16)
13:         $\mathbf{U}_i \leftarrow$ first $K$ smallest eigenvectors of $\mathcal{L}_{\text{reg},i}$
14:     end for
15: until $\Omega$ (Eq. 12) is converged or max iterations is reached
    // Discrete solution
16: return $\mathcal{Y} \leftarrow k$-means clustering on rows of $\mathbf{V}$
```

is convex w.r.t. each of $\{\mathbf{U}_m\}_{m=1}^M$ and $\mathbf{V}$. Since Ω is differentiable, it is easy to reach local optimum of the objective w.r.t. one variable when fixing others. Therefore, we propose to alternatively update $\mathbf{V}$, $\{\mathbf{U}_m\}_{m=1}^M$ and $\{\alpha_{ij}\}_{j \ge i}^M$. This update can converge to the local saddle with mild constraints (refer to [24] for detail). We will justify the convergence of our approach in Section 4.4 experimentally.

Initialization. We initialize each $\mathbf{U}_i$ using Laplacian embedding of the corresponding feature type, and assign equal weights to different costs.

Minimization: optimizing $\mathbf{V}$. To minimize the objective function Ω, it can be transformed to Eq. 13 using the linearity of matrix trace:

$$\Omega = \text{tr}\left(\mathbf{V}^T \mathcal{L}_{\mathbf{V}} \mathbf{V}\right) + \sum_{i=1}^{M} \alpha_{ii}^{\gamma} \mathcal{Q}_{ii}, \tag{13}$$

where

$$\mathcal{L}_{\mathbf{V}} = \sum_{j=i+1}^{M} \sum_{i=1}^{M} \alpha_{ij}^{\gamma} \mathcal{L}_{ij}, \tag{14}$$

and only the first term is related to $\mathbf{V}$. Under the orthonormal constraints of $\mathbf{V}$ (Eq. 12), $\mathbf{V}$ can be updated by performing spectral embedding of $\mathcal{L}_{\mathbf{V}}$, *i.e.*, by seeking the first K smallest eigenvectors of $\mathcal{L}_{\mathbf{V}}$.

Minimization: optimizing $\mathbf{U}_i$. To minimize the objective function Ω, it can be transformed to Eq. 15 using the linearity and cyclicity of matrix trace, where we let $\alpha_{ij} = \alpha_{ji}, \forall j < i$:

$$\Omega = \text{tr}\left(\mathbf{U}_i^T \mathcal{L}_{\text{reg},i} \mathbf{U}_i\right) + \mathcal{C}_i, \tag{15}$$

where

$$\mathcal{L}_{\text{reg},i} = \alpha_{ii}^{\gamma} \mathcal{L}_i - \sum_{j \ne i} \alpha_{ij}^{\gamma} \text{sym}\left(\mathbf{U}_j \mathbf{U}_j^{\text{T}} \mathbf{V} \mathbf{V}^{\text{T}}\right), \tag{16}$$

and

$$\mathcal{C}_i = \sum_{j \ge h, j \ne i} \sum_{h \ne i} \alpha_{hj}^{\gamma} \mathcal{Q}_{hj} + K \sum_{j \ne i} \alpha_{ij}^{\gamma}. \tag{17}$$

Since $\mathcal{C}_i$ is not related to $\mathbf{U}_i$, under the orthonormal constraints of $\mathbf{U}_i$ (Eq. 12), $\mathbf{U}_i$ can be updated by performing spectral embedding of $\mathcal{L}_{\text{reg},i}$, *i.e.*, by seeking the first K smallest eigenvectors.

Maximization: optimizing α_{ij}. It becomes a maximization problem w.r.t. α_{ij}. Applying the Lagrange multiplier method, we can obtain the closed-form of α_{ij}^{γ} as

$$\alpha_{ij}^{\gamma} = \frac{\mathcal{Q}_{ij}^{\frac{\gamma}{1-\gamma}}}{\left(\sum_{q=p}^{M} \sum_{p=1}^{M} \mathcal{Q}_{pq}^{\frac{1}{1-\gamma}} \right)^{\gamma}}. \tag{18}$$

Because $\gamma \in [0, 1)$, Eq. 18 shows that larger costs will be assigned with larger weights. As a result, larger disagreements will be suppressed across heterogeneous features in the process of total cost minimization. Further analyzing Eq. 18, we can see that, when $\gamma \to 0$, different weights will come close to each other; when $\gamma \to 1$, the weight of the largest cost will tend to be 1, while other weights will approach 0; $0 < \gamma < 1$ achieves a trade-off weighting. We will also discuss the influence of parameter γ in the experiments.

We show our complete solution in Algorithm 1. As can be seen, the computational complexity within each iteration of our method mainly relies on $M+1$ times of eigendecomposition (Lines 9 and 13), which can be efficiently solved by state-of-the-art eigensolvers.

4. Experiments

4.1. Datasets and Experimental Setting

To evaluate our multi-feature clustering, we conduct experiments on three image datasets: UCI Digits [2], Oxford Flowers [21], UC Merced Land Uses [38], and a video dataset: Body Motions [26].

UCI Digits. We integrate multiple feature types for handwritten digit recognition on the UCI Digit Dataset, which consists of features of handwritten numerals ('0'–'9') extracted from $2,000$ Dutch utility maps [2]. Each

Feature Type	Body Motions		Oxford Flowers		UC Merced Land Uses		UCI Digits	
	Feature	Dimension	Feature	Dimension	Feature	Dimension	Feature	Dimension
1	HOG	4000	Color	500	LLC 1×1	1024	FOU	76
2	MBH	4000	Shape	1000	LLC 2×2	4096	FAC	216
3	–	–	Texture	700	LLC 4×4	16384	KAR	64
4	–	–	–	–	pHOG	680	PIX	240
5	–	–	–	–	GIST	512	ZER	47
6	–	–	–	–	Color Histograms	784	MOR	6

Table 1. The image/video datasets used in our experiment and their feature descriptors.

digit category contains 200 samples. These digits are represented by six types of features: (1) Fourier coefficients of the character shapes (FOU); (2) profile correlations (FAC); (3) Karhunen-Loeve coefficients (KAR); (3) pixel averages in 2×3 windows (PIX); (5) Zernike moments (ZER); and (6) morphological features (MOR).

Oxford Flowers. The Oxford Flower Dataset is composed of 17 flower categories, with 80 images for each category [21]. Each image is described by different visual features using color, shape, and texture.

UC Merced Land Uses. The UC Merced Land Use dataset [38] contains 21 classes of aerial orthoimagery, with 100 images each category. For local visual features, we represent each image as three pools of LLCs (locality-constrained linear codes) over dense SIFTs with 1×1, 2×2, and 4×4 partitions [34]. For global visual features, we extract pHOG [5], GIST [22] and Color Histograms [23].

Body Motions. Appearance and motion features complement each other for body motion description and recognition in video data. Therefore, we combine such two feature types for action clustering. The human body motion dataset, which is introduced by UCF101 [26], contains 16 categories of human body actions and 1910 videos in total. For appearance features, each video is described by dense appearance trajectories based on Histogram of Oriented Gradients (HOG); while for motion features, each video is represented as dense motion trajectories based on Motion Boundary Histograms (MBH) [29].

We summarize the feature descriptors in Table. 1 for the four datasets, including feature type IDs and feature dimensions. In all experiments, we compute pairwise image similarities using Gaussian kernel (Eq. 1). The bandwidth parameter σ is equal to the median of the pair-wise χ distances (for Oxford Flowers [21])[1] or Euclidean distances (for UCI Digits [2], UC Merced Land Uses [38] and Body Motions [26]) of corresponding feature descriptors.

Except for the parameter sensitivity experiment, we fix the parameter γ to 0.33 for al experiments. As adopted in [15, 6, 12], we evaluate clustering performance using two standard measures: Clustering accuracy and normalized mutual information (NMI) from 10 random runs. Generally, the higher the measures, the better the performance.

4.2. Baseline Algorithms

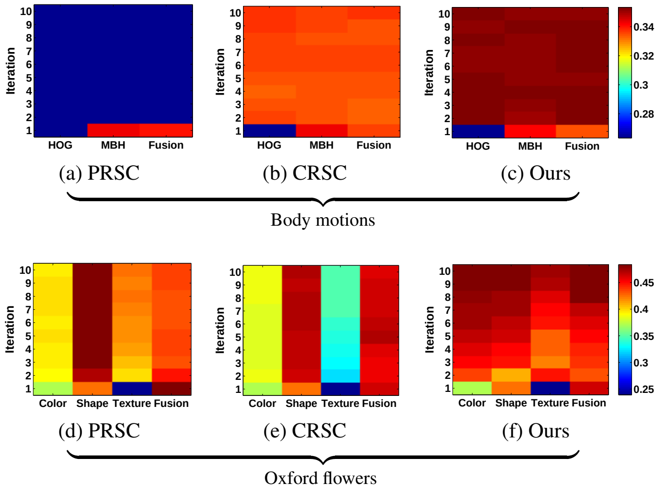
To validate the performance of the proposed multi-feature clustering approach, we compare it with various baselines:

- **Single Feature Type Spectral Clustering** (SC(#)): running spectral clustering [20] with graph Laplacian derived from a single feature type.
- **Kernel Averaging Spectral Clustering** (KASC): averaging normalized kernel matrix derived from individual feature types, followed by applying spectral clustering [20] with corresponding Laplacian. The kernel normalization is obtained by $(ker)^{\frac{1}{dim}}$, where ker denotes a feature kernel matrix, and dim denotes the feature dimension.
- **Centroid Co-regularized Spectral Clustering** (CRSC): pushing all spectral embeddings of different feature types close to a centroid embedding using data-data similarity matrices (Eq. 4) [15], followed by k-means with the centroid embedding. We set the parameters in this algorithm to be 0.01 as suggested.
- **Pairwise Co-regularized Spectral Clustering** (PRSC): pushing pairwise spectral embeddings of different feature types close to each other using data-data similarity matrices (Eq. 4) [15], followed by k-means clustering with embedding concatenation. We set the parameter in this algorithm to be 0.01 as suggested .
- **Multi-Modal Spectral Clustering** (MMSC): learning a shared graph Laplacian from different feature types [6], followed by NMF-based spectral clustering [9]. We report the best results by tuning the parameter of this algorithm in the range from 10^{-2} to 10^{2} with incremental step $10^{0.2}$ as suggested.
- **Affinity Aggregation Spectral Clustering** (AASC): aggregating affinities of different feature types with opti-

[1] χ^2 distances for different flower features are provided online: http://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html

Method	Body Motions		Oxford Flowers		UC Merced Land Uses		UCI digits	
	Accuracy	NMI	Accuracy	NMI	Accuracy	NMI	Accuracy	NMI
SC (1)	0.273 ± 0.009	0.263 ± 0.005	0.343 ± 0.013	0.371 ± 0.009	0.381 ± 0.011	0.459 ± 0.012	0.679 ± 0.044	0.649 ± 0.015
SC (2)	0.312 ± 0.010	0.342 ± 0.004	0.404 ± 0.016	0.425 ± 0.007	0.364 ± 0.020	0.440 ± 0.017	0.631 ± 0.048	0.622 ± 0.016
SC (3)	–	–	0.257 ± 0.010	0.239 ± 0.009	0.387 ± 0.010	0.449 ± 0.006	0.692 ± 0.087	0.652 ± 0.042
SC (4)	–	–	–	–	0.093 ± 0.004	0.242 ± 0.003	0.710 ± 0.053	0.660 ± 0.027
SC (5)	–	–	–	–	0.336 ± 0.007	0.397 ± 0.005	0.569 ± 0.021	0.500 ± 0.009
SC (6)	–	–	–	–	0.234 ± 0.006	0.269 ± 0.006	0.420 ± 0.028	0.469 ± 0.008
KASC	0.301 ± 0.013	0.328 ± 0.009	0.370 ± 0.012	0.403 ± 0.009	0.294 ± 0.010	0.349 ± 0.009	0.709 ± 0.042	0.668 ± 0.016
PRSC [15]	0.275 ± 0.007	0.263 ± 0.005	0.419 ± 0.013	0.435 ± 0.008	0.368 ± 0.022	0.447 ± 0.010	0.769 ± 0.049	0.728 ± 0.020
CRSC [15]	0.317 ± 0.020	0.335 ± 0.012	0.449 ± 0.019	0.461 ± 0.009	0.395 ± 0.011	0.468 ± 0.006	0.770 ± 0.036	0.713 ± 0.011
MMSC [6]	0.266 ± 0.008	0.305 ± 0.010	0.416 ± 0.015	0.427 ± 0.012	0.123 ± 0.012	0.265 ± 0.007	0.731 ± 0.011	0.675 ± 0.008
AASC [12]	0.250 ± 0.011	0.292 ± 0.008	0.410 ± 0.028	0.422 ± 0.014	0.226 ± 0.009	0.291 ± 0.007	0.683 ± 0.047	0.649 ± 0.018
Ours	**0.322 ± 0.015**	**0.352 ± 0.010**	**0.493 ± 0.039**	**0.484 ± 0.022**	**0.404 ± 0.021**	**0.482 ± 0.015**	**0.800 ± 0.102**	**0.785 ± 0.049**

Table 2. Comparisons of various baselines with the proposed approach.



(a) PRSC (b) CRSC (c) Ours

Body motions

(d) PRSC (e) CRSC (f) Ours

Oxford flowers

Figure 1. Iteration comparisons of PRSC, CRSC and our method w.r.t. NMI performance of each feature embedding and the fusion result on the body motion and Oxford flower datasets. Best viewed in color.

mized weights [12], followed by applying normalized cut [25] with corresponding Laplacian.

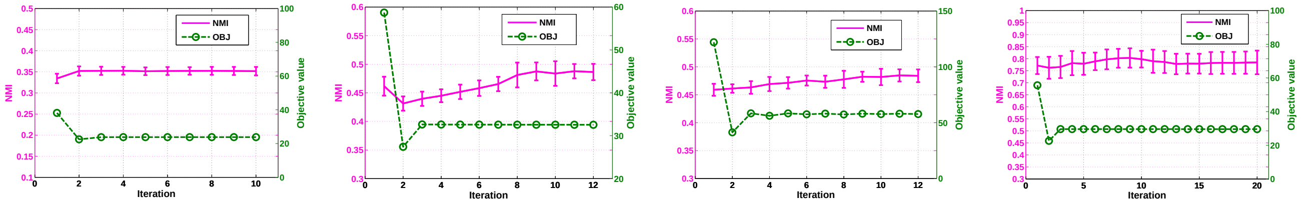
4.3. Experimental Results

We compare our method with baseline methods in Table 2. For the experiment of Body Motion clustering, there are only two types of features: motion features (MBH) and appearance features (HOG). Spectral clustering results show that motion features perform better than appearance features in human body motion clustering. Although most compared algorithms perform better than spectral clustering on the poorer feature type, they are difficult to beat spectral clustering on the better feature type. For example, only CRSC performs better than the better feature type. In contrast, thanks to our new formulation for multi-feature fusion in Eq. 12, our approach can enhance the poorer feature embedding with iterations (Algorithm 1), thus generating a good clustering result. As can be seen in Table 2, our approach is not only better than the compared algorithms, but also can compete against the result of the better feature type.

Similarly, our approach also outperforms all compared methods on the other benchmark datasets, which further verifies the effectiveness of the proposed multi-feature clustering. Regarding flower clustering, all the compared methods perform better than spectral clustering on a single feature type. However, our approach can achieve much higher accuracy and NMI values than the other methods. For scene clustering, spectral clustering performs poorly on pHOG and color histograms. Despite such poor features, our approach still achieves a noticeable performance gain thanks to heterogeneous feature fusion. In handwritten digit grouping, since each single feature type reaches a good spectral clustering, the compared multi-feature clustering methods all obtain commendable results. Again, our approach achieves the most significant improvement than other compared methods. All the results obtained with our approach can benefit from effectively unveiling and fusing complementary information from heterogeneous features by optimizing Eq. 12.

To further illustrate the advantage of our approach, we study how different regularization methods influence the performance of each feature embedding, as well as the performance of the fusion result. As shown in Fig. 1, we compare the most related work PRSC and CRSC with our approach on the body motion and Oxford flower datasets.

PRSC adopts pairwise regularization among different modality-specific Laplacian embeddings using data-data similarity matrices. Form Fig. 1 (a) and (d), we can see that, PRSC is sensitive to the poorer feature types, *e.g.*, HOG features in body motion dataset and color/texture features in Oxford flower dataset. In the initialization, *i.e.*, the first iteration, the fusion result approaches/exceeds the performance of the best feature type. However, with more iterations, the regularization may lead to a worse result. In such a case, it is not a good choice for multi-feature clustering. Similarly, CRSC also leverages data-data similarity matrices to perform regularization. But it aims to force each modality-specific Laplacian embedding towards a consen-



(a) Body Motions (b) Oxford Flowers (c) UC Merced Land Uses (d) UCI Digits

Figure 2. Convergence study of our algorithm: NMI and objective values with iterations.

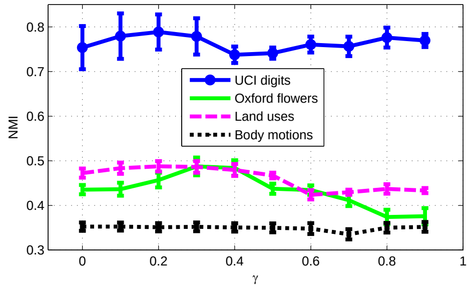


Figure 3. Parameter study of our algorithm: NMI values with different parameters.

Feature Order	Accuracy	NMI
1 2 3	0.493 ± 0.039	0.484 ± 0.022
3 2 1	0.471 ± 0.018	0.474 ± 0.012
3 1 2	0.490 ± 0.029	0.489 ± 0.017
2 3 1	0.458 ± 0.026	0.464 ± 0.015
2 1 3	0.458 ± 0.017	0.471 ± 0.009
1 3 2	0.491 ± 0.029	0.484 ± 0.019

Table 3. Performance of flower clustering with different feature input orders using our algorithm.

sus embedding. From Fig. 1 (b) and (e), we can see that, CRSC is not very sensitive to poor feature types. However, it cannot effectively enhance the poor feature types. As shown in Fig. 1 (e), successive regularization does not improve the bad performances of color and texture features. In such a case, CRSC is unable to bring different modality-specific Laplacian embeddings close enough, which finally influences the consensus result. On the contrary, we do not directly use data-data similarity matrices for regularization. By projecting each data-data similarity matrix onto the universal feature embedding, we relax the data-data similarity matrix from each feature type to obtain the regularized data-cluster similarity matrix (using Eq. 5) for our regularization framework. As shown in Fig. 1 (c) and (f), although our method may not perform better than the best individual feature initially (Fig. 1 (c)), the regularization can gradually reduce the disagreements among different feature embeddings to refine the performance, and finally enhance the fusion result.

4.4. Convergence Analysis

To evaluate the convergence of the proposed solution in Algorithm 1, Fig. 2 shows the objective function value together with its NMI performance indicator over iterations, with $\gamma = 0.33$. The dashed line shows the objective values, while the solid line shows the NMI values. As can be seen, the objective function first moves down then upwards and flattens. After less than 20 iterations, the algorithm will converge to a saddle, which meets minimax optimization of the objective function in Eq. 12. Besides, it is interesting to notice that, although the objective function value has almost no change after several iterations, the performance can still benefit from the min-max iteration, *e.g.*, the results shown in Fig. 2 (b) and (e). This further verifies the effectiveness of our proposed algorithm. When testing other parameter values, we observe that, $\gamma < 0.5$ can generally generate a converged solution. When $\gamma \geq 0.5$, we choose to stop the iteration in less than 20 iterations and output the final result.

4.5. Sensitivity of Parameters

In our objective function in Eq. 12, we have only one parameter γ to control the weights. As another factor, our proposed method relies on the input order of different features as shown in lines 10-13 of Algorithm 1. The sensitivity of the related factors will be studied in this section.

Parameter γ. As formulated in the objective function of Eq. 12, our proposed approach only has one parameter γ to balance different cost terms. γ is in the interval $[0, 1)$. We plot NMI performance curve w.r.t. parameter γ in Fig. 3. As can be seen, different values of γ do not influence much for appearance and motion feature fusion on the body motion dataset. The reason may be that these two types of features can regularize each other, and the weight assignments are not so critical in such a case. For each of the other datasets, the result are not very sensitive to γ.

Feature Order. To study how the input order of feature types influences clustering performance, we test our algorithm using different feature input orders on the Oxford flower dataset. We enumerate all six possible permutations of feature types, as shown in the first column of Table 3. The corresponding feature IDs are given in Table 1. All the results (in Table 3) perform better than the baseline methods (as shown in Table 2), and the result is not sensitive to the feature input order.

5. Conclusion

Multi-feature clustering is a challenging problem as it is difficult to find a clustering result agreeable to all feature modalities. To find the consensus, we explore an loss function consisting of both the unary term based on the cost of the Laplacian embedding of each individual feature modality and the pairwise disagreement term between any pair of feature modalities. To optimize the objective function, we propose a minimax formulation by minimizing the maximum loss. Our multi-feature clustering approach has only one parameter and does not need to specify the fusion weights. Our multi-feature clustering results on four image and video datasets show superior performance when compared with the state-of-the-art methods.

Acknowledgment

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

References

[1] Z. Akata, C. Thurau, C. Bauckhage, et al. Non-negative matrix factorization in multimodality data for segmentation and label prediction. In *CVWW*, 2011. 2

[2] K. Bache and M. Lichman. UCI machine learning repository, 2013. 4, 5

[3] S. Bickel and T. Scheffer. Multi-view clustering. In *ICDM*, volume 4, pages 19–26, 2004. 2

[4] M. Blaschko and C. Lampert. Correlational spectral clustering. In *CVPR*, 2008. 1, 2

[5] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *CIVR*, pages 401–408, 2007. 5

[6] X. Cai, F. Nie, H. Huang, and F. Kamangar. Heterogeneous image feature integration via multi-modal spectral clustering. In *CVPR*, pages 1977–1984, 2011. 1, 2, 5, 6

[7] K. Chaudhuri, S. M. Kakade, K. Livescu, and K. Sridharan. Multi-view clustering via canonical correlation analysis. In *ICML*, pages 129–136, 2009. 2

[8] V. R. de Sa, P. W. Gallagher, J. M. Lewis, and V. L. Malave. Multi-view kernel construction. *Mach. Learn.*, 79(1-2):47–71, 2010. 2

[9] C. Ding, T. Li, and M. I. Jordan. Nonnegative matrix factorization for combinatorial optimization: Spectral clustering, graph matching, and clique finding. In *ICDM*, pages 183–192, 2008. 5

[10] X. Guo, D. Liu, B. Jou, M. Zhu, A. Cai, and S.-F. Chang. Robust object co-detection. In *CVPR*, 2013. 2

[11] Y. Guo. Convex subspace representation learning from multi-view data. In *AAAI*, 2013. 1, 2

[12] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen. Affinity aggregation for spectral clustering. In *CVPR*, pages 773–780, 2012. 1, 2, 5, 6

[13] Y. Jiang, J. Liu, Z. Li, P. Li, and H. Lu. Co-regularized plsa for multi-view clustering. In *ACCV*, pages 202–213, 2012. 2

[14] A. Kumar and H. D. III. A co-training approach for multi-view spectral clustering. In *ICML*, pages 393–400, 2011. 2

[15] A. Kumar, P. Rai, and H. D. III. Co-regularized multi-view spectral clustering. In *NIPS*, pages 1413–1421, 2011. 1, 2, 3, 5, 6

[16] T. Lange and J. M. Buhmann. Fusion of similarity data in clustering. In *NIPS*, 2005. 2

[17] J. Liu, C. Wang, J. Gao, and J. Han. Multi-view clustering via joint nonnegative matrix factorization. In *SDM*, 2013. 2

[18] B. Long, S. Y. Philip, and Z. M. Zhang. A general model for multiple view unsupervised learning. In *SDM*, pages 822–833, 2008. 2

[19] H. Lütkepohl. *Handbook of matrices*. John Wiley & Sons, 1996. 2

[20] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. In *NIPS*, volume 2, pages 849–856, 2001. 2, 5

[21] M. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *CVPR*, volume 2, pages 1447–1454, 2006. 4, 5

[22] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001. 5

[23] F. Palermo, J. Hays, and A. A. Efros. Dating historical color images. In *ECCV*, pages 499–512, 2012. 5

[24] R. T. Rockafellar. Saddle-points and convex analysis. In *Differential Games and Related Topics*, pages 109–128. North-Holland, 1971. 4

[25] J. Shi and J. Malik. Normalized cuts and image segmentation. *TPAMI*, 22(8):888–905, 2000. 2, 6

[26] 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. 4, 5

[27] G. F. Tzortzis and C. Likas. Multiple view clustering using a weighted combination of exemplar-based mixture models. *TNN*, 21(12):1925–1938, 2010. 2

[28] B. Wang, J. Jiang, W. Wang, Z.-H. Zhou, and Z. Tu. Unsupervised metric fusion by cross diffusion. In *CVPR*, pages 2997–3004, 2012. 1

[29] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *CVPR*, pages 3169–3176, 2011. 5

[30] H. Wang, F. Nie, and H. Huang. Multi-view clustering and feature learning via structured sparsity. In *ICML*, 2013. 2

[31] H. Wang, F. Nie, H. Huang, and C. Ding. Heterogeneous visual features fusion via sparse multimodal machine. In *CVPR*, 2013. 1

[32] H. Wang, J. Yuan, and Y. Tan. Combining feature context and spatial context for image pattern discovery. In *ICDM*, pages 764–773, 2011. 2

[33] H. Wang, J. Yuan, and Y. Wu. Context-aware discovery of visual co-occurrence patterns. *TIP*, 23(4):1805–1819, 2014. 2

[34] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, 2010. 5

[35] X. Wang, B. Qian, J. Ye, and I. Davidson. Multi-objective multi-view spectral clustering via pareto optimization. In *SDM*, 2013. 2

[36] T. Xia, D. Tao, T. Mei, and Y. Zhang. Multiview spectral embedding. *TSMC-B*, 40(6):1438–1446, 2010. 2

[37] J. Yang, Z. Wang, Z. Lin, X. Shu, and T. Huang. Bilevel sparse coding for coupled feature spaces. In *CVPR*, pages 2360–2367, 2012. 2

[38] Y. Yang and S. Newsam. Spatial pyramid co-occurrence for image classification. In *ICCV*, 2011. 4, 5

[39] S. Yu, L.-C. Tranchevent, X. Liu, W. Glanzel, J. A. Suykens, B. De Moor, and Y. Moreau. Optimized data fusion for kernel k-means clustering. *TPAMI*, 34(5):1031–1039, 2012. 1, 2

[40] J. Yuan and Y. Wu. Context-aware clustering. In *CVPR*, 2008. 2

[41] D. Zhou and C. J. Burges. Spectral clustering and transductive learning with multiple views. In *ICML*, pages 1159–1166, 2007. 2