The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Reliable Multi-View Clustering

Hong Tao,[1] Chenping Hou,[1] Xinwang Liu,[2] Dongyun Yi,[1] Jubo Zhu[1]
[1]College of Science, National University of Defense Technology, Changsha, 410073, Hunan, China.
[2]School of Computer, National University of Defense Technology, Changsha, 410073, Hunan, China.
{taohong.nudt, hcpnudt}@hotmail.com, xinwangliu@nudt.edu.cn, dongyun.yi@gmail.com, ju_bo_zhu@aliyun.com

Abstract

With the advent of multi-view data, multi-view learning (MVL) has become an important research direction in machine learning. It is usually expected that multi-view algorithms can obtain better performance than that of merely using a single view. However, previous researches have pointed out that sometimes the utilization of multiple views may even deteriorate the performance. This will be a stumbling block for the practical use of MVL in real applications, especially for tasks requiring high dependability. Thus, it is eager to design *reliable* multi-view approaches, such that their performance is never degenerated by exploiting multiple views. This issue is vital but rarely studied. In this paper, we focus on clustering and propose the Reliable Multi-View Clustering (RMVC) method. Based on several candidate multi-view clusterings, RMVC maximizes the worst-case performance gain against the best single view clustering, which is equivalently expressed as no label information available. Specifically, employing the squared χ^2 distance for clustering comparison makes the formulation of RMVC easy to solve, and an efficient strategy is proposed for optimization. Theoretically, it can be proved that the performance of RMVC will never be significantly decreased under some assumption. Experimental results on a number of data sets demonstrate that the proposed method can effectively improve the *reliability* of multi-view clustering.

Introduction

In many real-world applications, such as image retrieval and cross language text classification, the same object can be represented by multiple different features (Xu, Tao, and Xu 2013). For example, an image can be characterized by different descriptors and news can be translated into various languages. This kind of data is known as multi-view data, and each feature representation corresponds to a view (Hou et al. 2010; Sun 2011). Multi-view learning (MVL), which aims to improve the learning performance by exploiting the information from different views, has become an important research direction. Unlike single-view algorithms that concatenate all views into a big one to meet the setting, MVL designs advanced methods of combining multiple views to achieve the performance improvement (Xu, Tao, and Xu 2013). For instance, in (Tao et al. 2017a; Nie, Cai, and Li 2017), weights are allocated to different views automatically and promising results are obtained.

It is generally recognized that the performance of MVL algorithms will be improved by utilizing more views. However, the experimental results of some researches (Bickel and Scheffer 2004; Yang et al. 2013; Zhang et al. 2015) show that, sometimes the use of multiple views may degenerate the performance. This problem is also reflected in our experiments. As shown in Table 1, compared with the best single-view normalized cut (Ncut) (Shi and Malik 2000) method, the previous proposed multi-view clustering approaches (Kumar, Rai, and Daume 2011; Kumar and Daumé 2011; Cai et al. 2011; Cai, Nie, and Huang 2013; Li et al. 2015) obtain lower clustering accuracy on at least two data sets. Such phenomena are contrary to the original intention of MVL and will hinder the practical use of MVL in real applications, especially tasks requiring high dependability. Thus, it is vital to have *reliable* MVL approaches, whose performance is *never statistically significantly worse than* that of merely using a single view.

Though there are already many studies on MVL, little work has been done explicitly about its *reliability*. In this paper, we focus on clustering and propose the Reliable Multi-View Clustering (RMVC) method. To the best of our knowledge, our work is the first to directly study on the reliability of MVL. More concretely, our goal is to utilizing multiple views to obtain a clustering which is not worse than that of using any single view. Specifically, the final clustering is produced by maximizing the performance gain in the worst case based on several candidate multi-view clusterings. To obtain the objective function, there are two key issues to be solved. On one hand, the widely used clustering performance measures (e.g., clustering accuracy and normalized mutual information (NMI)), are typically discontinuous and hard to analyze. Using these measures directly causes trouble in optimization. Instead, the squared χ^2 distance between partitions (Meilă 2012), which is quadratic and can be analyzed conveniently, is employed for comparing clusterings. On the other hand, when there is no ground truth, it is hard to find out the best clustering in the original single-view outputs. Based on the candidate multi-view clusterings, a equivalent clustering is defined and can be solved with a bound linear least squares optimization. The optimization of RMVC can be decomposed into two subproblems,

i.e., a small-scale convex linearly constrained quadratic programming and a nonnegative orthogonal matrix factorization. Both subproblems can be solved efficiently. Theoretically, RMVC is provably reliable when one of the candidate multi-view clustering algorithms realizes the ground-truth clustering. Experimental results on a number of multi-view data sets validate that RMVC improves the reliability of multi-view clustering.

In the following, we first introduce some notations and definitions. Next, the proposed RMVC method and its theoretical analysis are presented. Then, experimental results follows. Finally we conclude this paper.

Notations and Definitions

Let $\mathcal{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]$ denote a set of points. A clustering is a partition of the n points into disjoint and nonempty subsets $\{\mathcal{C}_l, \mathcal{C}_2 \cdots, \mathcal{C}_K\}$, which are called *clusters*. Denote $|\mathcal{C}_k|$ as n_k, then $\sum_{k=1}^{K} n_k = n$. A clustering can be represented as a $n \times K$ normalized cluster indicator matrix $\mathbf{Y}$ with $\mathbf{Y}_{ik} = n_k^{-1/2}$ if $i \in \mathcal{C}_k$ and 0 otherwise. The columns of $\mathbf{Y}$ stand for the indicator vectors of the K clusters, and they are mutually orthogonal. In the future, we will refer to a clustering by its matrix representation.

For two different clusterings (of the same set of points) with normalized cluster indicator matrices $\mathbf{Y}_1 \in \mathbb{R}^{n \times K_1}$ and $\mathbf{Y}_2 \in \mathbb{R}^{n \times K_2}$ (K_1 and K_2 may be not equal), the χ^2 distance between them is defined as follows.

Definition 1. *(Meilă 2012) Let $\|\cdot\|_F$ represent the Frobenius norm and the superscript $(\cdot)^T$ denotes matrix transposition. Then the χ^2 distance between $\mathbf{Y}_1$ and $\mathbf{Y}_2$ is defined as*

$$d^2_{\chi^2}(\mathbf{Y}_1, \mathbf{Y}_2) = \|\mathbf{Y}_1\mathbf{Y}_1^T - \mathbf{Y}_2\mathbf{Y}_2^T\|_F^2. \quad (1)$$

It can be found that $d^2_{\chi^2}$ is a quadratic function, making it a convenient instrument in deductions. It has been proved that $d^2_{\chi^2}$ distance is equivalent to the Misclassification Error distance (d_{ME}), which corresponds to the well-known clustering accuracy. Compared with $d^2_{\chi^2}$, d_{ME} is not everywhere differentiable and is theoretically much harder to analyze (Meilă 2012).

The Proposed Method

In this section we first present the problem setting of RMVC and its formulation, and then give the solution.

Problem Setting and Formulation

Suppose we are given a data set with V views. On each view, a single-view clustering algorithm (e.g. Normalized cut (Shi and Malik 2000)) is performed, thereby producing V single-view clustering results $\{\mathbf{Y}_0^{(v)}\}_{v=1}^V$. On the other hand, we run m multi-view clustering algorithms to obtain m multi-view clusterings $\{\mathbf{Y}_1, \cdots, \mathbf{Y}_m\}$. Besides, the m clusterings can also be obtained by implementing a multi-view clustering algorithm with different parameters, or a hybrid of the above two. Based on the given m multi-view clusterings, our task is to find a clustering $\mathbf{Y}$ that is not worse than $\mathbf{Y}_0^{(v)}, \forall 1 \le v \le V$.

Unlike classification or regression, the label vector or normalized cluster indicator matrix of each clustering is not unique. In fact, for each unique clustering with K clusters, there are $K!$ equivalent representations. Thus, it is unrealistic to directly use the Euclidean distance between different $\mathbf{Y}$s to measure the clustering performance (Meilă 2012). Moreover, the commonly used clustering evaluation metrics, e.g., clustering accuracy and NMI, are typically non-differentiable and non-convex. Using them for performance measure will make the resultant formulation difficult to solve. Instead, the squared χ^2 distance with appealing properties, is employed for comparing clusterings. The smaller the squared χ^2 distance between a clustering and the ground truth is, the better the clustering is. Concretely, the performance gain of $\mathbf{Y}$ against the best single-view results in $\{\mathbf{Y}_0^{(v)}\}_{v=1}^V$ is measured by the difference of squared χ^2 distance between clusterings, i.e.,

$$\max_{\mathbf{Y} \in \mathcal{Y}} \big(\min_{1 \le v \le V} d^2_{\chi^2}(\mathbf{Y}_0^{(v)}, \mathbf{Y}^*) - d^2_{\chi^2}(\mathbf{Y}, \mathbf{Y}^*) \big), \quad (2)$$

where $\mathbf{Y}^*$ refers to the ground-truth clustering, and $\mathcal{Y}$[1] is the feasible region of $\mathbf{Y}$.

To solve Eq. (2), the difficulty lies in the fact that the ground-truth $\mathbf{Y}^*$ is unknown. Otherwise, it is trivial to get the solution $\mathbf{Y} = \mathbf{Y}^*$. To alleviate this challenge, $\boldsymbol{\alpha} = [\alpha_1; \alpha_2; \cdots; \alpha_m] \ge \mathbf{0}$ is assumed to be the weights of multi-view clusterings $\{\mathbf{Y}_1, \cdots, \mathbf{Y}_m\}$. The larger the weight is, the closer the clustering is to the ground-truth $\mathbf{Y}^*$. Using these candidate multi-view clusterings to approximate the ground truth, one optimizes the following functional instead:

$$\max_{\mathbf{Y} \in \mathcal{Y}} \sum_{i=1}^{m} \alpha_i \big(\min_{1 \le v \le V} d^2_{\chi^2}(\mathbf{Y}_0^{(v)}, \mathbf{Y}_i) - d^2_{\chi^2}(\mathbf{Y}, \mathbf{Y}_i) \big). \quad (3)$$

Note that the minimization operation increases the difficulty of the problem, because it may return different single-view $\mathbf{Y}_0^{(v)}$ for different candidate multi-view clustering $\mathbf{Y}_i$. To solve this problem, we define $\mathbf{Y}_0$ such that $\forall 1 \le i \le m$

$$d^2_{\chi^2}(\mathbf{Y}_0, \mathbf{Y}_i) = \min_{1 \le v \le V} d^2_{\chi^2}(\mathbf{Y}_0^{(v)}, \mathbf{Y}_i). \quad (4)$$

It can be found that $\mathbf{Y}_0$ is a equivalent expression of the best single-view clustering.

As a matter of fact, when there is no prior knowledge on the candidate multi-view clustering algorithms, it is difficult to know their weights (i.e., $\boldsymbol{\alpha}$) explicitly. To make the proposal more practical, $\boldsymbol{\alpha}$ is assumed to be from a simplex $\mathcal{M} = \{\boldsymbol{\alpha} | \mathbf{1}^T\boldsymbol{\alpha} = 1; \boldsymbol{\alpha} \ge \mathbf{0}\}$. Since there is no way to determine the relative importance of multiple multi-view clustering algorithms, we aim to optimize the worst-case performance gain as follows,

$$\max_{\mathbf{Y} \in \mathcal{Y}} \min_{\boldsymbol{\alpha} \in \mathcal{M}} \sum_{i=1}^{m} \alpha_i \big(d^2_{\chi^2}(\mathbf{Y}_0, \mathbf{Y}_i) - d^2_{\chi^2}(\mathbf{Y}, \mathbf{Y}_i) \big). \quad (5)$$

[1] $\mathcal{Y}$ is the set of possible normalized cluster indicator matrices.

Relation to Other Approaches

Reliability is a common concern of many disciplines of machine learning. In semi-supervised learning, where exploiting the unlabeled data does not necessarily bring positive effect, there are also researches working on this topic (Li and Zhou 2011; 2015; Li, Kwok, and Zhou 2016; Li, Zha, and Zhou 2017). Specifically, based on several low-density separators, (Li and Zhou 2011) constructed safe semi-supervised SVMs (S4VM). Later, SAFE semi-supervised Regression (SAFER) was considered as a geometric projection issue (Li, Zha, and Zhou 2017).

Although S4VM and SAFER also study on the reliability, our proposed RMVC is different from them in two aspects. Firstly, their tasks are different. S4VM and SAFER are designed for classification and regression, respectively, while RMVC focus on clustering. Unlike classification and regression, the label vector of a clustering is not unique, thus the comparison between different clusterings cannot be made directly on the label vectors. Secondly, S4VM and SAFER are single-view methods while RMVC is in the scope of MVL, which leads to the differences in the number of baseline results. Both S4VM and SAFER only need to handle one baseline result, while RMVC is required to find out the best one from a set of single-view results. This add difficulty to RMVC, because there is no ground-truth clustering available.

Another related work is clustering ensembles (Zhou 2012; Xie and Sun 2013; Tao et al. 2017b), which aim to obtain a better final partition by combining different component clusterings. With totally different motivation, our primary goal is to make the utilization of multiple views "reliable", whereas this has not been considered in previous ensemble clustering approaches.

Solution

Substituting the χ^2 distance between clusterings defined in Eq. (1) into Eq. (5), the optimization problem becomes into

$$\max_{\mathbf{Y}\in\mathcal{Y}} \min_{\boldsymbol{\alpha}\in\mathcal{M}} \sum_{i=1}^{m} \alpha_i \big(\|\mathbf{Y}_0\mathbf{Y}_0^T - \mathbf{Y}_i\mathbf{Y}_i^T\|_F^2 - \|\mathbf{Y}\mathbf{Y}^T - \mathbf{Y}_i\mathbf{Y}_i^T\|_F^2 \big). \tag{6}$$

It can be found that Eq. (6) is quartic with respect to $\mathbf{Y}$. Thus, it is hard to optimize for $\mathbf{Y}$ directly in the max-min problem and we employ a two-step strategy. That is to say, we first take $\mathbf{A} = \mathbf{Y}\mathbf{Y}^T$ as a variable and then decompose $\mathbf{A}$ into the product of $\mathbf{Y}$ and its transpose. Specifically, the optimization of Eq. (6) is decomposed into two subproblems:

$$\max_{\mathbf{A}=\mathbf{A}^T} \min_{\boldsymbol{\alpha}\in\mathcal{M}} \sum_{i=1}^{m} \alpha_i \big(\|\mathbf{A}_0 - \mathbf{A}_i\|_F^2 - \|\mathbf{A} - \mathbf{A}_i\|_F^2 \big), \tag{7}$$

$$\min_{\mathbf{Y}\in\mathcal{Y}} \|\mathbf{Y}\mathbf{Y}^T - \hat{\mathbf{A}}\|_F^2, \tag{8}$$

where $\mathbf{A}_0 = \mathbf{Y}_0\mathbf{Y}_0^T$ and $\mathbf{A}_i = \mathbf{Y}_i\mathbf{Y}_i^T$ correspondingly, and $\hat{\mathbf{A}}$ denotes the optimal solution of Eq. (7).

As seen from the above deductions, if $\mathbf{A}_0$ is obtained, the exact form of $\mathbf{Y}_0$ is not necessary for solving Eq. (7). In the following, we first calculate $\mathbf{A}_0$ and then solving subproblems in Eq. (7) and Eq. (8).

Calculating $\mathbf{A}_0$ According to Eq. (4), we have $\forall 1 \le i \le m$,

$$\begin{aligned} & \|\mathbf{A}_0\|_F^2 - 2Tr(\mathbf{A}_i^T\mathbf{A}_0) + \|\mathbf{A}_0^{(v)}\|_F^2 \\ = & \min_{1\le v\le V} \{\|\mathbf{A}_0^{(v)}\|_F^2 - 2Tr(\mathbf{A}_i^T\mathbf{A}_0^{(v)})\} + \|\mathbf{A}_0^{(v)}\|_F^2, \end{aligned} \tag{9}$$

where $\mathbf{A}_0^{(v)} = \mathbf{Y}_0^{(v)}(\mathbf{Y}_0^{(v)})^T$, and $Tr(\cdot)$ is the matrix trace.

Define $vec(\cdot)$ as a operation that outputs a vector whose elements are taken column-wise from a matrix, and $mat(\cdot)$ as its inverse operation. For convenience, denote $\mathbf{a} = vec(\mathbf{A})$ for arbitrary matrix $\mathbf{A}$. Note that $\|\mathbf{A}\|_F^2 = \mathbf{a}^T\mathbf{a}$, and for symmetric matrices $\mathbf{A}$ and $\mathbf{B}$, $Tr(\mathbf{A}\mathbf{B}) = \mathbf{a}^T\mathbf{b}$. Denote $q_i = \min_{1\le v\le V} \{\|\mathbf{A}_0^{(v)}\|_F^2 - 2Tr(\mathbf{A}_i^T\mathbf{A}_0^{(v)})\}$, then we have

$$\mathbf{a}_0^T\mathbf{a}_0 - 2\mathbf{a}_i^T\mathbf{a}_0 = q_i, 1 \le i \le m. \tag{10}$$

Eliminating the quadratic term $\mathbf{a}_0^T\mathbf{a}_0$, it arrives at

$$2(\mathbf{a}_i - \mathbf{a}_j)^T\mathbf{a}_0 = q_j - q_i, 1 \le i < j \le m. \tag{11}$$

Let $\mathbf{P} = 2[(\mathbf{a}_2 - \mathbf{a}_1)^T; \cdots; (\mathbf{a}_m - \mathbf{a}_1)^T; \cdots; (\mathbf{a}_m - \mathbf{a}_{m-1})^T]$, and $\mathbf{q} = [q_1 - q_2; \cdots; q_1 - q_m; \cdots; q_{m-1} - q_m]$, then,

$$\mathbf{P}\mathbf{a}_0 = \mathbf{q}. \tag{12}$$

Eq. (12) has many solutions since it is an underdetermined equation system. If there exists a $\hat{\mathbf{Y}}_0 \in \{\mathbf{Y}_0^{(v)}\}_{v=1}^V$, such that $\hat{\mathbf{Y}}_0 = \arg\min_{1\le v\le V} d_{\chi^2}^2(\mathbf{Y}_0^{(v)}, \mathbf{Y}_i)$ $(i = 1, \cdots, m)$, then $\hat{\mathbf{Y}}_0$ is a solution to Eq. (12). Hence, we first judge whether there is such a solution, if not, then solve Eq. (12).

As shown in Eq. (7), as long as the squared Frobenius norm of the difference between $\mathbf{A}_0$ and $\mathbf{A}_i$ equals to $\min_{1\le v\le V} d_{\chi^2}^2(\mathbf{Y}_0^{(v)}, \mathbf{Y}_i)$ $(i = 1, \cdots, m)$, which solution of Eq. (12) is adopted has no effect on the resolution of Eq. (7).

In this paper, we suppose there are at least two points in each cluster. According to the definition of normalized cluster indicator matrix $\mathbf{Y}$, it is easy to derive that the elements of $\mathbf{Y}\mathbf{Y}^T$ are within $[0, \frac{1}{2}]$. Therefore, if $\mathbf{A}_0$ is strictly equal to $\mathbf{Y}_0\mathbf{Y}_0^T$, then $0 \le \mathbf{a}_0 \le \frac{1}{2}$. Thus, to suit the original intention better, we aim to solve $\mathbf{a}_0$ with the following formulation:

$$\min_{0\le\mathbf{a}\le\frac{1}{2}} \|\mathbf{P}\mathbf{a} - \mathbf{q}\|_2, \tag{13}$$

which is a bounded linear least squares problem and can be efficiently solved with the *lsqlin* function in the MOSEK package[2]. Once $\mathbf{a}_0$ is solved, $\mathbf{A}_0 = mat(\mathbf{a}_0)$.

Solving Eq. (7) Problem (7) is a convex-concave optimization since the objective function is convex with respect to $\boldsymbol{\alpha}$ and concave with respect to $\mathbf{A}$. Problems of this kind can be solved by gradient descent algorithms such as the infeasible start Newton method (Ghosh and Boyd 2003). However, the efficiency of gradient descent algorithms is not appealing (Nesterov 2013). In the following, we show that Eq.

[2]https://www.mosek.com/resources/downloads

(7) can be transformed into a small-scale convex linearly constrained quadratic programming.

Note that the objective function in Eq. (7) is differentiable, and there is no other constraints on $\mathbf{A}$. Thus, the partial derivative of the optimal solution $\hat{\mathbf{A}}$ is zero and a closed-form solution can be obtained. That is,

$$\hat{\mathbf{A}} = \sum_{i=1}^{m} \alpha_i \mathbf{A}_i. \tag{14}$$

Substituting Eq.(14) into Eq.(7), the resulting formulation is only related to $\boldsymbol{\alpha}$:

$$\min_{\boldsymbol{\alpha}\in\mathcal{M}} \left\| \sum_{i=1}^{m} \alpha_i \mathbf{A}_i - \mathbf{A}_0 \right\|_F^2. \tag{15}$$

Further, expanding the quadratic form in Eq.(15), it can be rewritten as

$$\min_{\boldsymbol{\alpha}\in\mathcal{M}} \boldsymbol{\alpha}^T \mathbf{F} \boldsymbol{\alpha} - \mathbf{v}^T \boldsymbol{\alpha}, \tag{16}$$

where $\mathbf{F} \in \mathbb{R}^{m\times m}$ is a matrix with $\mathbf{F}_{ij} = Tr(\mathbf{A}_i \mathbf{A}_j^T)$, $\forall 1 \le i, j \le m$ and $\mathbf{v} = [2Tr(\mathbf{A}_1\mathbf{A}_0^T); \cdots; 2Tr(\mathbf{A}_m\mathbf{A}_0^T)]$. It is easy to verify that $\mathbf{F}$ is positive semi-definite, and the constraint set of $\boldsymbol{\alpha}$ ($\mathcal{M} = \{\boldsymbol{\alpha} | \mathbf{1}^T\boldsymbol{\alpha} = 1; \boldsymbol{\alpha} \ge \mathbf{0}\}$) is a simplex, therefore, Eq.(16) is a convex linearly constrained quadratic programming. In this step, constructing $\mathbf{F}$ and $\mathbf{v}$ consumes $\mathcal{O}(n^2m^2)$, and solving Eq. (16) spends $\mathcal{O}(m^3)$. Since the number of candidate multi-view clusterings m is usually small, Eq.(16) can be efficiently solved by employing state-of-the-art optimization solvers, such as the MOSEK package. After obtaining the optimal solution $\hat{\boldsymbol{\alpha}}$, then the optimal $\hat{\mathbf{A}} = \sum_{i=1}^{m} \hat{\alpha}_i \mathbf{A}_i$.

Solving Eq. (8) Note that we have constrained $\mathbf{Y}$ to be a normalized cluster indicator matrix. It is hard to decompose $\hat{\mathbf{A}}$ into the product of $\mathbf{Y}$ and its transpose and make $\mathbf{Y}$ in the exact form of normalized cluster indicator matrix simultaneously. Thus, we relax the constraint of $\mathbf{Y}$ in Eq. (8) to be $\mathbf{Y}^T\mathbf{Y} = \mathbf{I}, \mathbf{Y} \ge \mathbf{0}$, where $\mathbf{I}$ denotes the identity matrix. That is to say, we aim to solve the following problem

$$\hat{\mathbf{Y}} = \underset{\mathbf{Y}\ge\mathbf{0}, \mathbf{Y}^T\mathbf{Y}=\mathbf{I}}{\arg\min} \left\| \mathbf{Y}\mathbf{Y}^T - \hat{\mathbf{A}} \right\|_F^2. \tag{17}$$

With simple mathematical operations, problem (17) is equivalent to

$$\max_{\mathbf{Y}\ge\mathbf{0}, \mathbf{Y}^T\mathbf{Y}=\mathbf{I}} Tr(\mathbf{Y}^T\hat{\mathbf{A}}\mathbf{Y}). \tag{18}$$

This optimization problem can be solved by utilizing an iterative algorithm (Cai et al. 2011):

$$\mathbf{Y}_{ij} \leftarrow \mathbf{Y}_{ij} \sqrt{\frac{(\hat{\mathbf{A}}\mathbf{Y})_{ij}}{(\mathbf{Y}\boldsymbol{\beta})_{ij}}}, \quad \boldsymbol{\beta} = \mathbf{Y}^T\hat{\mathbf{A}}\mathbf{Y}. \tag{19}$$

The algorithmic complexity of this iterative algorithm is $\mathcal{O}(Tn^2K)$, where T is the number of iterations.

Once $\hat{\mathbf{Y}}$ is obtained, it is projected back to normalized clustering indicator matrix by setting $\hat{\mathbf{Y}}_{ik} = n_k^{-1/2}$ if $k = \underset{1\le j\le K}{\arg\max} \hat{\mathbf{Y}}_{ij} (i = 1, \cdots, n)$, and 0 otherwise. Algorithm 1 summarizes the pseudocode of the proposed method.

Algorithm 1 Reliable Multi-View Clustering

Input: Single-view clustering results $\{\mathbf{Y}_0^{(v)}\}_{v=1}^V$ and candidate multi-view clustering results $\{\mathbf{Y}_i\}_{i=1}^m$.
Output: The learned clustering result $\hat{\mathbf{Y}}$.
Procedure:
1: Construct $\mathbf{P}$ and $\mathbf{q}$, and solve the bounded linear least squares problem in Eq. (13) to obtain $\mathbf{A}_0$.
2: Construct $\mathbf{F}$ with $\mathbf{F}_{ij} = Tr(\mathbf{A}_i\mathbf{A}_j^T)$, $\forall 1 \le i, j \le m$, and $\mathbf{v} = [2Tr(\mathbf{A}_1\mathbf{A}_0^T); \cdots; 2Tr(\mathbf{A}_m\mathbf{A}_0^T)]$.
3: Solve the convex quadratic optimization Eq. (16) and obtain the optimal solution $\hat{\boldsymbol{\alpha}}$.
4: Perform the iterative algorithm Eq. (19) to get the final cluster indicator matrix $\hat{\mathbf{Y}}$.

Theoretical Analysis

In this section, we provide some analysis of the proposed proposal. Denote $\hat{\mathbf{Y}}$ abusively as the optimal solution of Eq. (5) and denote $\sum_{i=1}^{m} \alpha_i \big(d^2_{\chi^2}(\mathbf{Y}_0, \mathbf{Y}_i) - d^2_{\chi^2}(\mathbf{Y}, \mathbf{Y}_i)\big)$ as $g(\mathbf{Y}, \mathbf{Y}_0, \boldsymbol{\alpha})$. Assume that the ground-truth clustering can be realized by one of the multi-view clustering algorithms, i.e., $\mathbf{Y}^* \in \{\mathbf{Y}_i\}_{i=1}^m$, we can prove that the performance of $\hat{\mathbf{Y}}$ is not worse than that of $\{\mathbf{Y}_0^{(v)}\}_{v=1}^V$.

Theorem 1. *If the ground-truth clustering $\mathbf{Y}^* \in \{\mathbf{Y}_i\}_{i=1}^m$, then $d^2_{\chi^2}(\hat{\mathbf{Y}}, \mathbf{Y}^*) \le \min_{1\le v\le V} d^2_{\chi^2}(\mathbf{Y}_0^{(v)}, \mathbf{Y}^*)$.*

Proofs of theorems in this paper are in the supplemental material. Theorem 1 shows that the proposed RMVC method is provably reliable when the ground-truth clustering can be realized by one of the candidate multi-view clustering algorithms. Note that this is a sufficient rather than necessary condition for RMVC. In other words, RMVC may still work when the ground truth is not among the candidate multi-view clusterings.

Since the condition given in Theorem 1 is difficult to attain in reality, in the following, we will study how the performance of RMVC will be affected when the condition is not satisfied. Specifically, let $\bar{\mathbf{Y}} \in \{\mathbf{Y}_i\}_{i=1}^m$ satisfy

$$\bar{\mathbf{Y}} = \underset{1\le i\le m}{\arg\min} \left\| \mathbf{Y}_i\mathbf{Y}_i^T - \mathbf{Y}^*(\mathbf{Y}^*)^T \right\|_F^2. \tag{20}$$

Denote $\boldsymbol{\epsilon} = \mathbf{Y}^*(\mathbf{Y}^*)^T - \bar{\mathbf{Y}}\bar{\mathbf{Y}}^T$ as the residual to reflect the degree of violation. The following theorem provides some insight into the robustness of RMVC when the condition in Theorem 1 is violated.

Theorem 2. *If $Tr\big((\mathbf{Y}_0\mathbf{Y}_0^T - \mathbf{Y}_0^{(v)}(\mathbf{Y}_0^{(v)})^T)\boldsymbol{\epsilon}\big) \ge 0$, $\forall 1 \le v \le V$, where $\mathbf{Y}_0$ is defined as $d^2_{\chi^2}(\mathbf{Y}_0, \mathbf{Y}_i) = \min_{1\le v\le V} d^2_{\chi^2}(\mathbf{Y}_0^{(v)}, \mathbf{Y}_i) (i = 1, \cdots, m)$, then the increased loss of the proposed method against $\{\mathbf{Y}_0^{(v)}\}_{v=1}^V$, i.e., $\frac{1}{n^2}\big(d^2_{\chi^2}(\hat{\mathbf{Y}}, \mathbf{Y}^*) - \min_{1\le v\le V} d^2_{\chi^2}(\mathbf{Y}_0^{(v)}, \mathbf{Y}^*)\big)$, is at most $\min\{\frac{\|\boldsymbol{\epsilon}\|_1}{n^2}, \frac{\|\boldsymbol{\epsilon}\|_F}{n}\}$.*

When the required reliability condition is violated, as illustrated in Theorem 2, if the calculated $\mathbf{Y}_0$ satisfies

the above condition, then the worst-case increased loss of RMVC is only related to the norm of the residual. In other words, the robustness of RMVC is largely related to the quality of candidate multi-view clusterings. To improve the quality of candidate clusterings, one may choose clusterings with large between-group scatter and small within-group scatter.

Experiment

In this section, we conduct experiments to validate the effectiveness of the proposed method. The multi-view data sets used in the experiments cover diverse domains, including six text data sets, seven image data sets, and one image-text data set.

Data Set Description

The six text data sets are 3sources[3] (3sou), BBC3/4views (BBC3/4) and BBCSport2/3/4views[4] (BSpt2/3/4). **3sou** data set was collected from three well-known online news sources: BBC, Reuters, and The Guardian, corresponding to 3 views. Each view is a term-document matrix. We use the 169 stories reported in all three sources for our experiment. The stories were manually categorized into 6 classes according to the primary section headings. By splitting the single-view BBC and BBCSport news articles from 5 topical areas into related segments of text, 5 multi-view data sets were constructed, i.e., **BBC3/4** and **BSpt2/3/4**. The same with 3sources, each view is a term-document matrix.

Image data sets employed in the experiments contain Caltech101[5], Corel[6], UIUC Sport Event data set[7] (Event), the multi-feature digits dataset[8] (Digits), the indoor scene database[9] (Indoor), Microsoft Research Cambridge Volume 1[10] (MSRC), and Scene UNderstanding database[11] (SUN).

Following (Li et al. 2015), 7 widely used classes with 441 images were selected from the **Caltech101** object recognition database. The resulting dataset is referred to as **Cal7**. Six visual features are extracted for each image: LBP (256) (Ojala, Pietikäinen, and Mäenpää 2002), pyramid HOG (680) (Dalal and Triggs 2005), GIST (512) (Oliva and Torralba 2001), SURF (200) (Bay, Tuytelaars, and Van Gool 2006), SIFT (200) (Lowe 2004), and wavelet texture (WT, 32) (Manjunath and Ma 1996), where numerals in the parentheses denote the dimensions of different views.

Corel consists of 5000 images from 50 different categories. Each category has 100 images. The features are color histogram (9), edge direction histogram (18) and WT (9).

Event contains 1579 images belonging to 8 sports event categories. Except that the dimensions of SURF and SIFT features are increased to 500, the same six kinds of features with Caltech7 are extracted for Event data set.

Digits data set is comprised of 2,000 data points from 0 to 9 digit classes, with 200 data points for each class. There are six public features available: 76 Fourier coefficients of the character shapes, 216 profile correlations, 64 Karhunen-love coefficients, 240 pixel averages in 2×3 windows, 47 Zernike moments and 6 morphological features.

The original **Indoor** database contains 67 indoor categories. We choose 5 categories (*auditorium*, *buffet*, *classroom*, *cloister* and *elevator*) with total 621 images for our experiments and extract the same features with Caltech7.

For **MSRC** data set, we follow (Cai et al. 2011) to select 7 classes (*tree*, *building*, *airplane*, *cow*, *face*, *car*, *bicycle*), and each class has 30 images. Color moment (48), LBP (256), HOG (100), SIFT (200), GIST (512) and CENTRIST (1302) (Wu and Rehg 2011) features are extracted.

We randomly choose 10 classes from the 397 well-sampled subset of the **SUN** database (Xiao et al. 2016) and each class has 100 images. We refer to the sampled subset as **SUN1k**. Three pre-extracted features: SIFT (6300), HOG (6300) and texton histogram (10752), are adopted.

The **NBA-NASCAR Sport** (**NNSpt**) image-text dataset is collected by (Sun 2011), including 420 NBA images and 420 NASCAR images. Each image is normalized to be a 32×32-sized gray image, thus the image view has dimension of 1024. The attached short text is preprocessed and each text has a 296-dimensional TFIDF (Salton and Buckley 1988) feature.

Experimental Setting

Our proposed method is compared with the following methods.

Best single-view normalized cut (BestNcut) (Shi and Malik 2000). On each view, single-view Ncut is performed and the best results are reported.

Co-regularized multi-view spectral clustering (CoRegSC) (Kumar, Rai, and Daume 2011). The approach employs co-regularization to make the clusterings in different views agree with each other. We implement the centroid-based co-regularization approach.

Co-trained multi-view spectral clustering (CoTrainSC) (Kumar and Daumé 2011). CoTrainSC utilizes the spectral embedding from one view to modify the graph structures in other views. By iteratively applying this procedure, the clusterings of multiple views tend towards consensus.

Multi-modal spectral clustering (MMSC) (Cai et al. 2011). MMSC learns a commonly shared graph Laplacian matrix by minimizing both spectral clustering error of each view and the distances between the common clustering indicator matrix and each single-view one.

Multi-view spectral clustering (MVSC) (Li et al. 2015). MVSC aims to accelerate the multi-view spectral clustering process by approximating the similarity graphs using bipartite graphs.

Robust multi-view K-means clustering (RMKMC) (Cai, Nie, and Huang 2013). RMKMC integrates data's multiple representations via structured sparsity-inducing norm to make it more robust to data outliers.

[3] http://mlg.ucd.ie/datasets/3sources.html

[4] http://mlg.ucd.ie/datasets/segment.html

[5] http://www.vision.caltech.edu/Image_Datasets/Caltech101/

[6] http://www.cais.ntu.edu.sg/~chhoi/SVMBMAL/

[7] vision.stanford.edu/lijiali/event_dataset/

[8] https://archive.ics.uci.edu/ml/datasets/Multiple+Features

[9] http://web.mit.edu/torralba/www/indoor.html

[10] https://www.microsoft.com/en-us/research/project/image-understanding/

[11] http://vision.princeton.edu/projects/2010/SUN/

Table 1: Clustering results in terms of clustering accuracy (mean ± std.). Symbols '✓/◊/▼' denote respectively that the corresponding multi-view method is better/tied/worse than the best single-view normalized cut by the paired t-test with confidence level 0.05. The win/tie/loss counts are summarized in the last row, and the method with the smallest number of losses is bolded.

Data sets	BestNcut	CoRegSC	CoTrainSC	MMSC	MVSC	RMKMC	RMVC
3sou	.461(.017)	.453(.032)◊	.473(.031)✓	.391(.024)▼	.462(.034)◊	.467(.067)◊	.503(.000)✓
BBC3	.356(.015)	.466(.035)✓	.491(.027)✓	.402(.040)✓	.472(.021)✓	.579(.051)✓	.427(.000)✓
BBC4	.389(.001)	.425(.017)✓	.390(.025)◊	.336(.023)▼	.597(.021)✓	.565(.061)✓	.419(.000)✓
BSpt2	.495(.001)	.700(.018)✓	.660(.026)✓	.727(.003)✓	.476(.025)▼	.559(.072)✓	.746(.000)✓
BSpt3	.528(.003)	.593(.029)✓	.581(.026)✓	.628(.020)✓	.547(.054)✓	.597(.091)✓	.603(.000)✓
BSpt4	.533(.043)	.718(.044)✓	.607(.047)✓	.723(.011)✓	.669(.040)✓	.339(.074)▼	.724(.000)✓
Cal7	.717(.001)	.735(.034)✓	.752(.049)✓	.724(.027)◊	.765(.045)✓	.683(.076)▼	.800(.000)✓
Corel	.203(.004)	.365(.009)✓	.281(.010)✓	.294(.005)✓	.176(.003)▼	.256(.007)✓	.258(.001)✓
Event	.447(.002)	.420(.021)▼	.491(.027)✓	.557(.001)✓	.359(.014)▼	.360(.025)▼	.516(.014)✓
Digits	.930(.000)	.850(.064)▼	.853(.070)▼	.982(.000)✓	.883(.060)▼	.777(.082)▼	.937(.017)✓
Indoor	.538(.026)	.438(.009)▼	.556(.038)✓	.645(.014)✓	.498(.020)▼	.412(.029)▼	.647(.000)✓
MSRC	.777(.003)	.875(.055)✓	.878(.059)✓	.876(.028)✓	.681(.048)▼	.825(.075)✓	.873(.018)✓
SUN1k	.334(.016)	.422(.024)✓	.442(.018)✓	.411(.009)✓	.390(.015)✓	.363(.020)✓	.423(.000)✓
NNSpt	.656(.000)	.749(.000)✓	.613(.000)▼	.858(.158)✓	.989(.000)✓	.987(.000)✓	.761(.000)✓
Ave.	.526	.586	.576	.611	.569	.555	.617
win/tie/loss against BestNcut		10/1/3	11/1/2	11/1/2	7/1/6	8/1/5	**14/0/0**

The implementations of Ncut[12], CoRegSC[13], CoTrainSC[14], MMSC[15] and RMKMC[16] are downloaded from their authors' homepages. All algorithms are tested with MATLAB R2013b, and our method also use MOSEK 7.1. The experiments are conducted on a work station with 12 cores (2.10 GHz for each) and 96.0 GB RAM memory.

Except for RMKMC, all the other above mentioned methods need to construct affinity graphs. Two points are connected if at least one of them is among the k nearest neighbors of the other in the Euclidean distance and k is set to be 9 empirically. The edge weight is calculated using Gaussian Kernel, where the bandwidth parameter is set as the mean squared Euclidean distance between sample pairs. For CoRegSC, CoTrainSC and MMSC, their trade-off parameters are selected from $\{0.01, 0.1, 1, 10, 100\}$, and the best results are reported. For MVSC, the number of salient points is set as 10% of the total number of examples. For the proposed RMVC, we use 3 multi-view clustering results, which are produced by CoRegSC, CoTrainSC and MMSC respectively. We use the same method with MMSC to initialize the iterative algorithm in Eq. (19). k-means is employed to get the final discrete clustering for approaches based on spectral clustering. As the results of all algorithms depend on the initial conditions, we repeat 50 times for all methods and report the average results and the standard deviation. The clustering performance is evaluated in terms of clustering accuracy (ACC) and NMI.

[12]https://www.cis.upenn.edu/ jshi/software/

[13]http://www.umiacs.umd.edu/~abhishek/code_coregspectral.zip

[14]http://www.umiacs.umd.edu/~abhishek/code_cospectral.zip

[15]http://www.escience.cn/system/file?fileId=67628

[16]http://www.escience.cn/system/file?fileId=67658

Clustering Results

Table 1 shows the results of ACC and Fig. 1 presents the results of NMI. We have following observations.

In terms of average performance, all multi-view methods achieve higher ACC than best single-view Ncut. In addition, our proposed RMVC obtains the highest average ACC.

CoRegSC, CoTrainSC and MMSC achieve good performance for all three performance measures. They all obtains at least 10 wins against BestNcut. However, the performance of CoRegSC is dramatically degenerated on Event, Digits, and Indoor; CoTrainSC loses on Digits, Indoor and NNSpt; and MMSC is defeated by BestNcut on 3sou, BBC3, BBC4 and Cal7 with respect to ACC or NMI.

Though the average ACC of both MVSC and RMKMC are higher than that of BestNcut, their performance is not satisfying. MVSC causes serious performance degeneration on one text data sets and five image data sets, and the clustering results of RMKMC are significantly worse than BestNcut on five data sets.

Our proposed RMVC significantly outperforms the best single-view Ncut on all data sets for ACC and wins 13 cases in terms of NMI. What is more important is that RMVC does not seriously decrease the performance.

Note that the reliability condition presented in Theorem 1 is a sufficient rather than necessary condition for RMVC, thus RMVC may still work when the condition is not fulfilled. This has been verified by the experimental results. As shown in Table 1, none of the candidate multi-view clustering algorithms (CoRegSC, CoTrainSC and MMSC) reaches 100% accuracy, i.e., none of them realizes the ground-truth clustering on all data sets. Yet, RMVC still achieves better or comparable clustering results comparing with the BestNcut.

Since RMVC is formulated with the χ^2 distance that relates to ACC, it is natural that it obtains expected results in terms of ACC. Surprisingly, as shown in Fig. 1, when eval-

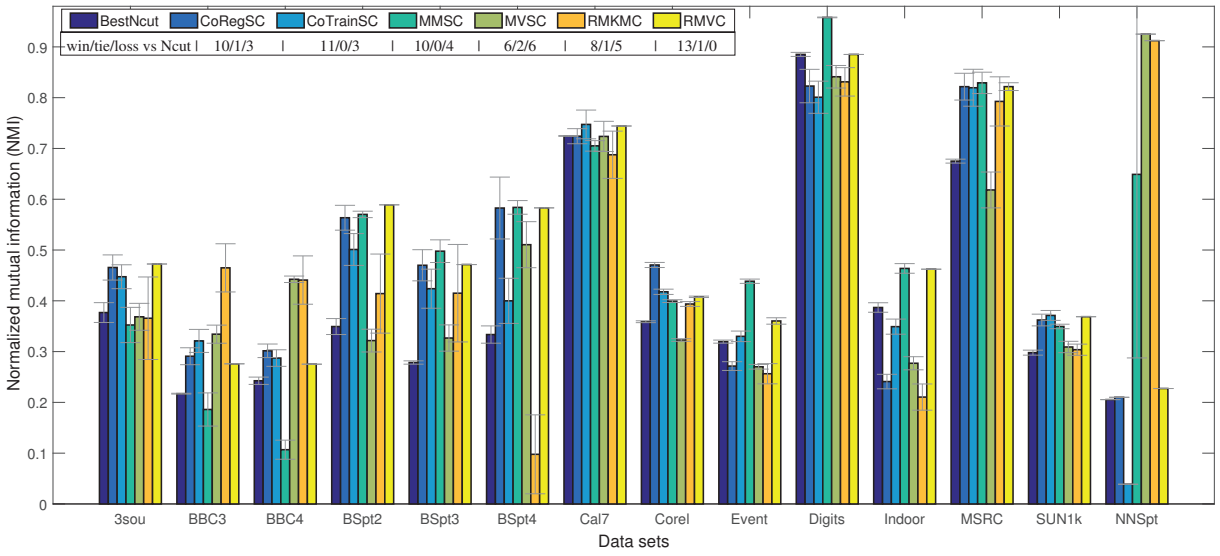

Figure 1: Comparison of clustering results with respect to NMI. The bars show the average values and the standard deviation is presented by the error bar. The results of multi-view methods are compared with the best single-view Ncut by the paired t-test with confidence level 0.05. The win/tie/loss counts are summarized in the top of the figure.

Table 2: Average CPU time (in seconds) for all the methods on four representative data sets. The running time of RMVC is the sum of RMVC(c) and RMVC(p), where RMVC(c) and RMVC(p) denote the time of computing candidate clusterings and solving Eq. (5) respectively.

Data sets	BBC3	Corel	Digits	SUN1k
Ncut	1.93	24.52	5.23	1.32
CoRegSC	20.25	913.64	58.53	32.37
CoTrainSC	23.12	631.26	104.07	12.25
MMSC	3.17	85.10	10.59	1.96
MVSC	553.31	48.67	28.56	420.83
RMKMC	75.95	121.92	35.27	123.47
RMVC(c)	46.54	1620.00	173.19	46.58
RMVC(p)	7.15	48.83	26.35	4.31

uating with NMI, our method not only ensures reliable performance on all data sets, but also achieves competitive performance for most data sets. This reflects that RMVC has a certain degree of robustness to the change of performance measures.

In summary, the proposed method RMVC effectively improves the reliability of multi-view clustering and obtains highly competitive performance with state-of-the-art approaches. Moreover, its performance is robust to the violation of reliability condition and the change of evaluation metrics to some extent.

Running Time

Table 2 displays the running time of all methods on four representative data sets. The running time of RMVC is the sum of the time of computing candidate clusterings by CoRegSC, CoTrainSC and MMSC (RMVC(c)), and the time of post-processing by solving Eq. (5) (RMVC(p)). As shown from the results, Ncut is the fastest, because it does not need iteration, whereas all the compared multi-view algorithms need. It is shown that the optimization procedure of solving Eq. (5) in RMVC is efficient. Thus, by performing RMVC, we can obtain a more reliable clustering in a few extra time.

Conclusion

Although multi-view learning has flourished, little work has done to make the performance not worse than that of single views explicitly. In this paper, we try to address the reliability of multi-view clustering and propose the RMVC method. RMVC exploits several candidate multi-view clusterings to maximize the worst-case performance gain against the best single view clustering. Measured in the χ^2 distance, the final formulation is solved efficiently with a small-scale convex linearly constrained quadratic programming and a non-negative orthogonal matrix factorization. The reliability of RMVC is provable when one of the candidate multi-view cluster learners realizes the ground truth. Comprehensive experiments validate the ability of RMVC in obtaining reliable clustering. In the future, we will work on finding a reliability condition which can be judged more easily.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61473302, 61503396). Chenping Hou is the corresponding author of this article.

References

Bay, H.; Tuytelaars, T.; and Van Gool, L. 2006. SURF: Speeded up robust features. In *ECCV*. 404–417.

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

Cai, X.; Nie, F.; Huang, H.; and Kamangar, F. 2011. Heterogeneous image feature integration via multi-modal spectral clustering. In *CVPR*, 1977–1984.

Cai, X.; Nie, F.; and Huang, H. 2013. Multi-view k-means clustering on big data. In *IJCAI*, 2598–2604.

Dalal, N., and Triggs, B. 2005. Histograms of oriented gradients for human detection. In *CVPR*, 886–893.

Ghosh, A., and Boyd, S. 2003. Minimax and convex-concave games. Technical Report EE392o, Standford University.

Hou, C.; Zhang, C.; Wu, Y.; and Nie, F. 2010. Multiple view semi-supervised dimensionality reduction. *Pattern Recognition* 43(3):720–730.

Kumar, A., and Daumé, H. 2011. A co-training approach for multi-view spectral clustering. In *ICML*, 393–400.

Kumar, A.; Rai, P.; and Daume, H. 2011. Co-regularized multi-view spectral clustering. In *NIPS*, 1413–1421.

Li, Y. F., and Zhou, Z. H. 2011. Towards making unlabeled data never hurt. In *ICML*, 1081–1088.

Li, Y.-F., and Zhou, Z.-H. 2015. Towards making unlabeled data never hurt. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(1):175–188.

Li, Y.; Nie, F.; Huang, H.; and Huang, J. 2015. Large-scale multi-view spectral clustering via bipartite graph. In *AAAI*, 2750–2756.

Li, Y. F.; Kwok, J. T.; and Zhou, Z. H. 2016. Towards safe semi-supervised learning for multivariate performance measures. In *AAAI*, 1816–1822.

Li, Y.-F.; Zha, H.-W.; and Zhou, Z.-H. 2017. Learning safe prediction for semi-supervised regression. In *AAAI*, 2217–2223.

Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60(2):91–110.

Manjunath, B. S., and Ma, W.-Y. 1996. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(8):837–842.

Meilă, M. 2012. Local equivalences of distances between clusterings - a geometric perspective. *Machine Learning* 86(3):369–389.

Nesterov, Y. 2013. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Nie, F.; Cai, G.; and Li, X. 2017. Multi-view clustering and semi-supervised classification with adaptive neighbours. In *AAAI*, 2408–2414.

Ojala, T.; Pietikäinen, M.; and Mäenpää, T. 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(7):971–987.

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

Salton, G., and Buckley, C. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management* 24(5):513–523.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* 22(8):888–905.

Sun, S. 2011. Multi-view laplacian support vector machines. In *ADMA*, 209–222.

Tao, H.; Hou, C.; Nie, F.; Zhu, J.; and Yi, D. 2017a. Scalable multi-view semi-supervised classification via adaptive regression. *IEEE Transactions on Image Processing* 26(9):4283–4296.

Tao, Z.; Liu, H.; Li, S.; Ding, Z.; and Fu, Y. 2017b. From ensemble clustering to multi-view clustering. In *IJCAI*, 2843–2849.

Wu, J., and Rehg, J. M. 2011. CENTRIST: A visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33(8):1489–1501.

Xiao, J.; Ehinger, K. A.; Hays, J.; Torralba, A.; and Oliva, A. 2016. SUN database: Exploring a large collection of scene categories. *International Journal of Computer Vision* 119(1):3–22.

Xie, X., and Sun, S. 2013. Multi-view clustering ensembles. In *ICMLC*, 51–56.

Xu, C.; Tao, D.; and Xu, C. 2013. A survey on multi-view learning. *arXiv:1304.5634*.

Yang, Y.; Song, J.; Huang, Z.; Ma, Z.; Sebe, N.; and Hauptmann, A. G. 2013. Multi-feature fusion via hierarchical regression for multimedia analysis. *IEEE Transactions on Multimedia* 15(3):572–581.

Zhang, C.; Fu, H.; Liu, S.; Liu, G.; and Cao, X. 2015. Low-rank tensor constrained multiview subspace clustering. In *ICCV*, 1582–1590.

Zhou, Z.-H. 2012. *Ensemble Methods: Foundations and Algorithms*. Chapman & Hall/CRC.