Multi-View K-Means Clustering on Big Data

Xiao Cai, Feiping Nie, Heng Huang[*]
University of Texas at Arlington
Arlington, Texas, 76092
xiao.cai@mavs.uta.edu, feipingnie@gmail.com, heng@uta.edu

Abstract

In past decade, more and more data are collected from multiple sources or represented by multiple views, where different views describe distinct perspectives of the data. Although each view could be individually used for finding patterns by clustering, the clustering performance could be more accurate by exploring the rich information among multiple views. Several multi-view clustering methods have been proposed to unsupervised integrate different views of data. However, they are graph based approaches, *e.g.* based on spectral clustering, such that they cannot handle the large-scale data. How to combine these heterogeneous features for unsupervised large-scale data clustering has become a challenging problem. In this paper, we propose a new robust large-scale multi-view clustering method to integrate heterogeneous representations of large-scale data. We evaluate the proposed new methods by six benchmark data sets and compared the performance with several commonly used clustering approaches as well as the baseline multi-view clustering methods. In all experimental results, our proposed methods consistently achieve superiors clustering performances.

1 Introduction

With the rising of data sharing websites, such as Facebook and Flickr, there is a dramatic growth in the number of data. For example, Facebook reports about 6 billion new photo every month and 72 hours of video are uploaded to YouTube every minute. One of major data mining tasks is to unsupervised categorize the large-scale data [Biswas and Jacobs, 2012; Lee and Grauman, 2009; Dueck and Frey, 2007; Cai *et al.*, 2011], which is useful for many information retrieval and classification applications. There are two main computational challenges in large-scale data clustering: (1) How to integrate the heterogeneous data features to improve the performance of data categorizations? (2) How to reduce the computational cost of clustering algorithm for large-scale applications?

Many scientific data have heterogeneous features, which are generated from different data collection sources or feature construction ways. For example, in biological data, each human gene can be measured by different techniques, such as gene expression, Single-nucleotide polymorphism (SNP), Array-comparative genomic hybridization (aCGH), methylation; in visual data, each image/video can be represented by different visual descriptors, such as SIFT [Lowe, 2004], HOG [Dalal and Triggs, 2005], LBP [Ojala *et al.*, 2002], GIST [Oliva and Torralba, 2001], CENTRIST [Wu and Rehg, 2008], CTM [Yu *et al.*, 2002]. Each type of features can capture the specific information in the data. For example, in visual descriptors, CTM uses the color spectral information and hence is good for categorizing the images with large color variations; GIST achieves high accuracy in recognizing natural scene images; CENTRIST is good for classifying indoor environment images; HOG can describe the shape information of the image; SIFT is robust to image rotation, noise, illumination changes; and LBP is a powerful texture feature. It is crucial to integrate these heterogeneous features to create more accurate and more robust clustering results than using each individual type of features.

Although several graph based multi-view clustering algorithms were presented with good performance, they have the following two main drawbacks. On one hand, because all of them are graph based clustering method, the construction of data graph is a key issue. Using different kernels to build the graph will affect the final clustering performance a lot. Moreover, for some specific kernels, we have to consider the impact of the choice of parameters, such that the clustering results are sensitive to the parameters tuning. On the other hand, more important, due to the heavy computation of the kernel construction as well as eigen decomposition, these graph based methods cannot be utilized to tackle large-scale data clustering problem.

The classical K-means clustering is a centroid-based clustering method, which partitions the data space into a structure known as Voronoi diagram. Due to its low computational cost and easily parallelized process, the K-means clustering method has often been applied to solve large-scale data clustering problems, instead of the spectral clustering. However, the K-means clustering was designed for solving single-view data clustering problem. In this paper, we propose a new robust multi-view K-means clustering method to integrate het-

[*]Corresponding Author. This work was partially supported by NSF CCF-0830780, CCF-0917274, DMS-0915228, IIS-1117965.

erogeneous features for clustering. Compared to related clustering methods, our proposed method consistently achieves better clustering performances on six benchmark data sets. Our contributions in this paper are summarized in the following four folds:

(1) We propose a novel robust large-scale multi-view K-means clustering approach, which can be easily parallelized and performed on multi-core processors for big visual data clustering;

(2) Using the structured sparsity-inducing norm, $\ell_{2,1}$-norm, the proposed method is robust to data outliers and can achieve more stable clustering results with different initializations;

(3) We derive an efficient algorithm to tackle the optimization difficulty introduced by the non-smooth norm based loss function with proved convergence;

(4) Unlike the graph based algorithms, the computational complexity of our methods is similar to the standard K-means clustering algorithm. Because our method does not require the graph construction as well as the eigen-decomposition, it avoids the heavy computational burden and can be used for solving large-scale multi-view clustering problems.

2 Robust Multi-View K-Means Clustering

As one of most efficient clustering algorithms, K-means clustering algorithm has been widely applied to large-scale data clustering. Thus, to cluster the large-scale multi-view data, we propose a new robust multi-view K-means clustering (RMKMC) method.

2.1 Clustering Indicator Based Reformulation

Previous work showed that the G-orthogonal non-negative matrix factorization (NMF) is equivalent to relaxed K-means clustering [Ding *et al.*, 2005]. Thus, we reformulate the K-means clustering objective using the clustering indicators as:

$$\begin{aligned} &\min_{F,G} ||X^T - GF^T||_F^2 \\ &s.t.\ G_{ik} \in \{0,1\},\ \sum_{k=1}^{K} G_{ik} = 1,\ \forall i = 1, 2, \cdots, n \end{aligned} \tag{1}$$

where $X \in \mathbb{R}^{d \times n}$ is the input data matrix with n images and d-dimensional visual features, $F \in \mathbb{R}^{d \times K}$ is the cluster centroid matrix, and $G \in \mathbb{R}^{n \times K}$ is the cluster assignment matrix and each row of G satisfies the *1-of-K* coding scheme (if data point $\mathbf{x}_i$ is assigned to k-th cluster then $G_{ik} = 1$, and $G_{ik} = 0$, otherwise). In this paper, given a matrix $X = \{x_{ij}\}$, its i-th row, j-th column are denoted as $\mathbf{w}^i$, $\mathbf{w}_j$, respectively.

2.2 Robust Multi-View K-Means Clustering via Structured Sparsity-Inducing Norm

The original K-means clustering method only works for single-view data clustering. To solve the large-scale multi-view clustering problem, we propose a new multi-view K-means clustering method. Let $X^{(v)} \in \mathbb{R}^{d_v \times n}$ denote the features in v-th view, $F^{(v)} \in \mathbb{R}^{d_v \times K}$ be the centroid matrix for the v-th view, and $G^{(v)} \in \mathbb{R}^{n \times K}$ be the clustering indicator matrix for the v-th view. Given M types of heterogeneous features, $v = 1, 2, \cdots, M$.

The straightforward way to utilize all views of features is to concatenate all features together and perform the clustering algorithm. However, in such method, the important view of features and the less important view of features are treated equally such that the clustering results are not optimal. It is ideal to simultaneously perform the clustering using each view of features and unify their results based their importance to the clustering task. To achieve this goal, we have to solve two challenging problems: 1) how to naturally ensemble the multiple clustering results? 2) how to learn the importance of feature views to the clustering task? More important, we have to solve these issues simultaneously in the clustering objective function, thus previous ensemble approaches cannot be applied here.

When a multi-view clustering algorithm performs clustering using heterogeneous features, the clustering results in different views should be unique, *i.e.* the clustering indicator matrices $G^{(v)}$ of different views should share the same one. Therefore, in multi-view clustering, we force the cluster assignment matrices to be the same across different views, that is, the consensus common cluster indicator matrix $G \in \mathbb{R}^{n \times K}$, which should satisfy the *1-of-K* coding scheme as well.

Meanwhile, as we know, the data outliers greatly affect the performance of K-means clustering, because the K-means solution algorithm is an iterative method and in each iteration we need to calculate the centroid vector. In order to have a more stable clustering performance with respect to a fixed initialization, the robust K-means clustering method is desired. To tackle this problem, we use the sparsity-inducing norm, $\ell_{2,1}$-norm, to replace the ℓ_2-norm in the clustering objective function, *e.g.* Eq. (1). The $\ell_{2,1}$-norm of matrix X is defined as $||X||_{2,1} = \sum_{i=1}^{d} ||\mathbf{x}^i||_2$ (in other related papers, people also used the notation ℓ_1/ℓ_2-norm). The $\ell_{2,1}$-norm based clustering objective enforces the ℓ_1-norm along the data points direction of data matrix X, and ℓ_2-norm along the features direction. Thus, the effect of outlier data points in clustering are reduced by the ℓ_1-norm. We propose a new robust multi-view K-means clustering method by solving:

$$\begin{aligned} &\min_{F^{(v)},G,\alpha^{(v)}} \sum_{v=1}^{M} (\alpha^{(v)})^{\gamma} ||X^{(v)^T} - GF^{(v)^T}||_{2,1} \\ &s.t. G_{ik} \in \{0,1\},\ \sum_{k=1}^{K} G_{ik} = 1,\ \sum_{v=1}^{M} \alpha^{(v)} = 1, \end{aligned} \tag{2}$$

where $\alpha^{(v)}$ is the weight factor for the v-th view and γ is the parameter to control the weights distribution. We learn the weights for different types of features, such that the important features will get large weights during the multi-view clustering.

3 Optimization Algorithm

The difficulty of solving the proposed objective comes from the following two aspects. First of all, the $\ell_{2,1}$-norm is non-smooth. In addition, each entry of the cluster indicator matrix is a binary integer and each row vector must satisfy the *1-of-K* coding scheme. We propose new algorithm to tackle them efficiently.

3.1 Algorithm Derivation

To derive the algorithm solving Eq. (2), we rewrite Eq. (2) as

$$J = \min_{F^{(v)}, D^{(v)}, \alpha^{(v)}, G} \sum_{v=1}^{M} (\alpha^{(v)})^{\gamma} H^{(v)}, \tag{3}$$

where

$$H^{(v)} = Tr\{(X^{(v)} - F^{(v)}G^T)D^{(v)}(X^{(v)} - F^{(v)}G^T)^T\}\ . \tag{4}$$

$D^{(v)} \in \mathbb{R}^{n \times n}$ is the diagonal matrix corresponding to the v-th view and the i-th entry on the diagonal is defined as:

$$D_{ii}^{(v)} = \frac{1}{2\left\|\mathbf{e}^{(v)i}\right\|}, \quad \forall i = 1, 2, ..., n, \tag{5}$$

where $\mathbf{e}^{(v)i}$ is the i-th row of the following matrix:

$$E^{(v)} = X^{(v)^T} - GF^{(v)^T}. \tag{6}$$

The first step is fixing G, $D^{(v)}$, $\alpha^{(v)}$ and updating the cluster centroid for each view $F^{(v)}$.
Taking derivative of J with respect to $F^{(v)}$, we get

$$\frac{\partial J}{\partial F^{(v)}} = -2X^{(v)}\widetilde{D}^{(v)}G + 2F^{(v)}G^T\widetilde{D}^{(v)}G, \tag{7}$$

where

$$\widetilde{D}^{(v)} = (\alpha^{(v)})^{\gamma} D^{(v)}. \tag{8}$$

Setting Eq. (7) as 0, we can update $F^{(v)}$:

$$F^{(v)} = X^{(v)}\widetilde{D}^{(v)}G(G^T\widetilde{D}^{(v)}G)^{-1}. \tag{9}$$

The second step is fixing $F^{(v)}$, $D^{(v)}$, $\alpha^{(v)}$ and updating the cluster indicator matrix G.
We have

$$\begin{aligned} & \sum_{v=1}^{M} Tr\{(X^{(v)} - F^{(v)}G^T)\widetilde{D}(X^{(v)} - F^{(v)}G^T)^T\} \\ = & \sum_{v=1}^{M}\sum_{i=1}^{N} \widetilde{D}_{ii}^{(v)} ||\mathbf{x}_i^{(v)} - F^{(v)}\mathbf{g}_i||_2^2 \\ = & \sum_{i=1}^{N}(\sum_{v=1}^{M} \widetilde{D}_{ii}^{(v)} ||\mathbf{x}_i^{(v)} - F^{(v)}\mathbf{g}_i||_2^2) \end{aligned} \tag{10}$$

We can solve the above problem by decoupling the data and assign the cluster indicator for them one by one independently, that is, we need to tackle the following problem for the fixed specific i, with respect to vector $\mathbf{g} = [g_1, g_{2,} \cdots, g_K]^T \in \mathbb{R}^{K \times 1}$

$$\min_{\mathbf{g}} \sum_{v=1}^{M} \widetilde{d}^{(v)} ||\mathbf{x}^{(v)} - F^{(v)}\mathbf{g}||_2^2, \ s.t. g_k \in \{0, 1\}, \ \sum_{k=1}^{K} g_k = 1 \tag{11}$$

where $\widetilde{d}^{(v)} = \widetilde{D}_{ii}^{(v)}$ is the i-th element on the diagonal of the matrix $\widetilde{D}^{(v)}$. Given the fact that $\mathbf{g}$ satisfies *1-of-K* coding scheme, there are K candidates to be the solution of Eq. (11), each of which is the k-th column of matrix $I_K = [\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_K]$. To be specific, we can do an exhaustive search to find out the solution of Eq. (11) as,

$$\mathbf{g}^* = \mathbf{e}_k, \tag{12}$$

where k is decided as follows,

$$k = \arg\min_{j} \sum_{v=1}^{M} \widetilde{d}^{(v)} ||\mathbf{x}^{(v)} - F^{(v)}\mathbf{e}_j||_2^2\ . \tag{13}$$

The third step is fixing $F^{(v)}$, G, $\alpha^{(v)}$ and updating $D^{(v)}$ by Eq. (5) and Eq. (6).
The fourth step is fixing $F^{(v)}$, G, $D^{(v)}$ and updating $\alpha^{(v)}$.

$$\min_{\alpha^{(v)}} \sum_{v=1}^{M} (a^{(v)})^{\gamma} Tr\{H^{(v)}\}, \ s.t. \ \sum_{v=1}^{M} \alpha^{(v)} = 1, \ \ \alpha^{(v)} \geq 0 \tag{14}$$

where $H^{(v)}$ is also defined in Eq. (4). Thus, the Lagrange function of Eq. (14) is:

$$\sum_{v=1}^{M} (\alpha^{(v)})^{\gamma} H^{(v)} - \lambda(\sum_{v=1}^{M} \alpha^{(v)} - 1). \tag{15}$$

In order to get the optimal solution of the above subproblem, set the derivative of Eq. (15) with respect to $\alpha^{(v)}$ to zero. We have:

$$\alpha^{(v)} = \left(\frac{\lambda}{\gamma H^{(v)}}\right)^{\frac{1}{\gamma - 1}}\ . \tag{16}$$

Substitute the resultant $\alpha^{(v)}$ in Eq. (16) into the constraint $\sum_{v=1}^{M} \alpha^{(v)} = 1$, we get:

$$\alpha^{(v)} = \frac{\left(\gamma H^{(v)}\right)^{\frac{1}{1-\gamma}}}{\sum_{v=1}^{M} \left(\gamma H^{(v)}\right)^{\frac{1}{1-\gamma}}}\ . \tag{17}$$

By the above four steps, we alternatively update $F^{(v)}$, G, $D^{(v)}$ as well as $\alpha^{(v)}$ and repeat the process iteratively until the objective function becomes converged. We summarize the proposed algorithm in Alg. 1.

3.2 Discussion of The Parameter γ

We use one parameter γ to control the distribution of weight factors for different views. From Eq. (17), we can see that when $\gamma \to \infty$, we will get equal weight factors. And when $\gamma \to 1$, we will assign 1 to the weight factor of the view whose $H^{(v)}$ value is the smallest and assign 0 to the weights of the other views. Using such a kind of strategy, on one hand, we avoid the trivial solution to the weight distribution of the different views, that is, the solution when $\gamma \to 1$. On the other hand, surprisingly, we can take advantage of only one parameter γ to control the whole weights, reducing the parameters of the model greatly.

Algorithm 1 The algorithm of RMKMC

```
Input:
1. Data for $M$ views $\{X^{(1)}, \cdots, X^{(M)}\}$ and $X^{(v)} \in \mathbb{R}^{d_v \times n}$.
2. The expected number of clusters $K$.
3. The parameter $\gamma$.
Output:
1. The common cluster indicator matrix $G$
2. The cluster centroid matrix $F_{(v)}$ for each view.
3. The learned weight $\alpha^{(v)}$ for each view.
Initialization:
1. Set $t = 0$
2. Initialize the common cluster indicator matrix $G \in \mathbb{R}^{n \times K}$ randomly, such that $G$ satisfies the 1-of-K coding scheme.
3. Initialize the diagonal matrix $D^{(v)} = I_n$ for each view, where $I_n \in \mathbb{R}^{n \times n}$ is the identity matrix.
4. Initialize the weight factor $\alpha^{(v)} = \frac{1}{M}$ for each view.
repeat
    1. Calculate the diagonal matrix $\widetilde{D}^{(v)}$ by Eq. (8)
    2. Update the centroid matrix $F_{(v)}$ for each view by Eq. (9)
    3. Update the cluster indicator vector $\mathbf{g}$ for each data one by one via Eq. (12) and Eq. (13)
    4. Update the diagonal matrix $D^{(v)}$ for each view by Eq. (5) and Eq. (6)
    5. Update the weight factor $\alpha^{(v)}$ for each view by Eq. (17)
    6. Update $t = t + 1$
until Converges
```

3.3 Convergence Analysis

We can prove the convergence of the proposed Alg. 1 as follows: We can divide the Eq. (2) into four subproblems and each of them is a convex problem with respect to one variable. Therefore, by solving the subproblems alternatively, our proposed algorithm will guarantee that we can find the optimal solution to each subproblem and finally, the algorithm will converge to local solution.

4 Time Complexity Analysis

As we know, graph based clustering methods, like spectral clustering and etc., will involve heavy computation, *e.g.* kernel/affinity matrix construction as well as eigen-decomposition. For the data set with n images, the above two calculations will have the time complexity of $O(n^2)$ and $O(n^3)$ respectively, which makes them impractical for solving the large-scale image clustering problem. Although some research works have been proposed to to reduce the computational cost of the eigen-decomposition of the graph Laplacian [Yan *et al.*, 2009] [Sakai and Imiya, 2009], they are designed for two-way clustering and have to use the hierarchical scheme to tackle the multi-way clustering problem.

However, our proposed method is centroid based clustering method with the similar time complexity as traditional K-means. For K-means clustering, if the number of iteration is P, then the time complexity is $O(PKnd)$ and the time complexity of our proposed method is $O(PKndM)$, where M is the number of views and usually $P \ll n$, $M \ll n$ and $K \ll n$. In addition, in the real implementation, if the data is too big to store them in memory, we can extend our algorithm as an external memory algorithm that works on a chunk of data at a time and iterate the proposed algorithm on each data chunk in parallel if multiple processors are available. Once all of the data chunks have been processed, the cluster centroid matrix will be updated. Therefore, our proposed method can be used to tackle the very large-scale clustering problem.

Table 1: Data set summary.

Data sets	# of data	# of views	# of cluster
SensIT	300	2	3
Caltech7	441	6	7
MSRC-v1	210	6	7
Digit	2000	6	10
AwA	30475	6	50
SUN	10000	7	100

Because the graph based multi-view clustering methods cannot be applied to the large-scale image clustering, we did not compare the performance of our method with them in the experiments.

5 Experiments

In this section, we will evaluate the performance of the proposed RMKMC method on six benchmark data sets: SensIT Vehicle [Duarte and Hu, 2004], Caltech-101 [Li *et al.*, 2007], Microsoft Research Cambridge Volume 1(MSRC-v1) [Winn and Jojic, 2005] Handwritten numerals [Frank and Asuncion, 2010], Animal with attribute [Lampert *et al.*, 2009] and SUN 397 [Xiao *et al.*, 2010]. Three standard clustering evaluation metrics are used to measure the multi-view clustering performance, that is, Clustering Accuracy (ACC), Normalized Mutual Information(NMI) and Purity.

5.1 Data Set Descriptions

We summarize the six data sets that we will use in our experiments in Table 1.

SensIT Vehicle data set is the one from wireless distributed sensor networks (WDSN). It utilizes two different sensors, that is, acoustic and seismic sensor to record different signals and do classification for three types of vehicle in an intelligent transportation system. We download the processed data from LIBSVM [Chang and Lin, 2011] and randomly sample 100 data for each class. Therefore, we have 300 data samples, 2 views and 3 classes.

Caltech101 data set is an object recognition data set containing 8677 images, belonging to 101 categories. We chose the widely used 7 classes, *i.e.* Faces, Motorbikes, Dolla-Bill, Garfield, Snoopy, Stop-Sign and Windsor-Chair. Following [Dueck and Frey, 2007], we sample the data and totally we have 441 images. In order to get the different views, we extract LBP [Ojala *et al.*, 2002] with dimension 256, HOG [Dalal and Triggs, 2005] with dimension 100, GIST [Oliva and Torralba, 2001] with dimension 512 and color moment (CMT) [Yu *et al.*, 2002] with dimension 48, CENTRIST [Wu and Rehg, 2008] with dimension 1302 and DoG-SIF [Lowe, 2004] with dimension 128 visual features from each image.

MSRC-v1 data set is a scene recognition data set containing 8 classes, 240 images in total. Following [Lee and Grauman, 2009], we select 7 classes composed of tree, building, airplane, cow, face, car, bicycle and each class has 30 images. We also extract the same 6 visual features from each image with Caltech101 dataset.

Handwritten numerals data set consists of 2000 data points for 0 to 9 ten digit classes. (Each class has 200 data points.) We use the published 6 features to do multi-view clustering. Specifically, these 6 features are 76 Fourier coefficients of the character shapes (FOU), 216 profile correlations (FAC), 64 Karhunen-love coefficients (KAR), 240 pixel averages in 2×3 windows (PIX), 47 Zernike moment (ZER) and 6 morphological (MOR) features.

Animal with attributes is a large-scale data set, which consists of 6 feature, 50 classes, 30475 samples. We utilize all the published features for all the images, that is, Color Histogram (CQ) features , Local Self-Similarity (LSS) features [Shechtman and Irani, 2007], PyramidHOG (PHOG) features [Bosch *et al.*, 2007], SIFT features [Lowe, 2004], colorSIFT (RGSIFT) features [van de Sande *et al.*, 2008], and SURF features [Bay *et al.*, 2008].

SUN 397 dataset [Xiao *et al.*, 2010] is a published dataset to provide researchers in computer vision, human perception, cognition and neuroscience, machine learning and data mining, with a comprehensive collection of annotated images covering a large variety of environmental scenes, places and the objects. It consists of 397 classes with 100 images for each class. We conduct the clustering experiment on the top 100 classes via the 7 published features for all the 10000 images.The 7 visual features are color moment, dense SIFT, GIST, HOG, LBP, MAP and TEXTON.

5.2 Experimental Setup

We will compare the multi-view clustering performance of our method (RMKMC) with their corresponding single-view counterpart. In addition, we also compare the results of our method with the baseline method naive multi-view K-means clustering (NKMC), and affinity propagation (AP). In our method, when we ignore the weight learning for each type of visual features, the method degenerates to a simple version, called as simple MKMC (SMKMC). In order to see the importance of the weight learning, we also compare our method to this simple version method.

Before we do any clustering, for each type of features, we normalize the data first, making all the values in the range $[-1, 1]$. When we implement naive multi-view K-means, we simply use the concatenated normalized features as input for the classic K-means clustering algorithm. As for affinity propagation methods, we need to build the similarity kernel first. Due to the fact that linear kernel is preferred in large-scale problem, we use the following way to construct linear kernel.

$$w_{ij} = \mathbf{x}_i^T \mathbf{x}_j, \quad \forall i, j = 1, 2, ..., n, \tag{18}$$

Table 2: SensIT Vehicle data set

Methods	ACC	NMI	Purity
acoustic	0.5049 ± 0.030	0.1018 ± 0.023	0.5055 ± 0.029
seismic	0.5122 ± 0.047	0.1149 ± 0.046	0.5129 ± 0.046
NKMC	0.5449 ± 0.041	0.1375 ± 0.030	0.5465 ± 0.039
AP	0.3867 ± 0.000	0.0084 ± 0.000	0.3867 ± 0.000
SMKMC	0.5490 ± 0.040	0.1395 ± 0.032	0.5494 ± 0.040
RMKMC	$\mathbf{0.5504} \pm 0.049$	$\mathbf{0.1484} \pm 0.033$	$\mathbf{0.5542} \pm 0.044$

Table 3: Caltech101-7 data set.

Methods	ACC	NMI	Purity
LBP	0.5236 ± 0.021	0.4319 ± 0.006	0.6005 ± 0.008
HOG	0.5561 ± 0.052	0.5020 ± 0.035	0.6459 ± 0.038
GIST	0.5663 ± 0.032	0.4737 ± 0.024	0.6418 ± 0.028
CMT	0.3809 ± 0.015	0.2706 ± 0.021	0.4346 ± 0.010
DoG-SIFT	0.6125 ± 0.037	0.5637 ± 0.018	0.6673 ± 0.028
CENTRIST	0.6315 ± 0.058	0.5981 ± 0.046	0.7035 ± 0.044
NKMC	0.6587 ± 0.063	0.6561 ± 0.035	0.7458 ± 0.030
AP	0.5125 ± 0.000	0.3611 ± 0.1054	0.5170 ± 0.1290
SMKMC	0.6723 ± 0.058	0.6775 ± 0.034	0.7561 ± 0.026
RMKMC	$\mathbf{0.6797} \pm 0.053$	$\mathbf{0.6892} \pm 0.029$	$\mathbf{0.7595} \pm 0.027$

In addition, RMKMC has a parameter γ to control the weight factor distribution among all views. We search the logarithm of the parameter γ, that is, $log_{10\gamma}$ in the range from 0.1 to 2 with incremental step 0.2 to get the best parameters γ^*. Since all the clustering algorithms depend on the initializations, we repeat all the methods 50 times using random initialization and report the average performance.

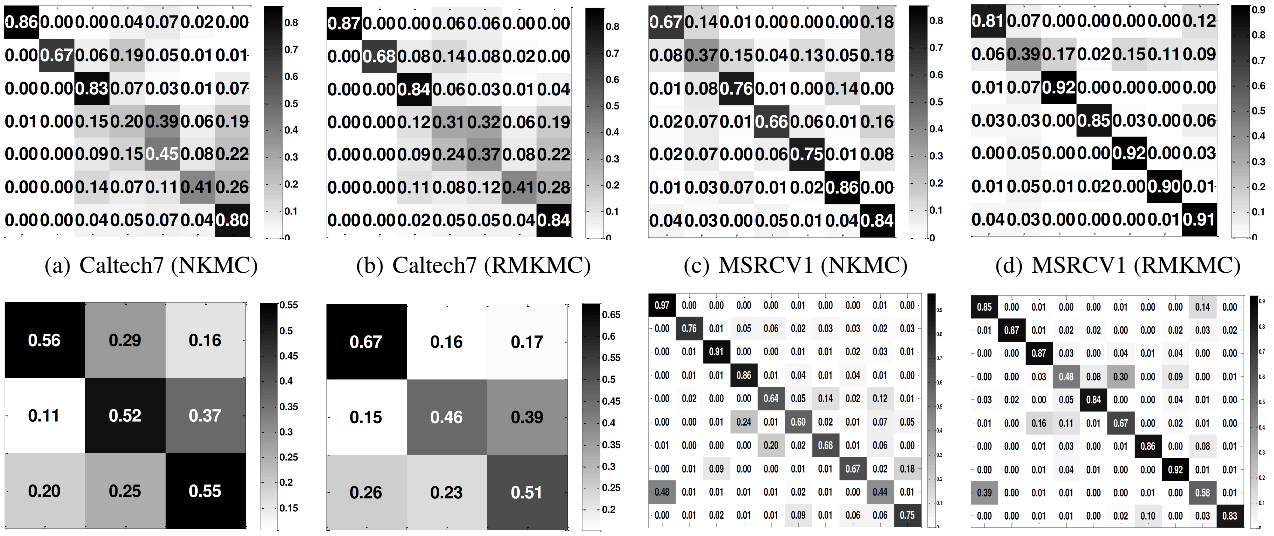
5.3 Clustering Results Comparisons

Table 2 demonstrates the clustering results on SensIT Vehicle data set. From it, we can see that although there are only two views (acoustic and seismic), compared with single-view K-means counterparts, our proposed RMKMC can boost the clustering performance by more than 10%. Our RMKMC can also beat NKMC and AP. Table 3 and Table 5 show the clustering results on regular size Caltech101-7, MSRC-v1 as well as Handwritten numerals data set. From it, we can see that with more feature views involved in, our method can improve the clustering performance even further. Also, on large-scale data set Animal with attribute, although doing clustering on a 50 class data set is hard, the performance of our method can still outperform that of the other compared methods as shown in Table 6.

We plot the confusion matrices of RMKMC and NKMC in terms of clustering accuracy in Fig. 1. Because the clustering numbers of AwA and SUN data sets are large, their confusion matrices cannot be plotted within one page. We skip these two figures. From both tables and figures, we can see that our proposed methods consistently beat the base line method on all the data sets.

6 Conclusion

In this paper, we proposed a novel robust multi-view K-means clustering methods to tackle the large-scale multi-view



(a) Caltech7 (NKMC) (b) Caltech7 (RMKMC) (c) MSRCV1 (NKMC) (d) MSRCV1 (RMKMC)

(e) sensIT (NKMC) (f) sensIT (RMKMC) (g) digit(NKMC) (h) digit(RMKMC)

Figure 1: The calculated average clustering accuracy confusion matrix for Caltech101, MSRCV1, SensIT Vehicle, and Handwritten numerals data sets.

Table 4: MSRC-v1 data set.

Methods	ACC	NMI	Purity
LBP	0.4726 ± 0.039	0.4156 ± 0.024	0.5087 ± 0.030
HOG	0.6361 ± 0.041	0.5669 ± 0.032	0.6610 ± 0.037
GIST	0.6283 ± 0.057	0.5523 ± 0.039	0.6511 ± 0.044
CMT	0.5076 ± 0.043	0.4406 ± 0.037	0.5307 ± 0.037
DoG-SIFT	0.4341 ± 0.036	0.3026 ± 0.028	0.4558 ± 0.030
CENTRIST	0.5977 ± 0.062	0.5301 ± 0.037	0.6205 ± 0.054
NKMC	0.7002 ± 0.085	0.6405 ± 0.057	0.7207 ± 0.073
AP	0.1571 ± 0.000	0.2890 ± 0.000	0.1714 ± 0.000
SMKMC	0.7423 ± 0.093	0.6940 ± 0.070	0.7652 ± 0.079
RMKMC	**0.8142** ± 0.087	**0.7776** ± 0.071	**0.8341** ± 0.073

Table 5: Handwritten numerals data set.

Methods	ACC	NMI	Purity
FOU	0.5560 ± 0.062	0.5477 ± 0.028	0.5793 ± 0.048
FAC	0.7078 ± 0.065	0.6791 ± 0.032	0.7374 ± 0.051
KAR	0.6898 ± 0.051	0.6662 ± 0.030	0.7149 ± 0.044
MOR	0.6143 ± 0.058	0.6437 ± 0.034	0.6428 ± 0.050
PIX	0.6945 ± 0.067	0.7030 ± 0.040	0.7235 ± 0.059
ZER	0.5348 ± 0.052	0.5123 ± 0.025	0.5684 ± 0.043
NKMC	0.7282 ± 0.067	0.7393 ± 0.039	0.7609 ± 0.059
AP	0.6285 ± 0.000	0.5940 ± 0.000	0.6600 ± 0.000
SMKMC	0.7758 ± 0.079	0.7926 ± 0.039	0.8106 ± 0.060
RMKMC	**0.7889** ± 0.075	**0.8070** ± 0.033	**0.8247** ± 0.052

Table 6: Animal with attribute data set.

Methods	ACC	NMI	Purity
CP	0.0675 ± 0.002	0.0773 ± 0.003	0.0874 ± 0.002
LSS	0.0719 ± 0.002	0.0819 ± 0.005	0.0887 ± 0.002
PHOG	0.0690 ± 0.004	0.0691 ± 0.003	0.0823 ± 0.004
RGSIFT	0.0725 ± 0.003	0.0862 ± 0.004	0.0889 ± 0.003
SIFT	0.0732 ± 0.003	0.0944 ± 0.005	0.0919 ± 0.004
SURF	0.0764 ± 0.003	0.0885 ± 0.003	0.0978 ± 0.004
NKMC	0.0802 ± 0.001	0.1075 ± 0.003	0.1007 ± 0.001
AP	0.0769 ± 0.001	0.0793 ± 0.003	0.0975 ± 0.001
SMKMC	0.0841 ± 0.005	0.1108 ± 0.005	0.1039 ± 0.005
RMKMC	**0.0943** ± 0.005	**0.1174** ± 0.005	**0.1140** ± 0.005

Table 7: SUN data set.

Methods	ACC	NMI	Purity
COLOR	0.0507 ± 0.003	0.1417 ± 0.003	0.0544 ± 0.003
DSIFT	0.0661 ± 0.002	0.1717 ± 0.002	0.0710 ± 0.002
GIST	0.0740 ± 0.002	0.2008 ± 0.002	0.0812 ± 0.004
HOG	0.0715 ± 0.003	0.1862 ± 0.003	0.0772 ± 0.003
LBP	0.0599 ± 0.002	0.1618 ± 0.002	0.0644 ± 0.002
MAP	0.0656 ± 0.003	0.1917 ± 0.003	0.0710 ± 0.004
TEXTON	0.0561 ± 0.002	0.1682 ± 0.002	0.0608 ± 0.002
NKMC	0.0546 ± 0.001	0.1507 ± 0.003	0.0591 ± 0.001
AP	0.0667 ± 0.001	0.1693 ± 0.003	0.0765 ± 0.001
SMKMC	0.0834 ± 0.003	0.2106 ± 0.003	0.0839 ± 0.003
RMKMC	**0.0927** ± 0.003	**0.2154** ± 0.003	**0.0922** ± 0.003

clustering problems. Utilizing the common cluster indicator, we can search a consensus pattern and do clustering across multiple visual feature views. Moreover, by imposing the structured sparsity $\ell_{2,1}$-norm on the objective function, our method is robust to the outliers in input data. Our new method learns the weights of each view adaptively. We also introduce an optimization algorithm to iteratively and efficiently solve the proposed non-smooth objective with proved convergence. We evaluate the performance of our methods on six multi-view clustering data sets.

References

[Bay *et al.*, 2008] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc J. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[Biswas and Jacobs, 2012] Arijit Biswas and David Jacobs. Active Image Clustering: Seeking Constraints from Humans to Complement Algorithms. *CVPR*, pages 2152–2159, 2012.

[Bosch *et al.*, 2007] Anna Bosch, Andrew Zisserman, and Xavier Muñoz. Representing shape with a spatial pyramid kernel. In *CIVR*, pages 401–408, 2007.

[Cai *et al.*, 2011] Xiao Cai, Feiping Nie, Heng Huang, and Farhad Kamangar. Heterogeneous image features integration via multi-modal spectral clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, pages 1977–1984, 2011.

[Chang and Lin, 2011] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM TIST*, 2(3):27, 2011.

[Dalal and Triggs, 2005] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR (1)*, pages 886–893, 2005.

[Ding *et al.*, 2005] Chris H. Q. Ding, Xiaofeng He, and Horst D. Simon. Nonnegative lagrangian relaxation of *k*-means and spectral clustering. In *ECML*, pages 530–538, 2005.

[Duarte and Hu, 2004] Marco F. Duarte and Yu Hen Hu. Vehicle classification in distributed sensor networks. *J. Parallel Distrib. Comput.*, 64(7):826–838, 2004.

[Dueck and Frey, 2007] Delbert Dueck and Brendan J. Frey. Non-metric affinity propagation for unsupervised image categorization. In *ICCV*, pages 1–8, 2007.

[Frank and Asuncion, 2010] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[Lampert *et al.*, 2009] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958, 2009.

[Lee and Grauman, 2009] Yong Jae Lee and Kristen Grauman. Foreground focus: Unsupervised learning from partially matching images. *International Journal of Computer Vision*, 85(2):143–166, 2009.

[Li *et al.*, 2007] Fei-Fei Li, Robert Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

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

[Ojala *et al.*, 2002] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):971–987, 2002.

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

[Sakai and Imiya, 2009] Tomoya Sakai and Atsushi Imiya. Fast spectral clustering with random projection and sampling. In *MLDM*, pages 372–384, 2009.

[Shechtman and Irani, 2007] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *CVPR*, 2007.

[van de Sande *et al.*, 2008] Koen E. A. van de Sande, Theo Gevers, and Cees G. M. Snoek. Evaluation of color descriptors for object and scene recognition. In *CVPR*, 2008.

[Winn and Jojic, 2005] John M. Winn and Nebojsa Jojic. Locus: Learning object classes with unsupervised segmentation. In *ICCV*, pages 756–763, 2005.

[Wu and Rehg, 2008] Jianxin Wu and James M. Rehg. Where am i: Place instance and category recognition using spatial pact. In *CVPR*, 2008.

[Xiao *et al.*, 2010] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pages 3485–3492, 2010.

[Yan *et al.*, 2009] Donghui Yan, Ling Huang, and Michael I. Jordan. Fast approximate spectral clustering. In *KDD*, pages 907–916, 2009.

[Yu *et al.*, 2002] Hui Yu, Mingjing Li, HongJiang Zhang, and Jufu Feng. Color texture moments for content-based image retrieval. In *ICIP*, pages 929–932, 2002.