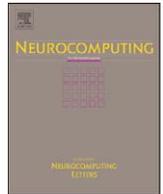




Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

On defining affinity graph for spectral clustering through ranking on manifolds

Tian Xia^{a,b,*}, Juan Cao^a, Yong-dong Zhang^a, Jin-tao Li^a

^a Center for Advanced Computing Technology Research, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China

^b Graduate University of the Chinese Academy of Sciences, Beijing 100039, China

ARTICLE INFO

Article history:

Received 3 November 2008

Received in revised form

20 January 2009

Accepted 1 March 2009

Communicated by D. Tao

Keywords:

Affinity graph

Spectral clustering

Ranking on manifolds

ABSTRACT

Spectral clustering consists of two distinct stages: (a) construct an affinity graph from the dataset and (b) cluster the data points through finding an optimal partition of the affinity graph. The focus of the paper is the first step. Existing spectral clustering algorithms adopt Gaussian function to define the affinity graph since it is easy to implement. However, Gaussian function is hard to depict the intrinsic structure of the data, and it has to specify a scaling parameter whose selection is still an open issue in spectral clustering. Therefore, we propose a new definition of affinity graph for spectral clustering from the graph partition perspective. In particular, we propose two consistencies: smooth consistency and constraint consistency, for affinity graph to hold, and then define the affinity graph respecting these consistencies in a regularization framework of ranking on manifolds. Meanwhile the proposed definition of affinity graph is applicable to both unsupervised and semi-supervised spectral clustering. Encouraging experimental results on synthetic and real world data demonstrate the effectiveness of the proposed approach.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

In recent years, spectral clustering has become one of the most popular modern clustering algorithms. It clusters data points through performing spectral analysis on the matrix derived from the data. Specifically, spectral clustering consists of two distinct stages: (a) construct an affinity graph from the dataset and (b) cluster the data points through finding an optimal partition of the affinity graph. Although a great deal of effort has been carried out addressing the latter, little progress has been made on defining affinity graph, whereas it encodes the intrinsic structure of the data, and plays an important role in spectral clustering.

Existing spectral clustering algorithms [2–5] adopt Gaussian function, i.e., $A(x_i, x_j) = \exp(-\|x_i - x_j\|^2 / 2\sigma^2)$, to define the affinity graph, since it is simple to implement. However, Gaussian function has a scaling parameter σ to be specified manually, and its selection is still an open issue in spectral clustering. In practice, σ is often set by an empirical value, such as σ is set as 0.05 of the maximal pairwise Euclidean distance among the dataset in normalized cut algorithm (NC) which is a representative spectral clustering algorithm [2,6]. This setting of σ makes spectral clustering be very sensitive to outliers. Manor et al. propose to

use a local scale rather than a global one for Gaussian function [7]; however, this algorithm has limited success on real world data although it works well on synthetic data.

More important, Gaussian function is hard to depict the intrinsic structure of the data from graph partition perspective, and makes spectral clustering perform badly on some data distributed in complex shape. To illustrate this, let us consider a toy example on two-moon data as shown in Fig. 1(a). The affinity graph constructed in NC is shown in Fig. 1(b), in the form of K-nearest neighborhood (KNN) graph. We can see that some data pairs distributed on separate moons are also linked in the affinity graph; it implies some wrong local neighborhood relationships, and thus the clustering result of NC is somehow biased as shown in Fig. 1(c). We also illustrate the example in semi-supervised case: four constraints including two must-link constraints and two cannot-link constraints are added as shown in Fig. 1(d); the affinity graph as shown in Fig. 1(e) is constructed through a representative distance metric learning method proposed in [8]; we can see that some very near neighbors are not linked in the affinity graph, and thus it produces bad clustering result as shown in Fig. 1(f). From this toy example, we can observe: (a) the success of spectral clustering depends greatly on the constructed affinity graph which encodes the intrinsic structure of the data; and (b) Gaussian function based affinity graph still has problems to depict the intrinsic structure of some data distributed in complex shape.

In this paper, we focus on the definition of affinity graph for spectral clustering in both unsupervised and semi-supervised

* Corresponding author at: Center for Advanced Computing Technology Research, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China.

E-mail address: txia@ict.ac.cn (T. Xia).

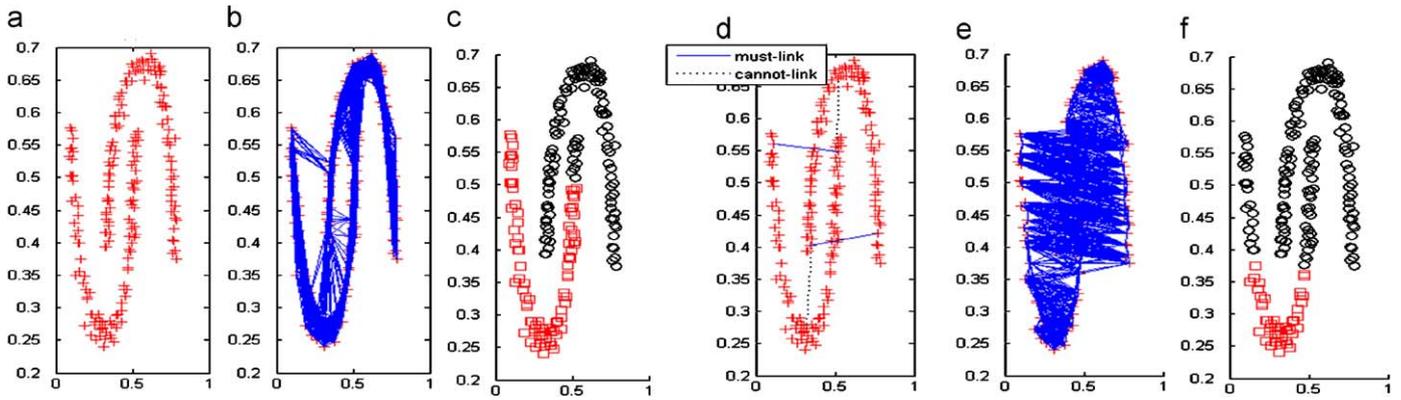


Fig. 1. (a) Two-moon dataset. (b) KNN graph based on Gaussian affinity graph when $K = 30$. (c) Clustering result of NC algorithm. (d) Two-moon dataset with two must-link constraints and two cannot-link constraints. (e) KNN graph based on the affinity graph learned through the distance metric learning method in [8], when $K = 20$. (f) Clustering result of NC algorithm based on the affinity graph learned through the distance metric learning method in [8].

cases. Unlike the methodology of finding optimal parameter for Gaussian function [7], we try to define the affinity graph from graph partition perspective. We deem that it is advantageous for spectral clustering, if the intrinsic structure induced by the affinity graph is helpful to a “meaningful” graph partition. In this way, we propose two consistencies: smooth consistency and constraint consistency, for affinity graph to hold, and then define the affinity graph respecting these consistencies in a regularization framework of ranking on manifolds (RoM). RoM is a consistency method widely used in semi-supervised learning [9,10], and also in unsupervised clustering [11]. Encouraging experimental results on synthetic and real data demonstrate the effectiveness of the proposed approach.

To summarize, the main contributions of this paper are twofold:

- (1) Propose to define the affinity graph from the graph partition perspective for spectral clustering in both unsupervised and semi-supervised cases. And two consistencies, i.e., smooth consistency and constraint consistency, are proposed for the affinity graph to hold.
- (2) Propose to define the affinity graph respecting the proposed two consistencies in a regularization framework based on RoM.

The rest of the paper is organized as follows. In Section 2, we provide a short review on spectral clustering. In Section 3, we present our definition of affinity graph for spectral clustering, i.e., RoM-AG. In Section 4, we give the experimental results. Finally, we give the conclusion and some discussions in Section 5.

2. Brief review of spectral clustering

Spectral clustering is a method that can cluster data points through performing spectral analysis on the affinity graph derived from the data. Traditional spectral clustering is an unsupervised algorithm. Besides that, spectral clustering also has developed semi-supervised algorithms, on which only some preliminary effort is known to us.

2.1. Unsupervised spectral clustering

Much of the research work on spectral clustering has focused on unsupervised case, in which no more information besides the data itself is provided. Spectral clustering clusters the data points through finding an optimal partition of the affinity graph

according to some criterion, and different implementations vary with different criteria [1], e.g., graph cut [2,3], random walks [4] and perturbation theory [5].

In this paper, we select the spectral clustering algorithm proposed by Yu et al. [3] to evaluate the performance of the proposed definition of affinity graph. This algorithm is derived from graph cut point of view. It has a complete optimization framework, and thus is more stable than other spectral clustering algorithms. We denote this algorithm as multiclass spectral clustering (MSC). The details of MSC is presented as follows:

Algorithm: MSC

Input: Set of data points $\chi = \{x_1, \dots, x_n\} \subset \mathbb{R}^d$, number of clusters K .

Output: Disjoint K partitioning $\{\chi_i\}_{i=1}^K$.

Method:

1. Construct the affinity matrix A using Gaussian function

$$A(x_i, x_j) = \exp(-\|x_i - x_j\|^2 / 2\sigma^2).$$
2. Compute the largest K eigenvectors v_1, \dots, v_K of the eigenproblem $D^{-1}Av = \lambda v$ in which D is a diagonal matrix with the sum of the i -th row of A on the diagonal.
3. Perform optimal discretization over the K eigenvectors.

We can see that MSC adopts Gaussian function to define the affinity graph. In this paper, we propose an improved algorithm of MSC, i.e., RoM-MSC, in which the affinity graph is defined from RoM.

2.2. Semi-supervised spectral clustering

Compared to unsupervised clustering, some supervision in the form of pairwise constraints is provided in semi-supervised clustering. Specially, two types of pairwise constraints proposed in [12] are often considered: must-link pair which specifies two samples should be in the same cluster; and cannot-link pair which specifies two samples should be in different clusters. Semi-supervised clustering usually has two general approaches called constraint-based and metric-based [13]. In constraint-based methods, the clustering algorithm itself is modified by adding constraints to the clustering objective function, and enforces constraints during the clustering process [14]. While in metric-based methods, the clustering algorithm itself is not modified, and only the distance metric is first learned from the constraints [15].

Although a great deal of effort has been carried on semi-supervised clustering, only some preliminary effort on semi-supervised spectral clustering is known to us. Yu et al. propose constrained spectral clustering (CSC) in [16]. CSC focuses on must-link constraints which are added to the optimization

problem of MSC as constraints. Therefore, it is a constraint-based semi-supervised clustering method. Kamvar et al. propose a metric-based semi-supervised spectral clustering algorithm in [17]. It utilizes Gaussian function to generate the affinity matrix, and then simply sets to 1 for entries of must-link pairs, to 0 for entries of cannot-link pairs. Not surprisingly, this method generally does not work very well since the effect of pairwise constraints is not propagated over all the data. Lu et al. propose to use affinity propagation to propagate the pairwise constraints for semi-supervised spectral clustering in [18].

In this paper, we extend MSC to semi-supervised case, and propose a metric-based semi-supervised spectral clustering algorithm, i.e., SRoM-MSC, which propagates the must-link constraints through RoM.

3. Definition of affinity graph for spectral clustering through RoM

In this section, we present the proposed definition of affinity graph for spectral clustering. Firstly, we give a short introduction to ranking on manifolds, from which we propose two assumptions of consistency for affinity graph to hold, and then define the affinity graph respecting these consistencies in a regularization framework of RoM. We denote the proposed definition of affinity graph as RoM-AG. Finally, two spectral clustering algorithms are proposed based on RoM-AG for unsupervised and semi-supervised cases, respectively.

3.1. Ranking on manifolds

Ranking on manifolds is widely used as a semi-supervised learning method which tries to learn a very smooth solution with respect to the intrinsic structure of the data [9,10]. One unsupervised clustering method through RoM is also proposed in [11]. In this paper, RoM is used to define affinity graph for spectral clustering, both in unsupervised and semi-supervised cases. Below is a brief summary of RoM.

Given a set of points $\chi = \{x_1, \dots, x_q, x_{q+1}, \dots, x_n\} \subset \mathbb{R}^d$, the first q points are the queries and the rest are the points that we want to rank according to their relevance to the queries. Let $f: \chi \rightarrow \mathbb{R}$ denote a ranking vector where each point x_i has a ranking value f_i , that is $f = [f_1, \dots, f_n]^T$. Define $y = [y_1, \dots, y_n]^T$, in which $y_i = 1$ if x_i is a query, and $y_i = 0$ otherwise. The entire algorithm of RoM is as follows:

1. Compute the affinity matrix $W(x_i, x_j) = \exp(-\|x_i - x_j\|^2 / 2\sigma^2)$, if $i \neq j$ and 0 otherwise.
2. Compute $S = D^{-1/2} W D^{-1/2}$ with $D_{ii} = \sum_{j=1}^n W_{ij}$ and $D_{ij} = 0$, $i \neq j$.
3. Iterate $f(t+1) = \alpha S f(t) + (1 - \alpha)y$ until convergence, $\alpha \in [0, 1)$.

Let f^* denote the limit of the sequence $\{f(t)\}$, the point x_i can be ranked according to its ranking score f_i^* . f^* also has a closed form, $f^* = (I - \alpha S)^{-1}y$. The iteration step in RoM can be understood as all points spreading their ranking scores to their neighbors via a weighted graph.

3.2. Assumptions of consistency for affinity graph

Since spectral clustering clusters data points through finding an optimal partition of the affinity graph, we deem that the definition of affinity graph for spectral clustering should be from the graph partition point of view. It means that not only the entry in affinity graph reflects the pairwise similarity, but also the intrinsic structure induced by the affinity graph should be helpful

to a “meaningful” graph partition, although spectral clustering algorithm itself is assumed to have some capability to find a “meaningful” graph partition.

In this way, we propose two assumptions of consistency for affinity graph to hold:

- *Smooth consistency*: nearby points are likely to have the similar distances on the manifold to all other points.
- *Constraint consistency*: must-link pair is likely to have relatively large affinity.

The smooth consistency assumes two points to be identical if they both have identical distances on the manifold to all other points. It measures the affinity of two points by taking their neighborhoods into account. The constraint consistency focuses on the effect of must-link constraint. Both of the two consistencies imply the intrinsic structure of the data. We will derive the proposed definition of affinity graph (RoM-AG) respecting these two consistencies in the next section.

3.3. Regularization framework

As aforementioned in Section 3.1, the learned ranking vector $f^* = [f_1, \dots, f_n]^T$ in RoM denotes the ranking scores of all points to the query point (the largest score ranks first). It can also be understood as an affinity vector which reflects the similarities between the query point and all the points. In this way, we extend the framework of RoM to define the affinity graph respecting the above two consistencies.

Given a set of points $\chi = \{x_1, \dots, x_n\} \subset \mathbb{R}^d$, let matrix $A \in \mathbb{R}^{n \times n}$ denote the affinity graph for spectral clustering derived on χ , and A_i denote the i -th row vector of A , A_i can be regarded as the ranking vector with x_i as the query point in RoM's framework. Define matrix $Y \in \mathbb{R}^{n \times n}$, and Y_i as the i -th row vector of Y , Y_i^T can be regarded as the vector y in RoM with x_i as the query point, thus $Y = I$ in this case (for unsupervised case). The optimal objection is defined as below:

$$\mathbb{C}(A) = \frac{1}{2} \left(\sum_{i,j=1}^n W_{ij} \left\| \frac{1}{\sqrt{D_{ii}}} A_i - \frac{1}{\sqrt{D_{jj}}} A_j \right\|^2 + \mu \sum_{i=1}^n \|A_i - Y_i\|^2 \right), \quad (1)$$

where $W_{ij} = \exp(-\|x_i - x_j\|^2 / 2\sigma^2)$, if $i \neq j$ and 0 otherwise. D is a diagonal matrix with the sum of the i -th row of W on the diagonal. $\mu > 0$ is the regularization factor. Let \mathcal{A} denote the set of $n \times n$ matrices with nonnegative entries, and then the optimal affinity graph is

$$A^* = \arg \min_{A \in \mathcal{A}} \mathbb{C}(A). \quad (2)$$

The first term of the right-hand side in Eq. (1) is the smooth constraint which focuses on the smooth consistency proposed in Section 3.2. It means that nearby points should have similar distances to all other points. The second term is the fitting constraint which focuses on the constraint consistency. It prefers to give a large value to the affinity of must-link pair. Concretely, in unsupervised case, $Y = I$, it means that any point and itself are a must-link pair, this is called self must-link constraint. In semi-supervised case, $Y_{ij} = 1$ if (x_i, x_j) is must-link pair or $i = j$, and $Y_{ij} = 0$ otherwise, this specification respects both real must-link and self must-link constraints. As aforementioned in Section 3.1, RoM can be understood as all points spreading their ranking scores to their neighbors via a weighted graph. In a similar way, the regularization framework in Eq. (1) can be interpreted as all points spreading their affinity to their neighbors via a weighted graph, and it respects the proposed two consistencies simultaneously.

The unconstrained optimization problem in Eq. (2) can be solved as follows:

$$\begin{aligned} \frac{\partial \mathcal{C}}{\partial A} \Big|_{A=A^*} &= A^* - SA^* + \mu(A^* - Y) = 0 \\ \Rightarrow A^* - \frac{1}{1+\mu}SA^* - \frac{\mu}{1+\mu}Y &= 0, \end{aligned} \quad (3)$$

where $S = D^{-1/2}WD^{-1/2}$, let $\alpha = 1/1 + \mu$, $\beta = \mu/1 + \mu$, note that $\alpha + \beta = 1$, then

$$\begin{aligned} (I - \alpha S)A^* &= \beta Y, \\ A^* &= \beta(I - \alpha S)^{-1}Y. \end{aligned} \quad (4)$$

For affinity graph, it is equivalent to

$$A^* = (I - \alpha S)^{-1}Y. \quad (5)$$

Finally, since the affinity graph should be symmetric, we define RoM-AG as

$$A^* = (I - \alpha S)^{-1}Y + ((I - \alpha S)^{-1}Y)^T. \quad (6)$$

RoM-AG is also non-negative, the proof is given as below:

Proof of RoM-AG being non-negative: Since the eigenvalues of S in $[-1, 1]$ and $\alpha \in (0, 1)$, we have the eigenvalues of αS in $(-1, 1)$. Hence

$$\lim_{n \rightarrow \infty} \sum_{t=0}^n (\alpha S)^t = (I - \alpha S)^{-1}. \quad (7)$$

Based on Eq. (7), $(I - \alpha S)^{-1}$ can be regarded as the sum of infinite non-negative matrices, so $(I - \alpha S)^{-1}$ is non-negative. RoM-AG is non-negative as a consequence of that $(I - \alpha S)^{-1}$ and Y both are non-negative.

In order to illustrate the effect of RoM-AG with respect to the two consistencies, two toy examples are given in Fig. 2. Fig. 2(b) illustrates the KNN graph based on RoM-AG of two-moon dataset as shown in Fig. 2(a), Fig. 2(d) illustrates the KNN graph based on RoM-AG of XOR dataset with two must-link constraints as shown

in Fig. 2(c). From Fig. 2, we can see that RoM-AG respects the two consistencies, and reflects the intrinsic structure of the data from the graph partition perspective on the two toy datasets.

3.4. Algorithms

In this section, two spectral clustering algorithms based on RoM-AG are proposed. The first one is an improved algorithm of MSC which adopts RoM-AG as the affinity graph, we denote it as RoM-MSC. RoM-MSC is an unsupervised spectral clustering algorithm.

Algorithm: RoM-MSC

Input: Set of data points $\chi = \{x_1, \dots, x_n\} \subset \mathbb{R}^d$, number of clusters K .

Output: Disjoint K partitioning $\{Z_i\}_{i=1}^K$.

Method:

1. Compute the affinity matrix $A = A^*$ using Eq. (6) with $Y = I$.
2. Perform step 2 and step 3 of MSC.

The second one is SRoM-MSC which is for semi-supervised case, it is summarized as below:

Algorithm: SRoM-MSC

Input: Set of data points $\chi = \{x_1, \dots, x_n\} \subset \mathbb{R}^d$, number of clusters K , set of must-link pairs φ where $(x_i, x_j) \in \varphi$ implies x_i and x_j should be in the same cluster.

Output: Disjoint K partitioning $\{Z_i\}_{i=1}^K$.

Method:

1. Compute the transitive closure of φ , that is φ' , thus $\varphi \subseteq \varphi'$.
2. Construct the $n \times n$ matrix Y , $Y_{ij} = 1$ if $(x_i, x_j) \in \varphi'$ or $i = j$, and $Y_{ij} = 0$ otherwise.
3. Compute the affinity matrix $A = A^*$ using Eq. (6).
4. Perform step 2 and step 3 of MSC.

We can see that based on RoM-AG, unsupervised and semi-supervised spectral clustering algorithms can be unified in one uniform framework; two algorithms have different specifications on Y .

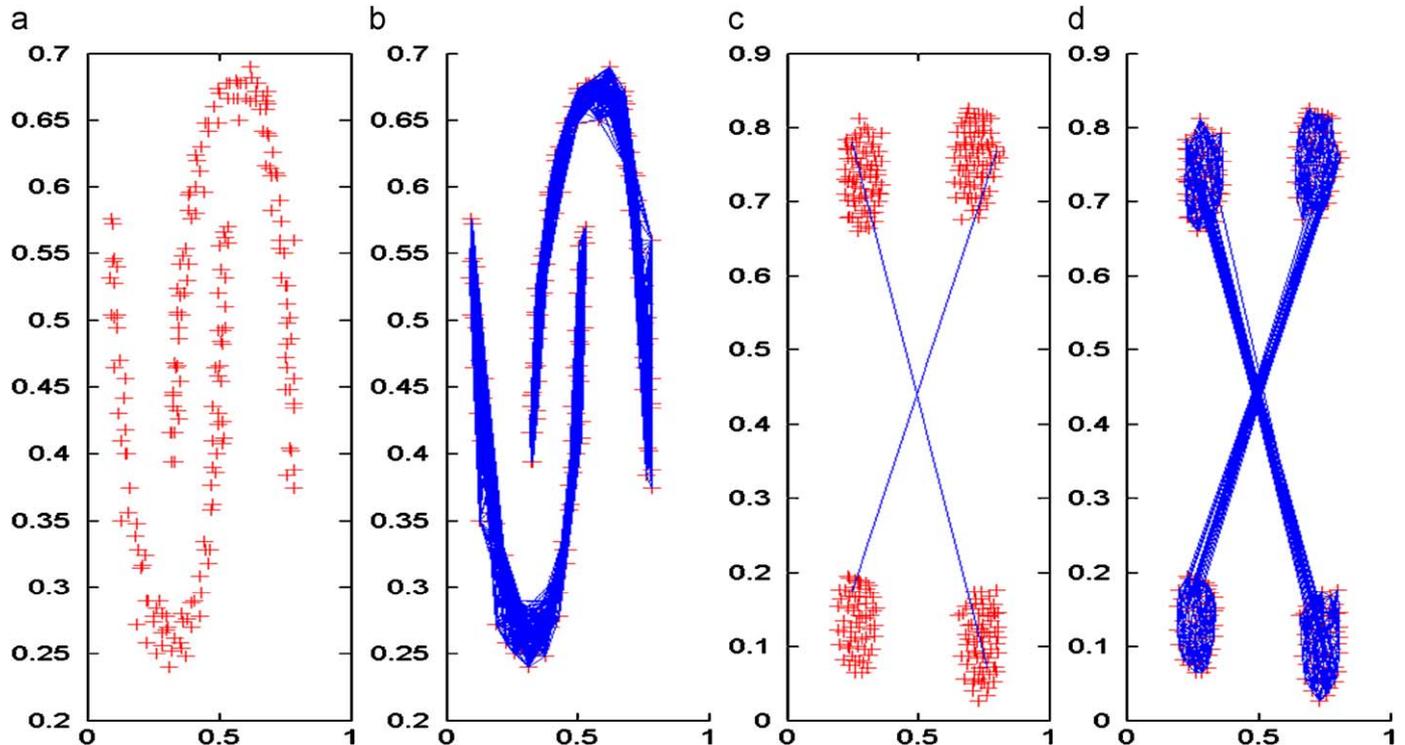


Fig. 2. (a) Two-moon dataset. (b) KNN graph based on RoM-AG when $K = 30$. (c) XOR dataset with two must-link constraints. (d) KNN graph based on RoM-AG when $K = 10$.

3.5. Time complexity analysis

Compared with conventional Gaussian function based affinity graph, RoM-AG has an additional computation of inverting the matrix $I - \alpha S$; it has complexity of $O(n^3)$, where n denotes the number of samples. This is the penalty of RoM-AG for its better performance. In practice, we often compute $(I - \alpha S)^{-1}$ in an iterative process: $A(t+1) = \alpha SA(t) + (1 - \alpha)Y$, $A(0) = Y$; the convergence of this process can be proved directly from the above proof of RoM-AG being non-negative. The iterative solution can reduce computational cost, since it has complexity of $O(n^2T)$, where T denotes the iterative times.

4. Experiments

4.1. Parameters setting for RoM-AG

There are two parameters σ and α to be set in RoM-AG. Parameter σ is simply set by an empirical value as normalized cut algorithm does, i.e., 5% of the maximal pairwise Euclidean distance among the dataset. It is easy to implement, and RoM-AG based spectral clustering algorithm is less sensitive to this parameter than Gaussian affinity graph based spectral clustering algorithm; we will demonstrate it in our experimental results.

Parameter $\alpha \in (0, 1)$; from Eq. (1) and $\alpha = 1/(1 + \mu)$, we can see that the larger the α is, the lesser the affinity graph respects the constraint consistency. There are two cases for the setting of α ; in the case of unsupervised clustering, we set $\alpha = 0.99$ since there is no constraint consistency in this situation, while in the case of semi-supervised clustering, we set it according to the average pairwise Euclidean distance of must-link pairs. Let D_{ij} denote the Euclidean distance between x_i and x_j , φ denote the set of must-link pairs, for one must-link pair $(x_i, x_j) \in \varphi$; we think the smaller the D_{ij} is, the lesser the information this must-link pair provides for clustering, since nearby points are very likely to be must-link, so we should assign a large value to α in this case, while a large D_{ij} means the opposite. Following the above intuition, we define α as

$$\alpha = \frac{1}{1 + \overline{D_{ij}^\varphi} / \overline{D_{ij}}}, \quad (8)$$

where $\overline{D_{ij}^\varphi} = (1/|\varphi|) \sum_{(x_i, x_j) \in \varphi} D_{ij}$, $\overline{D_{ij}} = (1/(n(n-1))) \sum_{i,j=1, i \neq j}^n D_{ij}$.

4.2. Baseline algorithms and evaluation metrics

Baseline algorithms: To evaluate the effectiveness of RoM-AG for spectral clustering, we compare two algorithms, i.e. RoM-MSC and SRoM-MSC, to conventional spectral clustering algorithms and some popular unsupervised and semi-supervised clustering algorithms. In the case of unsupervised clustering, we compare RoM-MSC to K-Means, multiclass spectral clustering (MSC) [3], and self-tuning spectral clustering (SSC) [7]. In the case of semi-supervised clustering, we compare SRoM-MSC to constrained complete-link (CCL) [15], constrained spectral clustering (CSC) [16], and spectral learning (SL) [17]. CCL is a constrained complete-link algorithm, it is firstly proposed as a method of transforming instance-level constraints to space-level constraints in semi-supervised clustering, and it is one of the most popular metric-based semi-supervised clustering algorithms. We run MSC, CSC in our experiments by using the MATLAB codes published by Yu at [19], and the MATLAB codes of SSC is published by Zelnik-Manor at [20].

Evaluation metrics: As for cluster evaluation, following [15], we use constrained rand index (CRI) defined as

$$CRI = \#correct \text{ free decisions} / \#total \text{ free decisions}. \quad (9)$$

A decision is considered correct if the clustering algorithm agrees with the real clustering. According to Eq. (9), we use Eqs. (10) and (11) for unsupervised and semi-supervised clustering separately.

$$CRI = \frac{\#CD}{N(N-1)/2}, \quad (10)$$

$$CRI = \frac{\#CD - \#ML}{N(N-1)/2 - \#ML}, \quad (11)$$

where $\#CD$ denotes the number of correct decisions, $\#ML$ denotes the number of must-link constraints, N denotes the number of data points.

In our experiments, K-Means adopts random sampling to select the initial cluster centroids, and we compute the average CRI for evaluation after having performed K-Means 50 times.

4.3. Experimental results

In order to evaluate RoM-AG, we evaluate the proposed algorithms, i.e., RoM-MSC and SRoM-MSC, on both synthetic data and real world data. In the case of semi-supervised clustering, the generation of must-link constraints is as follows. Given the number of must-link constraints, assign it to all the clusters according to the cluster size. Within one cluster, must-link constraints are generated by randomly choosing data pairs. For a specific number of must-link constraints, we perform each clustering algorithm 10 times, and take the average CRI as the evaluation criterion.

Synthetic data: In this experiment, we consider three popular synthetic datasets, i.e. two-moon dataset, spiral dataset, and two-density dataset. Two-moon dataset and spiral dataset are both widely used in semi-supervised learning [9,10,21], two-density dataset is used in [7]. Note that these datasets all lie on manifolds, and thus methods which assume spherical shape for the data like K-Means will fail.

Fig. 3 illustrates the results on the three synthetic datasets. We can see that K-Means fails on all the three datasets, since it assumes spherical shape for the data. RoM-MSC achieves the best performance, and outperforms MSC on all the datasets; it demonstrates that RoM-AG can improve the performance of MSC a lot on the three synthetic datasets. Note that SSC achieves similar performance to RoM-MSC except on spiral dataset; we will demonstrate that SSC has limited success on real world data in the following experiments.

UCI data: In this experiment, we consider five datasets from the UCI machine learning repository [22]: Ionosphere, Iris, Glass, Wine, and Letters. For Letters, we choose three clusters: {I, J, L} from it to construct our dataset. Table 1 summarizes the properties of the datasets.

Table 2 shows the performance comparison in the case of unsupervised clustering. We can see:

- MSC outperforms K-Means on most of the datasets except Glass and Wine. This demonstrates that Gaussian affinity graph based spectral clustering has fundamental advantages over K-Means, but its performance is a little not stable since MSC is sensitive to the scaling parameter.
- The local scaling method adopted in SSC has achieved limited success on real world data.
- RoM-MSC outperforms K-Means, MSC, and SSC on all the datasets. This demonstrates that RoM-AG is better than

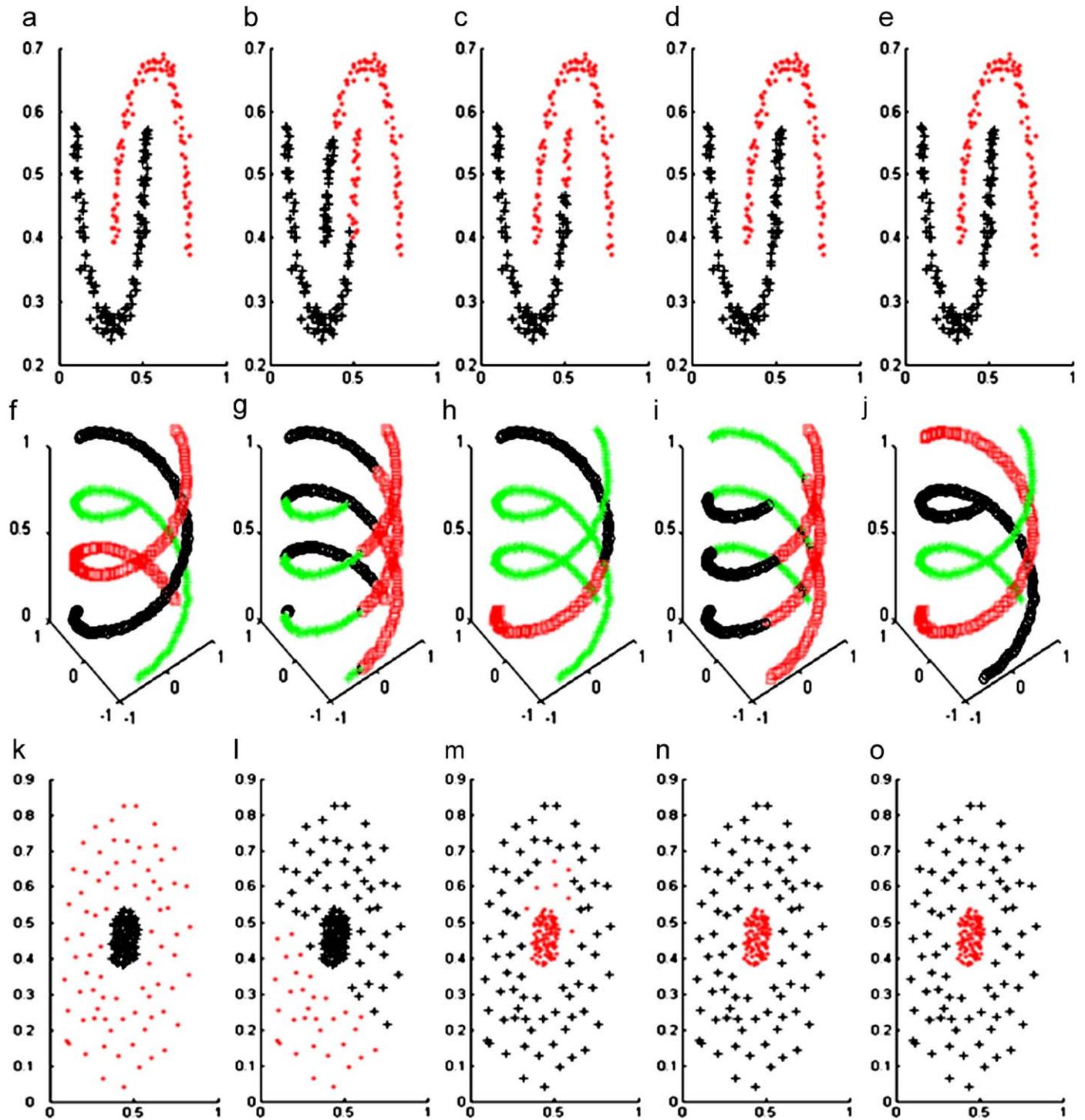


Fig. 3. Performance comparison on synthetic data. (a) Two moon, (b) K-means: CRI = 0.608, (c) MSC: CRI = 0.859, (d) SSC: CRI = 1.000, (e) RoM-MSC: CRI = 1.000, (f) spiral, (g) K-means: CRI = 0.554, (h) MSC: CRI = 0.721, (i) SSC: CRI = 0.555, (j) RoM-MSC: CRI = 1.000, (k) two density, (l) K-means: CRI = 0.602, (m) MSC: CRI = 0.924, (n) SSC: CRI = 1.000, and (o) RoM-MSC: CRI = 1.000.

Table 1
Properties of UCI datasets.

	Ionosphere	Iris	Glass	Wine	Letter-IJL
No. of instances	351	150	214	178	2263
No. of attributes	34	4	9	13	16
No. of clusters	2	3	6	3	3

Table 2
Unsupervised clustering results on UCI data measured by CRI.

	Ionosphere	Iris	Glass	Wine	Letter-IJL
K-Means	0.584	0.798	0.671	0.705	0.607
MSC	0.65	0.886	0.667	0.676	0.661
SSC	0.609	0.889	0.675	0.663	0.615
RoM-MSC	0.69	0.892	0.691	0.706	0.681

Gaussian affinity graph to depict the intrinsic structure of data, and RoM-MSc is less sensitive to the scaling parameter than MSC does.

- Some datasets have relatively low CRI, such as Letter-IJL, because the attributes of the dataset are not suitable for clustering. In this case we can try to perform semi-supervised clustering by adding must-link constraints to improve the clustering results.

Tables 3 and 4 show the performance comparisons on Letter-IJL and Wine separately in the case of semi-supervised clustering. Here #ML means the number of must-link constraints.

From Tables 3 and 4, we can see:

- The performance of CCL fluctuates with different #ML, since its propagation of must-link constraints pays more attention to constraint consistency, and can not respect the smooth consistency well.
- CSC outperforms CCL on the two datasets, and its performance improves with the increasing of #ML, but the improvement is limited.
- SL has the similar performance with CSC.
- SRoM-MSc produces better clusters compared to CCL, CSC, and SL. Its performance increases with the increasing #ML, and the increment is reasonable.

UMist face data: In this experiment, we consider UMist face data [23] which consist of 575 grayscale face images from 20 subjects; each face image has a size of 112×92 on pixel level. The UMist face data used in our experiment are downloaded from [24], and we regard the face images from one subject as a cluster. Table 5 shows the performance comparison on UMist face data in the case of unsupervised clustering. We can see that RoM-MSc achieves the best performance on this dataset.

20-newsgroup data: In this experiment, we consider a text dataset: a modified 20-newsgroup dataset downloaded from [24]. It is a tiny version of the 20-newsgroup data [25] with binary occurrence data for 100 selected keywords across 16 242 postings. Although the data are organized into 20 different newsgroups, we can separate them into four clusters according to the highest level of the topic naming, i.e., comp, rec, sci, and talk. We randomly select 500 samples from each cluster, and thus our dataset consists of 2000 samples from four clusters, each sample has dimensionality of 100. As in [10], we use $W_{ij} = \exp(-(1 - \langle x_i, x_j \rangle / (\|x_i\| \|x_j\|)) / (2\sigma^2))$ in computing S in Eq. (6). Table 6 shows the performance comparison on 20-newsgroup data in the case of

Table 3
Semi-supervised clustering results on Letter-IJL measured by CRI.

	#ML = 0	#ML = 50	#ML = 100	#ML = 150	#ML = 200
CCL	0.453	0.452	0.601	0.518	0.592
CSC	0.661	0.662	0.664	0.668	0.668
SL	0.661	0.663	0.665	0.667	0.670
SRoM-MSc	0.681	0.768	0.831	0.886	0.889

Table 4
Semi-supervised clustering results on Wine measured by CRI.

	#ML = 0	#ML = 10	#ML = 20	#ML = 30	#ML = 40
CCL	0.604	0.575	0.616	0.538	0.624
CSC	0.676	0.668	0.682	0.715	0.716
SL	0.676	0.677	0.680	0.703	0.711
SRoM-MSc	0.706	0.707	0.727	0.751	0.765

Table 5
Unsupervised clustering results on UMist face data measured by CRI.

	K-Means	MSC	SSC	RoM-MSc
CRI	0.885	0.919	0.876	0.956

Table 6
Unsupervised clustering results on 20-newsgroup data measured by CRI.

	K-Means	MSC	SSC	RoM-MSc
CRI	0.527	0.582	0.310	0.728

Table 7
Semi-supervised clustering results on 20-newsgroup data measured by CRI.

	#ML = 0	#ML = 50	#ML = 100	#ML = 150	#ML = 200
CCL	0.263	0.283	0.305	0.362	0.369
CSC	0.582	0.591	0.598	0.603	0.611
SL	0.582	0.588	0.595	0.613	0.624
SRoM-MSc	0.728	0.735	0.769	0.783	0.803

unsupervised clustering. Table 7 shows the performance comparison on 20-newsgroup data in the case of semi-supervised clustering. We can see that both RoM-MSc and SRoM-MSc achieve the best performance on this dataset.

Evaluation of parameter setting: As aforementioned in Section 4.1, parameter α in RoM-AG is set according to Eq. (8). In order to evaluate this setting of α , we illustrate the performance variation of SRoM-MSc with respect to α in Fig. 4. We plot the performance curve on 20-newsgroup dataset; similar phenomenon has also been observed for other datasets. From Fig. 4, we can see that when α increases from 0 to 1, the performance curve exhibits a shape with a peak, and the performance of SRoM-MSc is near the one of RoM-MSc when α is near 1. As discussed in Section 4.1, this is because the must-link constraints are nearly ignored in RoM-AG when α is very near 1. We can also observe that the selected α according to Eq. (8) is better than most of the sampling values of α when different numbers of must-link constraints are given.

5. Conclusion

In this paper, we focus on how to define affinity graph for spectral clustering. We propose to define it from the perspective of graph partition. We deem that it is advantageous for spectral clustering when the intrinsic structure induced by the affinity graph is helpful to a “meaningful” graph partition. In this way, we propose two consistencies: smooth consistency and constraint consistency, for affinity graph to hold, and then define the affinity graph respecting these consistencies in a regularization framework of ranking on manifolds. Meanwhile the proposed definition of affinity graph, i.e., RoM-AG, is applicable to both unsupervised and semi-supervised spectral clustering. Encouraging experimental results on synthetic and real world data demonstrate the effectiveness of the proposed approach.

There are still some open issues in our work. In defining RoM-AG in semi-supervised case, we only focus on must-link constraints, since we find that it is hard to integrate both must-link and cannot-link constraints in a single affinity graph in the framework of RoM. Simply setting negative affinity in RoM-AG for cannot-link pairs will produce bad results, and is not physically

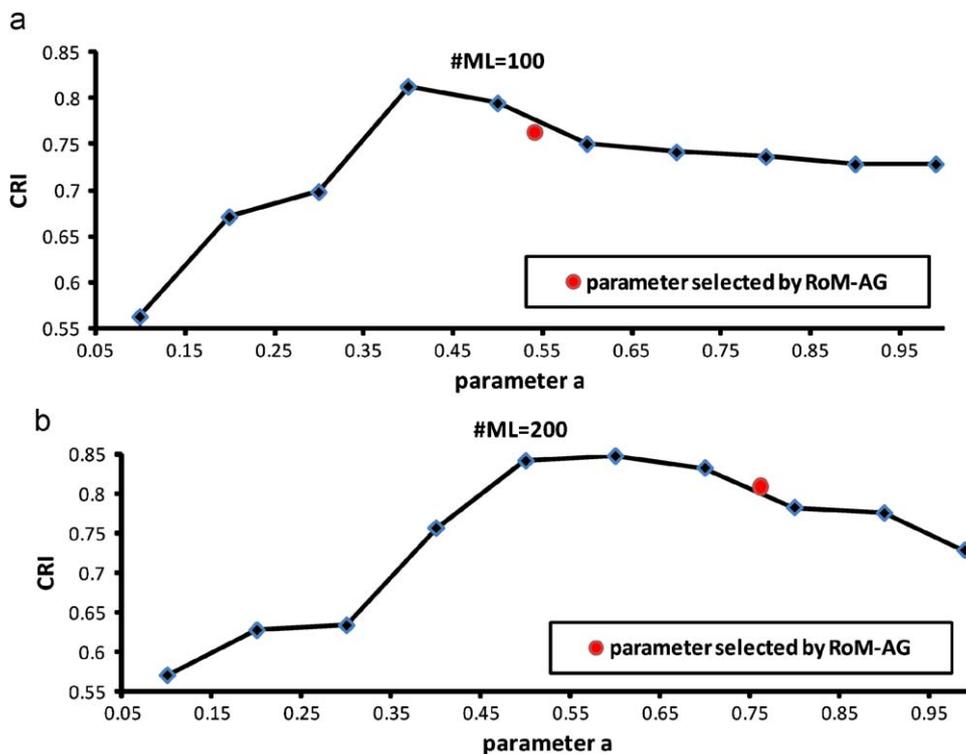


Fig. 4. Performance variation of SRoM-MSC with respect to a on 20-newsgroup data measured by CRI. (a) Performance curve on 20-newsgroup data with 100 must-link constraints. (b) Performance curve on 20-newsgroup data with 200 must-link constraints.

meaningful. In our future work, we plan to adopt multiple view spectral clustering technique [26] and patch alignment method [27,28] to address this problem. We firstly use RoM-AG to generate two affinity graphs for must-link and cannot-link constraints separately, and then reformulate them into a unified form based on the patch alignment formulation, and finally fuse them in a multiple view spectral clustering framework.

Acknowledgments

This work was supported by National Basic Research Program of China (973 Program, 2007CB311100), National High Technology and Research Development Program of China (863 Program, 2007AA01Z416), National Nature Science Foundation of China (60873165, 60802028), Beijing New Star Project on Science and Technology (2007B071), Co-building Program of Beijing Municipal Education Commission. We would like to express our appreciation to the editor and all anonymous reviewers for their insightful comments. We also thank Ping Luo and Liang Wang for their useful suggestions to this paper.

References

- [1] U.V. Luxburg, A tutorial on spectral clustering, Technical Report, Report no. TR-149, Max Planck Institute for Biological Cybernetics, August 2006.
- [2] J. Shi, J. Malik, Normalized cuts and image segmentation, *IEEE Transaction on Pattern Analysis and Machine Intelligence* 22 (8) (2000) 888–905.
- [3] S.X. Yu, J. Shi, Multiclass spectral clustering, in: *Proceedings of the Ninth International Conference on Computer Vision*, 2003, pp. 11–17.
- [4] M. Meila, J. Shi, A random walks view of spectral segmentation, in: *the Eighth International Workshop on Artificial Intelligence and Statistics*, 2001.
- [5] A. Ng, M. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: T.G. Dietterich, S. Becker, Z. Ghahramani (Eds.), *The 16th Annual Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, 2002, pp. 856–894.
- [6] <<http://www.cis.upenn.edu/~jshi/software/>>.
- [7] L. Zelnik-Manor, P. Perona, Self-tuning spectral clustering, in: L.K. Saul, Y. Weiss, L. Bottou (Eds.), *The 18th Annual Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, 2004, pp. 1601–1608.
- [8] E. Xing, A. Ng, M. Jordan, S. Russe, Distance metric learning with applications to clustering with side information, in: Becker, et al. (Eds.), *The 16th Annual Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, 2002, pp. 521–528.
- [9] D. Zhou, J. Weston, A. Gretton, O. Bousquet, B. Schölkopf, Ranking on data manifolds, in: S. Thrun, L. Saul, B. Schölkopf (Eds.), *The 17th Annual Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, 2003, pp. 169–176.
- [10] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, B. Schölkopf, Learning with local and global consistency, in: S. Thrun, L. Saul, B. Schölkopf (Eds.), *The 17th Annual Conference on Neural Information Processing Systems*, MIT Press, Cambridge, MA, 2003, pp. 321–328.
- [11] M. Breitenbach, G.Z. Grudic, Clustering through ranking on manifolds, in: *Proceedings of the 22nd International Conference on Machine Learning*, 2005, pp. 73–80.
- [12] K. Wagstaff, C. Cardie, Clustering with instance-level constraints, in: *Proceedings of the 17th International Conference on Machine Learning*, 2000, pp. 1103–1110.
- [13] S. Basu, M. Bilenko, R.J. Mooney, A probabilistic framework for semi-supervised clustering, in: *Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 59–68.
- [14] K. Wagstaff, C. Cardie, S. Rogers, S. Schroedl, Constrained K-means clustering with background knowledge, in: *Proceedings of the 18th International Conference on Machine Learning*, 2001, pp. 577–584.
- [15] D. Klein, S.D. Kamvar, C.D. Manning, From instance-level constraints to space-level constraints: making the most of prior knowledge in data clustering, in: *Proceedings of the 19th International Conference on Machine Learning*, 2002, pp. 307–314.
- [16] S.X. Yu, J. Shi, Segmentation given partial grouping constraints, *IEEE Transaction on Pattern Analysis and Machine Intelligence* 26 (2) (2004) 173–183.
- [17] S. Kamvar, D. Klein, C. Manning, Spectral learning, in: *Proceedings of the International Joint Conference on Artificial Intelligence*, 2003, pp. 561–566.
- [18] Z. Lu, M.A. Perpinan, Constrained spectral clustering through affinity propagation, in: *Proceedings of the Computer Vision and Pattern Recognition*, 2008.
- [19] <<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/stellayy/code.html>>.
- [20] <<http://www.ee.technion.ac.il/~lihi/>>.
- [21] X. Zhu, Z. Ghahramani, J. Lafferty, Semi-supervised learning using Gaussian fields and harmonic functions, in: *Proceedings of the 20th International Conference on Machine Learning*, 2003, pp. 912–919.

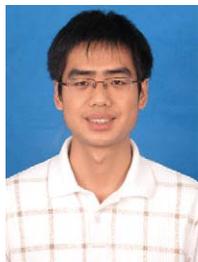
- [22] <<http://mllearn.ics.uci.edu/MLRepository.html>>.
- [23] D.B. Graham, N.M. Allinson, Characterizing virtual eigensignatures for general purpose face recognition, in: H. Wechsler, P.J. Phillips, V. Bruce, F. Fogelman-Soulie, T. S. Huang (Eds.), Face Recognition: From Theory to Applications, in: NATO ASI Series F, Computer and Systems Sciences, vol. 163, 1998, pp. 446–456.
- [24] <<http://www.cs.toronto.edu/~roweis/data.html>>.
- [25] <<http://people.csail.mit.edu/jrennie/20Newsgroups/>>.
- [26] D. Zhou, C.J.C. Burges, Spectral clustering and transductive learning with multiple views, in: Proceedings of the 24th International Conference on Machine Learning, 2007, pp. 1159–1166.
- [27] T. Zhang, D. Tao, X. Li, J. Yang, A unifying framework for spectral analysis based dimensionality reduction, in: Proceedings of the International Conference on Neural Networks, 2008, pp. 1670–1677.
- [28] T. Zhang, X. Li, D. Tao, J. Yang, Local coordinates alignment and its linearization, ICONIP (1) (2007) 643–652.



Yong-dong Zhang, born in 1973, is a Ph.D. associate professor. His major fields include image processing and video processing.



Jin-tao Li, born in 1962, is a professor and Ph.D. supervisor. His major fields include multimedia processing and VR technology.



Tian Xia, born in 1980, received his B.S. in Electrical Engineering from University of Science and Technology of China, China in 2004. Currently, he is a Ph.D. candidate. His research interests focus on multimedia retrieval and machine learning.



Juan Cao, born in 1980, is a Ph.D. candidate. Her research interests focus on semantic video retrieval.