A Robust Clustering Algorithm Based on Aggregated Heat Kernel Mapping

Hao Huang*, Shinjae Yoo† Hong Qin*, and Dantong Yu†
*Department of Computer Science, Stony Brook University
Email: haohuang@cs.stonybrook.edu, qin@cs.stonybrook.edu
†Computational Science Center, Brookhaven National Laboratory
Email: sjyoo@bnl.gov, dtyu@bnl.gov

***Abstract*—Current spectral clustering algorithms suffer from both sensitivity to scaling parameter selection in similarity matrix construction, and data perturbation. This paper aims to improve robustness in clustering algorithms and combat these two limitations based on heat kernel theory. Heat kernel can statistically depict traces of random walk, so it has an intrinsic connection with diffusion distance, with which we can ensure robustness during any clustering process. By integrating heat distributed along time scale, we propose a novel method called Aggregated Heat Kernel (AHK) to measure the distance between each point pair in their eigenspace. Using AHK and Laplace-Beltrami Normalization (LBN) we are able to apply an advanced noise-resisting robust spectral mapping to original dataset. Moreover it offers stability on scaling parameter tuning. Experimental results show that, compared to other popular spectral clustering methods, our algorithm can achieve robust clustering results on both synthetic and UCI real datasets.**

***Keywords*-Spectral analysis; Diffusion processes; Green's function methods**

I. Introduction

Clustering analysis is one of the most important unsupervised knowledge exploration tools in knowledge discovery and data mining. It is especially of value when we have no or limited prior knowledge about the data being acquired or the clustered results are needed to be fed into succeeding phases of the data analysis pipeline.

However, clustering analysis is of little use if the clustered results are radically-different when the scaling parameters of clustering algorithms are slightly modified or even with very little data perturbation (noise or outliers). We call such susceptibility the sensitivity of clustering algorithms, and one of the most desirable properties of clustering algorithms is robustness. In particular, the robustness of clustering algorithms should be measured in the following aspects: (1) not sensitive to any small change of parameters; (2) not sensitive to data perturbation; (3) non-degraded performance even with significant noise level or less-correct parameter settings; and (4) competitive and comparable results when comparing with those less-robust clustering algorithms without any data perturbation and with correct parameter settings. Robust clustering algorithms are highly desirable to combat both **scaling parameter tuning sensitivity** and **noise sensitivity**. With these robustness properties, we can reliably analyze data and conduct other data-driven tasks in succeeding analysis steps. The robustness property is equally significant for domain experts who do not have strong machine learning background as they become much more comfortable in utilizing robust clustering algorithms. It is imperative to develop robust clustering algorithms [5], and this paper serves this pressing need.

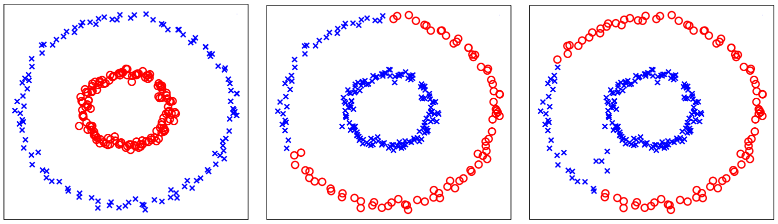
Towards robustness, researchers have explored various techniques, including robust statistics [14], noise in-sensitive regression [3], and noise robust clustering [17]. However, robust clustering approaches considering both parameter tuning sensitivity and noise sensitivity are rather rare. In fact, as shown in Figure 1, scaling parameter tuning of spectral clustering may affect the quality of clusters significantly, moreover, in Figure 1(c) we can see that both tuning sensitivity and data perturbation are correlated to each other.

This paper proposes a unified probabilistic method based on diffusion theory, in this way we try to avoid the influence of both scaling parameter tuning and data perturbation. Since we concentrate on global distribution when we conduct clustering, the embedded structure must be invariant to local perturbation (noise or outliers), and they should be determined only by visible neighborhood while avoiding negative effects from changing scaling parameters. Heat kernel, as the fundamental solution of heat diffusion on manifolds, offers a statistical description on random walk, so it can be employed to build a diffusion map based on global information. In this paper, we unite spectral clustering and heat diffusion theory together and show that it facilitates robustness to both scaling parameter tuning and data perturbation.

A. Motivation

For similarity measure, we typically employ Gaussian kernel as it is one of the most widely-used metric. As shown in Figure 1, it is a well known problem that the scaling parameter, σ, of Gaussian kernel for the affinity matrix has significant impact on discovering embedded structure because σ determines whether two points are considered similar (neighbor) or not [25]. Although several methods have been proposed to address this problem (e.g., [30], [17]), it remains challenging to find a certain range which is

1550-4786/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDM.2011.15



(a) $\sigma = 0.22$ (b) $\sigma = 0.23$ (c) $\sigma = 0.22$ with noises

Figure 1. The sensitivity example of Spectral Clustering Algorithm (NJW) with respect to scaling parameter σ and noises. The small perturbation of scaling parameter or data points gives rise to radically-different results in spectral clustering.

large enough to maintain optimal, yet data-dependent performance. The second challenge in using spectral clustering is the clustering quality with respect to data noise. As noted in [18], spectral clustering is less sensitive to data perturbation than popular K-means algorithms. Yet, depending on the application domain or inappropriate preprocessing of data, spectral clustering can still be susceptible to data noise [31], which tends to make clustering parameter selection even more difficult, especially when making use of scaling parameter σ of Gaussian kernel. Since parameter selection can be significantly affected by the noise level of data, we must address robust spectral clustering in terms of parameter selection and noise simultaneously.

To overcome such difficulties of spectral clustering, we consider heat equation in diffusion theory, which has the built-in robustness of data perturbation and an intrinsic relationship with spectral clustering. Diffusion distance is based on Markov matrix which is a stochastic matrix representing a random walk on graph [19], it can consider up to t steps out of all the possible paths bridging any two points, which makes it much more robust than geodesic distance [6]. Diffusion distance has a potential to be more robust to data perturbation via a family of diffusion maps [6]. In this paper, we focus on heat kernel [13] which offers a natural mechanism to express diffusion distance through heat dissipation process. Heat kernel makes use of not only eigenvectors but also eigenvalues, which give us insight regarding the relative importance of eigenvectors. Inspired by the concept of heat kernel diffusion distance, a more stable clustering algorithm could be designed in terms of data perturbation because it considers multiple paths like diffusion maps. Typically, any diffusion method often starts with some local observation (e.g., Euclidean distance) which is then refined into a global metric (e.g., geodesic or heat kernel distance) through propagation. Nonetheless, existing methods still need to make non-intuitive decisions at various stages for selecting neighbors, global similarity, and embedded reconstruction. As a result, burdensome parameter selection is unavoidable in the current state-of-the-art.

B. Contribution

This paper articulates a novel unsupervised robust spectral clustering method to combat the problem of scaling parameter tuning and data perturbation. It is built on top of spectral clustering and heat kernel theory for robust diffusion with the following contributions:

(1) We derive a robust heat kernel by integrating all time scales of heat kernel into one single term, namely Aggregated Heat Kernel (AHK) (Section III). As a result, we removed the time scaling parameter of heat kernel and design a complete robust clustering algorithm. We discuss the connection of this kernel with other popular robust clustering approaches.

(2) We investigate the best matching normalization approaches for our proposed AHK, which is critical in parameter sensitivity and noise robustness. Laplace-Beltrami Normalization (LBN) [6] is another key ingredient in our clustering framework, which has a very close relationship with diffusion theory and spectral clustering as well. We integrate LBN into our clustering framework rather than the standard graph Laplacian normalization, so that we can recover Riemannian manifold structure regardless the density distribution of dataset.

(3) Our novel clustering algorithm (Section IV), combining Aggregated Heat Kernel with the best matching normalization approaches, delivers robust clustering results in terms of both parameter selection and noise level.

(4) We systematically evaluate the proposed algorithm with several closely-related baseline clustering algorithms on a number of synthetic and benchmark datasets (Section V). We focus on the sensitivity of parameter selections (e.g., both global and local scaling parameters of Gaussian kernel) and the sensitivity of noise level. Our experimental results confirm that the proposed algorithm produces not only competitive results of carefully-tuned baselines on non-noisy datasets but also outperforms existing results with noisy or off-the-sweet-spot parameters.

II. Background and Diffusion Theory

Since our new method is founded upon both spectral clustering and heat diffusion, we shall briefly review the basic idea of spectral clustering, diffusion maps, and heat equation, and address the weakness of existing approaches.

A. Spectral Clustering

```
Algorithm 1: SpectralClustering($X$,$k$)
  Input: Input data $X \in R^{n \times m}$, and $k$ is the number of clusters
  Output: Cluster assignments of $n$ instances
  Compute the affinity matrix $W \in R^{n \times n}$ where $W(i,j) = exp(||x(i) - x(j)||/2\sigma^2)$ ;
  Compute the diagonal matrix $D \in R^{n \times n}$ where $D(i,i) = \sum_{j=1}^{n} W(i,j)$ and $D(i,j) = 0$ if $i \neq j$;
  Compute the graph Laplacian $L$ where $L_{nn} = D - W$, $L_{rw} = I - D^{-1}W$ or $L_{sym} = I - D^{-1/2}WD^{-1/2}$ ;
  Compute the first $k$ eigenvectors $\psi$ of $L$, $\psi = \{\psi(1), \psi(2), \ldots, \psi(k)\}$ ;
  Re-normalize the rows of $\psi \in R^{n \times k}$ into $Y_{ij} = \psi(i,j)/(\sum_q \psi(i,q)^2)^{1/2}.$ ;
  Run $k$-means with $Y \in R^{n \times k}$ ;
```

Among several kinds of clustering algorithms, we focus on spectral clustering, which has gained popularity in the last decade in data mining community because of its ability to discover embedded data structure. Spectral clustering (Algorithm 1) has been known as one of the most popular clustering algorithms nowadays. It has strong connection with graph cutting, in the way that spectral clustering uses eigenspace to solve relaxed forms of the balanced graph partitioning problem [22]. Another aspect of spectral clustering is that, it can capture the manifold structure of data as shown in Figure 1, which is difficult or impossible to achieve for other popular k-means or similar algorithms.

However, there are two challenges in spectral clustering. First, the selection of scaling parameter σ of affinity matrix computation could affect the clustering results radically (Figure 1) because this parameter determines the neighborhood. Second, it is still sensitive to noise. For instance in Figure 1(c), with only a few noisy instances, the clustering result is quite different and the optimal range of scaling parameter σ is also changed.

B. Diffusion Maps

In 2006, Coifman et al. [6] designed a framework based on diffusion process to consider both eigenvalues and eigenvectors. The non-negativity property of affinity matrix W allows us to normalize it into a Markov transition matrix $P = D^{-1}W$ where the states of the corresponding Markov process are data points, which enables us to analyze it as random walk. It is straightforward to calculate the transition probability, $p_t(i,j)$ (the probability of transition from i to j after t steps or time) using entries from P. The diffusion distance between two points at time scale t is

$$D_t^2(i,j) = \sum_k [\frac{(p_t(i,k) - p_t(j,k))^2}{\phi_1(k)}], \tag{1}$$

where $\phi_1(z)$ is the stationary distribution of the random walk (trivial left eigenvector). So the diffusion maps at time scale t project the data point to m dimensional eigenspace as

$$\Psi_t : x \to [\lambda_1^t \psi_1(x), \lambda_2^t \psi_2(x), ..., \lambda_m^t \psi_m(x)], \tag{2}$$

where λ_i are eigenvalues and ψ_i are the corresponding right eigenvectors of P [20]. In this way the diffusion distance between two points becomes

$$D_t^2(x,y) = \sum_{i=1}^{m} [\lambda_i^{2t}(\psi_i(x) - \psi_i(y))^2]. \tag{3}$$

By projecting the data to diffusion space, the effect of scaling parameter in Gaussian similarity is reduced. However, the scaling parameter t in diffusion space is still very essential in terms of the transitive connectivity: small scaling t makes the loosely-connected graph into slightly stronger connection within t connections, while large scaling t makes the graph tend to be more strongly-connected. In 2009, Richards et al. [27] proposed multiscale diffusion distance, which considers all possible paths between each point pair in diffusion space across all time scales t, so that multiscale diffusion distance is more robust to the structure at different time scales. To do this, λ_i^t of Equation (2) is replaced by

$$\sum_{t=1}^{\infty} \lambda_i^t = \lambda_i/(1 - \lambda_i). \tag{4}$$

So they eliminated the effect of different time scales.

C. Heat Equation

Our proposed work is strongly inspired by heat kernel theory [13] and its attractive properties. For instance, it is symmetric, positive semi-definite, multiscale, and stable. Moreover, it can be interpreted as the transition density function of Brownian motion [29], which is the most fundamental continuous time Markov process.

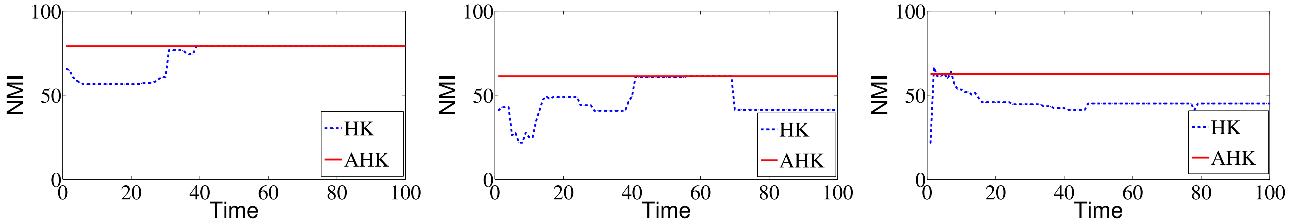
Specifically, the heat equation is associated with normalized graph Laplacian, L_{rw}, which can be defined by

$$\frac{\partial H_t}{\partial t} = -L_{rw} H_t, \tag{5}$$

where $H_t = e^{-tL_{rw}}$ is the heat kernel on Riemannian manifold $\mathcal{M}$ and t is the time scaling parameter [10]. For $L_{rw} = \psi' \Lambda \psi$, the heat kernel can be re-written as follows:

$$H_t(x,y) = \sum_{i=1}^{n} [e^{-\lambda_i t} \psi_i(x) \psi_i(y)], \tag{6}$$

where $H_t(x,y)$ represents the amount of heat being transferred from x to y in time t given a unit heat source at x.



(a) $\sigma = 0.15$

(b) $\sigma = 0.16$

(c) $\sigma = 0.28$

Figure 2. The sensitivity of time scaling parameter t on Iris dataset is shown in NMI (Section V). We use random walk normalization for all three examples.

III. Aggregated Heat Kernel

A. Aggregated Heat Kernel

As discussed earlier, heat kernel is multiscale. For small t, the function $H_t(x, *)$ is mainly determined by the nearby neighborhood of x, and this area grows bigger as t increases. In other words, for small t, $H_t(x, *)$ only reflects local properties of the area around x but large t captures the properties from larger area or even global structure. But this additional one more degree of freedom makes it difficult to determine t in any algorithm (Figure 2) because we have little clue about how to find the best t, which is similar to the scaling parameter σ of Gaussian similarity. In other words, the clustering result could become sensitive due to this time scaling selection.

We propose a new approach by integrating the entire time scale from zero to infinity on heat kernel, which is called **Aggregated Heat Kernel** (AHK).

$$\mathcal{H}(x, y) = \int_{t=0}^{\infty} H_t(x, y)dt = \sum_{i=1}^{n}[(1/\lambda_i)\psi_i(x)\psi_i(y)]. \quad (7)$$

AHK inherits many powerful properties from heat kernel. Among them, the most relevant ones to our current work include

- Symmetric: $\mathcal{H}(x, y) = \mathcal{H}(y, x)$.
- Semigroup identity: $\mathcal{H}(x, y) = \int_M \mathcal{H}(x, z)\mathcal{H}(y, z)dz$.
- Positive semi-definite: $\sum_{i,j} \mathcal{H}(x, y)c_i c_j \geq 0$, where $c_1, c_2, ..., c_n$ are real numbers.

From Figure 2 we observe that in conventional heat kernel the time scaling parameter t is also correlated with the scaling parameter σ and it needs to be carefully tuned. But AHK is better than traditional HK on most of the time parameters. AHK is originally defined by the anisotropic transition kernel such as L_{rw} but we could generalize AHK to $\mathcal{H}_{sym}$ of symmetric L_{sym} or $\mathcal{H}_{nn}$ of unnormalized L_{nn}.

B. Connections to AHK

In this subsection we built theoretical connections from AHK to the other existing popular techniques.

Inverse Laplacian: AHK, $\mathcal{H}$, is pseudo inverse or Moor-Penrose inverse [11]. By doing so, we achieve multiscale heat diffusion. Instead of doing pseudo inverse, we could directly inverse graph Laplacian matrix [17].

$$(I + \alpha L_{sym})^{-1}, \quad (8)$$

where α is the positive regularization parameter and I allows us to invert Laplacian matrix always. Note that, [17] used this direct inversion to get noise robust clustering results.

Commute Distance: Commute distance $C(x, y)$ between x and y is the expected random walk round trip travel time. AHK is also known as Green's function [26], which is closely related to commute distance (CD) or resistance distance. The Green's function is left inverse operator of Laplace operator, $\mathcal{H}_{rw} \cdot L_{rw} = I$. For $\mathcal{H}_{nn}$ constructed from unnormalized L_{nn}, commute distance can be defined as

$$C(x, y) = vol(\mathcal{H}_{nn}(x, x) + \mathcal{H}_{nn}(y, y) - 2\mathcal{H}_{nn}(x, y)), \quad (9)$$

where $vol = \sum_{i=1}^{n} D(i, i)$. Just like AHK, commute distance also considers all possible length, paths and their weights, which is more robust than the shortest path. Note that, commute distance can also be expressed by the random walk or symmetric graph Laplacian normalization [26].

Multiscale Diffusion Map: Commute distance is also related to diffusion distance. By replacing Equation (7) into the above equation, we get

$$C(x, y) = vol \sum_{i=2}^{n}[(1/\lambda_i)(\psi_i(x) - \psi_i(y))^2], \quad (10)$$

and also multiscale diffusion distance can be defined by:

$$\sum_{t=0}^{\infty} D_t^2(x, y) = \sum_{i=1}^{m}[1/(1 - \lambda_i^2)(\psi_i(x) - \psi_i(y))^2]. \quad (11)$$

Both commute distance and diffusion distance look similar but they have different eigenvalue weighting and different Laplacian normalization.

Multiscale diffusion distance [27] can also be represented by $\sum_{t=0}^{\infty} \lambda_i^t = 1/\lambda_i$, which shares the same weighting with $\mathcal{H}$ but it is for distance weighting. If the time summation starts from $t = 1$, then it is exactly the same as the multiscale diffusion map (MDM) of Equation (4). Both of eigenvalue weighting (starting $t = 0$ and $t = 1$) will show quite similar weighting distribution anyway for $0.5 \leq \lambda \leq 2$, which is common for most of normalized graph Laplacian.

C. Normalization

Even though we made proper connections among similar approaches, most of them used different normalization without thorough evaluation. Therefore it is not clear what is the best way to normalize graph Laplacian matrix for our proposed $\mathcal{H}$. It is shown in [16] that if we assume uniform sampling of data points from a sub-manifold $\mathcal{M}$, the eigenvectors of L_{rw} with $\sigma \to 0$ and $n \to \infty$, tend to approximate Laplace-Beltrami operator on $\mathcal{M}$, which guarantees manifold structure reconstruction. However, in reality, the sampled data points tend to be nonuniform and show skewed density distributions, resulting in poor manifold structure reconstruction in AHK. To improve the distributional sensitivity of Random Walk (RW) normalization, we consider the following two additional normalizations:

$$W^{(\alpha)} = D^{-\alpha} W D^{-\alpha}, \tag{12}$$

$$L^{(\alpha)} = I - {D^{(\alpha)}}^{-1} W^{(\alpha)}, \tag{13}$$

where α is a normalization parameter and $D^{(\alpha)}$ is a diagonal matrix with the sum of row weight of $W^{(\alpha)}$.

- If $\alpha = 0$, $L^{(0)} = L_{rw}$ (Random Walk normalization).
- If $\alpha = 1/2$, then it is *Fokker-Planck* (FP) diffusion.
- If $\alpha = 1$, it is Laplace-Beltrami Normalization (LBN).

The relations among those three normalizations are well described in [6]. Depending on α, LBN can also be reduced to Random Walk or FP diffusion. In particular, we focus on LBN because it removes the influence of the dataset density and recovers manifold structures on $\mathcal{M}$ with the condition of both $\sigma \to 0$ and $n \to \infty$ [6]. In other words, the additional re-normalization of affinity matrix W enables us to reconstruct manifold structures better under non-uniform density distribution, so that our clustering results can be less sensitive to noise and scaling parameter sensitivity.

D. Comparison

Figure 3 shows the effects of different approaches and normalizations on 20 newsgroup text data (20ngC) (Section V). True inversion and commute distance show the worst results in separating three topics. Although they share the same Laplacian matrix inversion approaches, the results are quite different. Interestingly multiscale diffusion map shows the best separation among non-AHK approaches. In case of AHK, most of normalization approaches except unnormalized Laplacian reconstruct ball shape of topic distribution. The original Random Walk (RW) normalization shows the most mixture of three topics but as we add the additional normalization of Equation (12), we reconstruct better manifold structures. LBN shows the best coherent and condensed reconstruction quality. AHK with unnormalized Laplacian appears to have the ability of separation but the distance among documents are very close to each other compared to other normalizations. Symmetric normalization also shows very good separation and ball shape reconstruction but symmetric normalization is not anisotropic transition. For our future experiments, we mainly focus on LBN but we provide further detailed analysis across different datasets regarding different normalization effects and approaches in Section V.

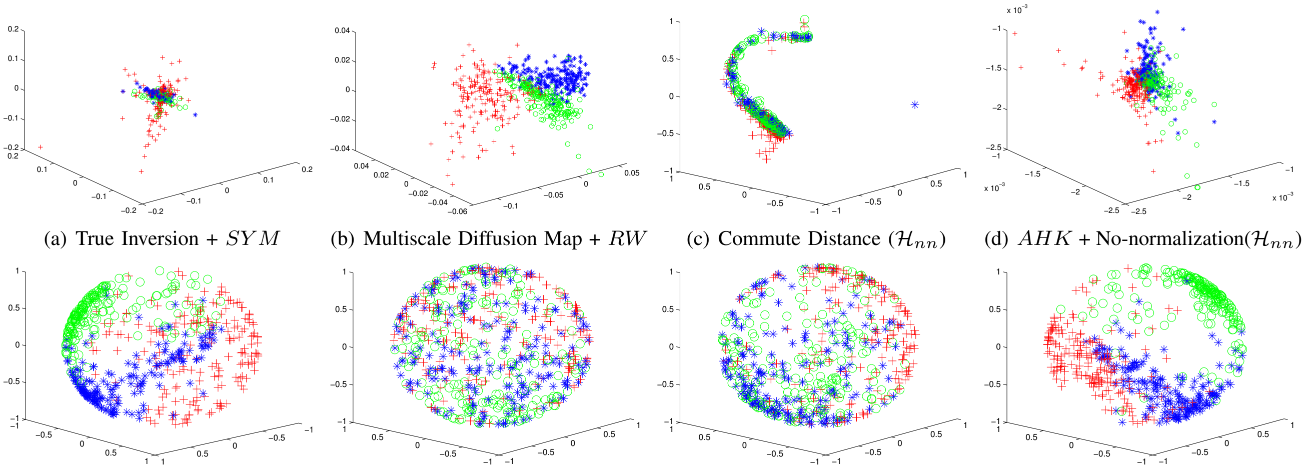
IV. New Algorithm

After investigating some nice properties of heat kernel, it now sets a stage for us to introduce a novel robust spectral clustering algorithm using both AHK and LBN (Algorithm 2), which is less sensitive to the scaling parameter selection and noise perturbation. Let X be a matrix of size $n \times m$, where n is the number of data points and m is the number of dimensions, our algorithm is detailed in Algorithm 2.

```
Algorithm 2: AHKClustering($X$,$k$,$\gamma$, xxx)
  Input: Input data $X \in R^{n \times m}$, $k$ is the number of
         clusters, $\gamma$ is an eigenvalue smoothing
         parameter, and xxx is a normalization method
  Output: Cluster assignments of $n$ instances
1 Construct Laplacian $L_{\text{xxx}}$ ;
2 Compute generalized eigenvectors $\psi(i)$ and
  corresponding eigenvalues $\lambda_i$, $i = 1, 2, ..., n.$ ;
3 Construct $\mathcal{H}_{\text{xxx}}$ matrix with $\psi(i)$ and $\lambda_i$, where
  $\mathcal{H}_{\text{xxx}}(x, y) = \sum_{i=2}^{n} [\frac{1}{\gamma + \lambda_i} \psi(i, x) \psi(i, y)]$;
4 Compute the first $k$ eigenvectors $\psi_s$ of $\mathcal{H}_{\text{xxx}}$,
  $\psi_s = \{\psi_s(1), \psi_s(2), \ldots, \psi_s(k)\}$;
5 Re-normalize the rows of $\psi_s \in R^{n \times k}$ into
  $Y_{ij} = \psi_s(i, j) / (\sum_q \psi_s(i, q)^2)^{1/2}.$ ;
6 Run $k$-means with $Y \in R^{n \times k}$ ;
```

We suggest to use LBN as normalization choice of our proposed algorithm. This algorithm undergoes a kind of data warping by using LBN (Step 1) and AHK (Step 2 and 3). Then we perform the second eigenvalue decomposition (Step 4) and then normalize its row (Step 5). k-means algorithm is used for final clustering. We assume that the entire graph is well-connected, so that the eigenvectors except the first one are included in Step 3. If unconnected, a threshold can be set to filter out the smaller eigenvalues and the corresponding eigenvectors. The eigenvector smoothing parameter γ of Step 3 is added to stabilize the affinity matrix computation.

Regarding computational complexity, eigenvalue decomposition is the most time consuming step, which will dominate our computation. There are many iterative methods to conduct eigenvalue decomposition (e.g., power iteration [2]), but in general finding the eigenvalues reduces to matrix multiplication by computing a symbolic determinant, which gives a running time of $O(n^3 + n^2 log^2 n)$ [24].



(a) True Inversion + SYM (b) Multiscale Diffusion Map + RW (c) Commute Distance ($\mathcal{H}_{nn}$) (d) AHK + No-normalization($\mathcal{H}_{nn}$)

(e) $AHK + SYM$ ($\mathcal{H}_{sym}$) (f) $AHK + RW$ ($\mathcal{H}_{rw}$) (g) $AHK + FP$ ($\mathcal{H}_{fp}$) (h) $AHK + LBN$ ($\mathcal{H}_{lbn}$)

Figure 3. Effects on embedded construction on 20ngC dataset is shown.

Table I
STATISTICS OF OUR EVALUATION DATASETS

	Data Set	# instances	# attributes	# clusters
1	Iris	150	4	3
2	Glass	214	9	6
3	PenDigits01	200	16	2
4	PenDigits17	200	16	2
5	PolBooks	105	105	3
6	UBMCBlog	404	404	2
7	AGBlog	1222	1222	2
8	20ngA	200	61188	2
9	20ngB	400	61188	2
10	20ngC	600	61188	3
11	20ngD	400	61188	4
12	FaceContour	266	2	3

V. Experimental Results

A. Experimental Setup

Dataset. To demonstrate the robustness of our proposed method, we evaluate our algorithm on one synthetic *FaceContour* dataset and seven UCI benchmark datasets including four text datasets, and three network datasets, summarized in Table I. Such diverse combination of data is intended for our comprehensive study. *Iris* dataset is a collection of three species of irises where one is linearly separable but the other two are not [9]. *Glass* includes types of glasses for criminological investigation [8]. We also test two handwritten pendigit data *PenDigits01* and *PenDigits17* with digit “0” vs. “1” for easy task and “1” vs. “7” for challenging task. *PolBooks* is a network data for co-purchasing pattern of 105 political books of three classes [21]. *UBMCBlog* and *AGBlog* are political blog connection network data [15][1]. *20ng* is 20 newsgroup text data [23]. *20ngA* includes 100 messages from misc.forsale and 100 messages from soc.religion.christian. *20ngB* and *20ngD* add 100 messages to each category and *20ngC* adds 200 from talk.politics.guns to 20ngB. To show noise robustness, we also add noise from 10% to 100% by 10% increment on both FaceContour and 20ngD. Noise in FaceContour is uniformly distributed. Noise in 20ngD comes from two other different news group talk.politics.guns and rec.sport.baseball.

Similarity Measure. We apply mainly Gaussian similarity as our similarity measure. For network data (PolBooks, UBMCBlog, and AGBlog), the affinity matrix is a binary link matrix where $A(i, j) = 1$ if there is an edge from i to j, and $A(i, j) = 0$ otherwise. By the nature of text data, cosine similarity metric is the only metric we apply for text and we use word counts as features except stop words and singleton words.

Baselines. We compare our results to five competitive clustering algorithms. For our basis of spectral clustering, we choose symmetric graph Laplacian spectral clustering (N-JW) [22]. To show parameter tuning sensitivity, we include Self-Tuning (ST) spectral clustering [32]. As our diffusion map baseline, Multiscale Diffusion Maps (MDM) [27] and Commute Distance (CD) [26] are considered. Finally, to compare noise robustness, we add Noise Robust Spectral Clustering (NR) [17].

Evaluation. Since we have the ground truth of labels for each data, we compare our clustered results with the labels. We use several popular evaluations in our experiment (e.g., purity, normalized mutual information (NMI)). Due to space limitation, NMI is used as our only evaluation metric among all being described because most of clustering algorithm papers make use of NMI as their primary evaluation metric. Detailed definition of NMI can be referred to [28].

Parameters. Other than scaling parameter σ of Gaussian similarity, our proposed algorithm has one eigenvalue smoothing parameter γ. In our experiments, if we have big enough $\sigma \geq 0.2$, we do not need to set γ but if we set

$\gamma = 0.01$, it makes our proposed algorithm stable even with very small σ and we apply the same γ to commute distance and it shows better stability as well. In our k-means implementation, we evaluate Within-Cluster Sum of Square (WCSS) scores of each random trials and we choose the best one out of 100 random trials [12].

As for both global and local scaling, we run experiments with all the sigma inside the range to test NMI average for each noise level using our algorithm and other five algorithms. We find the average performance along global scaling parameter σ ($[0.1, 8]$, with 0.1 as step size between 0.1 to 1 and 0.5 as step size between 1 to 8). For local scaling parameters, $[5, 50]$, 1 as step size is used. The only difference in local scaling lies at the number of parameters and selection of different algorithms. NR seeks σ and β based on the largest eigen-gap [17]. However, eigen-gap works poorly on our benchmark dataset and we use the same parameter for both σ and β. ST selects its only scaling parameter as σ_i (σ_j) where σ_i (σ_j) is the distance from point i (j) to its k^{th} nearest neighbor [32]. Our algorithm, as well as NJW and MDM, all follow the same way as ST for local scaling experiments (all the source code and datasets we have used is available at http://www.cs.sunysb.edu/∼huang3/).

B. AHK Normalization and Cosine Similarity Analysis

We evaluate different normalization methods for our proposed aggregated heat kernel (AHK). To avoid tuning scaling parameter σ, we adopt cosine similarity, which makes our proposed algorithm parameter-tuning-free. Table II documents the clustering results (NMI) of five different normalizations: no-normalization (NN), symmetric normalization (SYM), random walk (RW) normalization, Fokker-Planck (FP) diffusion, and Laplace-Beltrami normalization (LBN). These normalization methods are simply applied in Step one of Algorithm 2.

In Table II, LBN shows the best overall performance across different types of data. Specifically, LBN shows the best performance on text data and competitive performance on network datasets. On remaining dataset, it shows the best results along with SYM. From now on, AHK uses only LBN normalization. Compared with LBN, NN, RW and FP show relatively low performance but FP shows slightly better performance than RW, which supports the argument that the first normalization is helpful. Interestingly NN, RW and FP show quite worse performance on text data. These observations suggest that the density distribution plays a role in reconstructing manifold structures of real-world datasets and LBN is a better choice. No-normalization shows the worst performance among five approaches, which indicates the importance of normalization.

Table III summarizes six different approaches using cosine similarity. Our new AHK shows the best or very close to the best performances. MDM shows the second best on all but weak performance on text data. ST and NJW show quite similar performance on cosine similarity because of no σ tuning. Although commute distance shares similar motivation with MDM, CD appears to be worse than MDM especially on text data. NR shows the worst performance except PenDigits01 and PolBooks.

Table II
AHK NORMALIZATION COMPARISON USING COSINE SIMILARITY

Data Set	NN	RW	FP	SYM	LBN
Iris	8.8	40.6	40.6	60.8	**70.4**
PenDigits01	**100**	95.9	**100**	**100**	**100**
PenDigits17	2.3	0.9	13.8	**16.1**	**16.1**
PolBooks	56.9	56.9	54.0	56.7	**58.3**
UBMCBlog	5.7	**73.7**	40.7	**73.7**	72.8
AGBlog	1.1	41.1	0.0	**74.9**	70.2
20ngA	6.2	8.0	78.2	73.6	**80.8**
20ngB	1.9	0.0	37.2	67.8	**71.8**
20ngC	15.7	2.6	12.4	38.5	**67.2**
20ngD 0% noise	3.0	0.0	10.2	56.4	**61.7**
20ngD 50% noise	4.7	0.7	4.0	31.1	**49.0**
20ngD 100% noise	5.1	0.1	5.6	33.7	**43.7**
Average	17.6	26.7	33.1	57.0	**64.0**

Table III
COMPARISON AMONG SIX APPROACHES USING COSINE SIMILARITY

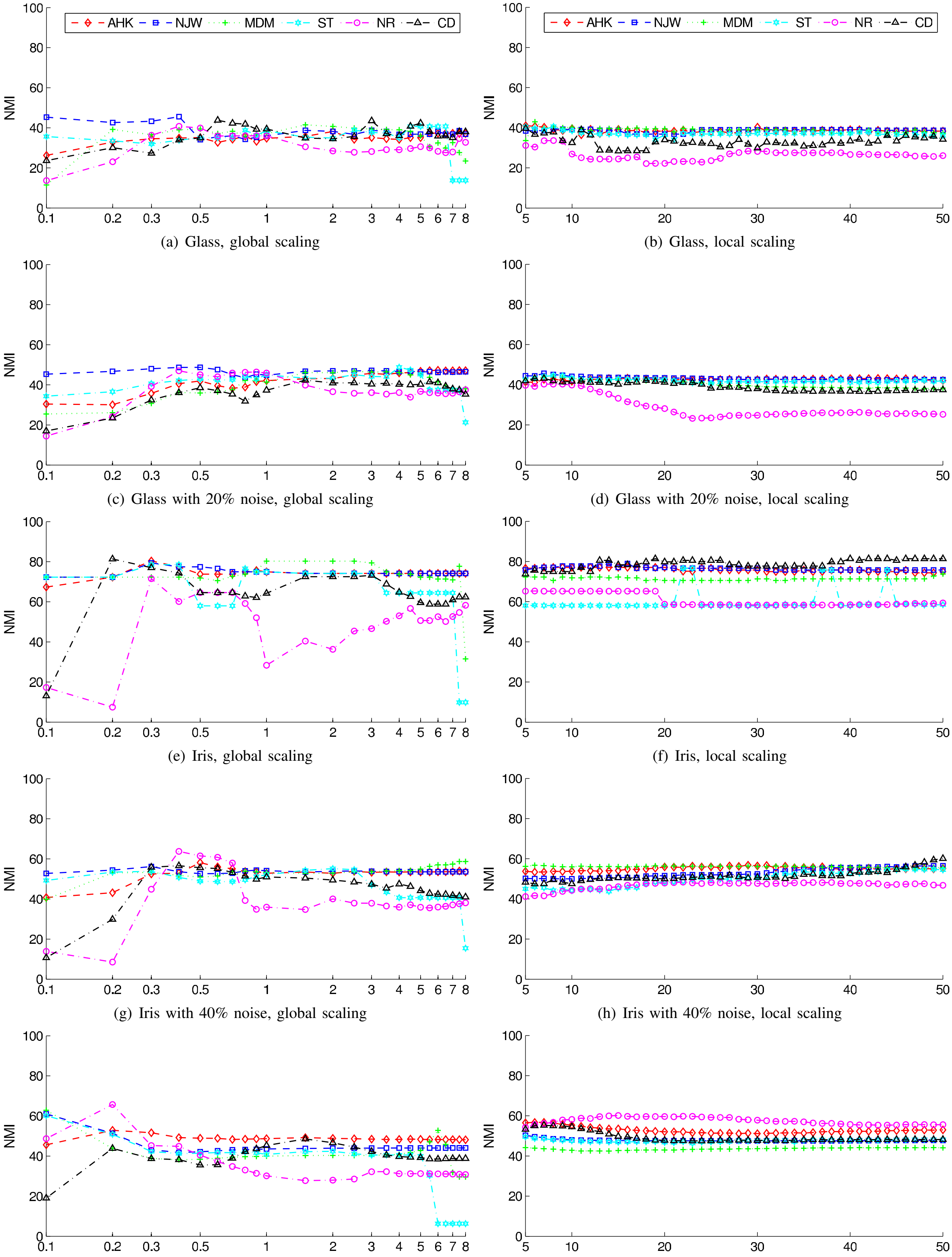
Data Set	NR	CD	NJW	ST	MDM	AHK
Iris	8.8	48.4	63.5	72.3	**93.1**	70.4
PenDigits01	**100**	95.9	**100**	**100**	**100**	**100**
PenDigits17	2.4	12.9	20.4	20.4	**20.7**	16.1
PolBooks	57.5	52.0	54.2	56.3	**58.7**	58.3
UBMCBlog	2.4	0.1	73.8	73.8	**74.9**	72.8
AGBlog	0.5	0.4	0.2	0.2	**71.7**	70.2
20ngA	2.4	0.7	75.9	75.9	78.2	**80.8**
20ngB	1.6	0.3	10.0	5.0	2.4	**71.8**
20ngC	2.2	1.7	34.9	34.4	38.2	**67.2**
20ngD 0% noise	2.4	0.0	56.8	55.4	53.5	**61.7**
20ngD 50% noise	2.8	2.0	38.5	41.5	42.1	**49.0**
20ngD 100% noise	2.6	0.2	39.5	39.5	38.8	**43.7**
Average	15.5	17.9	47.3	47.9	56.0	**64.0**

C. Robustness to Scaling Parameter

To systematically manifest the sensitivity of different algorithms on different scaling parameters, we test them respectively on a series of global and local scaling parameters. Datasets used here are Iris and Glass from UCI including 40% and 20% noise levels, and synthetic noisy dataset FaceContour with 40% noise level. For noisy dataset, we repeat randomization 20 times to get stable results. The quantitative results are shown in Figure 4. We can see that our new AHK algorithm is either less sensitive or at least comparable to other five algorithms using both global and local scaling parameters. Moreover, our algorithm is either the best or close to the best with noisy datasets and stays at the top.

D. Robustness to Noise

We conduct experiments on controlled noisy datasets to examine the performance of our algorithm and make



(a) Glass, global scaling

(b) Glass, local scaling

(c) Glass with 20% noise, global scaling

(d) Glass with 20% noise, local scaling

(e) Iris, global scaling

(f) Iris, local scaling

(g) Iris with 40% noise, global scaling

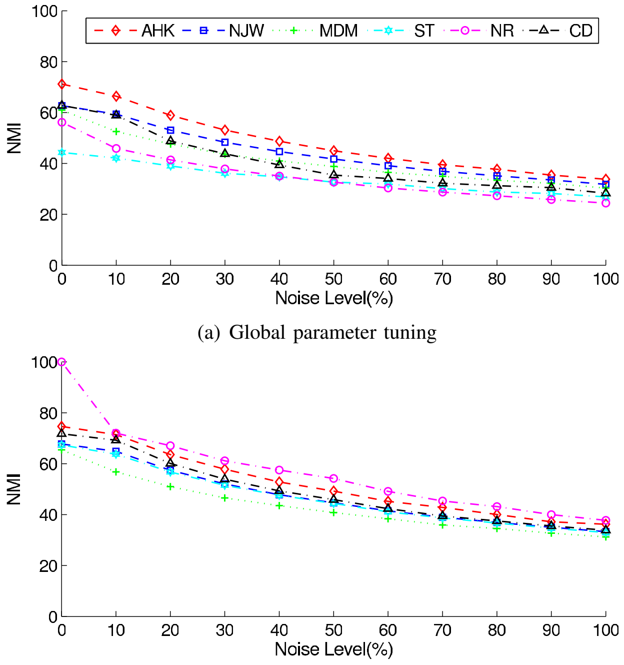
(h) Iris with 40% noise, local scaling

(i) FaceContour 40% noise, global scaling

(j) FaceContour 40% noise, local scaling

Figure 4. Comparison of six algorithms using different scaling parameters.

comparison with the other five algorithms. The data sets are FaceContour with uniformly-distributed noise of different noise levels (0%, 10%, 20%, $\cdots$, 100%). To show the noise robustness and avoid parameter tuning of scaling parameter, we average all the scaling parameters. The experimental results are documented in Figure 5. AHK indicates the best (global scaling) and the second best (local scaling) results as it always stays in the best candidate list. Although NR shows the best performance with local scaling, it is one of the worst performer on global scaling. Such fluctuating performance of NR is consistent throughout scaling experiments. Similar to NR, MDM works poorly on local scaling but MDM had shown stable performance in previous scaling experiments. Overall, AHK shows robust performance across different noise conditions including cosine similarity of Table III.



(a) Global parameter tuning

(b) Local parameter tuning

Figure 5. Algorithmic performance on different noise levels.

E. Discussion

Although our proposed clustering algorithm requires no further parameter tuning except σ, we can make it faster by dropping less-informative eigenvectors or we can fine tune special cases. In Step 3 of Algorithm 2, we may use smaller number of eigenvectors than n within the range of 300 to the number of data points. Normally, due to the dramatic value drop-down of eigenvalues, it is safe to choose from 300 to 500 for the data sets no larger than 10^4.

Choosing the number of eigenvectors k of Step 4 may also affect clustering results. We typically set this value as the number of clusters, as most spectral clustering algorithms take the same strategy. However, if the original data has strong manifold structures in k dimensions where k is the number of clusters, then other spectral cluster algorithms may fail to reconstruct original manifold structure but our proposed algorithm may be able to reconstruct this original manifold structure in the first k eigenvectors, which may produce worse results. Should such situation occurs, we could simply add one or two additional eigenvectors, which are expected to greatly improve the results. In reality, it did not happen on our benchmark dataset or it will not happen in high dimensional or noisy data because it is much more difficult to reconstruct original manifold structure.

VI. Related Work

Mean shift clustering [7] and spectral clustering [22] [31] [19] have shown good performance in some clustering tasks. However, both of them are sensitive to scaling parameters. To improve, Zelnik-Manor and Perona proposed to use local scale [32] which fully considers the local structure of dataset using neighbor adaptive scale. They introduced a local scaling value σ for each data point. However, the local scaling in [32] depends on distance between certain point p_i and its Q^{th} neighbor. So users still need to specify Q, which is also sensitive to the clustering result for tuning. Compared with the above methods, our method can maintain the similar performance which is insensitive to the scaling parameters.

In [17], the authors proposed a noise robust spectral clustering algorithm. But our experimental results have clearly demonstrated that our method has better performance. Recently in [4], M-estimation robust statistics is used in a robust path-based similarity measure which requires no local parameters to be set manually, nonetheless, prior knowledge of data domain is required. In contrast, users need no prior knowledge when using our algorithm.

VII. Conclusion

We have developed a new spectral clustering algorithm with robustness to both scaling parameter tuning and data perturbation. The mathematically-rigorous theory of our work, together with the new AHK algorithm, are originated from heat kernel and diffusion maps. In technical essence, our AHK permits reorganizing the spectral-embedded structure regardless sub-optimal scaling parameter selection, noise perturbation, and non-uniform density distribution. Extensive experiments and evaluations have demonstrated robust performance with our AHK algorithm in comparison with other popular spectral clustering algorithms. Immediate future work will be concentrated on constructing local and global coordinates with the goal of learning the intrinsic structure of data.

VIII. Acknowledgements

We gratefully thank all the anonymous reviewers for constructive suggestions toward paper improvement. This research is supported in part by NSF grants IIS-0710819, IIS-0949467, IIS-1047715, and IIS-1049448. It is also supported by a BNL LDRD project 10-001.

References

[1] L. A. Adamic and N. Glance. The political blogosphere and the 2004 us election. *In Proceedings of the 3rd international workshop on Link discovery*, pages 36–43, 2005.

[2] R. Badeau, B. David, and G. Richard. Fast approximated power iteration subspace tracking. In *Signal Processing, IEEE Transactions on 2005*, pages 2931–2941, 2005.

[3] G. Camps-Valls, L. Bruzzone, J. L. Rojo-Alvarez, and F. Melgani. Robust support vector regression for biophysical variable estimation from remotely sensed images. *Geoscience and Remote Sensing Letters, IEEE*, 3(3):339–343, 2006.

[4] H. Chang and D. Y. Yeung. Robust path-based spectral clustering. *Pattern Recognition*, 41:191–203, 2008.

[5] Y. Chi, X. Song, D. Zhou, K. Hino, and B. L. Tseng. On evolutionary spectral clustering. *ACM TKDD*, 3(4):1–30, 2009.

[6] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006.

[7] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[8] I. W. Evett and E. J. Spiehler. Rule induction in forensic science. 1987.

[9] R. A. Fisher. The use of multiple measurement in taxonomic problems. *Annual Eugenics*, 7(2):179–188, 1936.

[10] A. Grigor'yan. Estimates of heat kernels on riemannian manifolds. In *Proceedings on Spectral Theory and Geometry. ICMS Instructional Conference*, pages 140–225, 1999.

[11] I. Gutman and W. Xiao. The generalized inverse of the laplacian matrix and some applications. *Bulletin TCXXIX de lAcadmie serbe des sciences et des arts2004 Classe des sciences mathematiques et naturelles No 29*, pages 1–9, 2004.

[12] J. A. Hartigan and M. A. Wong. Algorithm as 136: a k-means clustering algorithm. *Appl. Stat.*, 28:100–108, 1978.

[13] E. Hsu. Stochastic analysis on manifolds. *Graduate Studies in Mathematics*, 38, 2002.

[14] P. J. Huber. *Robust Statistics*. New York:Wiley.

[15] A. Kale, A. Karandikar, P. Kolari, A. Java, T. Finin, and A. Joshi. Modeling trust and influence in the blogosphere using link polarity. *In Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*, 2007.

[16] S. Lafon, Y. Keller, and R. R. Coifman. Data fusion and multicue data matching by diffusion maps. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28:1784–1797, 2006.

[17] Z. Li, J. Liu, S. Chen, and X. Tang. Noise robust spectral clustering. In *Proceedings of IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

[18] U. V. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[19] M. Meila and J. Shi. A random walks view of spectral segmentation. *8th International Workshop on Artificial Intelligence and Statistics*, 2001.

[20] B. Nadler, S. Lafon, R. Coifman, and I. Kevrekidis. Diffusion maps, spectral clustering and eigenfunctions of fokkerplanck operators. *NIPS*, 2005.

[21] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E69*, 2004.

[22] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856, 2002.

[23] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchelle. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39:103–134, 2000.

[24] V. Y. Pan and Z. Q. Chen. The complexity of the matrix eigenproblem. In *STOC'99 Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 507–516, 1999.

[25] P. Perona and W. T. Freeman. A factorization approach to grouping. In *Proceedings of the 5th European Conference on Computer Vision*, pages 655–670, 1998.

[26] H. Qiu and E. R. Hancock. Clustering and embedding using commute times. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(11):1873–1890, 2007.

[27] J. W. Richards, P. E. Freeman, A. B. Lee, and C. M. Schafer. Accurate parameter estimation for star formation history in galaxies using sdss spectra. In *MNRAS,399*, pages 1044–1057, 2009.

[28] A. Strehl and J. Ghosh. Cluster ensembles ł a knowledge reuse framework for combining multiple partitions. In *J. Mach. Learn. Res.*, pages 583–617, 2003.

[29] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *SGP*, 2009.

[30] H. Valizadegan and R. Jin. Generalized maximum margin clustering and unsupervised kernel learning. *NIPS*, 19:1417–1424, 2007.

[31] D. Verma and M. Meila. Comparison of spectral clustering methods. *UW CSE Technical report*, 2001.

[32] L. Zelnik-manor and P. Perona. Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems 17*, pages 1601–1608, 2004.