JEET IU.2017; 10(3): 980-995
http://doi.org/11.5271/JEETIU.2017.15.00658

ISSN(Print) 1245-0153
ISSN(Online) 2093-7423

A New Sparse Graph Construction applied for Face Recognition

E.Talebi and M.Hosseini

Department of Engineering, Islamic Azad University, Buin-Zahra Branch, Iran

Designing and construction of a good graph, which can represent the data structure even in the presence of noise appropriately, is necessary for spectral clustering. In this paper, we propose a new graph construction method that can capture not only the global mixture of subspaces structure, but also the locally linear structure of the data by low rankness and sparseness. In the graph constructed by the proposed method, vertices are the data and edge weights (affinity matrix) are defined in form of $W = Z^T Z$, which matrix of Z is sparse representation of the data and is obtained based on a dictionary (D) using orthogonal matching pursuit (OMP) algorithm. Dictionary plays an important role in this study. Therefore, we have chosen it as a designed dictionary and adapt its atoms by K-SVD algorithm which contains special characters to fit data and make a good graph. We have also defined a new graph evaluation criterion and examined its performance by other existing criteria. Also, we used the extended YALE-B, COIL-100, ORL and the TDT2 as benchmark data sets and evaluated and compared the proposed graph construction method with other methods, e.g., *L1*-graph, non-negative low-rank and sparse (NNLRS) graph, Gaussian kernel graph and k-nearest-neighbor (KNN) graph, by new criterion and other evaluation criteria. Our simulations results confirm the superior performance of the proposed method compared to the others, especially when the data are noisy and contain outliers or lie in the intersection of multiple subspaces.

Keywords: *graph embedding, subspace learning, spectral clustering, sparse representation, k-SVD dictionary, orthogonal matching pursuit (OMP).*

1 Introduction

Since graphs include useful information about the data, nowadays are too taken into consideration. Many graph-oriented algorithms have been proposed that are based on the features which graphs are, and used for spectral clustering and subspace learning. Some works on manifold learning, e.g., Laplacian Eigenmaps [1], ISOMAP [2] and locally linear embedding [3], some others work on most popular subspace learning algorithms, e.g., principal component analysis (PCA) [4], linear discriminant analysis (LDA) [5], and locality preserving projections [6]. In addition, there is some semi-supervised learning algorithms that have been presented by Zhu *et al.* [7], Belkin and Niyogi [8], and all are dependent on the graphs which constructed in different methods.

Graph-oriented algorithms model the relationships between the data points as a graph, which nodes are data and pairwise edges between these nodes are the similarity between the corresponding data samples. In most papers, the graph representing matrix which is a block-diagonal matrix called affinity matrix. Based on this matrix a lot of algorithms have been presented for different tasks, e.g., data clustering and feature extraction. In most related studies, affinity matrix construction can be done in two ways. One is based on pairwise distances and using a criterion, e.g., Euclidean distance [9], and the other is based on reconstruction coefficients which assumes each data point represented as a linear combination of the other points. In fact, the second method implies the existence of same features in the intra-subspace points [9], [10]. When When data are without noise and particular problems,

graph construction can be done in simple way as [9], [10], [11] to recover the subspace structures. However, when the data are noisy and contain outliers or lie in the intersection of multiple subspaces, it seems hard to construct a good graph. In such circumstances, graph's affinity matrix contains significant connection weights in the inter-class data point's area. Removing these types of errors and connections is important in constructing a good graph and has been considered by many researchers, e.g., Sparse Subspace Clustering [12], Locally Linear Manifold Clustering [13], Fixed Rank Representation [14], Agglomerative Lossy Compression [15], Latent Low Rank Representation [16], *L1*-graph [17], Spectral Multi-Manifold Clustering [18], Low Rank Representation [19] and Least Squares Regression [20].Generally, good graph is a graph which have some characteristics as high discriminating power, low sparsity, low rankness and adaptive neighborhood to can reveal the true intrinsic dimensionality of the data and also capture certain global structures of the data as a whole (i.e. subspaces, multiple clusters, or manifolds) [21]. Most of the existing methods such as Locally Linear Reconstruction [22], L1-graph [23] and k-nearest neighbors can only capture the local structures and cannot capture the global structures of the data (i.e. the clusters), and also are very sensitive to local data noise and errors or at least in some cases (i.e. *L1*-graph) when no extra "clean data" are available, these methods may not be robust to noise and outliers [23]. Also, there are some

works as Non-negative low-rank and sparse (NNLRS) graph [24], which are both generative and discriminative. However, computational complexities and lack of good quality to reveal data structures are their problems.

Although, the existing graph construction methods have been somewhat successful, but finding a better way is still open. In this paper, a new method presented to construct a good graph based on the sparse representation of data which encoding each data sample over the dictionary. Dictionary training performed using K-SVD algorithm that leads to the best possible representations for each data with strict sparsity constraints. Also, sparse representation of data obtained using orthogonal matching pursuit (OMP) algorithm which proves to be the best, especially from a run-time point of view [23]. The K-SVD algorithm includes special characters which enable us to construct a good graph, e.g., an effective sparse representation, a Gauss–Seidel-like accelerated dictionary update procedure and direct trainability (by primary data that may be noisy and contain outliers or even lie in the intersection of multiple subspaces). This algorithm consecutively calculates the sparse representation of the data and updates the dictionary atoms to better fit the data. The update of the dictionary atoms is based on the singular value decomposition (SVD) and combined with an update of the sparse representations, thereupon accelerating convergence.

The remainder of this paper is organized as follows: Section 2 describes sparse coding along with K-SVD and OMP algorithms which play key role in the proposed method. Section 3 is about two well-known evaluation criteria, Accuracy (Ac) and Normalized Mutual Information (NMI). Moreover, our proposed method, which contains graph construction algorithm and a new evaluation criterion, is introduced in Section 4, and Section 5 reports the results of a series of experiments which examine the effectiveness of the proposed method. Finally, we provide some concluding remarks in Section 6.

2 The necessary preliminaries

The method which proposed in this paper is based on the sparse representation of data and its particular belongings. The key point of the proposed method that leads to construct a good graph is to apply K-SVD and OMP algorithms for training a dictionary and finding a sparse representation of data. Therefore, in the following we will explain them briefly.

Sparse representation of data is of the issues which recently too considered by researchers [32]. Sparse representation of data is the possibility of making data based on linear combination of a small number of atoms in the dictionary. It depends on the quality of the considered dictionary in justifying the characteristics of data. Thus choosing or making dictionary is of particular importance. The problem of sparse representation of a signal which commonly referred to as "atom decomposition" formulated as follows [32], [33] and can be solved by a "pursuit algorithm" that finds an approximated solution:

$$\min_z \|z\|_0 \quad \text{s.t.} \quad \|y - Dz\|_2 \le \varepsilon \qquad (1)$$

where $D \in R^{n \times k}$ is the dictionary matrix, $y \in R^n$ is a signal which can be represented as a sparse linear combination of some atoms, $z \in R^k$ contains the representation coefficients of the signal y, and $\| \ \|_p$ is l^p -norm.

A lot of applications can gain from sparse representation of signal, e.g., feature extraction, compression, signal denoising and inpainting [33]. In this paper, features of the sparse representation of data are used to construct a good graph.

- **K-SVD ALGORITHM**

K-SVD algorithm is introduced for training of dictionaries and contains flexibilities which make it possible to work in conjunction with any pursuit algorithm [32]. But the capability of K-SVD in well justifying the characteristics of signals attracted us to use this method for graph construction. This method is highly efficient, due to an effective sparse representation and a Gauss–Seidel-like accelerated dictionary update method. The K-SVD alternates between sparse representation of the data and updating the dictionary atoms to better fit the data. The update of the dictionary atoms is based on the singular value decomposition (SVD), rank one approximation, and combined with an update of the sparse representations, therefore, convergence is accelerated. K-SVD algorithm is used as [32] in the proposed method.

- **THE ORTHOGONAL MATCHING PURSUIT ALGORITHM**

Many pursuit algorithms have been proposed to approximate solutions in the problems of sparse representation, e.g., Matching Pursuit (MP), Basis Pursuit (BP), FOcal Underdetermined System Solver (FOCUSS) frames, Orthogonal Matching Pursuit (OMP) and etc. [34], [35], [36]. The OMP is greedy and a stepwise

forward selection algorithm which simply computes the sparse representation of data based on the inner product between the data and dictionary columns and deploying some least squares solvers. In each step, the column of dictionary that has the most correlation with the residual columns, was selected and add into the already selected columns. Some good aspects of this algorithm motivated us to use it in our graph construction procedure, e.g., robustness against noisy data and run-time point of view [23]. OMP algorithm is used as [23].

3 EVALUATION CRITERION

In this part of the paper, two well-known criteria are introduced for comparing the performance of the different graph construction algorithms in view point of clustering. First criterion is Ac (the Accuracy, sometimes called the purity) [41] and has been defined as Eq.2.

$$Ac = \frac{\sum_{i=1}^{N} \delta(\tilde{L}_{(i)}, Map(L, \tilde{L})_{(i)})}{N} \quad (2)$$

where L and $\tilde{L}$ respectively are the clustering result labels and the known thruth data labels, N is the number of data samples,), $Map(L,\tilde{L})$ is the Kuhn-Munkres algorithm results as the best mapping function, [40]. If A and B are the index sets of all samples in L and $\tilde{L}$, $Map(L,\tilde{L})$ permutes the index set A in L to match the index set B in $\tilde{L}$. Besides, $\delta(a,b)$ is a function that equals to 1 if and only if $a=b$. So, in fact, Ac computes the number of correct clustering.

The second criterion is NMI (Normalized Mutual Information) [42]. Let's consider $p(a)$ and $p(b)$ are the marginal probability distribution functions of A and B, $p(a,b)$ is the joint probability distribution function, $H(A)$ and $H(B)$ are the entropies. Therefore the mutual information between A and B is defined as:

$$MI(N,M) = \sum_{n\in N}\sum_{m\in M} p(n,m)\log_2\left(\frac{p(n,m)}{p(n)p(m)}\right) \quad (3)$$

and the normalizes mutual information would be as Eq. 4.

$$NMI(A,B) = \frac{MI(A,B)}{\max(H(A),H(B))} \quad (4)$$

4 THE PROPOSED METHOD

Our innovations contain two parts that respectively are the new graph construction method and a new evaluation criterion which considered in follows.

- **A New graph construction method (K-SVD graph)**

The method is based on the reconstruction coefficients which assumes each data point represented as a linear combination of a small number of atoms in a special dictionary. Dictionary is chosen as a designed dictionary and adapted its atoms by K-SVD algorithm as mentioned in the section 3.2. In the proposed method, affinity matrix is defined as bellow:

$$W = Z^T Z \quad (5)$$

Where matrix of Z is the sparse representation of data and is obtained based on K-SVD dictionary (D) using orthogonal matching pursuit (OMP) algorithm. A full description of the proposed algorithm is given in the bellow:

Target: *Constructing a K-SVD graph.*

Step1: *Initializing the dictionary matrix (D), sparseness coefficient (T_0), error threshold (e_{th}), lagrangian coefficient (λ) and noise gain (G).*

Step2: *Running the K-SVD algorithm as mentioned in section II to make dictionary (D).*

Step3: *Calculating the sparse representation of data (Z) based on D and orthogonal matching pursuit algorithm.*

Step4: *Constructing the affinity matrix as bellow:*

$W = Z^T Z$

This kind of dictionary selection as well as finding the reconstruction coefficients using OMP algorithm lead to construct a good graph which contains both of the global mixture of subspaces structure and the locally linear structure of the data and we called it K-SVD graph. This graph is sparse and low rank and also has high discriminating power and suitable robustness against noisy data. Obviously, the experimental results section can confirm these issues.

- **A new evaluation criterion (Graph Laplacian Eigen Space Evaluator, GLESE)**

We introduce a new criterion to compare more the effectiveness of the algorithms. In this evaluation method, we assume consecutively arranged the data samples

which are belonged to the same cluster. Therefore, apply the spectral decomposition procedure on data as follows:

***Target**: Spectral decomposition*

***Input**: Affinity matrix (W)*

***Step1**: Calculating the diagonal degree matrix (M) and laplacian matrix as follows.*

$$m_{ii} = \sum_j w_{ij} \quad , \qquad L = M - W$$

***Step2**: Then calculating the eigen vectors and eigen values as bellow.*

$$LV = SV$$

***Outputs:**Eigen vectors matrix (V) and diagonal matrix of Eigen values (S)*

After the spectral decomposition, the resulted Eigen vectors and eign values have information that used for spectral clustering [1], [9]. For example, Fiedler in [43] classified data into two groups using sign of second Eigen vector or NJW [22] uses *K* eigenvectors, corresponding to to the K largest eigenvalues, of the normalized affinity matrix of data to cluster data.

Thus, when data arranged consecutively, a good Eigen vector has meaningful structure. In better words, a good Eigen vector relatively has same value for samples which are in the same cluster. Thereupon, a good Eigen vector almost has template in time domain or has dominant harmonic in frequency domain. Figure 1 shows two samples Eigen vectors, that the bottom vector has dominant harmonic and known as a good Eigen vector.

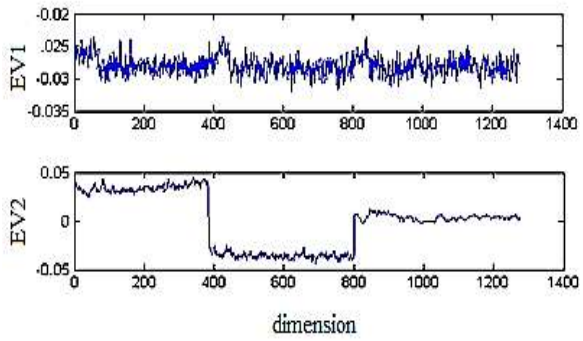


Figure 1: Sample Eigen vectors: ***EV1*** (without information), ***EV2*** (has template and dominant harmonic)

Consequently, the new criterion introduced as Eq.8 and we called it "Graph Laplacian Eigen Space Evaluator" (GLESE). GLESE evaluates information of each Eigen vector by comparing the strongest harmonic (maximum coefficient of DCT) with average of others. So assume that $V \in R^{M \times N}$ is Laplacian Eigen space and resulted after the spectral decomposition, and $v_i \in R^{1 \times N}$, $i = 1,2,\ldots,M$ is each Eigen vector of V, and $Q_i(k)$ is the absolute value of the discrete cosine transform of each Eigen vector and defines as Eq.6:

$$Q_i(k) = \left| \sum_{n=1}^{N} v_i(n) Cos(\frac{\pi(2n-1)(k-1)}{2N} \right|$$

$$i = 1,2,\ldots,M \ , \ k = 1,2,\ldots,N \qquad (6)$$

Now Eigen Vector Evaluator (EVE) and GLESE could be defined as:

$$EVE_i = \frac{mean(Q_i)}{\max(Q_i)} - \frac{1}{N} \ , \ i = 1,2,\ldots,M \qquad (7)$$

$$GLESE = \sum_{i=1}^{M} EVE_i \qquad (8)$$

Eq.8 evaluates all of the Eigen vectors that Eigen space has, and a good Eigen space (corresponding to a good graph) minimizes it. It is necessary that all transforms which have ability to establish the frequency of changes in the eigenvectors could be used. We decided to use DCT as a simple and real transform.

5 EXPERIMENTAL RESULTS

In this section, several different data sets from different machine learning applications have been selected for experiments. They are: the extended YALE-B face database [37] (contains 38 individuals and around 64 near near frontal images), COIL-100 database [38] (contains 7,200 color images of 100 objects), The ORL database [39] (contains 10 different images for each of 40 distinct subjects) and the TDT2 database [39] (consists of 11,201 on-topic documents which are classified into 96 semantic categories), These data sets contain different formats of data and were considered as the benchmarks in this work. In simulations, we considered our new graph evaluation criterion along with other criteria. Also, we select some of the most prominent graph constructing methods such as Gaussian kernel graph, L1-graph, non-negative low-rank and sparse (NNLRS) graph and k-nearest-neighbor (kNN) graph for comparison with our new graph constructing method. To constrauct our new graph, for all used databases, the dictionary used were of size 100×625 and initialized by DCT atoms, the value of noise gain, G, is selected to be 1.15, iteration number is 80 and lagrangian coefficient is $\lambda = 30 / sigma$. Where *sigma* is the variance of noise.

Sparse representation of data depends on the ability of the made dictionary in justifying the characteristics of a data. Therefore, the quality of the trained dictionary investigated at all stages by computing the mean

reconstruction error (RMSE: root mean square error). the quality of dictionary improved when dictionary was updated several times by K-SVD algorithm.

If using the made dictionary to construct a graph, resulting graph is denser, lower rank and better than a graph which may be made in other ways. Figure 2 indicates a specific example, where superiority of our graph is evident in comparison with Gaussian kernel graph.

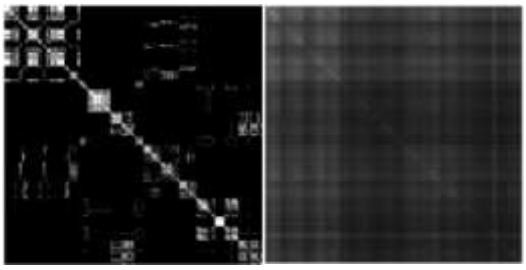


Figure 2: Affinity matrix of: Gaussian kernel graph (***left***) and proposed method (***right***)

Table 1: Clustering quality based on best Ac and corresponding NMI for different algorithms and databases

Data Sets	**Criterion**	**K-SVD graph**	**GK-graph**	**NNLRS-graph**	**L1-graph**	**kNN graph**
EX-YALE-B	Ac	***0.805***	0.398	0.801	0.702	0.558
	NMI	***0.814***	0.491	0.843	0.718	0.435
COIL-100	Ac	***0.827***	0.215	0.774	0.632	0.544
	NMI	***0.866***	0.482	0.852	0.681	0.571
ORL	Ac	***0.852***	0.187	0.684	0.803	0.453
	NMI	***0.837***	0.481	0.773	0.824	0.482
TDT2	Ac	***0.801***	0.191	0.787	0.743	0.524
	NMI	***0.889***	0.435	0.868	0.757	0.591

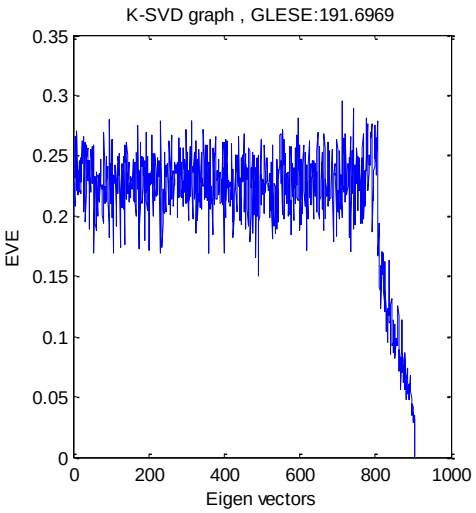
In this example, the rank of the affinity matrices respectively calculated 188 and 100. Rank of the proposed method calculated was lower because the necessary number of atoms in dictionary which could represent the data well was much less than the number of data samples. So, the affinity matrix obtained by the proposed method includes both of properties of the lower rank and more sparsity.

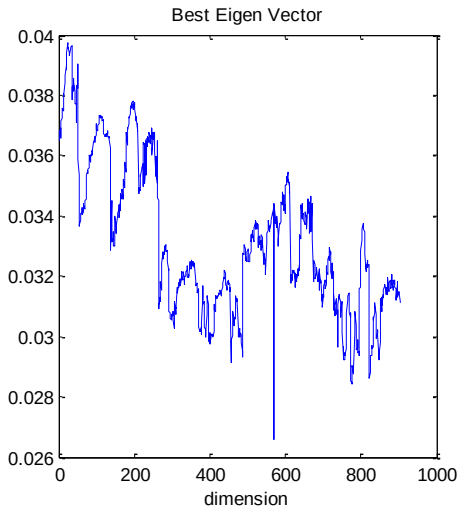
In the following sections the effectiveness of the proposed method by various analyses is examined. First, evaluate the effectiveness of the proposed method by comparing the results of spectral clustering using the clean data and then repeat it by the corrupted data. For this purpose, evaluate the performance of different algorithms using Ac and NMI as well as GLESE.

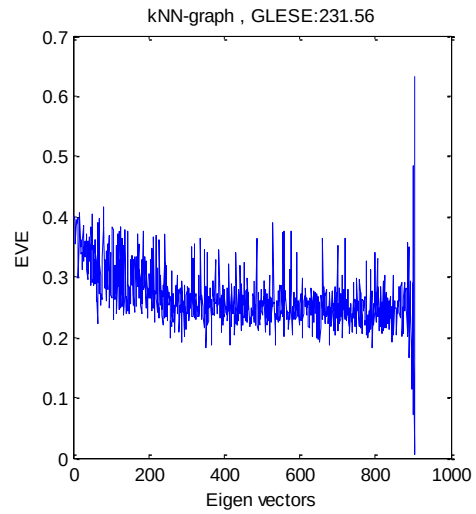
The clustering quality of the proposed method (K-SVD graph), Gaussian kernel graph (GK), l_1-graph, non-negative low-rank and sparse (NNLRS) graph and k-nearest-neighbor (kNN) graph are investigated in this subsection. Meantime, we used all four data sets for comparison.

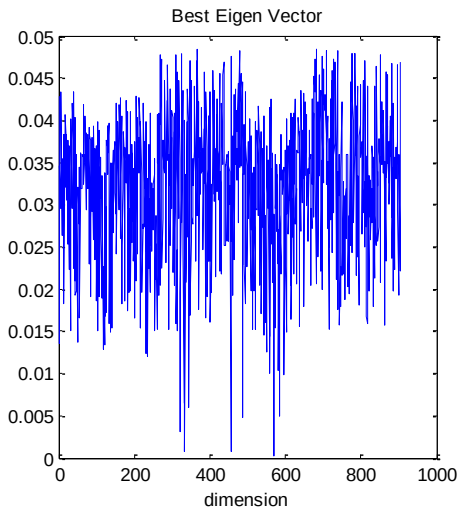
- **SPECTRAL CLUSTERING OF CLEAN DATA**

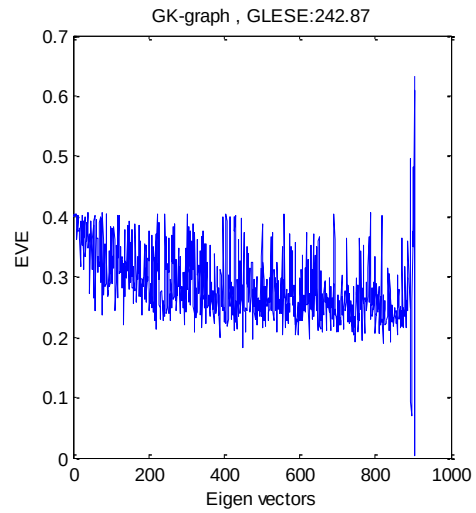
In this part of the experiments, the performance of the proposed method compared with others in the spectral clustering by the clean data. For this purpose, we used Ac and NMI as mentioned before. The comparative results on clustering accuracy are listed in Table 1, which confirm the superiority of the proposed algorithm in the facing different databases. As seen in Table 1, the clustering results of the proposed algorithm are better than those from all the others for both criteria. Also, NNLRS graph and l_1-graph are relatively better and more stable than GK-graph and kNN-graph. Meanwhile, all the mentioned results in Table 1 achieved when tuning well the parameters of the algorithms. Furthermore, we used the made criterion, GLESE, to compare the performance of the mentioned algorithms on data sets (Table 2). Meanwhile, for more comparison Figure 3 illustrates the EVE (Eq.7) of all Eigen vectors evaluation results and the GLESE (Eq.8) Eigen space evaluation results for all the mentioned algorithms on ExYale-B database.

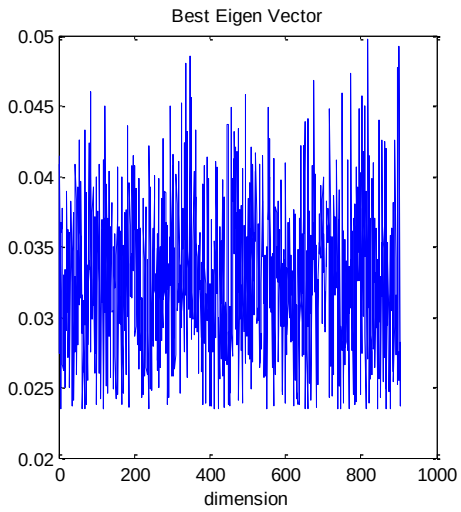

K-SVD graph , GLESE:191.6969
EVE
Eigen vectors


Best Eigen Vector
dimension


kNN-graph , GLESE:231.56
EVE
Eigen vectors


Best Eigen Vector
dimension


GK-graph , GLESE:242.87
EVE
Eigen vectors


Best Eigen Vector
dimension

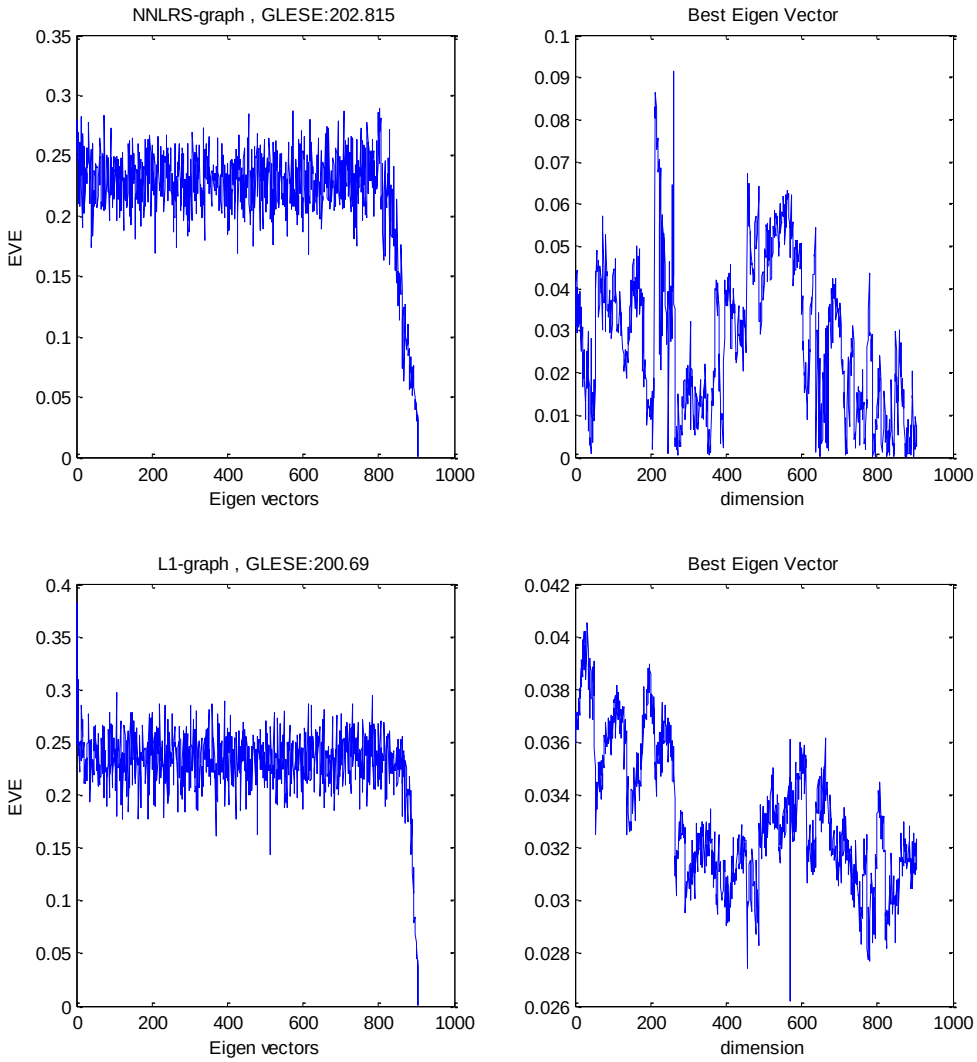


Figure 3: GLESE results on ExYale-B database (***left column***: Eigen vectors evaluation results (EVE) for different methods, GLESE of Eigen space evaluation result has been listed above the figure, ***right column***: the best Eigen vector with minimum EVE corresponding to the evaluation).

Table 2: GLESE results for different algorithms and databases

Data Sets	**K-SVD graph**	**GK-graph**	**NNLRS-graph**	**L1- graph**	**kNN graph**
EX-YALE-B	***191.69***	242.87	202.81	200.69	231.56
COIL-100	***181.22***	237.46	196.52	204.47	226.33
ORL	***195.15***	244.18	207.55	209.78	236.42
TDT2	***207.13***	263.71	223.12	244.90	256.87

As seen in Figure 3, the best Eigen vector corresponding to the proposed method has better template in time domain and strongest dominant harmonic in frequency domain than other algorithms. Hence, it has better information which can be used for classification. Generally, the GLESE results follow the Ac and NMI evaluations, and also confirm the superiority of the proposed method in comparison with others (Figure 3 and Table 2).

- **SPECTRAL CLUSTERING OF CORRUPTED DATA**

In this part of the paper, the performance of the proposed method in clustering by corrupted data was examined. For this purpose, the data sets corrupted with additive Gaussian noise. Performance of all algorithms was compared by Ac, NMI and GLESE. Results of the algorithms based on Ac and NMI are mentioned in Table 3 when peak signal to noise ratio (PSNR) chosen 6 dB (for more corruption). PSNR in this work calculated as Eq.9.

$$PSNR = 10\log_{10}\frac{1}{e^2} \tag{9}$$

where e is the mean squared error between original data and processed data. A comparison using GLESE for different PSNR and algorithms illustrated in Figure 4.

Table 3: Clustering quality based on best Ac and corresponding NMI for corrupted data (PSNR=6dB)

Data Sets	**Criterion**	**K-SVD graph**	**GK-graph**	**NNLRS-graph**	**L1- graph**	**kNN graph**
EX-YALE-B	Ac	***0.812***	0.125	0.512	0.631	0.326
	NMI	***0.815***	0.234	0.534	0.718	0.417
COIL-100	Ac	***0.819***	0.137	0.463	0.637	0.319
	NMI	***0.843***	0.236	0.527	0.564	0.423
ORL	Ac	***0.830***	0.124	0.378	0.513	0.349
	NMI	***0.842***	0.261	0.536	0.745	0.389
TDT2	Ac	***0.814***	0.147	0.512	0.612	0.337
	NMI	***0.862***	0.246	0.597	0.601	0.431

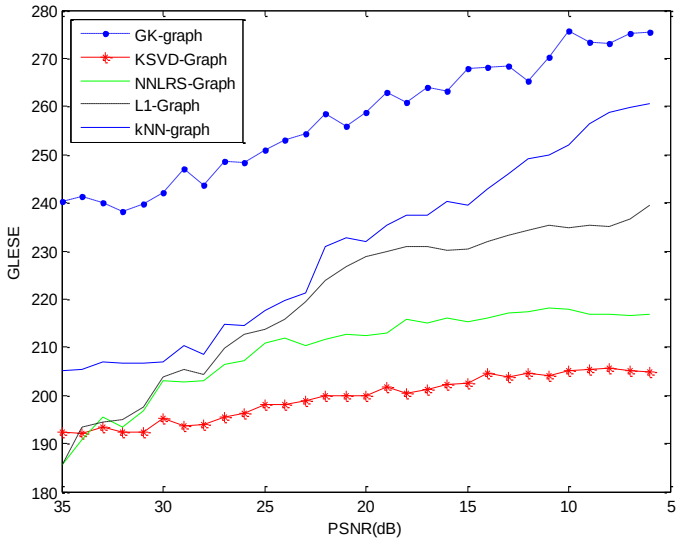


Figure 4: the GLESE comparative results for different algorithms

As seen in the Figure 4, although the performance of the proposed method is comparable with NNLRS-graph and *L1*-graph for larger PSNR, but it increases for lower PSNR which indicates its greater robustness against noisy data. Figure 5 shows this issue based on PSNR ratio for different noise level on a specific sample of ExYale-B database. In fact, Figure 5 shows the denoising capability of the K-SVD based data representation. These results achieved when data reconstructed after the encoding by K-SVDdictionary.

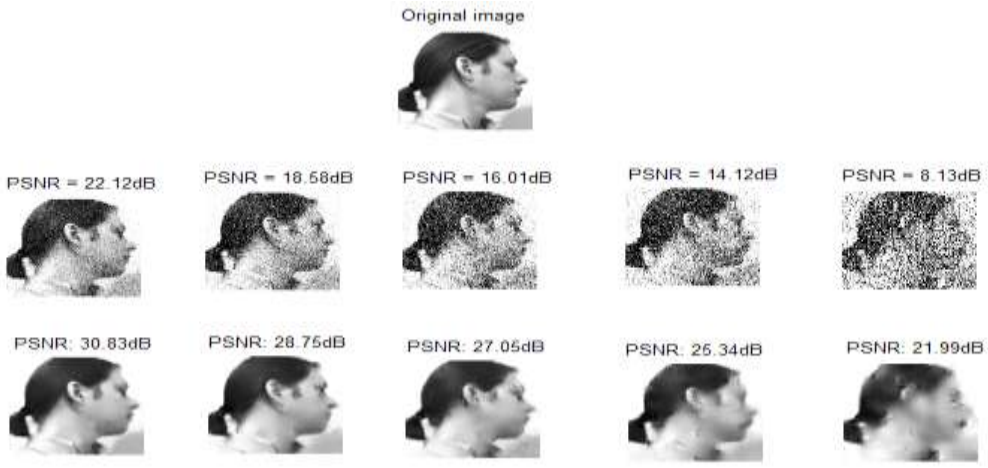


Figure 5: Robustness of the proposed method based on K-SVD in facing different noise level (***top row***: noisy data, ***bottom row***: reconstructed data based on K-SVD)

6 CONCLUSION

In this paper, we proposed a new graph construction method which can be used for development of the graph-based machine learning tasks. In this work, the weights of the affinity matrix are derived by nonnegative sparse representation of data based on K-SVD dictionary and using OMP algorithm which naturally contain special characters to fit data and make a good graph, especially when the data are noisy and contain outliers or lie in the intersection of multiple subspaces. Experimental results have shown the superior performance of the proposed method compared to others and also its sparseness, low rankness, high discriminating power and greater robustness against noisy data.

References:

[1] Belkin M, Niyogi P. Laplacian eigen maps for dimensionality reduction and data representation. *Neural Comput.* 2002; 15:1373–1396.

[2] Tenenbaum J, Silva V, Langford J. A global geometric frame work for nonlinear dimensionality reduction. *Science 2000;* 290:2319–2323.

[3] Roweis S, Saul L. Nonlinear dimensionality reduction by locally linear embedding. *Science 2000;* 290: 2323–2326.

[4] Joliffe I. *Principal Component Analysis*. New York: Springer-Verlag, 1986. pp. 1580–1584.

[5] Belhumeur P, Hespanda J, Kiregeman D. Eigen faces vs. fisher faces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell. 1997;* 19:711–720.

[6] He X, Niyogi P. Locality preserving projections. *Adv. NeuralInf. Process.Syst.2003;* 16:585–591.

[7] Zhu X, Ghahramani Z, Lafferty J. semi-supervised learning using gaussian fields and harmonic functions. in *Proc. Int. Conf. Ma-chine Learning*, 2003, pp. 912–919.

[8] Belkin M, Matveeva I, Niyogi P. Regularization and semi-supervised learning on large graphs. in *Proc. Int. Conf. Learning Theory*, 2004, pp. 624–638.

[9] Jordan I, Weiss Y. On spectral clustering: Analysis and an algorithm. in *Proc. of Advances in Neural Information Processing Systems*, 2002, pp. 849–856.

[10] Costa J, Hero A. Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *IEEE Transactions on Signal Processing*, 2004; 52: 2210–2221.

[11] He X, Niyogi P. Locality preserving projections. in *Proc. Of Advances in Neural Information Processing Systems*, 2003;16:153.

[12] Elhamifar E, Vidal R. Sparse subspace clustering. in *Proc. Of IEEE Conference on Computer Vision and Pattern Recognition*, 2009,pp. 2790–2797.

[13] Goh A, Vidal R. Segmenting motions of different types by unsupervised manifold clustering. in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 2032–2037.

[14] Risheng L, Lin Z, Su Z. Fixed-rank representation for unsupervised visual learning. in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[15] Rao S, Tron R, Vidal R, Ma Y. Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2010;* 32: 1832–1845.

[16] Liu G, Yan S. Latent low-rank representation for subspace segmentation and feature extraction. in *Proc. of IEEE International Conference on Computer Vision.* 2011, pp. 1615–1622.

[17] Cheng B, Yang J, Yan S, Fu Y, Huang T. Learning with ℓ1-graphfor image analysis. *IEEE Transactions on Image Processing*, 2010;19: 858–866.

[18] Yong W, Yuan Y, Yi W, Zhi-Hua W. Spectral clustering on multiple manifolds. *IEEE Transactions on Neural Networks*, 2011; 22: 1149–1161.

[19] Liu G, Lin Z, Yu Y. Robust subspace segmentation by low-rank representation. in *Proc. of the International Conference on Machine Learning*, 2010.

[20] Lu C.Y, Min Y, Zhao Z, Zhu Z, Huang Z, Yan S. Robust and efficient subspace segmentation via least squares regression. in *Proc. of European Conference on Computer Vision*, Springer Berlin Heidelberg, 2012, pp. 347–360.

[21] Wright J, Ma Y, Mairal J, Sapiro G, Huang T, Yan S. Sparse representation for computer vision and pattern recognition. Proceedings of IEEE, 2010,1031–1044.

[22] Jebara T, Wang J, Chang S. Graph construction and b-matching for semi-supervised learning. In ICML, 2009, pp 441–448.

[23] Cheng B, Yang Y, Yan S, Fu Y, Huang T. Learning with `L1-graph for image analysis. IEEE Trans. on Image Processing, 19(4), 2010.

[24] Zhuang L, Gao H, Non-Negative Low Rank and Sparse Graph for Semi-Supervised Learning, IEEE conf. on Computer vision(CVPR), 2012.

[25] Elhamifar E, Vidal R. Sparse subspace clustering: Algorithm, theory, and applications. Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012.

[26] Liu J, Ji S, Ye J. Multi-task feature learning via efficient l2;1-norm minimization. In Proc. of the 25th Conf. on Uncertainty in Artificial Intelligence. ACM, 2009.

[27] Liu G, Lin Z, Yu Y. Robust subspace segmentation by low-rank representation. in *Proc. of the International Conference on Machine Learning*, 2010.

[28] Guangcan L, Zhouchen L, Shuicheng Y, Ju S, Yong Y, Yi M. Robust recovery of subspace structures by low-rank

representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013; 35: 171–184.

[29] Yang J, Zhang Y. Alternating direction algorithms for ℓ1-problems in compressive sensing. *SIAM journal on scientific computing*, 2011; 33: 250–278.

[30] Plaku E, Kavraki L. Distributed computation of the *k*nn graph for large high-dimensional point sets. *Journal of Parallel and Distributed Computing*, 2007;67(3):346–359.

[31]Shakhnarovich, Darrell T, Indyk P, editors. *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*. The MIT Press, 2006.

[32] Aharon M, Elad M, Bruckstein A. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. IEEE transactions on signal processing, vol. 54, nov, 2006.

[33] Gorodnitsky I, Rao B. Sparse signal reconstruction from limited data using FOCUSS: A re-weighted norm minimization algorithm. IEEE Trans. Signal Process, 45: 600–616, 1997.

[34] Mallat S, Zhang Z. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.*, 1993; 41: 3397–3415.

[35] Chen S, Billings S, Luo W. Orthogonal least squares methods and their application to non-linear system identification. *Int. J. Contr.*, 1989;50:1873–96.

[36] Tropp J. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory*,2004; 50: 2231–2242.

[37] Lee K, Ho J, Kriegman D. Acquiring linear subspaces for face recognition under variable lighting IEEE Trans. Pattern Anal. Mach.Intell, 2005;27: 684–698.

[38] Nene S, Nayar S, Murase H. Columbia object image library (coil-20). *Technical Report CUCS-005-96*, 1996.

[39] Wright J, Yang A, Ganesh A, Sastry A, Ma Y. Robust face recognition via sparse representation IEEE Trans. on Pattern Analysis and Machine Intelligence, 2009;31:210–227.

[40] Chen S, Donoho D, Saunders M. Atomic decomposition by basis pursuit. Soc. Ind. Appl. Math. Rev., 2001;43:129–159.

[41] Zhao Y, Karypis G. Criterion functions for document clustering: experiments and analysis. Department of Computer Science, University of Minnesota, Tech. Rep., 2001.

[42] Cai D, He X, Han J. Document clustering using locality preserving indexing. IEEE Transactions on Knowledge and Data Engineering, 2005;17:1624–1637.

[43] Fiedler M. A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. Czechoslovak Mathematical Journ