Dimension Selected Subspace Clustering

Shuoyang Li[1], Yuhui Luo[2], Jonathon Chambers[3], and Wenwu Wang[1]

[1]Department of Electrical and Electronic Engineering, University of Surrey, UK
[2]Data Science Department, National Physical Laboratory, UK
[3]College of Science and Engineering, University of Leicester, UK

***Abstract*—Subspace clustering is a popular method for clustering unlabelled data. However, the computational cost of the subspace clustering algorithm can be unaffordable when dealing with a large data set. Using a set of dimension sketched data instead of the original data set can be helpful for mitigating the computational burden. Thus, finding a way for dimension sketching becomes an important problem. In this paper, a new dimension sketching algorithm is proposed, which aims to select informative dimensions that have significant effects on the clustering results. Experimental results reveal that this method can significantly improve subspace clustering performance on both synthetic and real-world datasets, in comparison with two baseline methods.**

I. Introduction

Subspace clustering is a powerful technique for learning class labels of non-linearly separable data which lie in a union of subspaces in a high-dimensional ambient space. A high dimensional dataset can be viewed as lying in a low-dimensional intrinsic subspace [1], and subspace clustering methods aim at grouping them according to their intrinsic subspace distributions. Subspace clustering has drawn increasing research attention recently, in applications involving high-dimensional data [2], such as image segmentation [3] and motion segmentation [4].

Spectral clustering, which is a widely used subspace clustering method, is shown to outperform traditional clustering approaches [5]. Unlike k-means [6] and fuzzy clustering [7] which rely on the distances between data points and centroids of the clusters, spectral clustering-based methods utilise distances or correlations among points, without regard to the centroids.

However, features of real-world datasets are sometimes of high dimension and large volume, which lead to a heavy computational load with the clustering algorithms. To reduce the computational cost with high dimensional datasets, a dimensionality reduction process can be performed. This can be achieved with classical techniques such as principal component analysis (PCA), however, it requires the calculation of the covariance matrix which can be computationally expensive for high dimensional data. Recently, sketching methods have been proposed to handle subspace clustering algorithms, such as the random sketching algorithm which relies on the restricted isometry property of a Johnson-Lindenstrauss transform [8] [9], and random sampling based algorithms [10]. Nevertheless, these dimension-reduction methods do not consider the structure of datasets. For example, in the Modified National Institute of Standards and Technology (MNIST) handwritten digits dataset [11], some pixels of the edge and corner areas are equal across different images, and appear to induce redundant dimensions.

The objective of this paper is to introduce a dimension selection algorithm which aims to remove such uninformative dimensions. Unlike conventional dimension-sketching methods, the proposed algorithm measures the change of the affinity matrix, and estimates the importance of different data dimensions. By eliminating the dimensions which have low effect on the clustering result, the dimensions which best represent the original data matrix are retained. Subspace clustering algorithms can then be performed with the dimension selected data matrix. With the proposed algorithm, the computational load can be substantially reduced, and the clustering accuracy can be retained favorably. The proposed algorithm can be used with a variety of subspace clustering algorithms such as Sparse Subspace Clustering (SSC) [12], Low-rank Representation (LRR) [13], and least squares regression [14]. It can also be used with a scalable subspace clustering method on a large number of data vectors such as sketched subspace clustering [9] and scalable sparse subspace clustering [15], to simplify the computation further. Experimental results are presented to show the effectiveness of the proposed algorithm.

II. Background

A. Subspace Clustering

Subspace clustering aims to determine the data segmentation according to subspace memberships. In the popular spectral clustering method, the pairwise similarities between data points are described in the affinity matrix and used to segment data into several parts, by making the intra-group similarities as high as possible, and the inter-group similarities as low as possible. To construct the affinity matrix, a coefficient matrix $\boldsymbol{C}$ is often obtained by optimising the following cost function

$$\min_{\boldsymbol{C}} \|\boldsymbol{C}\|_\ell + \frac{\lambda}{2}\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\|_F^2 \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{D\times N}$ is the data matrix, with each column being a data vector (i.e. data point). The ith column $\boldsymbol{c}_i$ of $\boldsymbol{C}$ contains the representation coefficients of the ith data vector $\boldsymbol{x}_i$ for all data vectors. The regularizer $\|\cdot\|_\ell$ here is a matrix norm,

This work was supported in part by an EPSRC IAA project EP/R511791/1 and an MoD DASA project MANTIS Phase 2.

e.g. the ℓ_1 norm for SSC, and the nuclear norm for LRR. A symmetric matrix $\boldsymbol{W}$, representing the pairwise affinities, can be constructed as $\boldsymbol{W} = |\boldsymbol{C}| + |\boldsymbol{C}^\top|$, where $|\cdot|$ takes the element-wise absolute value.

In conventional subspace clustering algorithms, spectral-clustering commonly serves as a post-processing step which obtains the clustering result based on the affinity matrix $\boldsymbol{W}$.

B. Spectral Clustering

Spectral clustering essentially treats data points as vertexes in a graph, and similarities between the data points as the weighted edges connecting vertexes. The graph needs to be cut into segments such that the edges connecting vertexes in different segments have the lowest weight [5].

To address this, a graph Laplacian matrix is introduced [16],

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W} \tag{2}$$

where $\boldsymbol{W}$ is the affinity matrix, whose (i, j)-th element evaluates the correlation between the i-th and j-th data points, $\boldsymbol{D}$ is a diagonal matrix whose (i, i)-th element is the sum of the i-th row/column of $\boldsymbol{W}$. For subspace clustering, $\boldsymbol{W}$ is generated by methods presented in Section II-A.

Assume there are k different categories, then k eigenvectors $\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_k$ corresponding to the k smallest eigenvalues of the normalized Laplacian matrix are obtained. Afterwards, a matrix $\boldsymbol{T} = [\boldsymbol{t}_1, \ldots, \boldsymbol{t}_k] \in \mathbb{R}^{N \times k}$ is constructed by assembling these k eigenvectors together. Then, a matrix $\boldsymbol{Y}$ whose rows have unit norm is obtained by normalizing $\boldsymbol{T}$. Finally, the clustering result is obtained by performing the conventional K-means algorithm on row vectors of $\boldsymbol{Y}$. In the process of spectral clustering, the vectors to be clustered are transformed from the data vectors $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ to the row vectors $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ of $\boldsymbol{Y}$, which can enhance the cluster-property of data [5].

C. Sketched Subspace Clustering

Traganitis and Giannakis [9] introduced a sketching method where the data matrix $\boldsymbol{X}$ is compressed by

$$\breve{\boldsymbol{X}} = \boldsymbol{X}\boldsymbol{R} \tag{3}$$

where $\boldsymbol{R}$ is a random Gaussian matrix and $\breve{\boldsymbol{X}}$ is the compressed matrix, which has fewer rows than $\boldsymbol{X}$. This algorithm is shown to have guaranteed Restricted Isometry Property (RIP).

In addition, the random selection of dimensions for subspace clustering has been proposed and discussed in [17] [10], and Heckel et al. [8] studied how dimensionality reduction affects the performance of subspace clustering algorithms. The above dimension reduction algorithms have shown their effectiveness. However, these algorithms do not distinguish which dimension is more informative than others which potentially degrades the clustering performance.

III. The Proposed Algorithm

For real-world data, there may be some dimensions (i.e. coordinates) of data points which do not affect the clustering result. As an example, Figure 1 shows digit images from the MNIST handwritten dataset [11]. Subspace clustering algorithms treat each image as a data point while the grey-level of each pixel as the value of a coordinate. The goal of subspace clustering is to cluster images of the same digit into one subspace, and to separate images of different digits into different subspaces simultaneously. From the figure, it is clear that some coordinates, such as edge and corner pixels, are equal across the images. As a result, if these coordinates are sampled by a random system, they would make no contribution to the clustering result, but consume computational resources.

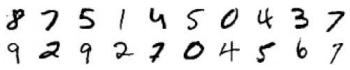

Fig. 1. Images from the MNIST handwritten digits dataset [11].

Similarly, as shown in Figure 2, there are two 1-dimensional subspaces lying in a 3-dimensional ambient space. To improve the computational efficiency, we want to cluster points in subspaces using only a few coordinates. Points in two subspaces can be distinguished using coordinates (y, z) and (x, z). However, the subspaces are identical on (x, y). If the selected dimensions are x and y, all columns will appear to lie in the same subspace. If we sample dimensions randomly, we would face the risk of losing important information, such as eliminating the z coordinates in this case, and the performance of subspace clustering is likely to be degraded.

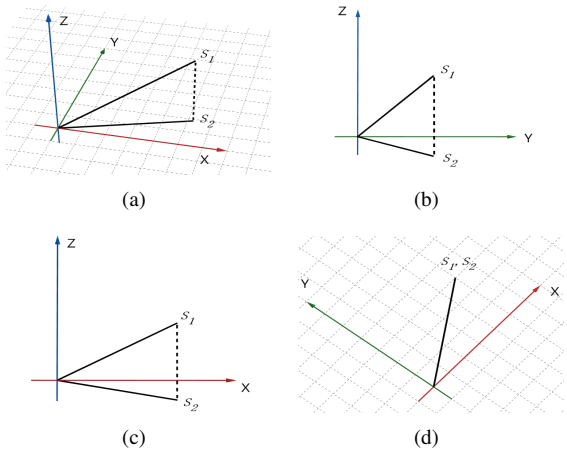

Fig. 2. (a) Two 1-dimensional subspaces in a 3-dimensional ambient space. (b) Two subspaces with coordinates (y, z). (c) Two subspaces with coordinates (x, z). (d) Two subspaces with coordinates (x, y).

A. Dimension Selection

The main objective of our algorithm is to ensure that the sketched dimensions are informative. Our method is to retain the data coordinates which have significant effects on the clustering results, and to eliminate coordinates which have low effects on clustering results.

As presented in Section II-B, the affinity matrix $\boldsymbol{W}$, which evaluates the pairwise correlations of points has a crucial

impact on the clustering result. In addition, in Section II-A, it has been introduced that $\boldsymbol{W}$ is formed in terms of $\boldsymbol{C} = [\boldsymbol{c}_1, ..., \boldsymbol{c}_N]$, which is constructed by coefficient vectors of each data point. For the ith input data point, from (1), we have

$$\boldsymbol{x}_i = \boldsymbol{X}\boldsymbol{c}_i \tag{4}$$

where $\boldsymbol{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_N]$, $\boldsymbol{c}_i = [c_{i1}, ..., c_{iN}]^T$.

Assume $\boldsymbol{X} \in \mathbb{R}^{D\times N}$ can be divided into two segments $\boldsymbol{X} = [\boldsymbol{X}_p^T \quad \boldsymbol{X}_q^T]^T$, where $\boldsymbol{X}_p = [\boldsymbol{x}_{p1}, ..., \boldsymbol{x}_{pN}]$, and $\boldsymbol{X}_q = [\boldsymbol{x}_{q1}, ..., \boldsymbol{x}_{qN}]$. The upper part $\boldsymbol{X}_p$ covers the 1st to kth coordinates while the lower part $\boldsymbol{X}_q$ covers the $(k+1)$th to Dth coordinates. $\boldsymbol{X}_p$ has full rank and consists of clusters of data in different subspaces, while the rank of $\boldsymbol{X}_q$ is 1, and $\boldsymbol{X}_q$ consists of trivial noise and redundant information. Then (4) becomes

$$\begin{bmatrix} \boldsymbol{x}_{pi} \\ \\ \boldsymbol{x}_{qi} \end{bmatrix} = c_{i1} \begin{bmatrix} \boldsymbol{x}_{p1} \\ \\ \boldsymbol{x}_{q1} \end{bmatrix} + \cdots + c_{iN} \begin{bmatrix} \boldsymbol{x}_{pN} \\ \\ \boldsymbol{x}_{qN} \end{bmatrix} \tag{5}$$

As $\boldsymbol{X}_p$ has full rank, any change of $\boldsymbol{X}_p$ will affect the solution of (5), including eliminating a row. On the other hand, as $rank(\boldsymbol{X}_p) = 1$, rows of $\boldsymbol{X}_q$ can be eliminated until only one row is left, and the solution to (5) will be almost unaffected.

Any data matrix $\boldsymbol{X}$ can be divided into one part with full rank like $\boldsymbol{X}_p$, and a few parts whose rank is 1, like $\boldsymbol{X}_q$. The coordinates of $\boldsymbol{X}_q$ can be eliminated, with negligible influence on the clustering accuracy. Those corner pixels in Figure 1, and coordinate (x, y) in Figure 2, resemble the coordinates of $\boldsymbol{X}_q$, which can be eliminated.

In our algorithm, the objective is to determine those coordinates which can affect entries of the affinity matrix. To achieve that, we estimate the importance of a coordinate by how much it affects the affinity matrix. Precisely, we investigate how much the affinity matrix is changed when the coordinate is eliminated.

Various metrics could be used to evaluate the change of the affinity matrix. In our algorithm, the importance of the i-th coordinate is evaluated by

$$\boldsymbol{T}_i = \|\boldsymbol{W} - \boldsymbol{W}_i\|_F^2 \tag{6}$$

where $\boldsymbol{W}_i$ is the generated affinity matrix when the i-th coordinate of the data matrix $\boldsymbol{X}$ is eliminated, that is,

$$\boldsymbol{x}_{\hat{i}j} = \boldsymbol{X}_{\hat{i}}\boldsymbol{c}_{\hat{i}j} \tag{7}$$

where $\boldsymbol{x}_{\hat{i}j} = [x_{j1}, ..., x_{j(i-1)}, x_{j(i+1)}, ..., x_{jD}]^T$, $\boldsymbol{X}_{\hat{i}} = [\boldsymbol{x}_{\hat{i}1}, ..., \boldsymbol{x}_{\hat{i}N}]$, $\boldsymbol{C}_{\hat{i}} = [\boldsymbol{c}_{\hat{i}1}, ..., \boldsymbol{c}_{\hat{i}N}]$, and $\boldsymbol{W}_i$ is constructed by $\boldsymbol{C}_i$, which is $\boldsymbol{W}_{\hat{i}} = |\boldsymbol{C}_{\hat{i}}| + |\boldsymbol{C}_{\hat{i}}^\top|$.

A large $\boldsymbol{T}_i$ indicates that the i-th coordinate is of high importance. In our algorithm we eliminate those dimensions with relatively small value of $\boldsymbol{T}_i$, and only retain a fixed number of dimensions.

To guarantee the selected dimensions are informative, the number of selected dimensions should be above the rank of subspaces ($d \geq r$). The dimension-selection algorithm can work with any spectral clustering based subspace clustering algorithm. It can also be combined with a volume-sketching algorithm to reduce the computational cost further.

B. Computational Cost

Computing $\boldsymbol{T}_i$ of the i-th dimension will incur additional load. The computational complexity involved in (6) is $\mathcal{O}(N^2)$, where N is the number of data points. This could be problematic when the number of input data points becomes high.

To address this issue, we randomly sample n data points (rather than using N points) when implementing the algorithm. Empirically, using a subset of the data points at sufficient scale does not affect the system performance substantially. Experimental results about this are presented in Section IV-C. It can be seen from Figure 2 that if we use a sample of points in two 1-dimensional subspaces, coordinate z would be retained and either of x, y would be eliminated, and the result of our algorithm would not change. Similarly, for digit image data from Figure 1, when a subset of images is used, edge and corner pixels would be eliminated and central pixels would be retained.

With the process of down-sampled input data, the computational cost of (6) becomes $\mathcal{O}(n^2)$ and $n \ll N$. When the proposed algorithm is used with conventional subspace clustering methods such as SSC, the complexity of the whole subspace clustering algorithm is $\mathcal{O}(dN^2 + Dn^2)$. It is clearly smaller than $\mathcal{O}(DN^2)$ of SSC when the input data are large ($D \gg d$, $N \gg n$).

IV. Evaluations

In this section, we present the experimental result of the proposed algorithm, in comparison with the random projection and random sketching algorithm, in terms of error-rate (%).

A. Benchmark Algorithms

In our experiments, the proposed dimension selection algorithm is implemented with conventional SSC [12]. Meanwhile, the randomly sketched subspace clustering [10] based on a random selection of data dimensions, and the Gaussian random projection based algorithm [18] based on the RIP are used as the benchmark algorithms.

B. Datasets and Parameter Settings

Both synthetic and real-world datasets are used.

Two synthetic dataset are vectors with elements of random numbers following Gaussian distribution with zero mean and unit variance. In the $1st$ synthetic dataset, the data have dimension $D = 1000$, subspaces have dimension $\bar{D} = 10$, and each cluster has 500 data points. For data in the same cluster, they are in the same 10-dimensional subspace. On the contrary, for data in different clusters, they are embedded in different 10-dimensional subspaces. To show the effectiveness of the proposed algorithm, $1/4$ of the data coordinates are noiseless entries, and $3/4$ of the data coordinates are filled with Gaussian noise, which is similar to random backgrounds of read-world images. In other words, $\boldsymbol{X} = [\boldsymbol{X}_p^T \quad \boldsymbol{X}_q^T]^T$, where $\boldsymbol{X} \in \mathbb{R}^{D\times N}$, $\boldsymbol{X}_p \in \mathbb{R}^{\frac{1}{4}D\times N}$ are noiseless data for clustering, and entries of $\boldsymbol{X}_q \in \mathbb{R}^{\frac{3}{4}D\times N}$ are Gaussian noise.

The $2nd$ synthetic dataset has the same cluster size, data dimension, and intrinsic dimension of subspaces. Whereas, different from the $1st$ dataset, there are no coordinates set as 'background part' of the image. The ambient space has D dimensions, and Gaussian noise is added to all the entries.

The MNIST handwritten digits dataset [11] is used as the real-world dataset. This dataset has 28×28 pixels of images. Each image represents a digit from 0 to 9, and is regarded as a 784 dimensional point in a space. Images from one of the 10 digits in this dataset should be clustered into the same linear subspace regardless of noise.

In the experiments of the proposed method, we set the sketched dimension $d = 20$ for synthetic data, $d = 60$ for the MNIST dataset, the sketched number of vectors $n = 20$ without further statement. Experiments have been carried out for 50 trials. Other settings of SSC and the benchmark algorithms are the same as used in the original references.

C. Influence of Sampled Points

As discussed in Section III-B, we proposed to sample n vectors when measuring the changing of the affinity matrix. Figure 3 shows the performance of the proposed algorithm

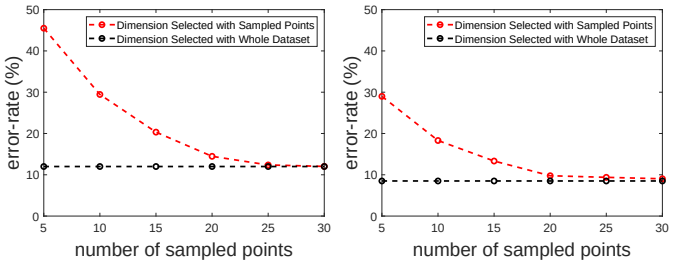

Fig. 3. Error-rate of the dimension selected algorithm when the number of data points is sampled (red), compared with the dimension selected algorithm with full data vectors (black), (a) on the synthetic dataset, (b) on the MNIST dataset.

with respect to the choice of n, in comparison with the proposed algorithm with full data vectors, N, on the $2nd$ synthetic dataset (left), and on the MNIST dataset (right).

It can be seen that when there are sufficient number of sampled data vectors ($n \geq 25$ in the experiments), the performance of the algorithm is retained. Such property still holds when testing with other synthetic datasets which have similar distributions in the number of samples per cluster, such as uniform distribution.

D. Experiments with Synthetic Data

Figure 4(a) shows the error rate changing with respect to the number of dimensions sketched, with the 1st synthetic dataset. When there are several noisy pixels which cannot be ignored, the proposed algorithm outperforms significantly the random sketching and the random projection algorithms. The error rate of the proposed algorithm approaches to zero when $d \geq 40$ in this case. This is because the reduction of noisy dimensions can reduce the interference of noises in generating

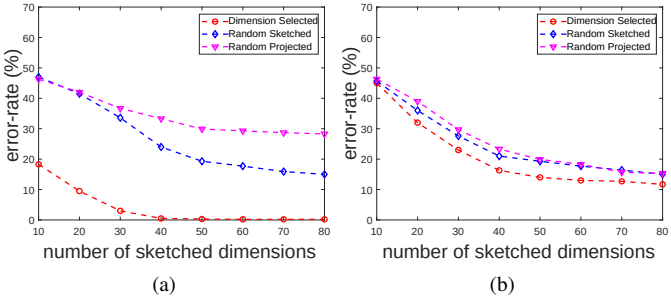

Fig. 4. Error-rate of varied dimensionality reduction algorithms (red, blue, and violet lines), with number of sketched dimensions changing, (a) on the $1st$ synthetic dataset, (b) on the $2nd$ synthetic dataset.

affinity matrices. Figure 4(b) shows the performance of the compared algorithms for the 2nd synthetic dataset, when there are data vectors from different subspaces with random noise. The performance of the proposed algorithm degrades in comparison with Figure 4(a). It is because there are noisy dimensions added to the 1st dataset, which can be eliminated by our method, but the Gaussian noise is added to all the entries of the 2nd dataset, which can not be clearly eliminated. Nevertheless, the proposed method still outperforms the two benchmark algorithms.

E. Experiments with Real-world Data

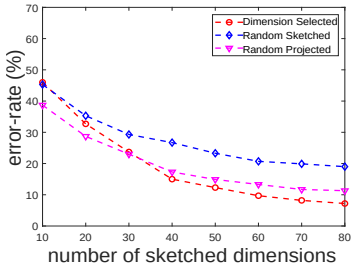

Fig. 5. Error-rate of the compared algorithms (red, blue, and violet lines), on the MNIST data, with the number of sketched dimensions changing.

Figure 5 shows the performance of the compared algorithms on the MNIST dataset. It can be observed that the proposed algorithm outperforms the two baseline algorithms when the number of the selected dimensions $d \geq 40$. There are edge and corner pixels which are the same across images. The proposed algorithm ignores the influence of such pixels, thus it obtains better performance than others with the MNIST image dataset.

V. Conclusion

We have presented a novel dimension selection algorithm for subspace clustering by selecting informative dimensions or removing noise. Experimental results show that the proposed algorithm can retain the clustering performance by using smaller number of dimensions in the data points. The proposed algorithm is flexible, and can be used with a variety of subspace clustering algorithms and volume sketching algorithms.

REFERENCES

[1] R. Vidal, "Subspace clustering," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 52–68, 2011.

[2] P. A. Traganitis, K. Slavakis, and G. B. Giannakis, "Large-scale subspace clustering using sketching and validation," *arXiv preprint arXiv:1510.01628*, 2015.

[3] A. Y. Yang, J. Wright, Y. Ma, and S. S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," *Computer Vision and Image Understanding*, vol. 110, no. 2, pp. 212–225, 2008.

[4] R. Vidal, R. Tron, and R. Hartley, "Multiframe motion segmentation with missing data using power factorization and GPCA," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 85–105, 2008.

[5] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[6] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881–892, 2002.

[7] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: The fuzzy c-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.

[8] R. Heckel, M. Tschannen, and H. Bölcskei, "Subspace clustering of dimensionality-reduced data," in *IEEE International Symposium on Information Theory*. IEEE, 2014, pp. 2997–3001.

[9] P. A. Traganitis and G. B. Giannakis, "Sketched subspace clustering," *arXiv preprint arXiv:1707.07196*, 2017.

[10] S. Li and W. Wang, "Randomly sketched sparse subspace clustering for acoustic scene clustering," *European Signal Processing Conference (EUSIPCO)*, 2018.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[12] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[13] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010, pp. 663–670.

[14] C.-Y. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, and S. Yan, "Robust and efficient subspace segmentation via least squares regression," *European Conference on Computer Vision (ECCV)*, pp. 347–360, 2012.

[15] X. Peng, L. Zhang, and Z. Yi, "Scalable sparse subspace clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 430–437.

[16] F. R. Chung, *Spectral Graph Theory*, Number 92. American Mathematical Soc., 1997.

[17] D. Pimentel-Alarcón, L. Balzano, and R. Nowak, "Necessary and sufficient conditions for sketched subspace clustering," in *54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2016, pp. 1335–1343.

[18] G. Li and Y. Gu, "Restricted isometry property of Gaussian random projection for finite set of subspaces," *arXiv preprint arXiv:1704.02109*, 2017.