Efficient Direct Structured Subspace Clustering

Wen-ming Cao[1], Rui Li[1], Sheng Qian[1], Si Wu[2], and Hau-San Wong[1](✉)

[1] Department of Computer Science, City University of Hong Kong,
Hong Kong, China
{wenmincao2-c,ruili52-c,sqian9-c}@my.cityu.edu.hk, cshswong@cityu.edu.hk
[2] School of Computer Science and Engineering,
South China University of Technology, Guangzhou, China
cswusi@scut.edu.cn

Abstract. Subspace clustering splits data instances that are drawn from special low-dimensional subspaces via utilizing similarities between them. Traditional methods contain two steps: (1) learning the affinity matrix and (2) clustering on the affinity matrix. Although these two steps can alternatively contribute to each other, there exist heavy dependencies between the performance and the initial quality of affinity matrix. In this paper, we propose an efficient direct structured subspace clustering approach to reduce the quality effects of the affinity matrix on performances. We first analyze the connection between the affinity and partition matrices, and then fuse the computation of affinity and partition matrices. This fusion allows better preserving the subspace structures which help strengthen connections between data points in the same subspaces. In addition, we introduce an algorithm to optimize our proposed method. We conduct comparative experiments on multiple data sets with state-of-the-art methods. Our method achieves better or comparable performances.

Keywords: Subspace clustering · Unsupervised learning

1 Introduction

Subspace clustering has been investigated and applied to various field including image content compress and feature representation learning [1], image segmentation [2] and gene expression profile clustering [3]. The high-dimensional characteristics of data potentially produce adverse effects on performances due to the noise disturbance [4]. Rather than being uniformly distributed across the high-dimensional space, these data often lie a union of low-dimensional subspace. For example, when clustering face images of a person under various illumination conditions or tracking a moving object in a continuously temporal sequence, we can observe that data points from a cluster are distributed in a low-dimensional space. Searching a union of low-dimensional subspaces to explore structures in

L. Cheng et al. (Eds.): ICONIP 2018, LNCS 11304, pp. 181–190, 2018.
https://doi.org/10.1007/978-3-030-04212-7_16

data helps alleviate negative effects of high-dimensional noise. Traditional clustering methods exploit proximity distances of data point belonging to each cluster. However, the above data distribute arbitrarily across the high-dimensional ambient space rather than around a centroid, which means traditional methods are not applicable to subspace clustering.

Subspace clustering approaches are grouped into [14]: algebraic-based methods, factorization-based methods, statistical-based methods and self expressive-based methods. The first type of methods converts subspace clustering problems by fitting and differentiating polynomials concerning the input matrix $\mathbf{X}$. The second type of methods obtains segmentations through searching a proper low-rank approximate factorization concerning $\mathbf{X}$. The third type of methods addresses subspace clustering issues by introducing a probabilistic assumption that data are sampled from a certain mixture of Gaussian distributions. The fourth type of methods contains two stages: learning the self-expressive matrix and performing clustering on the learnt self-expressive matrix. Among these methods, sparse subspace clustering (SSC) [10] and low-rank representations (LRR) [11] are representative. The difference between them lies in that SSC imposes ℓ_1-norm regularization on the affinity matrix, while LRR employs ℓ_*-norm regularization on the affinity matrix. Structured sparse subspace clustering (S^3C) [3,14] performs the affinity matrix learning and clustering by introducing a regularization to quantify the disagreement between the affinity and partition matrices.

Unlikely S^3C, we first analyze the connection between the affinity and partition matrices. This connection makes it possible the fusion of computing the affinity and partition matrices, where the transformation on the partition matrix is used as a substitute for the affinity matrix. Thus, the subspace clustering problem is cast as finding a proper partition matrix without explicitly computing the affinity matrix. Besides, since both the affinity and partition matrices are dedicated to capturing the segmentation of data, the above fusion allows better preserving the subspace structural information, which strengthens connections between data instances that are drawn from the same subspace. The contributions of this paper are summarized as follows:

1. We introduce a subspace clustering framework based on the connection between the affinity and partition matrices to avoid learning the affinity matrix.
2. We design an algorithm to optimize the objective function of our method, and conduct comparative experiments with state-of-the-art subspace clustering methods. Our method achieves better or at least comparable performances.

2 Related Work

We review existing related literatures on various kinds of subspace clustering methods. SSC [10] learns sparse representations by imposing ℓ_1-norm regularization on the affinity matrix. LRR [11] finds low-rank representations by employing

ℓ_*-norm regularization on the affinity matrix. Although both SSC and LRR construct the self-expressive matrix in the iteration manner, they perform badly in dealing with noisy data. Low rank subspace clustering (LSRC) derives a closed-form solution by recovering a clean dictionary. Efficient Dense Subspace Clustering (EDSC) [25] employs ℓ_2-norm regularization to handle noise and outliers. Correlation adaptive subspace segmentation (CASS) [18] employs the TraceLasso norm [21] on the representation matrix. Dohyung *et al.* [19] propose the nearest subspace neighbor algorithm to automatically identify neighbor data instances that are most likely drawn from the same subspace. Yang *et al.* [20] propose ℓ_0-induced sparse subspace clustering to discover the subspace-sparse representation for arbitrary distinct underlying subspaces. In [27], Ren *et al.* propose a weighted adaptive mean shift algorithm to reduce effects of the sparsity caused by high-dimensional data and the noise features. Friedman *et al.* [28] assign proper weights of features based on importance of features on clustering.

The combination of deep learning and subspace clustering has emerged recently. In [5,6], autoencoder is utilized to learn latent representations for clustering. The difference between them is that the former encodes representations from raw data, while the latter treats the predefined affinity matrix as the input of neural network. In [7], deep autoencoder is adopted to provide initialization for deep embedding for clustering. Deep subspace clustering with sparsity prior [8] clusters deep representations, which incorporate the locality for reconstructing input and structured global prior. Improved Deep Embedded Clustering [9] integrates the clustering loss and autoencoder reconstruction loss into a unified framework, and considers the structural preservation when performing clustering and feature learning jointly. Additionally, a self-expressive matrix is learnt by utilizing a fully connected layer in [26] with latent representations as input.

3 Methodology

In this section, we present our proposed subspace clustering approach. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ be a set of N data instances $\{\mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^N$ that are sampled from a special union of k subspaces $\{S_j\}_{j=1}^k$, *i.e.*, $S_1 \bigcup S_2 \bigcup \cdots \bigcup S_k$, of dimensions $\{d_j\}_{j=1}^k$ in $\mathbb{R}^d$. Let $\mathbf{X}_j \in \mathbb{R}^{d \times N_j}$ be a sub-matrix of X of rank d_j with $N_j > d_j$ data points that lie in S_j with $\sum_{j=1}^k N_j = N$. The subspace clustering aims at partitioning data instances in the $\mathbf{X}$ into their corresponding subspaces.

3.1 Efficient Direct Structured Subspace Clustering

The objective function of a traditional clustering method is given by:

$$\min \sum_{i=1}^{N} \sum_{j=1}^{k} \mathbf{P}_{ji} \left\| \mathbf{X}_{\cdot i} - \mathbf{C}_{\cdot j} \right\|^2 = \min \left\| \mathbf{X} - \mathbf{CP} \right\|_F^2, \tag{1}$$

where $\mathbf{X}_{\cdot i}$ and $\mathbf{C}_{\cdot j}$ denote the i^{th} and j^{th} columns of $\mathbf{X}$ and $\mathbf{C}$, respectively. $\mathbf{C} \in \mathbb{R}^{d \times k}$ and $\mathbf{P} \in \{0, 1\}^{k \times N}$ denote the cluster center and partition matrices.

The solution to Eq. (1) can be obtained in a closed form [23] which is $\mathbf{C} = \mathbf{X}\mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1}$. Thus, the above optimization problem is equivalent to:

$$\min \left\| \mathbf{X} - \mathbf{X}\mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1}\mathbf{P} \right\|_F^2, \ s.t., \ \mathbf{P}^T\mathbf{1}_k = \mathbf{1}_N, \tag{2}$$

where $\mathbf{1}_k \in \mathbb{R}^k$ and $\mathbf{1}_N \in \mathbb{R}^N$ are column vectors whose entries are 1.

Spectral clustering-based subspace clustering learns the affinity matrix under the assumption that any data point can be approximated by a linear combination of other data points, *i.e.*, $\mathbf{X} = \mathbf{X}\mathbf{Z}$, where $\mathbf{Z}$ is the coefficient matrix whose entity $\mathbf{Z}_{ij}$ measures the similarity between the i^{th} and j^{th} data points. When data points are corrupted by noise, the subspace clustering problem is formulated by

$$\min_{\mathbf{Z},\mathbf{E}} \|\mathbf{Z}\|_1 + \lambda \|\mathbf{E}\|_\ell, s.t., \mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E}, \ diag(\mathbf{Z}) = 0, \tag{3}$$

where $\mathbf{E} = [\epsilon_1, \cdots, \epsilon_N]$ is a noise matrix with $\{\|\epsilon_i\|_2 \leq \xi\}_{i=1}^N$, and $\xi > 0$. λ is a trade-off parameter. The constraint $diag(\mathbf{Z}) = 0$ is introduced to avoid the solution of $\mathbf{Z}$ deteriorating into an identity matrix.

From Eqs. (2) and (3), the coefficient matrix $\mathbf{Z}$ is obtained by

$$Z = (\mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1}\mathbf{P}) \odot (\mathbf{1}_N\mathbf{1}_N^T - \mathbf{I}) \tag{4}$$

where $\odot$ is an elementwise product. $\mathbf{I}$ is an identity matrix with a proper size.

Structured sparse subspace clustering [3] holds that both $\mathbf{P}$ and $\mathbf{Z}$ attempt to capture the segmentation of data. In other words, $\mathbf{Z}$ and $\mathbf{P}$ have related zero patterns, *i.e.*, for each $\mathbf{Z}_{ij} \neq 0$, the i^{th} and j^{th} data instances should be split into the same subspace, and $\mathbf{P}_{\cdot i} = \mathbf{P}_{\cdot j}$, where $\mathbf{P}_{\cdot i}$ and $\mathbf{P}_{\cdot j}$ denote the i^{th} and j^{th} columns of matrix $\mathbf{P}$. The subspace structured measure [14] is introduced to evaluate the disagreement between $\mathbf{Z}$ and $\mathbf{P}$ as follows

$$\|\Upsilon \odot \mathbf{Z}\|_0 = \sum_{i,j:\mathbf{P}_{\cdot i} \neq \mathbf{P}_{\cdot j}} \mathbb{I}_{\{|\mathbf{Z}_{ij} \neq 0|\}}, \tag{5}$$

where $\Upsilon_{ij} = \frac{1}{2}\|\mathbf{P}_{\cdot i} - \mathbf{P}_{\cdot j}\|^2$, and $\mathbb{I}$ is an indicator function. Since there are two possible values for Υ_{ij}, *i.e.*, $\Upsilon_{ij} \in \{0, 1\}$, we can utilize the number of nonzero entries in $\mathbf{Z}$ to quantify the cost of Eq. (5) when the i^{th} and j^{th} data instances are split into different subspaces. Since the above measure results in a NP-hard problem, a subspace structured norm [14] of $\mathbf{Z}$ concerning $\mathbf{P}$ is defined by

$$\|\mathbf{Z}\|_P = \beta \|\Upsilon \odot \mathbf{Z}\|_1 = \beta \sum_{ij} |\mathbf{Z}_{ij}| (\frac{1}{2}\|\mathbf{P}_{\cdot i} - \mathbf{P}_{\cdot j}\|^2). \tag{6}$$

Combining Eqs. (3), (4) and (6), we can obtain the expression by

$$\min_{\mathbf{P},\mathbf{E}} \|\mathbf{Z}\|_{1,P} + \lambda \|\mathbf{E}\|_\ell, s.t., \mathbf{X} = \mathbf{X}((\mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1}\mathbf{P}) \odot (\mathbf{1}_N\mathbf{1}_N^T - \mathbf{I})) + \mathbf{E}, \tag{7}$$

where

$$\|\mathbf{Z}\|_{1,P} = \sum_{ij} \left| ((\mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1}\mathbf{P}) \odot (\mathbf{1}_N\mathbf{1}_N^T - \mathbf{I}))_{ij} \right| (1 + \frac{\beta}{2}\|\mathbf{P}_{\cdot i} - \mathbf{P}_{\cdot j}\|^2) \tag{8}$$

3.2 Optimization

Given the objective function in Eq. (7), we solve it by introducing a Lagrange multiplier $\boldsymbol{Y}^1$, and obtain the augmented Lagrange as follows:

$$L(\mathbf{P}, \mathbf{E}, \boldsymbol{Y}^1) = \|\mathbf{Z}\|_{1,P} + \lambda \|\mathbf{E}\|_\ell + < \boldsymbol{Y}^1, \mathbf{X} - \mathbf{XZ} - \mathbf{E} > + \frac{\mu}{2}(\|\mathbf{X} - \mathbf{XZ} - \mathbf{E}\|_F^2). \tag{9}$$

When $\mathbf{E}$ and $\boldsymbol{Y}^1$ are fixed, $\mathbf{P}$ is updated via:

$$\mathbf{P}_{t+1} \Leftarrow \mathbf{P}_t[(\mathbf{X}^T\mathbf{X} + \mathbf{I})(\mathbf{X}^T(\mathbf{X} - \mathbf{E}_t - \frac{1}{\mu_t}\mathbf{Y}_t^1)) \odot (\mathbf{1}_N\mathbf{1}_N^T - \mathbf{I})]. \tag{10}$$

Given $\mathbf{P}$, the update of $\mathbf{Z}_{t+1}$ can be obtained from Eq. (8) naturally. When $\mathbf{P}$ and $\boldsymbol{Y}^1$ are fixed, the update of $\mathbf{E}$ can be obtained by:

$$\mathbf{E}_{t+1} = \arg\min_{\mathbf{E}} \frac{\lambda}{\mu_t}\|\mathbf{E}\|_\ell + \frac{1}{2}\left\|\mathbf{E} - \mathbf{X} - \mathbf{XZ}_{t+1} + \frac{1}{\mu_t}\boldsymbol{Y}_t^1\right\|_F^2. \tag{11}$$

When $\mathbf{E}$ and $\mathbf{P}$ are fixed, $\boldsymbol{Y}^1$ is updated as follows:

$$\boldsymbol{Y}_{t+1}^1 \Leftarrow \boldsymbol{Y}_t^1 + \mu_{t+1}(\mathbf{X} - \mathbf{XZ}_{t+1} - \mathbf{E}_{t+1}) \tag{12}$$

The optimization algorithm is shown in the Algorithm 1 below.

Algorithm 1. Efficient Direct Structured Subspace Clustering

```
Input: Data matrix $\mathbf{X}$, $MAXITE$, $\lambda$, $\beta$ and $\rho$;
Output: Partition Matrix $\mathbf{P}$.
Initialize: $\mathbf{P}$ with k-means clustering, $c_{nt} = 1$, $\mathbf{Z} = \mathbf{0}$;
while $c_{nt} \leq MAXITE$ do
    Update the noise matrix $\mathbf{E}$ via Eq. (11);
    Update the partition matrix $\mathbf{P}$ via Eq. (10);
    Update $\boldsymbol{Y}^1$ via Eq. (12);
    Update $\mu_{t+1} \leftarrow \rho\mu_{t+1}$
    $c_{nt} \leftarrow c_{nt} + 1$;
end while
```

4 Experiment

We conduct comparative experiments on multiple data sets with the following methods: LRR [13], LatLRR [12], BDLRR [16], CASS [18], LSR1 [17], LSR2 [17], LRSC [15], SSC [4], KSSC [22], BDSSC [16], AE+SSC [26], SSCOPM [24], EDSC [25], AE+EDSC [26], S^{3}C [14] and DSC [26]. Data sets contain the Extended Yale B, ORL, COIL20 and COIL100. We quantify performances by exploiting clustering error (Err), normalized mutual information (NMI) and subspace-preserving rate (SPR). We repeat 10 times for each algorithm and record the averages.

4.1 Evaluation Metrics

ERR is defined as below:

$$ERR(r, \hat{r}) = 1 - \max_{\pi} \frac{1}{N} \sum_{i=1}^{N} 1_{\{\pi(\hat{r}_i)=r_i\}} \tag{13}$$

where r and $\hat{r}$ denote the ground-truth and the estimated labels, respectively.

NMI is defined as follows:

$$NMI(r, \hat{r}) = \frac{\sum_{i=1}^{c} \sum_{j=1}^{c} r_{ij} \log(\frac{n \times r_{ij}}{r_i \hat{r}_j})}{\sqrt{(\sum_{i=1}^{c} r_i \log(\frac{r_i}{n}))(\sum_{j=1}^{c} \hat{r}_j \log(\frac{\hat{r}_j}{n}))}} \tag{14}$$

where r_{ij} denotes the number of data points shared by the i^{th} and j^{th} classes.

SPR quantifies the extent to which a clustering solution satisfies the subspace-preserving property, and is defined as follows:

$$SPR = \frac{1}{N} \sum_{j=1} (\sum_{i=1} (G_{ji} \cdot |Z_{ji}|)/(\|Z_{j\cdot}\|_1)) \tag{15}$$

where $G_{ji} \in \{0, 1\}$ is the ground-truth affinity. $Z_{j\cdot}$ is the j^{th} row of the learnt affinity matrix Z, and $\|\cdot\|_1$ is the ℓ_1-norm.

4.2 Experimental Analysis

In this part, we present comparative experimental results. First, we show trends of cost function values of our proposed method concerning iterations in Fig. 1. Cost values of our method decrease quick in the first 20 iterations, and remain steady after 30 iterations. It means that our proposed method can converge fast. In particular, the number of iterations is less than 15 for COIL20 and COIL100. Second, we present the comparisons between our method and other subspace clustering methods on the Extended Yale B in Table 1 in terms of Err and NMI. From this table, we have the following observations:

1. Among traditional methods, S^3 achieves the best performances, which is followed by EDSC and LRSC. The advantage of S^3 attributes to the special structured-norm that allows learning structured sparse representations for each data point. EDSC benefits from Frobenius-norm to obtain denser connections between data points, while LRSC benefits from the learning of a clean self-expressive dictionary that reduce negative effects caused by noise.
2. Deep subspace clustering methods achieve better performances than traditional methods. For example, AE+SSC and AE+EDSC outperform SSC and EDSC, respectively. It demonstrates that representations learnt by deep networks can capture the abstract structural information to contribute to performance improvement. Besides, DSC achieves better performances than AE+SSC and AE+EDSC. The reason is that DSC utilizes a self-expressive layer to learn pairwise affinity between all the data points effectively.
3. Our proposed method achieves better performances than traditional methods including S^3, EDSC and LRSC. It verifies that the combination of computing

the affinity and partition matrices produces positive effects on performance improvement. In addition, our method achieves better or comparable performances, compared with DSC. It indicates that the above combination can conduce to the learning of informative structural representations in the coefficient matrix, which helps improve performances.

Third, we show the effects of computing the affinity and partition matrices in a unified framework concerning SPR by varying the trade-off parameter β obtained in the above data sets in Fig. 2. From this figure, we notice that when the value of β increases in the range between 0 and 10, SPR rises quickly before remaining steady or showing slight decrease trends. It is worthy mentioning that the horizontal axis in Fig. 2 is in the log scale rather than in the linear scale. Besides, it is noted that the SPR obtained by our proposed method is higher than that by S^3 when β changes between 0.3 and 10. It means that our method can achieve better subspace-preserving property, which is brought about by the combination of computing the affinity and partition matrices simultaneously.

Table 1. Performance comparison on extended Yale B w.r.t. (Err %) and (NMI %)

No. Sub.	5		15		25		35	
Criterion	Err	NMI	Err	NMI	Err	NMI	Err	NMI
LRR	13.94	84.14	23.22	71.65	27.92	70.58	41.85	58.45
LatLRR	6.90	88.90	32.47	62.47	32.76	62.76	38.75	60.75
BDLRR	12.97	85.38	31.58	64.86	34.67	61.25	35.76	63.68
CASS	21.25	68.89	33.65	62.38	36.45	62.29	36.57	62.58
LSR1	13.87	75.22	37.64	59.87	40.23	63.12	40.42	62.94
LSR2	13.91	76.91	37.01	61.21	40.89	63.04	39.77	64.03
LRSC	7.59	95.86	15.84	85.99	11.46	89.57	14.79	92.05
SSC	4.32	96.38	13.13	88.69	26.22	68.28	28.55	65.65
KSSC	7.58	91.64	14.49	86.75	16.55	82.75	20.48	80.37
BDSSC	27.5	63.74	38.46	48.85	42.47	41.95	38.47	47.35
AE+SSC	11.76	86.76	18.65	79.26	18.72	83.48	22.13	78.01
SSC-OPM	9.68	93.45	16.22	86.65	18.89	84.40	20.29	83.46
EDSC	5.11	94.22	7.63	89.68	10.67	90.82	13.10	89.66
AE+EDSC	4.45	96.68	6.70	90.25	10.27	90.75	13.28	88.26
S^3C	3.41	95.89	7.54	90.86	9.59	91.89	14.67	88.47
DSC	1.80	97.38	**2.30**	94.45	**2.38**	**94.95**	2.85	93.75
Proposed	**1.62**	**98.55**	**2.29**	**95.26**	2.44	94.44	**2.65**	**94.03**

Fourth, we show the comparative clusterings between our proposed method and other methods obtained in ORL, COIL20 and COIL100 in terms of Err and NMI in Figs. 3 and 4, respectively. From these figures, we observe our proposed method can still achieve best performance than other clustering methods

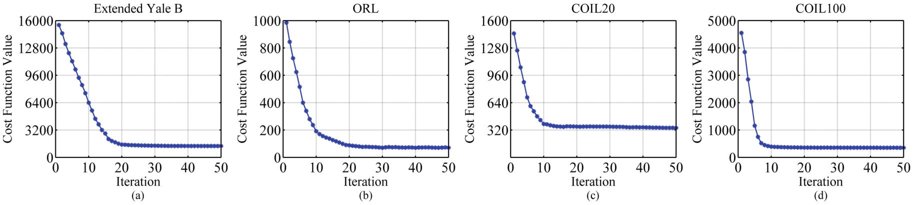

Fig. 1. Trends of cost function values with respect to iterations on Extended Yale B, ORL, COIL20 and COIL100

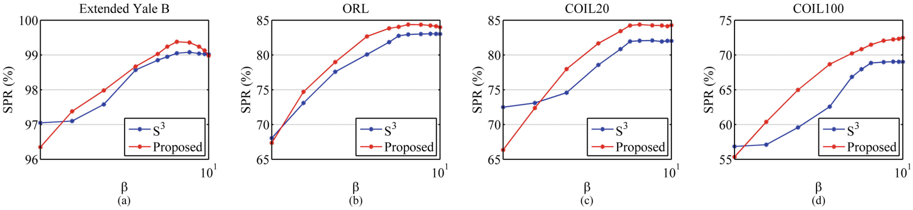

Fig. 2. Trends of SPR concerning β on Extended Yale B, ORL, COIL20 and COIL100

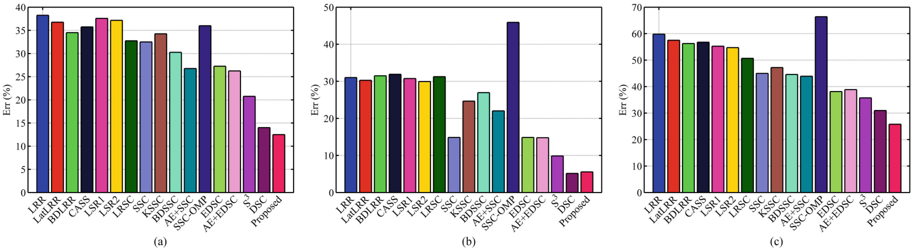

Fig. 3. Clustering Error (%) in ORL (a), COIL20 (b) and COIL100 (c)

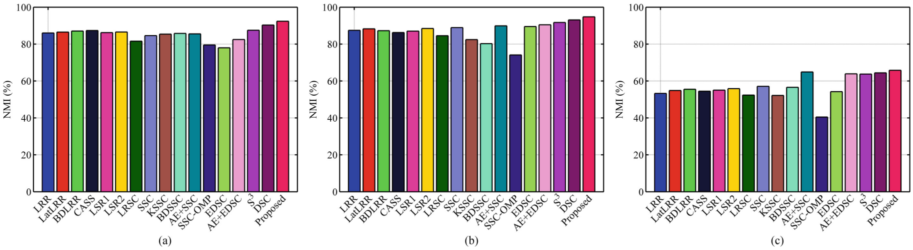

Fig. 4. NMI (%) in ORL (a), COIL20 (b) and COIL100 (c)

including traditional methods and deep subspace clustering methods except on COIL20, where DSC outperforms our method. The reason behind is that our method may consider the affinity between data points from different subspaces when attempting to preserve subspace structural property.

5 Conclusion

In this paper, we propose a direct subspace clustering method without learning the affinity matrix. We analyze the connection between the affinity matrix and partition matrix, and incorporate the computation of the affinity and partition matrices into a unified framework. Besides, we design an algorithm to optimize the objective function of our proposed method. Finally, we conduct extensive comparative experiments on multiple data sets with other representative subspace clustering methods. Our proposed method achieves better or comparable performances, which demonstrates the effectiveness of our method.

Acknowledgment. The work described in this paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China [Project No. CityU 11300715), and a grant from City University of Hong Kong [Project No. 7004884].

References

1. Zhao, C., Zhang, J., Ma, S., Fan, X., Zhang, Y., Gao, W.: Reducing image compression artifacts by structural sparse representation and quantization constraint prior. IEEE Trans. Circuits Syst. Video Technol. **27**(10), 2057–2071 (2017)
2. Lai, T., Wang, H., Yan, Y., Chin, T., Zhao, W.: Motion segmentation via a sparsity constraint. IEEE Trans. Intell. Transp. Syst. **18**(4), 973–983 (2017)
3. Li, C., You, C., Vidal, R.: Structured sparse subspace clustering: a joint affinity learning and subspace clustering framework. IEEE Trans. Image Process. **26**(6), 2988–3001 (2017)
4. Elhamifar, E., Vidal, R.: Sparse subspace clustering: algorithm, theory and applications. IEEE Trans. Patt. Anal. Mach. Intell. **35**(11), 2765–2781 (2013)
5. Ma, Y., Shang, C., Yang, F., Huang, D.: Latent subspace clustering based on deep neural networks. In: 6th International Symposium on Advanced Control of Industrial Processes, pp. 502–507. IEEE Press, Hiroshima (2014)
6. Tian, F., Gao, B., Cui, Q., Chen, E., Liu, T.: Learning deep representations for graph clustering. In: 28th AAAI Conference on Artificial Intelligence, pp. 1293–1299. AAAI Press, Québec (2014)
7. Xie, J., Girshick, R., Farhadi, A.: Unsupervised deep embedding for clustering analysis. In: 33rd International Conference on Machine Learning, New York, pp. 478–487 (2016)
8. Peng, X., Xiao, S., Feng, J., Yau, W., Yi, Z.: Deep subspace clustering with sparsity prior. In: 25th International Joint Conference on Artificial Intelligence, New York, pp. 1925–1931 (2016)
9. Guo, X., Gao, L., Liu, X., Yin, J.: Improved deep embedded clustering with local structure preservation. In: 26th International Joint Conference on Artificial Intelligence, Melbourne, pp. 1753–1759 (2017)
10. Elhamifar, E., Vidal, R.: Sparse subspace clustering. In: 22nd IEEE Conference on Computer Vision and Pattern Recognition, pp. 2790–2797. IEEE Press, Florida (2009)
11. Liu, G., Lin, Z., Yu, Y.: Robust subspace segmentation by low-rank representation. In: 27th International Conference on Machine Learning, Haifa, pp. 663–670 (2010)

12. Liu, G., Yan, S.: Latent low-rank representation for subspace segmentation and feature extraction. In: 13th International Conference on Computer Vision, pp. 1615–1122. IEEE Press, Barcelona (2011)
13. Liu, G., Lin, Z., Yan, S., Sun, J., Yu, Y., Ma, Y.: Robust recovery of subspace structures by low-rank representation. IEEE Trans. Patt. Anal. Mach. Intell. **35**(1), 171–184 (2013)
14. Li, C., Vidal, R.: Structured sparse subspace clustering: a unified optimization framework. In: 28th IEEE Conference on Computer Vision and Pattern Recognition, pp. 277–286. IEEE Press, Boston (2015)
15. Vidal, R., Favaro, P.: Low rank subspace clustering (LRSC). Patt. Recogn. Lett. **43**(1), 47–61 (2014)
16. Feng, J., Lin, Z., Xu, H., Yan, S.: Robust subspace segmentation with block-diagonal prior. In: 27th IEEE Conference on Computer Vision and Pattern Recognition, pp. 3818–3825. IEEE Press, Ohio (2014)
17. Lu, C.-Y., Min, H., Zhao, Z.-Q., Zhu, L., Huang, D.-S., Yan, S.: Robust and efficient subspace segmentation via least squares regression. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7578, pp. 347–360. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33786-4_26
18. Lu, C., Lin, Z., Yan, S.: Correlation adaptive subspace segmentation by trace lasso. In: 14th International Conference on Computer Vision, pp. 1345–1352. IEEE Press, Sydney (2013)
19. Park, D., Caramanis, C, Sanghavi, S.: Greedy subspace clustering. In: 28th Conference on Neural Information Processing Systems, Montréal, pp. 2753–2761 (2014)
20. Yang, Y., Feng, J., Jojic, N., Yang, J., Huang, T.S.: ℓ^0-sparse subspace clustering. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 731–747. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_45
21. Grave, E., Obozinski, G., Bach, F.: Trace lasso: a trace norm regularization for correlated designs. In: 25th Conference on Neural Information Processing Systems, Granada, pp. 2187–2195 (2011)
22. Patel, V., Vidal, R.: Kernel sparse subspace clustering. In: 22nd International Conference on Pattern Recognition, pp. 2849–2853. IEEE Press, Paris (2014)
23. De la Torre, F., Kanade, T.: Discriminative cluster analysis. In: 23rd International Conference on Machine Learning, Pittsburgh, pp. 241–248 (2006)
24. You, C., Robinson, D., Vidal, R.: Scalable sparse subspace clustering by orthogonal matching pursuit. In: 29th IEEE Conference on Computer Vision and Pattern Recognition, pp. 3918–3927. IEEE Press, Las Vegas (2016)
25. Ji, P., Salzmann, M., Li, H.: Efficient dense subspace clustering. In: IEEE Winter Conference on Application of Computer Vision, pp. 461–468. IEEE Press, Steamboat Springs (2014)
26. Ji, P., Zhang, T., Li, H., Salzmann, M., Reid, I.: Deep subspace clustering networks. In: 32nd Conference on Neural Information Processing Systems, Montréal, pp. 24–33 (2017)
27. Ren, Y., Domeniconi, C., Zhang, G., Yu, G.: A weighted adaptive mean shift clustering algorithm. In: SIAM International Conference on Data Mining, pp. 794–802. SIAM Press, Pennsylvania (2014)
28. Friedman, J.H., Meulman, J.J.: Clustering objects on subsets of attributes. J. Royal Stat. Soc. Ser. B (Stat. Methodol.) **66**, 815–849 (2004)