Structure-Aware Subspace Clustering

Simin Kou, Xuesong Yin, Yigang Wang, Songcan Chen, Tieming Chen, *Member, IEEE*, and Zizhao Wu

Abstract—Subspace clustering has attracted much attention because of its ability to group unlabeled high-dimensional data into multiple subspaces. Existing graph-based subspace clustering methods focus on either the sparsity of data affinity or the low rank of data affinity. Thus, the quality of data affinity plays an essential role in the performance of subspace clustering. However, the real-world data are generally high-dimensional, complex, and heterogeneous multi-source data, so that the data affinity learned by these methods cannot be completely dependent. Moreover, since these approaches always ignore the intrinsic structure of data, their grouping effect is relatively low. In this paper, we propose a novel unsupervised algorithm, called Structure-Aware Subspace Clustering (SASC), to address the above issues. SASC considers local and global correlation structures simultaneously to capture the intrinsic structure. Further, it integrates the captured structure into representation learning to gain a relatively precise data affinity. It is powerful to promote an all-around grouping effect and enhances the robustness and applicability of subspace clustering. Experiments on various benchmark datasets, including bioinformatics, handwritten digit, object image, and speech signal, demonstrate the effectiveness of the proposed algorithm.

Index Terms—Multi-source data, affinity graph, intrinsic structure, structure-aware, subspace clustering, representation learning

1 Introduction

In many real-world applications, such as computer vision, motion segmentation and bioinformatics recognition, the data are usually provided in the high-dimensional form, i.e., digital sequences, images, videos, etc. Such high-dimensional data can be well depicted by a union of multiple low-dimensional subspaces. Subspace clustering (SC) is an effective technique to solve the clustering problem of high-dimensional data. Specifically, SC aims to identify the subspaces where high-dimensional data points are located, and to group the data points into the corresponding subspaces [1]. For example, in a video of an intersection with multiple cars passing through, if one wants to separate each moving car, different subspaces are needed to describe the movement of different cars in the video. SC finds the latent subspaces suitable for each group of high-dimensional objects, and simultaneously groups these moving cars into different subspaces to achieve the segmentation of moving cars. Generally, SC is formulated as:

Problem (Subspace Clustering). Given a data matrix $X=[x_1,\ldots,x_n]\in\mathbb{R}^{m\times n}$, each column of X is sampled from a union of K subspaces $\{C_i\}_{i=1}^{K}$. SC aims to identify the subspaces and segment each data point x_i into its corresponding subspace.

Many researchers have early endeavored to study the SC problem through iterative methods [2]-[4], algebraic methods [5]-[7] and statistical methods [8]-[10]. The three traditional methods have limited ability to deal with complex samples, like data with noise, degeneracy, or coupled structure. In recent years, graph-based subspace clustering methods [11]-[22] have attracted more attention. The main difference between these methods is the construction of affinity graphs.

Graph-based approaches can be divided into supervised and unsupervised categories. Supervised methods [23] use the sample labels to construct a penalty map to directly count the affinity. Due to the expensive labor cost of sample labeling, unsupervised methods have been more widely applied and extended to estimate the affinity between samples through the Euclidean distance metric or the dictionary representation. The method of using the Euclidean distance metric is generally described by the neighbor graph expressing the local structure[24]. In recent years, methods based on the dictionary representation have become more popular. The dictionary is generally the dataset itself, so it is further derived into subspace clustering based on self-representation, and the affinity map is obtained by minimizing the reconstruction loss of the data self-representation and imposing representation constraints with different properties. Representation constraints in existing methods include the sparse representation [11],[17],[19],[20], the low-rank representation [12], [14],[25], least-squares regression [13],[18],[26], and the smooth representation [15],[25],[28]. These methods tend to focus only on a single representation between data, i.e., global or local. However, real-world data generally have complex structures and noise. The single representation is highly dependent on the data, which can easily lead to overfitting and out-of-sample problems. And the sparse representation does not consider the correlation between samples, which makes intra-subspace sparse and has a certain negative impact on the clustering accuracy. Moreover, noise and outliers may cause learned relationships between samples to be spurious, resulting in the poor

- *Simin Kou, Xuesong Yin, Yigang Wang and Zizhao Wu are with the Department of Digital Media Technology, Hangzhou Dianzi University, Zhejiang, China, 310018. E-mail: siminkou@hdu.edu.cn, yinxs@hdu.edu.cn, yigang.wang@hdu.edu.cn, wuzizhao@hdu.edu.cn*
- *Songcan Chen is with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Jinagsu, China. E-mail: s.chen@nuaa.edu.cn*
- *Tieming Chen is with the College of Computer Science and Technology, Zhejiang University of Technology, Zhejiang, China. E-mail: tmchen@zjut.edu.cn*

Corresponding author: Xuesong Yin, Yigang Wang

clustering performance. In summary, comprehensive data representation, the high correlation between samples, and mitigation or elimination of noise interference are all factors that have a positive impact on existing graph-based subspace clustering methods.

Motivated by this, we propose a novel unsupervised subspace clustering algorithm, called Structure-Aware Subspace Clustering (SASC), as shown in Fig. 1, which incorporates the structure representation into the subspace learning, and thus obtains a framework for comprehensive affinity capturing and subspace clustering. SASC efficiently combines three structure representations, including the adaptive local structure, the global structure based on maximizing scatter, and the global correlation structure by ridge regression approach. It aims to accurately perceive the intrinsic structure of data, gaining an affinity graph with high discriminative power for subspace learning. The experimental results show that, compared with other state-of-the-art methods, the proposed SASC algorithm is less dependent on multi-source data, and shows consistently good performance. Moreover, the clustering results obtained by SASC are relatively stable and robust to noise. In what follows, we summarize the main contributions of this paper as:

(1) Under the premise of considering data correlation, the proposed SASC can simultaneously capture the global and local structures, making full use of the complementarity of the composite structure. It can perceive the intrinsic structure of the high-dimensional data, which is valuable for expressing the affinity between samples with relative accuracy and reducing the sensitivity to multi-source data and noise interference. Therefore, our model is robust and effectively solves the problems of overfitting and out-of-sample, which may lead to better generation to real-world applications.

(2) SASC formulates a new grouping measure method based on the grouping effect, extracting the main information in a global scope by maximizing scatter. It can reduce the redundancy of the global representation when considering data correlation. In addition, we introduce this method into subspace representation learning together with local grouping and global correlation grouping, which captures more comprehensive structural information. Thus, SASC formulates an effective affinity measure for subspace clustering based on adaptive local grouping and global correlation grouping.

(3) An adaptive local structure-aware method is introduced, which not only ensures the accuracy of the local representation, but also better adapts to multi-source data. And it does not need to increase the manually adjusted parameters. Hence, this method can improve the adaptability and robustness of the algorithm in real-world applications, and the simplicity of the model is better maintained.

(4) The resulting structure graph can comprehensively describe the intrinsic relationship between samples with good interpretability and discriminability. Each internal element of this graph can directly reveal the affinity between any two samples. Such a structure graph is a general intrinsic one, and several popular graph-based approaches can be viewed as the special cases of this method to some extent. It can be naturally extended to other research fields, such as semi-supervised learning, multi-view learning, and so on.

(5) Compared with the methods using sparse representation and low-rank representation, SASC does not require any iterative computation and has a lower computational cost. In addition, simple scatter optimization leads to computational tractability and low model complexity.

We summarize the symbols used in this paper in Table 1. The remainder of the paper is organized as follows. Section 2 gives a brief overview of existing graph-based subspace clustering methods. In Section 3, we present the proposed SASC algorithm, provide the solution method and show some analysis. In Section 4, we perform the compared experiments with several baselines on 15 benchmark datasets. Finally, the conclusions are presented in Section 5.

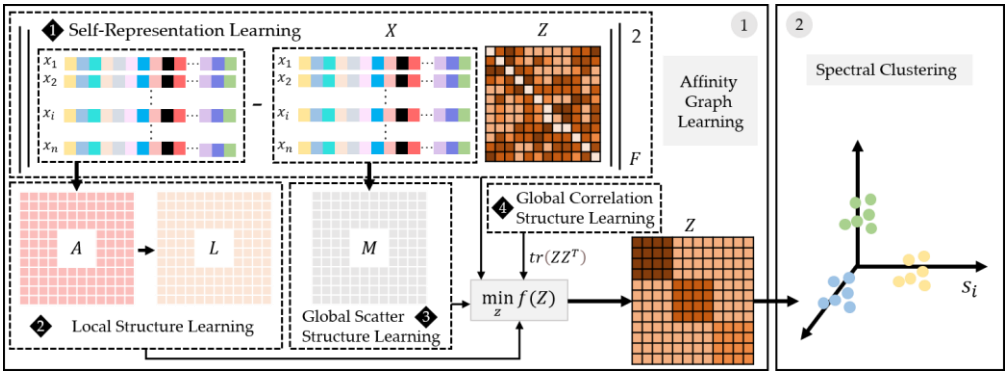


Fig. 1. The framework of the proposed Structure-Aware Subspace Clustering (SASC) algorithm.

2 Related Work

Affinity graph plays an essential role in the methods of graph-based subspace clustering. It is used to represent the affinity between any samples in a dataset and can be described by two representations, that is, global and local. In this paper, we uniformly refer to them as structure representation. Generally, the local structure only depicts partial samples with correlation, and the global one describes the affinity between all the samples in the dataset. Current popular graph-based methods are based on the Euclidean distance metric or the dictionary representation. The representative work and its key characteristics are shown in Fig. 2.

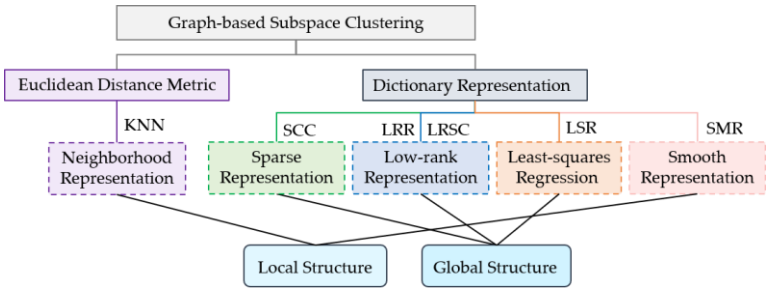


Fig. 2. Organization chart and classic algorithms of graph-based subspace clustering methods. Graph-based subspace clustering is usually divided into two categories: the Euclidean distance measurement and the dictionary representation. In these two kinds of approaches, some algorithms capture the local structure to learn the representation, while others employ the global structure to learn the representation.

The methods based on dictionary representation generally construct the affinity graph under the assumption of self-representation [11], [28]. Self-representation means that any sample in the same subspace can be obtained by a linear combination of other data samples, and this problem can be defined as:

$$\min_{Z} \|X - XZ\|_F^2 + \mu\Phi(Z), \quad s.t. Z \in \Psi, \tag{1}$$

where $X \in \mathbb{R}^{m\times n}$ is the original data, μ is a trade-off parameter, and $\Phi(Z)$ and Ψ are the regularizer and constraint set on the self-representation coefficient matrix Z.

TABLE 1
SUMMARY OF SYMBOLS

Symbols	Definitions
X	Data matrix
Z	Coefficient matrix
Z^T	The transpose of Z
L	Laplacian matrix
e	A vector with all elements of 1
A	Weight matrix
I	Identity matrix
H	Global scatter matrix
M	Negative matrix of H
Q	Label guidance matrix
P^{-1}	The inverse of matrix P
$tr(P)$	The trace of matrix P
$\|P\|_1$	ℓ_1-norm of matrix P
$\|P\|_*$	Nuclear-norm of matrix P
$\|P\|_F$	Frobenius-norm of matrix P

The main difference in the kind of methods is the regularizer, and different regularizers can help to obtain different representations for expressing the affinity graph. For example, using ℓ_1-norm as a regularizer can gain the sparse representation. SSC[11] obtains the sparse representation by solving the ℓ_1-norm minimization problem. Specifically, in the SSC model, the regularization term can be expressed as $\Phi(Z) = \|Z\|_1$. Because of its sparsity, SSC has been further expanded. EnSC [16] exploits the ℓ_1-norm and ℓ_2-norm mixed model to find the optimal self-expressive coefficient. S^{3}C[17] incorporates sparse representation learning and spectral clustering into a unified optimization scheme to alternately calculate the representation matrix and the segmentation matrix. S^{3}COMP[19] exploits a dropout technique into the self-representation model to address the connectivity problem related to SSC. SSC+E[20] uses the entropy-norm of the affinity matrix as the regularization term to obtain the entropy representation. Brbić *et al.* [21] introduced two S_0/ℓ_0 quasi-norms to achieve non-convex regularized LRSSC, capturing local and global structures of data.

ℓ_1-norm-based methods select as few samples as possible in the dataset to represent the data, and thus it can effectively eliminate the connection between samples belonging to different subspaces. The sparse representation, however, has weak expressive ability in the same subspace and cannot establish the correlation between samples, so it has a certain negative impact on the clustering accuracy. In order to utilize the correlation between samples to improve the clustering accuracy, the low-rank representation and the grouping effect are introduced to obtain an affinity graph with tighter intra-subspace. They leverage the nuclear-norm and ℓ_2-norm, respectively, and captured the structure representation using them are all global. LRR[12] seeks the low-rank representation by solving the nuclear-norm-based minimization issue. The regularization term in the LRR objective function can be formulated as $\Phi(Z) = \|Z\|_*$. LRSC[14] groups the data corrupted by noise or gross errors into different subspaces by solving a non-convex nuclear-norm optimization problem. The above two methods can capture the global structure by using the nuclear-norm. Wen *et al.* [40] incorporated the distance regularization and rank constraint into LRR to respect the global and local information of data. Least-squares regression (LSR) [13] is a classic algorithm based on the grouping effect, which segments highly correlated data together. Since LSR assigns consistent weights to all representations, it may be detrimental to inter-subspace discrimination. Therefore, Hu *et al.* [15] proposed an enforced grouping effect, introducing the local information into the original grouping effect and capturing the smooth representation, but they ignored the global correlation structure of data. SSRSC[18] introduces the non-negative constraint and the scalar constraint into the LSR model to avoid the problem of negative elements and forced symmetry of the affinity matrix. SOGFS[24] performs local structure learning and neighbor representation selection simultaneously, and can reduce the influence of redundant features in the original data on neighbor representation selection. Fu *et al.* [41] introduced a projection distance penalty into double LRR to capture the global and local structure. To improve the robustness, they exploited the Laplace rank constraint as a regularizer and incorporated it into the cost function. Wei *et al.* [42] constructed an adaptive graph to respect the local struc-

ture, which is merged into LSR to improve the sparsity of the obtained coefficient matrix. Qin *et al*. [43] regarded enforced block diagonal subspace clustering as an optimization problem to learn the expected similarity, and obtained the representation by solving the radial basis function kernel. Guo *et al*. [44] proposed a multi-view subspace clustering model induced by rank consistency to gain consistent low-rank structure information between view-specific self-expression matrices.

There are some works [30]-[32] that combine SSC and LRR to satisfy the inter-clustering separability and intra-clustering compactness of samples. However, the results obtained by ℓ_1-norm and nuclear-norm are approximate solutions. The ℓ_1-norm will underestimate the high-amplitude components[33], and the nuclear-norm may lead to biased results[34]. Therefore, some works introduce non-convex optimization. For example, Jiang *et al.* [35] replaced the nuclear-norm with the Ky Fan p-k-norm. Zhang *et al.* [36] introduced the Schatten-q norm which is closer to the rank function. Zheng *et al.* [37] proved that the smaller the value of p in the l_p ($0 \leqslant p < 1$) constraint, the more accurate the obtained solution is.

Although the above approaches improve the performance of SC to some extent, they suffer from the following issues. The sparse representations obtained by SSC-based methods are weakly expressive in the same subspace and cannot establish the correlation between samples. LRR and its variants have relatively strict assumptions on data distribution, which must theoretically be used on independent linear subspaces, and require a high time overhead for iterative optimization. Moreover, these approaches cannot perceive the intrinsic structure of data. To address the above issues, in the next section, we propose a novel algorithm named SASC to simultaneously consider global and local structures for SC.

3 Structure-Aware Subspace Clustering

In this paper, we aim to learn the all-around and precise affinity graph, and design the learning process as follows: (a) self-representation learning, (b) local structure learning, (c) global scatter structure learning, and (d) global correlation structure learning. SASC inherits from the problem (1), and the regularizer is improved as $\Phi(Z) = \Phi_1(Z) + \Phi_2(Z) + \Phi_3(Z)$. Specifically, we use the grouping effect to consider highly correlated samples as a bridge, thus presenting a novel global group measure, which achieves comprehensive structure awareness by combining global and local structures with different properties. From the global correlation structure considered by the classic grouping effect, we further expand the adaptive local structure awareness and the global scatter structure awareness based on it.

3.1 Biased Estimated Grouping Effect

The grouping effect [13],[15] is a very important approach in the subspace learning based on dictionary representation, which is used to find highly correlated samples. Specifically, suppose x_i and x_j are two arbitrary original samples. z_i and z_j are the corresponding self-expression coefficient of x_i and x_j respectively. The grouping effect thinks that the relationship of any two samples is consistent with that of their self-expression coefficients. It can be defined as

$$\|x_i - x_j\|_2^2 \to 0 \implies \|z_i - z_j\|_2^2 \to 0 \tag{2}$$

Generally, some samples in the real data X are highly linearly correlated, that is, X is not full rank. It makes the matrix X^TX that can be used for affinity measurement tends to be non-singular. At this time, the calculation error of $(X^TX)^{-1}$ will become larger, resulting in the lack of stability and reliability of the obtained self-expression coefficient matrix Z, and also increasing the risk of overfitting. In order to solve this problem, a bias can be added to X^TX, in which the unbiasedness can be sacrificed for higher accuracy to better describe the data. The biased grouping process can be expressed as:

$$\phi_1(Z) = \sum_{i=1}^{n}\sum_{j=1}^{n}\|z_i - z_j\|_2^2 + \frac{1}{n}\|Z^T e\|_2^2 \tag{3}$$

e is an n-dimensional column vector with all element values of 1. Further, function (2) is transformed into:

$$\phi_1(Z) = tr(ZZ^T) \tag{4}$$

Equations (4) and (3) are used to formulate the global correlation structure of data. Since the expression of Equation (4) is simpler, we use it to capture the global correlation structure in our algorithm. By minimizing (4), the samples are guaranteed to be highly correlated, while avoiding overfitting and out-of-sample problems.

3.2 Adaptive Local Structure Awareness

If x_i and x_j are close neighbors, this relationship represents as $x_i \in N_k(x_j)$. According to the grouping effect, z_i and z_j share the same neighbor relationship with x_i and x_j, that is:

$$x_i \in N_k(x_j) \Rightarrow z_i \in N_k(z_j). \tag{5}$$

It can help to learn the affinity graph via the local neighbor structure between samples, to improve the local grouping accuracy. However, existing nearest neighbor representations generally have a strong dependence on data. In order to dynamically adjust the neighbor relationship, we introduce an adaptive neighbor representation learning method. Referring to (4), the objective function of adaptive local grouping is defined as:

$$\phi_2(Z) = tr(ZLZ^T) \tag{6}$$

where the matrix $L = D - A$ is a Laplacian matrix. A denotes the weight matrix on an adaptive neighbor graph and is defined later in Equation 9. The matrix D is a diagonal one whose diagonal entries are sums of the corresponding column (or row) of A. By minimizing problem (6), we expect that if two samples x_i and x_j are close, their corresponding representations z_i and z_j are close to each other. By constructing such an adaptive neighbor graph L, our algorithm can respect the local structure and automatically captures the precise sub-connected components with the given data.

Let a_{ij} be the nearest neighbor relationship between x_i and x_j. According to the property of k-nearest neighbors, when there is a k-nearest neighbor relationship on x_i and

x_j, a_{ij} can be set to $0 < a_{ij} \leq 1$, otherwise a_{ij}=0. In addition, the neighbor relationship between x_i and any other sample can be represented by a_i vector. The following objective function is obtained:

$$\min_{A} \sum_{j=1}^{n} \left\| x_i - x_j \right\|_2^2 a_{ij}, \ \ s.t. \, a_i^T e = 1, 0 \leq a_{ij} \leq 1. \quad (7)$$

To increase adaptability and avoid trivial solutions [24], constraints on $A = \{a_{ij}\}^{n \times n}$ need to be imposed:

$$\min_{A} \sum_{j=1}^{n} a_{ij}^2, \ \ s.t. \, a_i^T e = 1, 0 \leq a_{ij} \leq 1. \quad (8)$$

From (7) and (8), the overall objective function for calculating the optimal adjacency graph A is:

$$\min_{A} \sum_{j=1}^{n} \left(\left\| x_i - x_j \right\|_2^2 a_{ij} + \lambda a_{ij}^2 \right), \quad s.t. \ a_i^T e = 1, 0 \leq a_{ij} \leq 1. \quad (9)$$

Let $d_{ij} = \left\| x_i - x_j \right\|_2^2$, the objective function (9) can be further rewritten as:

$$\min_{A} \sum_{j=1}^{n} \left(\frac{1}{\lambda} d_{ij} a_{ij} + \lambda a_{ij}^2 \right), \quad s.t. \ a_i^T e = 1, 0 \leq a_{ij} \leq 1. \quad (10)$$

And $\sum_{j=1}^{n} a_{ij} = 1$, then (10) can be converted into:

$$\min_{A} \left\| a_i + \frac{1}{2\lambda} d_i \right\|_2^2, \quad s.t. \ a_i^T e = 1, 0 \leq a_{ij} \leq 1. \quad (11)$$

According to (11), we construct the Lagrangian function:

$$L(a_i, \delta, \theta_i) = \frac{1}{2} \left\| a_i + \frac{1}{2\lambda} d_i \right\|_2^2 - \delta(a_i^T e - 1) - \theta_i a_i^T. \quad (12)$$

Differentiating (12) with respect to a_{ij} and setting the differential to zero gives:

$$\frac{\partial L}{\partial a_{ij}} = a_{ij} + \frac{1}{2\lambda_i} d_{ij} - \delta - \theta = 0. \quad (13)$$

According to the KKT condition, $\theta_i a_{ij} = 0$ can be obtained, and then combined with (13), we gain:

$$\left(a_{ij} + \frac{d_{ij}}{2\lambda_i} \right) a_{ij} - \delta a_{ij} = 0. \quad (14)$$

According to (14) as (15), we obtain the adjacency graph A. Noting that, all elements in A are all non-negative:

$$a_{ij} = \left(-\frac{d_{ij}}{2\lambda_i} + \delta \right)_+. \quad (15)$$

It can be seen from (15) that in addition to the original sample, a_{ij} is also related to the parameters δ and λ. Generally, setting the nearest neighbor parameter k is much easier than manually adjusting the regularization parameter λ. The reason is that k is an integer with a clear meaning, that is, the number of samples that are close to the current sample. Therefore, in order to make the algorithm more robust and convenient, the above-mentioned parameters δ and λ can be adapted to the given original sample X, instead of being specified from subjective experience or randomness. This problem can be addressed with the k-nearest neighbor of the original sample. Putting $a_{i,k} > 0$ and $a_{i,k+1} = 0$ into (15) respectively, can obtain:

$$\begin{cases} -\frac{1}{2\lambda_i} d_{i,k} + \delta > 0 \\ -\frac{1}{2\lambda_i} d_{i,k+1} + \delta \leq 0 \end{cases}. \quad (16)$$

From $a_i^T e = 1$, we can get:

$$\sum_{j=1}^{k} \left(-\frac{d_{i,k}}{2\lambda_i} + \delta \right) = 1 \Rightarrow \ \delta = \frac{1}{k} + \frac{1}{2k\lambda_i} \sum_{j=1}^{k} d_{ij}, \quad (17)$$

and then gain the parameter λ. Subsequently, we put the obtained δ from (17) into (16):

$$\begin{cases} \frac{1}{k} + \frac{1}{2k\lambda_i} \sum_{j=1}^{k} d_{ij} > \frac{d_{i,k}}{2\lambda_i} \\ \frac{1}{k} + \frac{1}{2k\lambda_i} \sum_{j=1}^{k} d_{ij} \leq \frac{d_{i,k+1}}{2\lambda_i} \end{cases} \Rightarrow \begin{cases} \lambda_i > \frac{k}{2} d_{i,k} - \frac{1}{2} \sum_{j=1}^{k} d_{ij} \\ \lambda_i \leq \frac{k}{2} d_{i,k+1} - \frac{1}{2} \sum_{j=1}^{k} d_{ij} \end{cases}$$

$$\Rightarrow \frac{k}{2} d_{i,k} - \frac{1}{2} \sum_{j=1}^{k} d_{ij} < \lambda_i \leq \frac{k}{2} d_{i,k+1} - \frac{1}{2} \sum_{j=1}^{k} d_{ij}. \quad (18)$$

We set $\lambda_i = \frac{k}{2} d_{i,k+1} - \frac{1}{2} \sum_{j=1}^{k} d_{ij}, (i = 1,2, \cdots, n)$, and λ is the mean of $\{\lambda_1, \lambda_2, \cdots, \lambda_n\}$. λ can represent as:

$$\lambda = \frac{1}{n} \sum_{j=1}^{n} \left(-\frac{k}{2} d_{i,k+1} - \frac{1}{2} \sum_{j=1}^{k} d_{ij} \right). \quad (19)$$

Putting (17) and (19) into (15) obtain the optimal adjacency graph $\tilde{A} = [a_{ij}] \in \mathbb{R}^{n \times n}$, and then diagonalizing $\tilde{A}$ get the corresponding adjacency matrix $A^* = \left(|\tilde{A}| + |\tilde{A}^T| \right)/2$. After that, taking its diagonal elements gain the degree matrix $D = \sum_{j=1}^{n} A_{ij}^*$. Finally, the adaptive neighbor graph $L = D^{-1/2}(D - A^*)D^{-1/2}$ is calculated by A^* and D. Such a graph can capture the local structure that satisfies the grouping effect in the original sample, to ensure the local invariance between similar samples.

3.3 Global Scatter Structure Awareness

Subspace clustering is to segment high-dimensional data into multiple subspaces for data classification. It is true that the local neighbor relationship plays an important role in determining whether two samples can be divided into a class, but relying only on the local structure is not very reliable. The pivotal reason is that its sensitivity to outliers is lower than the global structure. If the local and global structures are considered simultaneously, the important information of the data can be recovered to the maximum extent in the subspace, which has a positive influence on improving the clustering accuracy.

PCA [38] is a classical global representation learning method, which maps high-dimensional data to lowdimensional subspaces through basis transformation, and expects the retained projection dimension to contain the largest information. In other words, this is a process of maximizing scatter on the global range. Hence, it can extract the main information of the data with the least amount of information loss. Based on PCA, we propose a global structure representation of high-dimensional data,

namely the global scatter structure. We capture such a global structure $H = I - (1/n)ee^T$ by maximizing the total scatter S_t of the original sample X, and describe the problem as:

$$\max_H X^T H X . \tag{20}$$

Let $M = -H$, we can transfer problem (20) into:

$$\min_M X^T M X. \tag{21}$$

The key to unsupervised learning is the prior assumption of the consistency, which indicates [45]: (1) nearby samples may have the same cluster label; and (2) samples with the same structure are likely to have the same cluster label. Generally, KNN-based methods rely on the first assumption of the local consistency, while PCA-based ones depend on the second assumption of the global consistency. Thus, if any two samples x_i and x_j have the same structure, their corresponding representations z_i and z_j also have the same structure, which satisfies the global consistency. Moreover, it has been proved that the global consistency of representations satisfies the global scatter graph M [45,46]. In order to introduce the global scatter information in representation learning of the affinity, the objective function is defined as:

$$\phi_3(Z) = tr(ZMZ^T) . \tag{22}$$

3.4 Structure-Aware Subspace Clustering

The SASC algorithm comprehensively perceives the intrinsic structure of the data through three grouping effects oriented to the above different structural representations. It aims to improve the discriminative power and accuracy of the sample affinity graph and perform efficient subspace clustering analysis. We define the complete objective function of SASC as:

$$\begin{aligned} f(Z) &= \|X - XZ\|_F^2 + \phi_1(Z) + \phi_2(Z) + \phi_3(Z) \\ &= tr\big((X - XZ)(X^T - Z^T X^T)\big) \\ &\quad + \alpha tr(ZLZ^T) + \beta tr(ZMZ^T) + \gamma tr(ZZ^T), \end{aligned} \tag{23}$$

where α, β and γ are trade-off coefficients, and they affect the learning of the local structure, global scatter structure and global correlation structure, respectively. Using the convex programming strategy, obtaining the optimal self-expression coefficient matrix $\tilde{Z}$ requires differentiating (23) with respect to Z and setting it to zero:

$$-X^T X + X^T XZ + \alpha ZL + \beta ZM + \gamma Z = 0. \tag{24}$$

Further, in order to avoid numerical instability, L is required to be strictly positive[15]. Hence, we replace L into $L^* = L + \epsilon I$, where $0 < \epsilon \le 1$ and I is the identity matrix. It gains $\tilde{Z}$ by putting L^* into (24) and solving:

$$\big(X^T X + \gamma I\big)\tilde{Z} + \tilde{Z}\big(\alpha L^* + \beta M\big) = X^T X. \tag{25}$$

According to [39], we finally get a unique solution $\tilde{Z}$ by the Bartels-Stewart algorithm. Then, using $\tilde{Z}$ calculates the optimal affinity graph Z^*:

$$Z^* = \frac{\left|\tilde{Z}\right| + \left|\tilde{Z}^T\right|}{2}. \tag{26}$$

In summary, the overall process of the proposed structure-aware subspace clustering algorithm (SASC) is summarized as follows:

Algorithm 1. Structure-Aware Subspace clustering(SASC)

Input: The original data $X = [x_1, \cdots, x_n] \in \mathbb{R}^{m \times n}$, the number of nearest neighbors k, trade-off parameters α, β and γ.
Output: Results of clustering, strictly positive adaptive neighbor graph L^*, optimal global scatter graph M^*, and the optimal affinity graph Z^*.

1. **Adaptive neighbor structure awareness:**
   Build an optimal adjacency matrix A^* by (15), (17) and (19); Calculate the adaptive neighbor graph L and gain the optimal neighbor graph $L^* = L + \epsilon I$ by the enforced positive strategy.
2. **Global scatter structure awareness:**
   Use (21) to get the optimal global scatter graph M^* of the original sample.
3. By (25), learn the optimal self-expression coefficient matrix $\tilde{Z}$ via the joint grouping effects.
4. By $\tilde{Z}$ and (26), calculate the optimal affinity graph Z^*.
5. Use spectral clustering on Z^* to get the clustering results.

3.5 Computational Complexity

In problem (25), solving Z is a standard Sylvester equation, which has a unique solution by the Bartels-Stewart algorithm [39]. Therefore, the computational complexity of our algorithm is $O(n^3)$, where n is the number of the samples. In LRR and its variants, since the singular value thresholding (SVT) is used to obtain a low-rank representation, their computational complexity is $O(Tn^3)$ where T is the number of iterations. Moreover, the complexity of most methods based on the mixture of ℓ_1-norm and nuclear-norm regularizations also is $O(Tn^3)$. SASC does not require iterative calculation, so it is faster than these approaches.

3.6 Extension of SASC

In semi-supervised learning, a given dataset is usually composed of a small number of labeled samples and a large number of unlabeled samples. Specifically, given a data matrix X=[x_1,…,x_n]$\in \mathbb{R}^{m \times n}$, where $x_i \in \mathbb{R}^{m \times 1}$ denotes the i-th sample, the first l data points are labeled and the rest ones are unlabeled. Suppose these labeled data points belong to c classes. We formulate an $l \times c$ indicator matrix F where f_{ij} = 1 if x_i belongs to the j-th class and f_{ij} = 0 otherwise. Therefore, we can define a label guidance matrix Q as follows:

$$Q = \begin{pmatrix} F_{l \times c} & 0 \\ 0 & I_{n-l} \end{pmatrix} \tag{27}$$

where $I_{n\text{-}l}$ denotes an (n-l)× (n-l) identity matrix. With the label guidance matrix Q, we can expand our SASC to the semi-supervised scenario and define the following objective function:

$$\min_Z \left\|X - XQZ\right\|_F^2 + \alpha tr(ZLZ^T) + \beta tr(ZMZ^T) + \gamma tr(ZZ^T) \tag{28}$$

Multi-view SC method aims to segment multi-view data into underlying clusters. Suppose that X^1,…,X^p are the data matrices of p views and X^d is the d-th view data. The objective function of multi-view SASC is defined as

$$\begin{aligned} \min_Z \left\|X^d - X^d Z^d\right\|_F^2 &+ \alpha tr(Z^d L_d (Z^d)^T) \\ &+ \beta tr(Z^d M_d (Z^d)^T) + \gamma tr(Z^d (Z^d)^T) \end{aligned} \tag{29}$$

The extension of SC methods to semi-supervised or multi-view scenarios is an interesting research direction, which will be our next research topic.

4 Experiments

In this section, we evaluate the performance of the SASC algorithm on 15 benchmark datasets that compare with 12 baseline methods and then analyze several experimental results to present our advantages.

4.1 Datasets

To validate the effectiveness and robustness of the proposed SASC algorithm on different kinds of data, we consider four types: bioinformatics, handwritten digits, object images, and speech signals. Each type contains several publicly available datasets.

4.1.1 Bioinformatic Data

We select seven bioinformatics datasets including gene sequences and lesion images. The number of samples, features and classes of these datasets are shown in Table 2.

Chest is a dataset of the X-ray images that record whether patients have pneumonia. The other datasets are all gene sequences with different feature dimensions. For example, SRBCT is a tumor gene sequence dataset with 88 samples, each with 2308 gene expressions, of which 5 samples are identified as non-SRBCT. We employ 83 samples that are diagnosed with SRBCT, they are divided into four categories: EWS, RMS, NB, and BL. The COVID19 is a dataset of X-ray images from patients with Covid-19 disease, healthy, lung opacity, or viral pneumonia.

TABLE 2
Datasets Description

Datasets	Samples	Features	Classes
CLL_SUB	111	11340	3
GLI_85	85	22283	2
Lung	203	3312	5
TOX_171	171	5748	4
SRBCTML	83	2308	4
Chest	4933	1024	2
COVID19	4800	1024	4
USPSML	1854	256	10
MFD	2000	649	10
COIL20	1440	1024	20
PalmData25	2000	256	100
UMIST	575	1025	20
Faces96	3016	4096	151
Isolet1	1560	617	26
Isolet5	1559	617	26

4.1.2 Handwritten Digit Data

Handwritten digits data contains 10 classes that are the digits from 0 to 9. This paper utilized two classic databases: (1) USPS is a grayscale image library, and made by the United States Postal Service; (2) MFD is a multi-view database. The details of sampled datasets from USPS and MFD are illustrated in Table 2.

4.1.3 Object Image Data

We use two object image datasets: (1) the daily-object image dataset COIL20; (2) the natural-object palm image dataset PalmData25; (3) UMIST is a face image dataset with 20 subjects, and each subject demonstrates various images with different angles and head gestures; and (4) Faces96 contains 3016 face images with various changes and is sampled from 151 people who are from different races. The details of these datasets are summarized in Table 2.

4.1.4 Speech Signal Data

Isolet is a speech signal library with 26 English letters pronounced twice by 150 speakers. A string is used to identify each speaker, which has 617 dimensions that depict the gender, initial, a unique digit, and other features. The 150 individuals were divided into five groups, named Isolet1, Isolet2, Isolet3, Isolet4 and Isolet5. We select two subsets, Isolet1 and Isolet5, for experiments. The number of samples, features, and classes of the two subsets are demonstrated in Table 2.

The 15 datasets used in the experiments are high-dimensional. In particular, the dimension of bioinformatics datasets is higher than that of other types of datasets. Each dataset has been transformed into an explicit data matrix. From these explicit data matrices, we can observe that although these high-dimensional data matrices are not sparse, the values of their features differ greatly. In the CLL_SUB dataset with 11340 dimensions, for instance, the values of the features vary from 10 to 100000. Similar observations can be obtained from other bioinformatics datasets. Although the feature values of several image datasets do not vary as much as those of bioinformatics datasets, their feature values also vary in the range of 0 to 255. Hence, these data generally have complex structures.

4.2 Baselines and Evaluation Metrics

As presented in Section 1, the proposed algorithm belongs to the unsupervised SC category. Thus, from a methodological perspective, we select 12 popular unsupervised methods as baselines for comparison. The 12 state-of-the-art approaches methods are SSC[11], LRR[12], LSR[13], LRSC[14], SMR[15], EnSC[16], S^3C[17], SSRSC[18], S^3COMP[19], SSC+E[20], l_0-LRSSC[21], and SOGFS[24]. Actually, the above approaches also follow such a comparison protocol to select baseline models. Compared with existing methods, although the proposed algorithm adds several recent methods [20],[21] as comparison baselines, datasets and evaluation metrics used in our algorithm still follow the comparison protocol.

To make the experiments fair enough, we use the same strategy to set parameters for all the methods. Specifically, we set a parameter candidate set $\Omega = \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}\}$ and the optimal parameters are searched from this grid. For the selected parameters, the best result of each approach is reported. All the experiments were run under the MATLAB2021a environment on a machine with 3.20 GHz CPU and 16GB RAM. Besides, since all the compared methods are unsupervised, we regard all samples included in each dataset as test ones. Thus, the experimental results we report are the clustering result of each compared algorithm on the entire dataset.

To evaluate the clustering performance of the algorithm, we used two common evaluation metrics in the experiments, including unsupervised clustering Accuracy

rate (ACC) and Normalized Mutual Information (NMI). The detailed definitions of ACC and NMI are described in [40].

4.3 Experimental Results

The experimental results on the four types of datasets are shown in Tables 3-4, respectively. The corresponding analysis of the results will be provided below.

4.3.1 Results on Bioinformatics Data

According to the experimental findings, we have drawn the following conclusions:

(1) There are few samples and classes on the datasets CLL_SUB and GLI_85, and their features have reached tens of thousands. As shown in Tables 3-4, we can observe that the ACCs of different methods on GLI_85 are about 70%, but the other results almost stay at a low level, revealing the negative impact of high feature dimension on clustering. Furthermore, the two datasets obtained similar results on existing graph-based clustering approaches, focusing on the regular constraints of various representations, neither of which significantly contributed to the clustering results, especially with the neighbor representations. The proposed SASC algorithm achieves the highest ACC on these two datasets, and is also comparable to the best results in the baseline algorithm in NMI, demonstrating the effectiveness of the algorithm on high-dimensional digital sequences.

TABLE 3
CLUSTERING RESULTS (ACC%) OF THE COMPARED APPROACHES ON 15 DATASETS

Datasets	SSC	LRR	LSR	LRSC	SMR	EnSC	S^3C	SSRSC	S^3COMP	SSC+E	l_0-RSSC	SOGFS	SASC
CLL_SUB	51.35	54.05	54.96	54.96	55.85	54.05	55.86	54.96	57.66	55.86	55.86	45.95	**61.26**
GLI_85	70.59	70.59	70.59	75.29	77.65	72.33	72.94	72.94	68.24	67.06	70.59	67.06	**83.53**
Lung	86.70	68.47	86.21	76.36	88.18	87.19	87.69	84.24	73.89	86.70	75.86	83.74	**88.67**
TOX_171	45.02	46.78	57.31	47.37	56.73	49.70	49.71	56.73	40.94	46.19	45.61	40.94	**58.48**
SRBCTML	41.96	43.37	55.42	53.01	55.42	54.56	56.63	60.24	61.45	48.19	67.06	65.06	**68.67**
Chest	65.74	70.40	87.72	85.43	84.98	73.78	72.80	86.66	75.65	70.09	**88.57**	56.55	85.93
COVID19	55.19	53.86	64.66	67.02	68.27	61.88	65.89	62.88	56.28	45.72	45.69	58.44	**70.33**
USPSML	64.06	68.29	70.44	66.24	75.51	61.54	72.06	69.42	61.25	68.07	61.22	65.10	**75.62**
MFD	92.50	93.15	90.10	89.30	94.55	74.06	94.75	91.95	77.35	73.00	72.05	64.00	**95.00**
COIL20	67.36	47.29	61.32	61.11	65.07	50.69	77.57	75.07	54.10	77.92	78.13	69.65	**88.13**
PalmData25	68.85	76.10	84.10	84.15	90.05	52.50	56.45	**98.45**	52.70	72.90	96.90	76.25	90.10
UMIST	61.21	50.43	55.3	50.26	66.08	64.17	62.14	66.95	52.17	59.13	52.34	57.21	**69.04**
Faces96	48.44	47.24	67.24	63.22	71.05	63.46	61.06	66.01	54.6	63.79	67.77	61.73	**77.02**
Isolet1	60.06	57.37	68.94	66.92	69.68	65.51	51.67	54.87	38.33	**72.50**	68.97	67.24	71.73
Isolet5	53.3	43.81	54.33	**59.14**	56.19	39.13	45.67	42.14	33.03	55.16	52.79	50.16	56.19

TABLE 4
CLUSTERING RESULTS (NMI%) OF THE COMPARED APPROACHES ON 15 DATASETS

Datasets	SSC	LRR	LSR	LRSC	SMR	EnSC	S^3C	SSRSC	S^3COMP	SSC+E	l_0-RSSC	SOGFS	SASC
CLL_SUB	17.02	26.22	26.31	26.31	**34.12**	26.08	28.30	26.31	29.78	**34.12**	22.84	27.82	29.39
GLI_85	17.67	23.86	23.86	25.10	31.20	16.78	26.07	26.07	15.88	15.69	23.86	20.88	**36.38**
Lung	66.98	51.57	64.42	52.02	68.33	64.72	65.34	59.79	49.85	67.65	53.72	60.11	**70.05**
TOX_171	26.40	21.10	30.47	25.56	35.05	28.25	26.78	**37.92**	19.22	22.01	23.94	13.75	33.36
SRBCTML	14.02	17.82	28.75	33.54	29.30	30.96	44.11	35.69	32.57	20.06	**56.95**	45.42	47.52
Chest	12.21	15.85	43.45	38.06	37.22	19.87	14.76	39.46	25.28	17.72	**45.35**	36.13	39.74
COVID19	27.83	28.44	42.28	42.88	43.67	35.58	33.39	36.86	32.79	28.8	28.71	32.49	**45.07**
USPSML	60.09	66.02	68.84	65.67	74.84	62.77	68.63	68.17	63.21	65.12	65.13	62.54	**75.71**
MFD	86.17	86.51	82.68	81.93	89.03	76.77	89.65	85.51	78.13	69.59	68.79	59.86	**89.73**
COIL20	77.28	60.77	72.49	73.42	73.09	53.81	89.74	86.75	60.72	87.85	88.62	76.06	**92.91**
PalmData25	89.75	92.59	95.53	94.00	96.71	59.65	75.86	99.65	74.20	91.31	99.31	90.68	96.77
UMIST	72.84	68.58	72.35	65.28	80.94	79.95	74.69	81.53	64.33	75.6	72.35	71.54	**84.85**
Faces96	71.04	66.42	84.53	76.15	83.54	78.36	85.57	85.87	76.83	83.84	84.69	85.03	**90.77**
Isolet1	73.30	71.21	77.74	75.67	78.95	77.44	67.33	67.13	49.20	79.25	73.53	78.65	**80.26**
Isolet5	70.37	63.47	68.64	68.79	72.31	47.2	61.75	59.77	44.68	71.49	63.76	67.85	**72.82**

TABLE 5
RUNNING TIME (S) OF EACH ALGORITHM ON TWELVE DATASETS

Datasets	SSC	LRR	LSR	LRSC	SMR	EnSC	S^3C	SSRSC	S^3COMP	SSC+E	l_0-RSSC	SOGFS	SASC
CLL_SUB	218.2	4.261	0.042	12.51	0.053	4.721	3.359	0.112	2.792	0.013	0.075	0.302	0.051
GLI_85	603.3	5.773	0.054	37.40	0.061	23.42	3.027	0.115	5.606	0.018	0.079	0.091	0.044
Lung	33.32	3.609	0.094	0.182	0.086	1.001	3.106	0.178	1.870	0.019	0.143	4.037	0.085
TOX_171	69.19	4.330	0.072	0.388	0.076	1.393	4.222	0.175	2.313	0.018	0.136	0.575	0.068
SRBCTML	8.329	0.683	0.048	0.403	0.051	0.285	0.422	0.068	0.499	0.009	0.066	0.078	0.049

Chest	482.4	175.5	4.957	3.912	32.38	27.55	598.1	96.46	37.98	16.28	314.7	37.05	28.33
USPSML	33.09	8.317	0.897	0.792	2.190	5.319	39.78	10.08	4.753	0.928	123.9	5.503	1.835
MFD	58.96	32.31	0.932	0.886	2.384	6.762	60.52	12.35	6.823	1.177	16.27	16.43	2.127
COIL20	46.81	65.16	0.682	0.866	1.341	3.555	158.6	6.019	6.970	0.521	6.242	9.914	1.273
PalmData25	39.31	8.816	2.984	3.046	4.043	11.43	156.3	13.62	19.18	1.649	17.54	8.112	4.041
Isolet1	35.50	22.31	0.929	1.086	1.702	11.61	36.39	7.255	6.431	0.645	63.03	11.77	1.658
Isolet5	42.04	22.84	0.946	1.099	1.767	12.22	38.17	7.295	6.378	0.662	64.84	13.07	1.742

(2) The algorithms with the sparse representation, such as SSC, S3C, and SSC+E, show better clustering performance than the algorithms using the low-rank representation. Obviously, ℓ_0-LRSSC considers both sparse and low-rank, and also gains relatively poor clustering results. The above-mentioned observations suggest that the low-rank representation is not conducive to the accurate identification of Lung and TOX_171 information. In addition, although LSR, SMR, SSRSC, and SOGFS have good clustering performance, the proposed SASC reports better clustering results by combining grouping effects with different structures.

(3) The clustering performances of each method on SRBCTML and Chest are quite different. For SRBCTML, algorithms that employ a single structure generally fail to obtain acceptable clustering accuracy. The ℓ_0-LRSSC represented by the combination of sparse and low-rank representations obtains better results, which shows that SRBCTML has higher requirements for the comprehensiveness of information. A breakthrough in clustering performance improvement cannot be achieved by considering either the local or the global structure. SASC achieves better results in terms of clustering accuracy, possibly benefiting from the new global grouping measure strategy we introduce, and better exploiting the key information of the sample. Chest is different from the other 5 datasets about gene sequences and expresses lesion information through pixels, so LSR, SMR and SSRSC perform better on Chest, which is comparable to ℓ_0-LRSSC considering the combined structure. The proposed SASC demonstrates comparable performance with the best method.

4.3.2 Results on Handwritten Digit Data

From Tables 3-4, we acquire the following conclusions:

(1) The proposed SASC algorithm reports the best clustering results on the USPSML and MFD, which verifies the effectiveness of the algorithm on subspace clustering tasks.

(2) We find that the performance of the two datasets on SOGFS shows the lowest level. It can be inferred that the neighbor relationship is not suitable for the recognition of handwritten digits. Although the proposed SASC also draws on the idea of neighbor in obtaining the local structure, we introduce the adaptive learning of the neighbor representations to capture the local structure using the highly correlated grouping effect. Hence, our algorithm still performs well, and the clustering results prove that the proposed SASC has wider adaptability and higher robustness.

4.3.3 Results on Object Image Data

This section draws the following conclusions from the data of routine objects and natural plants on COIL20 and PalmData25:

(1) The number of classes in the COIL20 is much smaller than that of features. The clustering results are significantly worse for LRR, EnSC and S^{3}COMP, but better for S^{3}C and l_0-LRSSC. In addition, it performs well on SSRSC and SSC+E, but the clustering performance is mediocre on several classic models such as SSC, LSR, LRSC and SMR. Our algorithm has the best clustering results on COIL20, which verifies the effectiveness of the proposed SASC for learning an object image representation.

(2) The number of classes on PalmData25 is slightly smaller than that of features. This dataset can achieve good results on most algorithms, especially on methods using the grouping effect, such as LSR, SMR, SSRSC. However, it performs poorly on sparse representation methods, such as EnSC, S^{3}C, and S^{3}COMP. Results obtained by SASC are comparable to the best results and outperform most state-of-the-art baseline algorithms.

4.3.4 Results on Speech Signal Data

As can be seen in Tables 3-4, the two sub-datasets basically perform similar results, from which we draw the following conclusions:

The S^{3}COMP algorithm using dropout has a relatively poor result on the recognition of speech signals with timing characteristics, which is lower than all other algorithms. Instead, it has achieved good results on several classic algorithms, such as SSC, LSR, LRSC and SMR. The clustering performance of SSC+E is slightly better than that of SSC. SASC shows the best performance on the two speech signal datasets, which verifies the effectiveness of the proposed algorithm on speech data.

In a word, the NMI value of our SASC is lower than that of the best baseline method on CLL_SUB, TOX_171, SRBCTML, Chest, and PalmData25 datasets. On the other hand, the ACC value of our SASC is higher than that of all baseline methods on CLL_SUB, TOX_171, and SRBCTML datasets. In other words, the proposed algorithm is completely defeated by the baseline method only on Chest and PalmData25 datasets. The ACC metric calculates how many samples are correctly clustered into the corresponding class. NMI represents the correlation between the predicted labels and ground-truth labels without a bias towards smaller clusters. The reason why NMI of SASC is inferior to that of the baseline algorithm on the five datasets may be the low correlation between the predicted labels and ground-truth labels, which is the dataset dependent. For example, the Chest dataset is different from the other 5 datasets about gene sequences and expresses lesion information through pixels, which is very

difficult to obtain the intrinsic structure of the image. At the same time, such images differ very little in appearance, and have a large number of repetitive elements. It means that distinguishing these images needs to rely on more small correlations, so ℓ_0-LRSSC considering the spare and low-rank structure is superior to other methods on the Chest dataset. The proposed SASC demonstrates comparable performance with ℓ_0-LRSSC. Since the number of classes on PalmData25 is slightly smaller than that of features, the global structure plays an important role in segmenting data into different clusters. Thus, approaches based on the global structure are superior to those focusing on the sparse representation or the local representation. SSRSC gains the best performance on this dataset. Although our SASC considers the global correlation structure, it fails to show better performance, in which the local structure has no positive effect on clustering performance.

Since the twelve baseline methods are state-of-the-art, it is difficult for each algorithm to show the dominant performance on the 15 datasets, including our algorithm. From the experiments, we find that our algorithm outperforms all compared methods on 10 datasets. Especially, its NMI is 13% higher than that of the state-of-the-art algorithm SSRSC on two speech signal datasets. Moreover, on USPSML and COIL20, NMI of our SASC is about 7% higher than that of SSRSC. Although NMI of our SASC is lower than that of the best baseline method on five datasets, its ACC is higher than that of all compared methods on 11 datasets. This indicates the effectiveness of the proposed SASC. In addition, two metrics can more fully reflect the performance of the algorithm.

Finally, Table 5 shows the computational time of all the methods on the twelve datasets. We can observe that although the proposed SASC algorithm is slower than LSR and SSC+E, it is faster than these algorithms based on ℓ_1-norm or nuclear-norm regularization.

4.3.5 Comparison with KNN-based method

Our algorithm constructs an adaptive neighbor graph to capture the local structure. On the other hand, many methods respect the local structure by building a k-nearest neighbor (KNN) graph. Suppose we use the KNN graph instead of the adaptive graph in our algorithm. For simplicity, the KNN-based SASC is called SKNN. In this section, we exploit experiments to verify the differences between the methods based on these two graphs. We only perform the validation on five datasets. Similar observations can be gained on the remaining datasets and are omitted. The experimental results are shown in Table 6. From Table 6, we can see that the proposed algorithm is slightly better than SKNN.

TABLE 6
CLUSTERING RESULTS OBTAINED BY TWO ALGORITHMS

Datasets		COVID19	COIL20	UMIST	Faces96	Isolet1
SKNN	ACC	69.05	81.11	67.30	72.51	68.14
	NMI	44.77	87.11	81.64	87.58	79.29
SASC	ACC	70.33	88.13	69.04	77.02	71.73
	NMI	45.07	92.91	84.85	90.77	80.26

4.3.6 Ablation Experiment

It is easy to see that when $\alpha = 0$ and $\beta = 0$, our SASC degrades to LSR [13]. If $\beta = 0$ and $\gamma = 0$, our algorithm only captures the adaptive local structure to seek the representation; and if $\alpha = 0$ and $\gamma = 0$, our algorithm learns a representation by considering the global structure. Next, we perform a group of experiments to discuss the effect of a single regularization term. We conduct ablation experiments on six datasets, and only give the NMI value of clustering. Similar observations can be obtained on the remaining datasets and are left out. The experimental results are presented in Table 7. As can be seen from Table 7, different regularization terms have different effects. Obviously, the local structure plays a more important role in learning a representation than the other two structures. It has been shown that due to the lack of sample labels, the local structure is more important than the global structure in unsupervised learning. Although the global structure and the global correlation structure are not as important as the local structure, they can also play an active role in learning the representation [13],[46],[47]. Hence, by capturing the adaptive local structure, the global structure, and the global correlation structure, our SASC can express the affinity between samples relatively accurately and reduce the sensitivity to multi-source data and noise interference.

TABLE 7
CLUSTERING RESULTS (NMI%) OBTAINED BY USING A REGULARIZER

Paremeters	$\alpha\neq0,\beta=0,\gamma=0$	$\alpha=0,\beta\neq0,\gamma=0$	$\alpha=0,\beta=0,\gamma\neq0$
COVID19	43.96	22.86	42.28
USPSML	74.61	46.27	68.84
COIL20	77.97	52.33	72.49
UMIST	83.01	60.07	72.35
Faces96	86.66	73.42	84.53
Isolet1	79.52	50.24	77.74

4.3.7 Experiments on noisy datasets

Although most SC methods based on the sparse representation or the low-rank representation claim that they are robust to noise, they ignore the experiments on noisy datasets. Such an experiment is indeed important, and therefore deserves further exploration. To this end, we test the impact of noise on our algorithm and all baseline methods in this section. To generate noisy data, we add salt & pepper noise with a density of 20% to the 15 datasets in our experiments. The results on 15 noisy datasets are respectively reported in Tables 8-9.

From Tables 8-9, we can obtain the following observations:

(1) Compared with the results on the clean datasets, the performance of all the algorithms on noisy datasets decreases to different extents. Obviously, noise interferes with learning an accurate representation, thus reducing the clustering accuracy. Although the performance of the proposed algorithm also decreases, its decline is relatively small in comparison to other baselines. On the Isolet1 and Isolet5 datasets, for example, our algorithm is almost free from noise. The proposed algorithm does not outperform the best baseline on clean datasets, such as TOX_171,

SRBCTML, and so on. However, when these datasets are contaminated with noise, our algorithm outperforms all baselines. Intuitively, this indicates that the proposed algorithm is relatively robust to noise.

(2) The performance of SMR, SOGFS and SSC+E decreases significantly on noisy datasets. Clearly, the three methods using the local structure or entropy norm to learn the representation are sensitive to noise. Although the approaches using the sparse representation or the low-rank representation are inferior to our algorithm on noisy datasets, they outperform the above three methods. Hence, they are robust to noise.

4.3.8 Parameter Selection

In this section, we will evaluate the parameter sensitivity of the proposed SASC algorithm and select the optimal parameters. The algorithm has three adjustable parameters α, β and γ, which represent the importance of the adaptive local structure, the global scatter structure and the global correlation structure, respectively. In general, there are two common strategies to select parameters. One is to select parameters from a predefined grid, which is called grid search. The other is to employ k-fold cross validation to set parameters. Baselines and other SC methods [15],[21,[24] employ grid search to select parameters because it is simpler. Following this line, we also exploit grid search to select three parameters in our algorithm.

TABLE 8
CLUSTERING RESULTS (ACC%) OF THE COMPARED APPROACHES ON 15 NOISY DATASETS

Datasets	SSC	LRR	LSR	LRSC	SMR	EnSC	S^{3}C	SSRSC	S^{3}COMP	SSC+E	l_0-RSSC	SOGFS	SASC
CLL_SUB	49.27	50.46	**54.95**	52.25	**54.95**	51.58	**54.95**	51.35	**54.95**	45.94	**54.95**	45.94	**54.95**
GLI_85	65.41	67.58	67.58	71.76	69.41	66.62	68.23	65.88	66.23	68.23	69.41	60.23	**72.94**
Lung	66.01	50.73	64.53	60.09	63.54	65.27	67.14	68.96	66.01	79.80	62.06	65.02	**82.59**
TOX_171	44.44	40.35	46.19	46.36	43.86	45.08	47.36	45.02	39.25	41.52	43.86	39.18	**47.95**
SRBCTML	39.39	40.96	51.81	46.98	50.60	46.14	46.98	53.01	55.42	43.37	54.21	46.98	**66.88**
Chest	63.31	66.04	74.01	73.67	50.41	68.24	66.67	72.44	72.69	60.22	**74.52**	52.11	73.22
COVID19	46.0	42.27	59.63	59.13	46.16	54.66	60.05	56.27	47.76	35.63	44.36	40.83	**61.89**
USPSML	58.08	63.97	61.75	60.48	58.46	58.46	64.52	62.19	54.5	29.63	54.8	33.17	**74.64**
MFD	67.75	71.75	73.1	77.1	66.2	71.1	73.44	70.62	58.95	32.55	59.15	56.3	**83.9**
COIL20	58.81	40.15	56.63	60.31	55.69	48.75	63.33	57.29	49.79	31.04	58.26	42.63	**82.08**
PalmData25	55.45	62.65	78.1	75.85	70.4	50.91	52.22	**82.64**	51.3	55.05	64.05	62.2	80.7
UMIST	51.52	44.34	53.73	47.65	48.52	53.04	51.82	52.34	48.95	46.61	47.47	52.0	**68.61**
Faces96	47.67	46.25	61.07	56.49	56.66	58.98	60.72	53.34	49.93	48.21	60.74	54.27	**71.68**
Isolet1	57.41	53.75	67.43	65.98	64.16	63.91	50.42	53.25	36.66	35.92	62.94	36.84	**70.01**
Isolet5	51.71	42.52	51.18	55.03	47.08	34.83	43.3	41.72	31.56	31.35	50.28	31.61	**55.35**

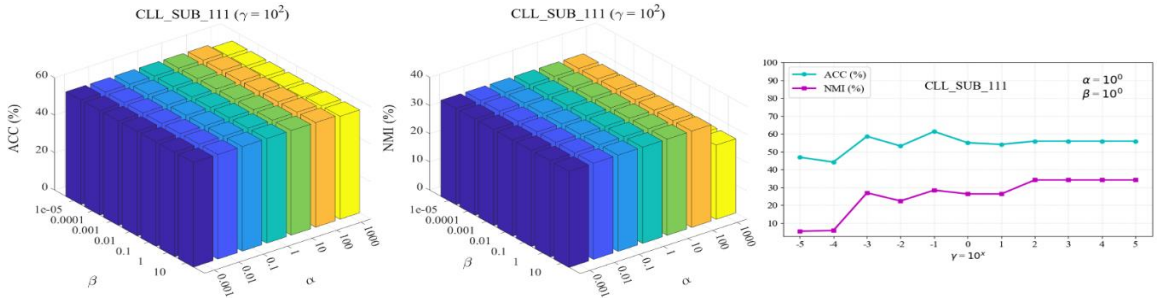
TABLE 9
CLUSTERING RESULTS (NMI%) OF THE COMPARED APPROACHES ON 15 NOISY DATASETS

Datasets	SSC	LRR	LSR	LRSC	SMR	EnSC	S^{3}C	SSRSC	S^{3}COMP	SSC+E	l_0-RSSC	SOGFS	SASC
CLL_SUB	13.71	14.72	21.83	15.06	21.83	15.06	24.49	18.01	24.08	8.02	21.83	13.13	**26.31**
GLI_85	6.79	7.36	7.36	24.93	9.63	7.72	18.40	19.96	14.83	9.15	9.63	6.87	**26.06**
Lung	29.28	27.31	46.87	46.67	46.29	30.23	42.15	47.6	43.02	53.47	42.85	32.99	**64.98**
TOX_171	16.04	14.34	22.04	21.53	22.57	17.19	20.91	21.16	13.45	10.84	20.91	12.46	**25.97**
SRBCTML	12.65	11.29	21.87	18.39	20.7	22.39	26.35	27.62	33.83	17.37	23.14	23.84	**44.12**
Chest	9.05	10.99	19.11	22.44	8.72	14.47	13.59	20.06	15.56	7.65	**23.99**	11.12	21.22
COVID19	23.93	21.58	32.27	31.60	29.43	31.81	27.49	27.79	25.61	12.08	27.42	20.67	**33.55**
USPSML	51.95	55.4	59.2	63.54	52.18	58.66	59.55	59.59	48.27	35.55	49.68	22.9	**73.65**
MFD	59.15	60.47	61.21	64.18	57.98	61.82	65.06	62.77	50.24	25.15	56.26	49.29	**72.66**
COIL20	70.23	57.77	68.6	72.76	68.99	50.99	73.33	70.96	58.11	39.7	71.57	56.78	**86.78**
PalmData25	75.89	83.39	89.59	92.02	87.63	56.02	63.2	**93.87**	73.81	76.19	80.61	82.31	92.77
UMIST	68.42	55.25	70.05	60.9	61.28	68.58	70.21	67.87	54.73	58.72	57.95	65.95	**81.73**
Faces96	70.78	65.53	82.16	66.36	77.28	77.51	81.15	77.25	74.04	73.79	79.35	79.51	**88.48**
Isolet1	71.72	67.43	76.07	73.87	74.81	74.27	66.38	65.07	47.68	40.61	70.57	44.98	**79.91**
Isolet5	68.97	61.51	66.40	67.08	64.87	46.08	59.67	58.47	42.57	31.70	61.72	39.52	**70.31**

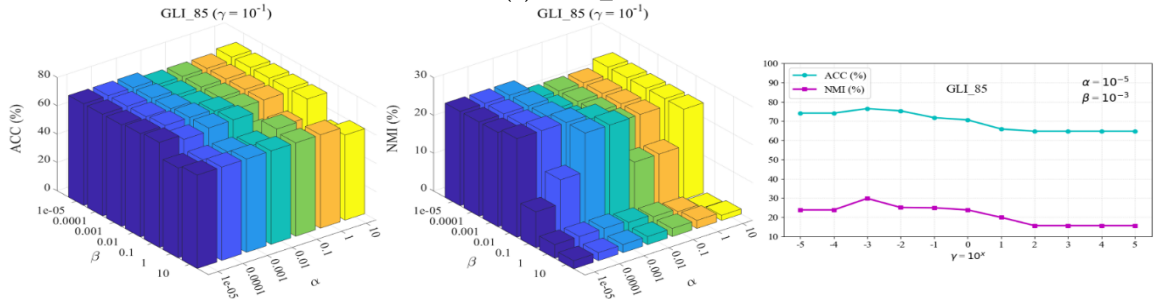
In the experiment, we set the parameter candidate set $\Omega = \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}\}$. Fixing the parameter γ, we execute the SASC on different datasets in the candidate set to explore the influence of the two structures on the clustering results of different data, determining the optimal α and β, and then find the best γ in the candidate set through a consistent strategy. Fig. 3 shows the impact of three parameters on the performance of our algorithm on six bioinformatics datasets. Similar observations can be obtained on the rest of datasets and thus omitted.

From Fig. 3, we can observe that α and β have a relatively large influence on SASC. Our algorithm seems insensitive to different values of γ. When α changes within

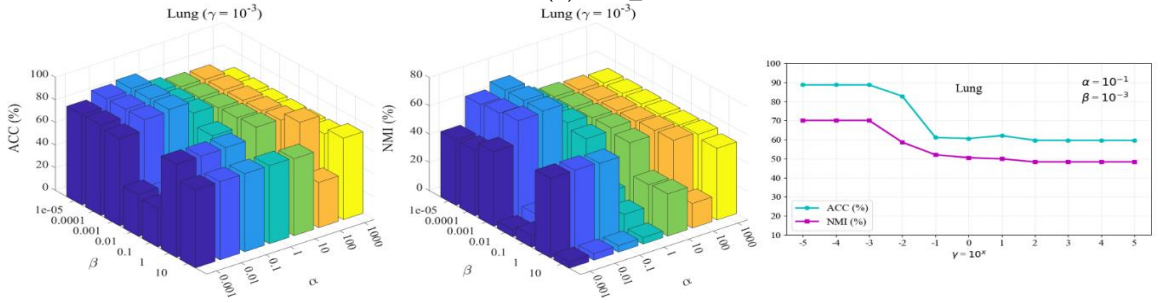
the range of $[10^{-2},10^{2}]$, our SASC achieves good performance. Generally, when α is greater than 10^{2} or less than 10^{-2}, the performance of our algorithm will decline. For instance, on the Lung dataset, SASC performs poorly when α is greater than 10^{2}. However, when it is less than this value, SASC becomes stable and gains better performance. As we can see, SASC achieves consistently good performance when β is changing from 10^{-5} to 10^{-3}. SASC becomes very stable when β is less than 10^{-3}. SASC is relatively stable on the parameter γ. In the experiments, the value of γ is set to the range of $[10^{-2},10^{1}]$.



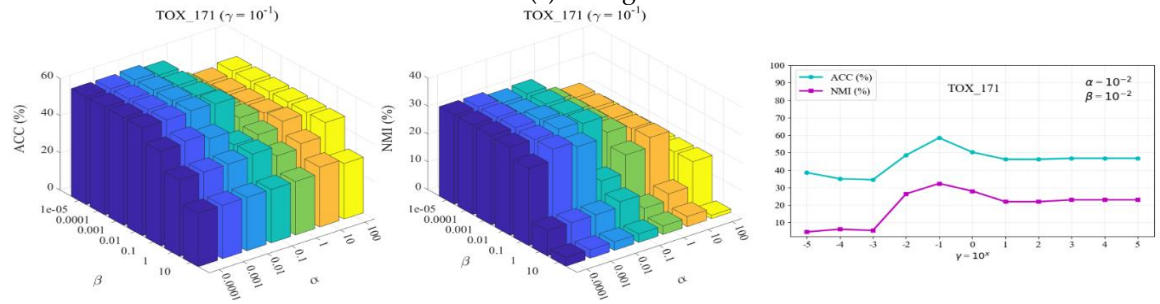
(a) CLL_SUB.



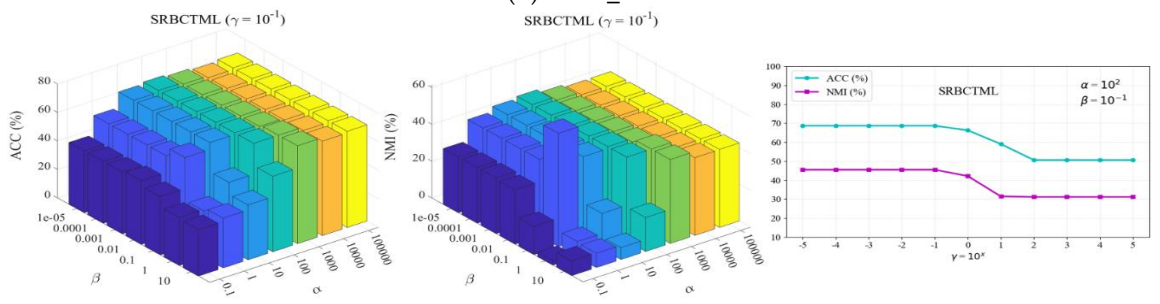
(b) GLI_85.



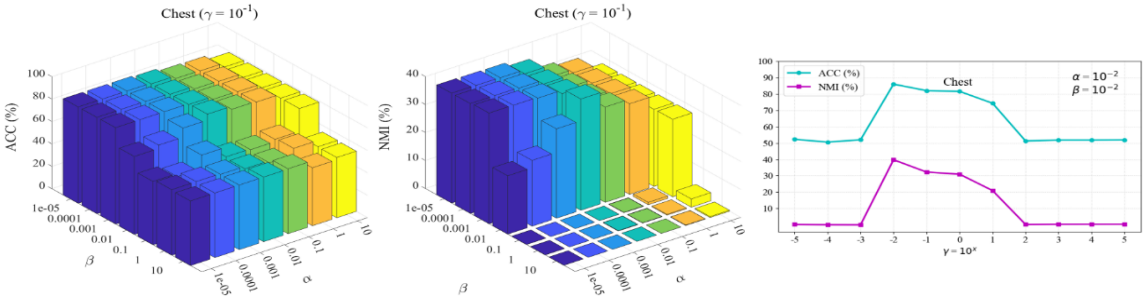
(c) Lung.



(d) TOX_171.



(e) SRBCTML.



(f) Chest.

Fig. 3. The relationship between clustering results and parameters α, β, γ on six bioinformatics datasets. In each of the subfigures (a),(b),(c), (d),(e) and (f), when γ is fixed, the left panel demonstrates the ACC value of our algorithm with different α and β; the middle panel shows NMI of our algorithm with different α and β; and the right panel shows ACC and NMI of our algorithm with different γ when α and β are fixed.

Obviously, if we set inappropriate values for these parameters, the performance of our algorithm will be greatly reduced. Such an issue also exists in other SC methods. Hence, the suitable values of the parameters are critical to SC and feature learning approaches.

5 Conclusions

We propose a novel subspace clustering framework named SASC, which simultaneously captures the local structure of the intra-cluster, the global correlation structure on the inter-cluster, and the global scatter structure. We also present a new grouping-measure approach that combines the grouping effect with various representation structures to perform subspace representation learning comprehensively and accurately. Extensive experiments demonstrate consistently good performance on various data, including bioinformatics, handwritten digits, object images, and speech signals. These results validate the effectiveness of the proposed SASC and its lower sensitivity on different data. In the future, there are two points that can be extended: (1) designing a more convenient way for parameter selection or adaptive parameter learning approach; (2) exploring analyzing methods that can be applied to large-scale data or developing effective data filtering strategies.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of this paper. This work was supported by Public-welfare Technology Application Research of Zhejiang under Grant LGG22F020032, and Key Research and Development Project of Zhejiang Province in China under Grant 2021C03137.

References

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

[2] P. Tseng, "Nearest q-flat to m points," *Journal of Optimization Theory and Applications*, vol. 105, no. 1, pp. 249-252, Apr. 2000.

[3] J. Ho, MH. Yang, J. Lim, KC. Lee and D. Kriegman, "Clustering appearances of objects under varying illumination conditions," in *CVPR*, pp. 11-18, 2003.

[4] T. Zhang, A. Szlam and G. Lerman, "Median k-flats for hybrid linear modeling with many outliers," in *IEEE 12th ICCV Workshops*, pp. 234-241, 2009.

[5] D. Lee and HS. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788-791, Oct. 1999.

[6] R. Vidal, Y. Ma and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1945-1959, Dec. 2005.

[7] P. Ji, M. Salzmann and HD. Li, "Shape Interaction Matrix Revisited and Robustified: Efficient Subspace Clustering with Corrupted and Incomplete Data," in *Proceeding of the ICCV*, pp. 4687-4695, 2015.

[8] A. Leonardis, H. Bischof and J. Maver, "Multiple eigenspaces," *Pattern Recognition*, vol. 35, no. 11, pp. 2613-2627, Nov. 2002.

[9] C. Archambeau, N. Delannay and M. Verleysen, "Mixtures of robust probabilistic principal component analyzers," *Neurocomputing*, vol. 71, no. 7-9, pp. 1274-1282, Mar. 2008.

[10] S. Rao, R. Tron, R. Vidal and Y. Ma, "Motion Segmentation in the Presence of Outlying, Incomplete, or Corrupted Trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 10, pp. 1832-1845, Oct. 2010.

[11] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765-2781, Nov. 2013.

[12] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171-184, Jan. 2013.

[13] C. Lu, H. Min, ZQ. Zhao, L. Zhu, DS. Huang and S. Yan, "Robust and Efficient Subspace Segmentation via Least Squares Regression," in *ECCV*, pp. 347-360, 2012.

[14] R. Vidal and P. Favaro, "Low rank subspace clustering (LRSC)," *Pattern Recognition Letters*, vol.43, pp.47-61, Jul. 2014.

[15] H. Hu, Z. Lin, J. Feng and J. Zhou, "Smooth representation clustering," in *Proceedings of the CVPR*, pp.3834-3841, 2014.

[16] C. You, CG. Li, D. P. Robinson and R. Vidal, "Oracle based active set algorithm for scalable elastic net subspace clustering," in *Proceedings of the CVPR*, pp.3928-3937, 2016.

[17] C. Li, C. You and R. Vidal, "Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework," *IEEE Trans. Image Process.*, vol.26, no. 6, pp. 2988-3001, Jun. 2017.

[18] J. Xu, M. Yu, L. Shao, W. Zuo, D. Meng, L. Zhang and D. Zhang, "Scaled simplex representation for subspace clustering," *IEEE Trans. Cybern.*, vol. 51, no. 3, pp. 1493-1505, Mar. 2021.

[19] Y. Chen, CG. Li and C. You, "Stochastic sparse subspace clustering," in *Proceedings of the CVPR*, pp. 4155-4164, 2020.

[20] L. Bai, and J. Liang, "Sparse subspace clustering with entropy-

norm," in *Proc. 37th Int. Conf. Mach. Learn.*, pp. 561-568, 2020.

[21] M. Brbic and K. Ivica, "ℓ_0-Motivated Low-Rank Sparse Subspace Clustering," *IEEE Trans. Cybern.*, vol. 50, no. 4, pp. 1711-1725, Apr. 2020.

[22] F. Nie, X. Wang, H. Huang, "Clustering and projected clustering with adaptive neighbors," in *Proceedings of Int. Conf. on Knowl. Discovery Data Mining*, pp. 977-986, 2014.

[23] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40-51, Jan. 2007.

[24] F. Nie, W. Zhu and X. Li, "Structured Graph Optimization for Unsupervised Feature Selection," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 3, pp. 1210-1222, Mar. 2021.

[25] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," *in Proceedings of the CVPR*, pp. 1615-1622, 2011.

[26] E. Dyer, C. Studer and RG. Baraniuk, "Subspace clustering with dense representations," in *Int. Conf. Acous. Speech Signal Process.*, 2013, pp. 3258-3262.

[27] L. Chen and G. Guo, "Ordered smooth representation clustering," *Int. J. Mach. Learn. & Cyber.*, vol. 10, no. 11, pp. 3301-3311, Nov. 2019

[28] X. Xiao and L. Wei, "Robust subspace clustering via latent smooth representation clustering," *Neural Processing Letters*, vol. 52, no. 2, pp. 1317-1337, Jul. 2020.

[29] S. Zhang, C. You, R. Vidal and CG. Li, "Learning a Self-Expressive Network for Subspace Clustering," in *Proceedings of the CVPR*, 2021, pp. 12393-12403.

[30] D. Luo, F. Nie, C. Ding and H. Huang, "Multi-subspace representation and discovery," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, pp. 405-420, 2011.

[31] Y. Wang, H. Xu, C. Leng, "Provable Subspace Clustering: When LRR meets SSC," in *Processing of the NeurIPS*, vol. 26, pp. 1-9, 2013.

[32] C. Li and R. Vidal. "A structured sparse plus structured low-rank framework for subspace clustering and completion," *IEEE Trans. on Signal Process.*, vol. 64, no. 24, pp. 6557-6570, Dec. 2016.

[33] I. Selesnick, "Sparse regularization via convex analysis," *IEEE Trans. on Signal Process.*, vol. 65, no. 17, pp. 4481-4494, Sept. 2017.

[34] V. Larsson and C. Olsson, "Convex low rank approximation," *Int. J. Comput. Vis.*, vol. 120, no. 2, pp. 194-214, Apr. 2016.

[35] W. Jiang, J. Liu, H. Qi and Q. Dai, "Robust subspace segmentation via nonconvex low rank representation," *Information Sciences*, vol. 340, pp. 144-158, May. 2016.

[36] X. Zhang, C. Xu, X. Sun and G. Baciu, "Schatten-q regularizer constrained low rank subspace clustering model," *Neurocomputing*, vol. 182, pp. 36-47, Mar. 2016.

[37] Z. Le, M. Arian, W. Haolei, XD. Wang and T. Long, "Does ℓ_p-minimization outperform ℓ_1-minimization?" *IEEE Trans. on Inf. Theory*, vol. 63, no. 11, pp. 6896-6935, Nov. 2017.

[38] S. Wold, K. Esbensen and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1-3, pp. 37-52, Aug. 1987.

[39] R. Bartels and G. Stewart, "Solution of the matrix equation $AX + XB = C$," *Communications of the ACM*, vol. 15, no. 9, pp. 820–826, Sept. 1972.

[40] J. Wen, X. Fang, Y. Xu, C. Tian, L. Fei, "Low-rank representation with adaptive graph regularization," *Neural Networks*, vol. 108, pp. 83–96, Aug. 2018.

[41] Z. Fu, Y. Zhao, D. Chang, X. Zhang, Y. Wang, "Double low-rank representation with projection distance penalty for clustering," *in Proceedings of the CVPR*, 2021, pp. 5320-5329.

[42] L. Wei, F. Zhang, Z. Chen, R. Zhou, C. Zhu, "Subspace clustering via adaptive least square regression with smooth affinities," *Expert Syst. Appl.*, vol. 239, pp. 107950, Jul. 2022.

[43] Y. Qin, H. Wu, J. Zhao, G. Feng, "Enforced block diagonal subspace clustering with closed form solution," *Pattern Recognit.*, vol. 130, pp. 108791, Oct. 2022.

[44] J. Guo, Y. Sun, J. Gao, Y. Hu, "Rank Consistency Induced Multiview Subspace Clustering via Low-Rank Matrix Factorization," *IEEE Trans. Neural. Netw. Learning Syst.*, vol. 33, no. 7, pp. 3157-3170, Jul. 2022.

[45] H. Wang, F. Nie, H. Huang, "Globally and Locally Consistent Unsupervised Projection," *in Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1328-1333.

[46] Y. Liu, Y. Liao, L. Tang, F. Tang, W. Liu, "General subspace constrained non-negative matrix factorization for data representation," *Neurocomputing*, vol. 173, pp. 224–232, Feb. 2016.

Simin Kou is currently a postgraduate in the School of Media and Design, Hangzhou Dianzi University, Hangzhou, China. Her interests include image processing, computer vision, Machine learning and pattern recognition.

Xuesong Yin received the Ph.D. degree in computer science from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2010. He is currently a Professor in the School of Media and Design, Hangzhou Dianzi University, Hangzhou, China. His current research interests include machine learning, data mining, and pattern recognition.

Yigang Wang received his M.S. and Ph.D. degrees in applied mathematics from Zhejiang University, Hangzhou, China. He is currently a Professor in the School of Media and Design, Hangzhou Dianzi University, Hangzhou, China. His interests include image processing, computer vision, pattern recognition and computer graphics.

Songcan Chen received the PhD degree in communication and information systems, in 1997, and then worked at NUAA in January 1986. Since 1998, as a full-time professor, he has been with the College of Computer Science & Technology at NUAA. His research interests include pattern recognition, machine learning and neural computing. He is also an IAPR Fellow.

Tieming Chen (M'01-M'04-F'11) is now a professor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. He is also a member of the ACM. His research interests include pattern recognition, computer vision and intelligence security.

Zizhao Wu received the Ph.D. degree in in the department of Computer science and technology from Zhejiang University, in 2013. Now he is an assistant professor in School of Media and Design, Hangzhou Dianzi University. His research interests include machine learning, computer graphics, and computer vision.