

Article

Robust Subspace Clustering with Block Diagonal Representation for Noisy Image Datasets

Qiang Li ¹, Ziqi Xie ² and Lihong Wang ^{2,*} ¹ School of Economics and Management, Yantai University, Yantai 264005, China² School of Computer and Control Engineering, Yantai University, Yantai 264005, China

* Correspondence: wanglh@ytu.edu.cn

Abstract: As a relatively advanced method, the subspace clustering algorithm by block diagonal representation (BDR) will be competent in performing subspace clustering on a dataset if the dataset is assumed to be noise-free and drawn from the union of independent linear subspaces. Unfortunately, this assumption is far from reality, since the real data are usually corrupted by various noises and the subspaces of data overlap with each other, the performance of linear subspace clustering algorithms, including BDR, degrades on the real complex data. To solve this problem, we design a new objective function based on BDR, in which $l_{2,1}$ norm of the reconstruction error is introduced to model the noises and improve the robustness of the algorithm. After optimizing the objective function, we present the corresponding subspace clustering algorithm to pursue a self-expressive coefficient matrix with a block diagonal structure for a noisy dataset. An affinity matrix is constructed based on the coefficient matrix, and then fed to the spectral clustering algorithm to obtain the final clustering results. Experiments on several artificial noisy image datasets show that the proposed algorithm has robustness and better clustering performance than the compared algorithms.

Keywords: subspace clustering; block diagonal representation; noisy image; outlier; $l_{2,1}$ norm



Citation: Li, Q.; Xie, Z.; Wang, L. Robust Subspace Clustering with Block Diagonal Representation for Noisy Image Datasets. *Electronics* **2023**, *12*, 1249. <https://doi.org/10.3390/electronics12051249>

Academic Editors: Yanwei Yu, Zhu Xiao and Ziqiang Yu

Received: 26 January 2023

Revised: 2 March 2023

Accepted: 2 March 2023

Published: 5 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Clustering [1] is an important analysis tool in the fields of data mining and machine learning, and is widely used in motion segmentation [2–4], text clustering [5], image segmentation [6,7], face recognition [8], and other practical applications [9]. In brief, clustering divides unlabeled data into several clusters according to the similarity of data points. Traditional clustering methods, such as K -means, hierarchical clustering [10], and density clustering [11,12], cannot perform well on high-dimensional data. Therefore, subspace clustering algorithms for high-dimensional data get more attention in many applications. For example, in spatiotemporal data analysis, huge volumes of image and video data are high-dimensional and noisy, subspace clustering can be employed as an effective method of data preprocessing, and the clustering results help to uncover the structure of the dataset.

In the view of spectral graph theory, the key to the success of subspace clustering is to construct a coefficient matrix that can correctly reflect the spatial distribution of the data. Ideally, the coefficient matrix has a block diagonal structure, and each block is a self-representative coefficient sub-matrix for a cluster of data. Sparse subspace clustering (SSC) [13,14] and low rank subspace clustering (LRR) [15] are two representative subspace clustering algorithms used in the last decade. SSC and LRR construct coefficient matrices C by sparse representation and low rank representation of data, respectively, and then obtain the affinity matrix W from the coefficient matrix, i.e., $W = (|C| + |C^T|)/2$, and finally W is fed into spectral clustering to obtain clustering results. In recent years, block diagonal representation (BDR) for subspace clustering algorithms [16,17] attracted attention due to its good clustering performance. BDR introduces Laplacian rank constraint into the

objective function, and obtains a more accurate block diagonal coefficient matrix than SSC and LRR. However, SSC, LRR, and BDR algorithms all belong to linear subspace clustering, the basic assumption of which is the self-expressiveness property. The self-expressiveness property assumes that each data point can be expressed as a linear combination of all the other points in the union of subspaces. Meanwhile, subspace-preserving is also a common assumption for linear subspace clustering, where each data point in a subspace can be linearly expressed by other points in the same subspace [18]. Finally, the data should be noise-free and subspaces should be independent of each other. However, these assumptions are usually far from reality, since in the process of data generation, transmission, storage, and even applications, data may be damaged, resulting in data containing noises, outliers, and even corruptions [19–25]. The corruptions make data deviate from the original data model. The subspaces corrupted by complex noises are beyond the self-expressiveness of independent linear subspaces, consequently, subspace clustering results will deviate from expectations. Thus, the linear subspace clustering algorithms degrade when facing corrupted datasets in reality.

Aiming at above problems, some efforts impose norm regularizations to handle various noises. Studies show that the Frobenius norm (F norm for short), l_1 norm, and $l_{2,1}$ norm can efficiently handle Gaussian noises, sparse noises, and outliers, respectively [19,26]. SSC used F norm and l_1 norm to deal with noise [13]. Favaro et al. [27] learned a new data dictionary by handling data errors through the F norm. LRR used $l_{2,1}$ norm to reduce the influence of data errors [19]. It is easy to find that these efforts achieve good results by assuming the type of errors as a prior and removing errors in the original input space by modeling them in their objective functions [28]. Unfortunately, the data in reality may be corrupted by complex noises [29], and modeling of errors should combine with more robust methods to handle the noises and uncover the subspace structure of the data set.

To solve this problem, Chen et al. [30] proposed a robust low-rank subspace clustering algorithm (RLRR), which adopts a probability density function to fit noise distribution. Liu et al. [31] proposed the latent low-rank representation subspace clustering (LatLRR) by reconstructing the representation dictionary to avoid the influence of noise and correctly represent the distribution of data subspaces. Zhang et al. [32] proposed Robust LatLRR on the basis of LatLRR to impose sparse constraints on the matrix. Similar to LatLRR, references [33,34] also learned a clean dictionary to find the data distribution structure and reduce the influence of data noise. Nie et al. [35] proposed a novel low-rank structural model for segmenting high-dimensional data. A new rank constraint is defined to learn a subspace indicator, which can capture different clusters directly from the data. These methods based on low-rank representation show good performance in dealing with the linear structure of data. Unfortunately, in practice, many datasets have nonlinear subspaces that overlap with each other. To solve this problem, the multiple kernel learning (MKL) is applied to solve nonlinear structural data modeling [36]. Moreover, low-rank kernel space clustering methods are studied to deal with the nonlinear structure of high-dimensional data [37]. These methods adaptively select or combine kernels suitable for the datasets from a set of kernels. However, if the dataset is corrupted by complex noises, maybe the selected kernel is not optimal.

Furthermore, there are efforts dealing with noises in various ways for subspace clustering. Qin et al. [28] proposed a method from an energy perspective to eliminate errors in the projected space rather than the input space. They defined an energy function to measure a block in the projected space, and found the correct block with the maximal energy to lead clustering. He et al. [38] proposed the subspace clustering algorithm via half-quadratic (SCHQ) to handle noisy data. SCHQ consisted of two parts, the first one adopted l_1 norm to obtain a sparse representation of data, and the second one maximized the correlation of low-dimensional representations among a given data point and other data points to reduce the damage of noisy data to coefficient representation. Wang et al. [39] proposed a robust block diagonal representation (RBDR) method, which used a penalty matrix to adaptively weigh the reconstruction error to handle noises without prior knowledge, and thus had

good clustering performance under noise conditions. The principles of these efforts are different from our study, and the performance of subspace clustering of some work (e.g., RBDR) will be compared in the following experiments.

In this study, we focus on the robust subspace clustering algorithms based on the Laplacian rank constraint, i.e., the block diagonal property. BDR assumes that the data contain Gaussian noises and fits the reconstruction errors with F norm. In fact, other types of noises, such as occlusions and outliers, may exist in the data. In this paper, $l_{2,1}$ norm is used to fit the possible outliers in the data, and is combined with the Laplacian rank constraint to improve the performance of the subspace clustering on more complex noisy datasets.

The main contributions of this paper are as follows:

- (1) The robust subspace clustering algorithm with block diagonal representation (OBDR) is proposed to handle noises, in which the noises are modeled using $l_{2,1}$ norm, and the Laplacian rank constraint is adopted to pursue a block diagonal structure of the subspace representation;
- (2) The objective function of the proposed algorithm is designed and a corresponding optimization process based on ADMM is given. The algorithm is presented and the time complexity is analyzed;
- (3) Experiments on artificial noisy digital dataset MNIST and face datasets (ORL and YaleB) show that OBDR is insensitive to noises.

Notation. We summarize the symbols and norms used in this paper, as shown in Table 1. We define the vector as italic lowercase letters, e.g., c , and the matrix as italic capital letters, e.g., C .

Table 1. The notations used in each algorithm.

Notations	Descriptions
$C_{i,:}$	the i -th row of C
$C_{:,j}$	the j -th column of C
C_{ij}	the (i,j) -th element of C
C^T	the transposed matrix of C
$Tr(C)$	the trace of the matrix C
$\text{diag}(C)$	a vector with its i -th element being the i -th diagonal element of C
$\text{Diag}(c)$	a diagonal matrix with its i -th diagonal element being i -th element of vector c
$\ C\ _F = \sqrt{\sum_{ij} C_{ij}^2}$	the Frobenius norm(or l_2 norm) of C
$\ C\ _{2,1} = \sum_i \ C_{i,:}\ _F$	the $l_{2,1}$ norm of C
$\ C\ _1 = \sum_{ij} C_{ij} $	the l_1 norm of C
$\ C\ _\infty = \max C_{ij} $	the l_∞ norm of C

The rest of this paper is organized as follows: In Section 2, we propose the objective function of the robust subspace clustering with block diagonal representation (OBDR) and illustrate the framework of the OBDR. In Section 3, we solve the optimization of the objective function by ADMM and present the algorithm in details. The experimental results are introduced in Section 4, and finally Section 5 summarizes the paper.

2. The Proposed Method

BDR adds Laplacian rank constraint, namely K -block diagonal regularizer, to its objective function to control the number of connected branches of the graph, which is constructed by using the matrix W as the affinity matrix. It is proved theoretically and

experimentally that a K -block diagonal regularizer can correctly represent the similarity of data and the coefficient matrix shows a block diagonal structure, which makes the BDR algorithm have good clustering performance [17].

For any matrix B , the K -block diagonal regularizer is defined as the sum of the k smallest eigenvalues of the corresponding Laplacian matrix of B (i.e., $L_B = \text{Diag}(B1) - B$),

$$\|B\|_K = \sum_{i=n-k+1}^n \lambda_i(L_B). \quad (1)$$

Lu [17] defined the BDR model as following:

$$\begin{aligned} \min_{Z,B} & \frac{1}{2} \|X - XZ\|_F^2 + \frac{\lambda}{2} \|Z - B\|_F^2 + \gamma \|B\|_K, \\ \text{s.t. } & \text{diag}(B) = 0, B \geq 0, B = B^T. \end{aligned} \quad (2)$$

On the basis of BDR, we remodel the reconstruction error E of data X , which may contain outliers, and propose the objective function of robust subspace clustering with block diagonal representation (OBDR) as follows:

$$\begin{aligned} \min_{Z,B,E} & \|E\|_\ell + \frac{\lambda}{2} \|Z - B\|_F^2 + \gamma \|B\|_K, \\ \text{s.t. } & \text{diag}(B) = 0, B \geq 0, B = B^T, E = X - XZ. \end{aligned} \quad (3)$$

$\|\cdot\|_\ell$ is an alternative regularization method for handling different types of noise. In general, F norm is selected to deal with Gaussian noises and slight sparse noises in the data, meanwhile, sparse noises can be better processed by $\|\cdot\|_1$. In the case of corruptions or outliers, $\|\cdot\|_{2,1}$ is a better choice. For different noises, appropriate regularization methods can effectively reduce the sensitivity of the algorithm to noisy data and improve the robustness of the algorithm.

Aiming at handling outliers, we select $\|\cdot\|_{2,1}$ to replace $\|\cdot\|_\ell$ in Equation (3),

$$\begin{aligned} \min_{Z,B,E} & \|E\|_{2,1} + \frac{\lambda}{2} \|Z - B\|_F^2 + \gamma \|B\|_K, \\ \text{s.t. } & \text{diag}(B) = 0, B \geq 0, B = B^T, E = X - XZ. \end{aligned} \quad (4)$$

Next, we use the augmented Lagrange multiplier method ALM [40,41] to rewrite Equation (4) and define the objective function $F(Z, B, E)$ as follows:

$$\begin{aligned} \min_{Z,B,E} & F(Z, B, E) = \frac{\mu}{2} \|X - XZ - E\|_F^2 + \|E\|_{2,1} + \frac{\lambda}{2} \|Z - B\|_F^2 + \gamma \|B\|_K, \\ \text{s.t. } & \text{diag}(B) = 0, B \geq 0, B = B^T. \end{aligned} \quad (5)$$

where μ and Λ are the penalty parameter and Lagrange multiplier, respectively. Equation (5) shows that, OBDR not only uses $l_{2,1}$ norm to handle outliers, but also uses F norm to deal with Gaussian noise and slight sparse noise.

Following [42,43], we summarize the framework of the OBDR in Figure 1. The objective function of OBDR retains the self-expressiveness and subspace preserving property in the first term of $F(Z, B, E)$, and the Laplacian rank constraint in the last term. Moreover, the noises are separated from the input data X in the first term of $F(Z, B, E)$ and modeled by the $l_{2,1}$ norm in the second term. The optimized self-expressive matrix Z is used to calculate the affinity matrix $W = (|Z| + |Z^T|)/2$, and then, W is fed to spectral clustering to get the clustering result.

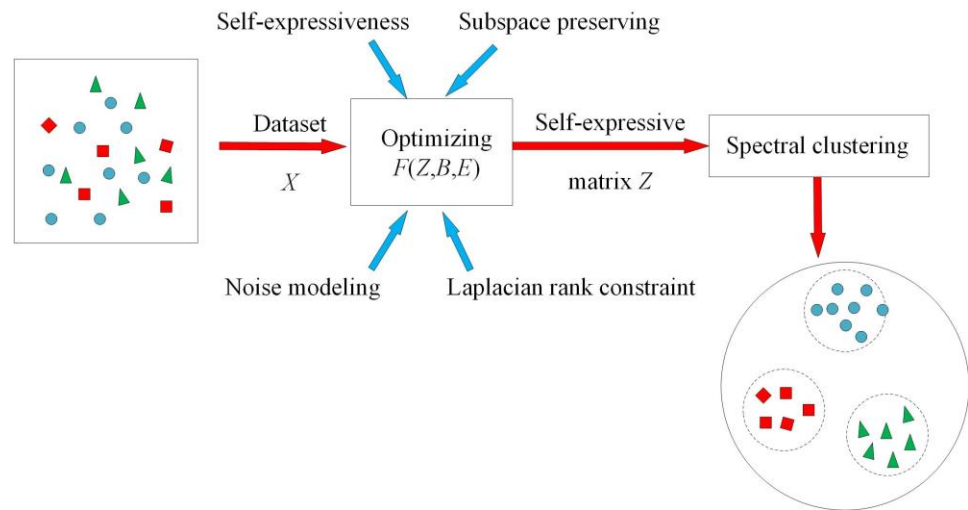


Figure 1. The framework of OBDR.

3. Optimization of the Objective Function

Firstly, $\|B\|_K$ is rewritten as a convex optimization problem [17],

$$\begin{aligned} \|B\|_K &= \min_V \langle L_B, V \rangle, \\ \text{s.t. } 0 &\preceq V \preceq I, \text{tr}(V) = K. \end{aligned} \quad (6)$$

Then, Equation (5) is equivalent to

$$\begin{aligned} \min J(Z, B, E) &= \frac{\mu}{2} \|X - XZ - E\|_F^2 + \frac{\Lambda}{\mu} \|E\|_{2,1} + \frac{\lambda}{2} \|Z - B\|_F^2 + \gamma \langle \text{Diag}(B1) - B, V \rangle \\ \text{s.t. } \text{diag}(B) &= 0, B \geq 0, B = B^T, 0 \preceq V \preceq I, \text{tr}(V) = K. \end{aligned} \quad (7)$$

Finally, the ADMM method is adopted to solve the variables E , Z , B , and V . The optimization consists of four sub-problems as follows:

- (1) Update E with fixed Z , B , and V
 E is updated as:

$$E^{t+1} = \underset{E}{\text{argmin}} \frac{\mu}{2} \|X - XZ - E\|_F^2 + \frac{\Lambda}{\mu} \|E\|_{2,1}. \quad (8)$$

To solve Equation (8), the following theorem is introduced.

Theorem 1 ([44]). Given a matrix $A = [a_1, a_2, \dots, a_i, \dots]$, solve the following problem:

$$\min_W \sigma \|W\|_{2,1} + \frac{1}{2} \|W - A\|_F^2 \quad (9)$$

then, the optimal solution W^* of Equation (9) can be solved column by column, and the i -th column of W^* as follows:

$$W^*(:, i) = \begin{cases} \frac{\|a_i\| - \sigma}{\|a_i\|} a_i, & \text{if } \sigma < \|a_i\|, \\ 0, & \text{otherwise.} \end{cases} \quad (10)$$

According to Theorem 1, we rewrite Equation (8) as:

$$E^{t+1} = \underset{E}{\text{argmin}} \frac{1}{2} \|E - (X - XZ + \frac{\Lambda}{\mu})\|_F^2 + \frac{1}{\mu} \|E\|_{2,1}. \quad (11)$$

Let $A = X - XZ + \frac{\Lambda}{\mu}$, and $A = [a_1, a_2, \dots, a_i, \dots]$, the optimal solution E^* is given:

$$E^*(:, i) = \begin{cases} \frac{\|a_i\|^{-\frac{1}{\mu}}}{\|a_i\|} a_i, & \text{if } \frac{1}{\mu} < \|a_i\|, \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

(2) Update Z with fixed E , B , and V
 Z is updated as:

$$Z^{t+1} = \underset{Z}{\operatorname{argmin}} \frac{\mu}{2} \|X - XZ - E + \frac{\Lambda}{\mu}\|_F^2 + \frac{\lambda}{2} \|Z - B\|_F^2. \quad (13)$$

Since $\|A\|_F^2 = \operatorname{Tr}(A^T A)$, we define the function f and transform it as follows:

$$\begin{aligned} f(Z) &= \frac{\mu}{2} \|X - XZ - E + \frac{\Lambda}{\mu}\|_F^2 + \frac{\lambda}{2} \|Z - B\|_F^2 \\ &= \frac{\mu}{2} \operatorname{Tr}((X - XZ - E + \frac{\Lambda}{\mu})^T (X - XZ - E + \frac{\Lambda}{\mu})) \\ &\quad + \frac{\lambda}{2} \operatorname{Tr}((Z - B)^T (Z - B)). \end{aligned} \quad (14)$$

Take the derivative of Equation (14), we obtain:

$$\frac{\partial f}{\partial Z} = \mu \left(X^T XZ - X^T X + X^T E - X^T \frac{\Lambda}{\mu} \right) + \lambda (Z - B). \quad (15)$$

Set Equation (15) to 0, then

$$(\mu X^T X + \lambda I)Z = \mu X^T X - \mu X^T E + X^T \Lambda + \lambda B.$$

We derive Z ,

$$Z = (\mu X^T X + \lambda I)^{-1} (\mu X^T X - \mu X^T E + X^T \Lambda + \lambda B). \quad (16)$$

(3) Update B with fixed E , Z , and V

B is updated as:

$$\begin{aligned} B^{t+1} &= \underset{B}{\operatorname{argmin}} \frac{\lambda}{2} \|Z - B\|_F^2 + \gamma \langle \operatorname{Diag}(B1) - B, V \rangle \\ \text{s.t. } \operatorname{diag}(B) &= 0, B \geq 0, B = B^T. \end{aligned}$$

According to the reference [17], the closed-form solution of B is given as:

$$B^{t+1} = \left[(\hat{A} + \hat{A}^T) / 2 \right]_+ \quad (17)$$

where $[A]_+ = \max(A, 0)$, $A = Z - \frac{\gamma}{\lambda} (\operatorname{diag}(V)1^T - V)$, $\hat{A} = A - \operatorname{Diag}(\operatorname{diag}(A))$.

(4) Update V with fixed E , B , and Z

V is updated as:

$$V^{t+1} = \underset{V}{\operatorname{argmin}} \langle \operatorname{Diag}(B1) - B, V \rangle \text{ s.t. } 0 \preceq V \preceq I, \operatorname{tr}(V) = K. \quad (18)$$

Equation (18) can be solved as follows [17]:

$$V^{t+1} = UU^T \quad (19)$$

where $U \in R^{n \times K}$ consists of the K eigenvectors associated with the first K smallest eigenvalues of $\operatorname{Diag}(B1) - B$.

The process of OBDR is summarized as Algorithm 1.

Algorithm 1 OBDR**Input:** X, λ, γ, ρ **Initialization:** $V = 0, B = 0, Z = 0, E = 0, \Lambda = 0, \mu = 10^{-3}, \mu_{\max} = 10^8, \varepsilon = 10^{-6}, t = 0, \text{Maxloop} = 500$.1. **WHILE** $t < \text{Maxloop}$ **DO**2. Update E by Equation (12);3. Update Z by Equation (16);4. Update B by Equation (17);5. Update V by Equation (19);6. Update $\Lambda, \Lambda = \Lambda + \mu(X - XZ - E)$;7. Update $\mu, \mu = \min(\rho\mu, \mu_{\max})$;8. if $\text{Max}(\|Z^{t+1} - Z\|_{\infty}, \|B^{t+1} - B\|_{\infty}, \|X - XZ - E\|_{\infty}) < \varepsilon$, break;9. $t = t + 1$;10. **END**.**Output:** Z, B

μ is the parameter for updating the Lagrange multiplier Λ , and its initial value is usually within $(10^{-8}, 10^{-3})$. Parameter ρ directly affects the update times of the algorithm. Generally, the larger ρ is, the higher the efficiency of the algorithm will be. However, large ρ will cause the variables to change greatly and miss the optimal solution; on the contrary, lowering ρ can bring high variable accuracies and good results but also takes a lot of time [45]. Considering both the updating efficiency and clustering results, we set the maximum number of iterations (Maxloop) in Algorithm 1 to 500 and set ρ according to the datasets to ensure an accurate coefficient matrix Z . Then the affinity matrix $W = (|Z| + |Z^T|)/2$ is calculated and fed into spectral clustering to get the clustering result.

The computation complexity of OBDR includes two parts, i.e., the process of updating E, Z, B , and V , and spectral clustering. Because the time complexity of eigenvector decomposition and spectral clustering is $O(n^3)$, and the time complexity of updating variables is $O(Tn^3)$, where T is iterative times, the total time complexity of OBDR is $O(Tn^3)$.

4. Experiments and Discussions

4.1. Experimental Datasets

In order to evaluate the robustness of the proposed algorithm, we test four artificial data sets under noises, including outliers, masking noises, Gaussian noises, and a mixture of various noises. We restrict our discussions and experiments to the following corruption processes:

- (1) Outliers: For a given data set, outliers are data that are beyond all subspaces of this data, i.e., outliers come from different data models, rather than a simple situation that is floating between subspaces [19], so we select small samples from different data sets to simulate outliers for a given data set;
- (2) Masking noises: A fraction of an image is masked by setting the elements of the masked position to 0 [46];
- (3) Additive Gaussian noises: $\tilde{x}|x \sim N(x, \sigma^2 I)$ for an image x [46].

The details of noisy data sets are described as follows:

- (1) Dataset1 for outliers (D_1)

We conduct experiments on the MNIST dataset for $k = \{2, 4, 6\}$ digits, respectively. For each k , we randomly select 100 images for each digit, down sample each image to 28×28 pixels and vectorize it as a vector of length 784. Therefore, the size of Dataset1 is $784 \times 100k$, for $k = \{2, 4, 6\}$. After that, we randomly replace a fraction $p = \{1\%, 3\%, 5\%, 10\%\}$ of Dataset1 with images from ORL, which were cropped to the same size as digits, i.e., 784-length vectors. Finally, we normalize each column vector to have a unit length. Two clusters in Dataset1 are shown in Figure 2. Dataset1 is denoted as D_1 for simplicity, $D_2 \sim D_4$ are defined similarly.



Figure 2. MNIST confused with a small number of ORL images.

(2) Dataset2 for masking noise (D_2)

On the ORL dataset, we conduct clustering experiments for $k = \{10, 20, 30\}$ people. For each k , we randomly pick k people from ORL, each of which has 10 images and the images are down sampled to 32×32 pixels. Then, we perform masking operations for each people. We randomly corrupt 4 out of 10 photos with mask size 5×5 at random positions, then a test dataset with 40% corrupted images is produced. Repeat the masking operations with mask sizes 8×8 and 10×10 to have two more test datasets. Furthermore, we test the noise-free dataset as a benchmark. Finally, we vectorize images and normalize each vector to have a unit length. Some images are shown in Figure 3.

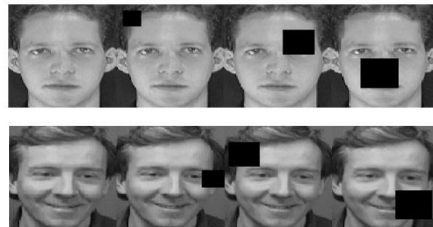


Figure 3. Masking photos from ORL.

(3) Dataset3 for Gaussian noise (D_3)

The extended YaleB dataset contains 38 people, each of which contains 64 frontal face images under different Lambertian conditions. We conduct clustering experiments on the YaleB for $k = \{3, 5, 8\}$ people. For each k , we randomly pick k people, each of which has 64 images. The images are down sampled to 32×32 pixels and vectorized as a vector of length 1024. Next, we randomly add Gaussian noise $N(0.1, 0.01)$ to a fraction $p = \{10\%, 20\%, 30\%\}$ of vectors. Similarly, we test the noise-free dataset as a benchmark. Some images from YaleB as shown in Figure 4.



Figure 4. YaleB dataset.

(4) Dataset4 for mixture noise (D_4)

In practice, a data set may be corrupted by various noises simultaneously and the performance of data clustering is seriously damaged [26,47]. On the basis of Dataset1~Dataset3, we add three kinds of data noises with a ratio of 1: 1: 1 to the ORL. Specifically, for each people, we randomly pick one image to add masking noise with mask size 10×10 and one image to add Gaussian noise $N(0.1, 0.01)$, respectively, and then replace one image with a cropped MNIST image with 32×32 pixels. So, there are three corrupted images for each people. On the Dataset4, we also conduct clustering experiments for $k = \{10, 20, 30\}$ people.

4.2. Experimental Setup

ACC [17] and NMI [48] were used to evaluate the clustering performance of algorithms under noises.

Let $U = [u_1, u_2, \dots, u_n]$, $V = [v_1, v_2, \dots, v_n]$ represent the real labels and the clustering labels of all data, respectively. ACC is defined as follows:

$$\text{ACC}(U, V) = \frac{1}{n} \sum_{i=1}^n \text{ind}(u_i, g(v_i)) \quad (20)$$

where u_i and v_i denote the real label and the clustering label of the i -th data point, respectively, $\text{ind}(x, y) = 1$ if $x = y$, and 0 otherwise. The function $g(\cdot)$ maximizes the ACC by permuting the labels V to match the labels U .

NMI is defined as follows:

$$\text{NMI}(U, V) = \frac{\sum_{i=1}^R \sum_{j=1}^T P(p_i \cap q_j) \log \frac{P(p_i \cap q_j)}{P(p_i)P(q_j)}}{-[\sum_{i=1}^R P(p_i) \log P(p_i) + \sum_{j=1}^T P(q_j) \log P(q_j)]/2}. \quad (21)$$

$P(c) = |c|/n$ denotes the probability that the data points belong to the cluster c , where $|c|$ is the cardinality of c , p_i is the set of data points with the real label u_i , while q_j is the set with the clustering label v_j . R and T represent the number of real labels and output labels of data, respectively.

The value ranges of ACC and NMI are $[0, 1]$, and the higher the ACC and NMI, the better the performance.

We picked five algorithms for comparison, and the algorithms are described as follows:

SSC [13]: The sparse subspace clustering algorithm, which uses sparse representation among data to obtain clustering results;

LRR [19]: The low rank subspace clustering algorithm, which uses low rank representation among data to obtain clustering results;

BDR [17]: Subspace clustering by block diagonal representation, the coefficient matrix obtained by BDR presents a block diagonal structure;

SBDR [16]: Structured block diagonal representation for subspace clustering. On the basis of BDR, SBDR added additional a subspace structure constraint;

RBDR [39]: Block diagonal representation learning for robust subspace clustering for handling data noise.

All the codes used in the experiment were released by the authors. For the parameter(s) in each algorithm, we searched the optimal values within the range of $[10^{-5}, 10^5]$ to ensure the good performance on most test sets. The parameters are shown in Table 2.

Table 2. The parameter values used in each algorithm.

Datasets	SSC	LRR	BDR	SBDR	RBDR	OBDR
D ₁	$\alpha = 20$	$\lambda = 0.45$	$\lambda = 70$ $\gamma = 0.1$	$\lambda = 100$ $\gamma = 10$ $\delta = 1$	$\lambda = 1 \times 10^{-4}$ $\gamma = 1 \times 10^{-4}$	$\lambda = 70$ $\gamma = 0.1$ $\rho = 2$
D ₂ , D ₄	$\alpha = 100$	$\lambda = 1.45$	$\lambda = 70$ $\gamma = 0.1$	$\lambda = 100$ $\gamma = 0.1$ $\delta = 1 \times 10^{-5}$	$\lambda = 1 \times 10^{-3}$ $\gamma = 1 \times 10^{-4}$	$\lambda = 70$ $\gamma = 0.1$ $\rho = 1.05$
D ₃	$\alpha = 20$	$\lambda = 0.85$	$\lambda = 10$ $\gamma = 0.1$	$\lambda = 70$ $\gamma = 0.1$ $\delta = 0.1$	$\lambda = 1 \times 10^{-4}$ $\gamma = 1 \times 10^{-5}$	$\lambda = 10$ $\gamma = 1 \times 10^{-3}$ $\rho = 1.45$

All experiments were run on a computer equipped with an Intel Core I3-10100 processor, 8 G memory, and Windows10 operating system.

4.3. Experimental Results

We run 10 trials on each test set, and the averaged ACC and NMI are shown in Tables 3 and 4, respectively, the best ACC and NMI on each test set are shown in bold.

Table 3. ACCs of algorithms on Dataset1~Dataset4.

Datasets	#Subjects	Noise Level	SSC	LRR	BDR	SBDP	RBDR	OBDR
D ₁	2	Clean	0.8685	0.8955	0.9230	0.9245 ¹	0.8425	0.8965
		1%	0.6355	0.8955	0.6715	0.8465	0.9230	0.8910
		3%	0.5566	0.7085	0.5110	0.5115	0.6945	0.8805
		5%	0.5090	0.5105	0.5090	0.5120	0.6945	0.8645
		10%	0.5735	0.5185	0.5720	0.5180	0.6870	0.6975
	4	Clean	0.6420	0.7368	0.7712	0.7548	0.6202	0.7560
		1%	0.6515	0.7232	0.6222	0.6352	0.6020	0.7183
		3%	0.5735	0.6038	0.6153	0.6295	0.5453	0.6358
		5%	0.5623	0.5968	0.6095	0.6213	0.4370	0.5930
		10%	0.3640	0.5777	0.4040	0.5590	0.2892	0.5767
	6	Clean	0.5712	0.6235	0.6077	0.6437	0.6100	0.6423
		1%	0.5495	0.5528	0.5828	0.5460	0.5828	0.6073
		3%	0.5113	0.5465	0.5758	0.5570	0.5758	0.5682
		5%	0.5183	0.5368	0.5452	0.6042	0.3832	0.5607
		10%	0.3023	0.4640	0.2582	0.4525	0.1840	0.5398
D ₂	10	Clean	0.8000	0.8760	0.8820	0.8820	0.5470	0.8830
		5 × 5	0.7440	0.8550	0.8590	0.8640	0.7990	0.8700
		8 × 8	0.6440	0.7800	0.7460	0.7690	0.6150	0.7830
		10 × 10	0.6480	0.7490	0.6120	0.6030	0.4980	0.6850
	20	Clean	0.7650	0.7940	0.8135	0.8155	0.3975	0.8350
		5 × 5	0.6660	0.7680	0.7835	0.7950	0.7070	0.8220
		8 × 8	0.6150	0.6910	0.6375	0.6560	0.5090	0.6905
		10 × 10	0.5935	0.6020	0.5085	0.5130	0.4205	0.5625
	30	Clean	0.7307	0.7903	0.8143	0.8370	0.2423	0.8543
		5 × 5	0.6370	0.7627	0.7913	0.7970	0.6747	0.8317
		8 × 8	0.5840	0.6530	0.6233	0.6670	0.4827	0.6923
		10 × 10	0.5850	0.6107	0.5047	0.5363	0.4120	0.5857
D ₃	3	Clean	0.6495	0.9638	0.9664	0.9303	0.3327	0.9824
		10%	0.6489	0.4898	0.7495	0.8059	0.3221	0.8729
		20%	0.6101	0.4394	0.7330	0.7479	0.2466	0.7947
		30%	0.5681	0.3351	0.6628	0.6016	0.1889	0.6686
	5	Clean	0.5962	0.7163	0.7987	0.7941	0.4210	0.8687
		10%	0.5300	0.6069	0.5966	0.7050	0.3210	0.7622
		20%	0.4206	0.4287	0.5572	0.5841	0.2567	0.5962
		30%	0.3859	0.3087	0.4563	0.5222	0.2212	0.5269
	8	Clean	0.5115	0.7094	0.8687	0.9127	0.2882	0.8468
		10%	0.5626	0.6160	0.6409	0.7411	0.2511	0.7568
		20%	0.4495	0.5039	0.6149	0.7000	0.1889	0.7080
		30%	0.4102	0.4127	0.5564	0.6221	0.1876	0.6268
D ₄	10		0.5880	0.2621	0.6580	0.6511	0.5315	0.6905
	20		0.6003	0.3285	0.5796	0.6121	0.4762	0.6188
	30		0.5682	0.3562	0.5802	0.6000	0.4935	0.6062

¹ The best ACC on each test set is shown in bold.

Table 4. NMIs of algorithms on Dataset1~Dataset4.

Datasets	#Subjects	Noise Level	SSC	LRR	BDR	SBDR	RBDR	OBDR
D ₁	2	Clean	0.4991	0.6078	0.6707	0.7173 ²	0.5649	0.6242
		1%	0.2302	0.5919	0.2876	0.5840	0.5528	0.6063
		3%	0.0858	0.3252	0.0066	0.0066	0.3298	0.5594
		5%	0.0047	0.0053	0.0047	0.0047	0.3259	0.4863
		10%	0.0649	0.0051	0.0044	0.0048	0.2860	0.2366
	4	Clean	0.4154	0.5842	0.6133	0.6821	0.4849	0.5507
		1%	0.5702	0.5688	0.5534	0.6065	0.4822	0.5292
		3%	0.5128	0.4777	0.5283	0.5803	0.3456	0.4590
		5%	0.4768	0.4641	0.5100	0.5631	0.1989	0.4442
		10%	0.0947	0.4222	0.4024	0.4744	0.0503	0.4182
	6	Clean	0.4255	0.5589	0.5623	0.6533	0.5623	0.5774
		1%	0.5468	0.5047	0.5325	0.6008	0.5325	0.5258
		3%	0.4859	0.4876	0.5037	0.5662	0.5037	0.4841
		5%	0.4844	0.4742	0.4644	0.5883	0.2605	0.4658
		10%	0.1254	0.3832	0.1475	0.4651	0.0221	0.4379
D ₂	10	Clean	0.8662	0.8976	0.8952	0.9047	0.6324	0.9176
		5 × 5	0.7726	0.8839	0.8793	0.8812	0.8229	0.8958
		8 × 8	0.6804	0.7917	0.7819	0.7981	0.6242	0.8008
		10 × 10	0.6564	0.7493	0.6346	0.6259	0.5225	0.6957
	20	Clean	0.8571	0.8732	0.8821	0.8941	0.5541	0.9009
		5 × 5	0.7643	0.8439	0.8635	0.8761	0.7861	0.8840
		8 × 8	0.7135	0.7644	0.7282	0.7479	0.6246	0.7646
		10 × 10	0.6943	0.6954	0.6264	0.6342	0.5572	0.6548
	30	Clean	0.8561	0.8829	0.9023	0.9144	0.4277	0.9221
		5 × 5	0.7745	0.8543	0.8813	0.8895	0.7913	0.8989
		8 × 8	0.7281	0.7602	0.7459	0.7766	0.6387	0.7900
		10 × 10	0.7228	0.7295	0.6626	0.6805	0.5974	0.7046
D ₃	3	Clean	0.4188	0.8712	0.8130	0.4870	0.2130	0.8851
		10%	0.4008	0.2657	0.5454	0.3702	0.2454	0.6236
		20%	0.3573	0.1741	0.4949	0.3082	0.1800	0.5637
		30%	0.3126	0.0200	0.4427	0.3215	0.1042	0.3843
	5	Clean	0.4224	0.7805	0.5714	0.6398	0.3714	0.8190
		10%	0.4703	0.5853	0.4749	0.4872	0.2749	0.5723
		20%	0.3817	0.3917	0.4856	0.4606	0.1856	0.4908
		30%	0.3463	0.2234	0.4470	0.4507	0.1470	0.4109
	8	Clean	0.4420	0.6417	0.5438	0.7066	0.2545	0.5757
		10%	0.5266	0.5351	0.4895	0.5647	0.2372	0.5979
		20%	0.4449	0.4240	0.4523	0.5982	0.1675	0.4959
		30%	0.3917	0.3253	0.4420	0.4658	0.1420	0.5457
D ₄	10		0.6362	0.3120	0.6299	0.6413	0.5500	0.6360
	20		0.6810	0.5065	0.6500	0.6488	0.5121	0.6541
	30		0.7030	0.5542	0.6741	0.6760	0.5935	0.6912

² The best NMI on each test set is shown in bold.

We observe that for each pair of algorithm and dataset, both the ACC and NMI decrease generally with the increasing of the noise levels, which verifies that the noise will hurt the performance of each algorithm. For the dataset D₁, in terms of ACC and NMI, no algorithm achieved the best performance in all test cases. The main reason for these phenomena is that D₁ is a dataset with outliers, which are data from different data models, rather than a simple situation that is floating between subspaces [19]. All the algorithms try to express the outliers in the subspaces of the main body of the data, but the randomness of the outliers and the distinct differences between them make the expression more uncertain, and then the performance of some algorithms decreases abruptly or even increases with

the increasing of the noise levels. For example, the ACC of SBDR decreases from 0.8465 to 0.5115 when the noise level increases from 1% to 3%, and the ACC of RBDR increases from 0.8425 to 0.9230 when the noise level increases from 0 to 1%. Similar things happen with the NMI results. However, with regard to ACC and NMI, the performance of OBDR decreases gradually and monotonically with the increasing noise levels, which shows the stability of OBDR on the noisy data with outliers. Additionally, among these algorithms, the SBDR and the OBDR are the top two algorithms in terms of both ACC and NMI. Specifically, in terms of ACC, the OBDR wins in six test cases, and the second best one is the SBDR, which wins in four cases. However, in terms of NMI, the SBDR wins in 11 cases, and the second best one is OBDR, which wins in 3 cases. The two algorithms work in different ways. The SBDR pursues the structure of subspaces at the expense of long running time (see Figure 5), and achieves the best NMIs in most cases on D_1 . However, the OBDR models the outliers and separates the noises from the subspaces, thus it has the second better performance. On the datasets D_2 and D_3 , the OBDR has significant advantages over the compared algorithms, including the SBDR with regard to ACC and NMI. As the dataset D_4 is concerned, when the data set is corrupted by various noises simultaneously, the OBDR wins in all the three cases in terms of ACC, while the SSC wins in two out of three cases with regard to NMI.

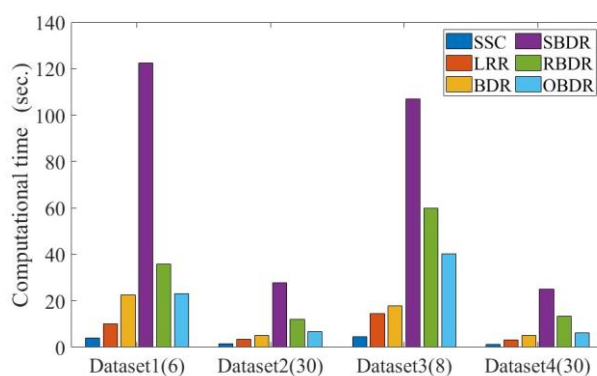


Figure 5. The computational time of algorithms on Dataset1~Dataset4.

Moreover, we report the wins/ties/losses (W/T/L) and average ranks to record times of best performance and overall performance of the algorithms, respectively. From Table 5, we observe that, in view of W/T/L, OBDR performs significantly better than other algorithms on the test datasets. Specifically, the OBDR wins in 28 and 19 out of the 42 cases in terms of ACC and NMI, respectively, which are higher than that of other algorithms. In terms of average rank, we observe that OBDR is better than the others. Lower ranks are better, and OBDR is the algorithm with the lowest average ranks in terms of ACC and NMI (average rank = 1.50 and 2.19, respectively), and SBDR is the second lowest one (average rank = 2.73 and 2.42, respectively). Meanwhile, we calculate the p -values of the Friedman test [49] for statistical comparison between the six algorithms. The results are shown in Table 5. In terms of both ACC and NMI, Friedman test results are significant at $\alpha = 0.05$ significance level. Thus, the six algorithms perform significantly different from each other at the test datasets.

Table 5. W/T/L, average ranks and p -values.

		SSC	LRR	BDR	SBDR	RBDR	OBDR	p -Value
ACC	W/T/L	0/0/42	6/0/36	1/1/40	5/0/37	1/1/40	28/0/14 ³	1.0×10^{-21}
	Av. ranks	4.65	3.65	3.33	2.73	5.24	1.50	
NMI	W/T/L	2/0/40	4/0/38	2/0/40	14/0/28	1/0/41	19/0/23	1.2×10^{-14}
	Av. ranks	4.12	3.55	3.52	2.42	5.20	2.19	

³ The best performance is shown in bold.

As shown in Tables 3–5, on the noisy data sets, both ACC and NMI of OBDR obtained the most times of best performance and the best average ranks, thus OBDR can handle various data noise corruptions better than the compared algorithms. Furthermore, we conduct pairwise comparisons using the Nemenyi multiple comparison test [50], as shown in Table 6. We observe that at the significance hypothesis level $\alpha = 0.05$, OBDR is significantly different from all compared algorithms in terms of ACC, and significantly different from all compared algorithms except SBDR in terms of NMI, indicating that the SBDR is also robust to data noises. In addition, RBDR does not show good clustering performance, possibly due to its special preprocessing methods of data sets [39]. Overall, the OBDR outperforms the compared algorithms with regard to the evaluation metrics.

Table 6. The p -values of using the Nemenyi multiple comparison test on Dataset1~Dataset4.

		SSC	LRR	BDR	SBDR	RBDR
ACC	LRR	0.0729	-	-	-	-
	BDR	0.0153	0.9952	-	-	-
	SBDR	3.4×10^{-5}	0.3353	0.6725	-	-
	RBDR	0.7094	0.0005	4.5×10^{-5}	1.1×10^{-8}	-
	OBDR	2.0×10^{-13}	7.8×10^{-6}	0.0001	0.0319	7.1×10^{-14}
NMI	LRR	0.7273	-	-	-	-
	BDR	0.6911	1.0000	-	-	-
	SBDR	0.0004	0.0623	0.0729	-	-
	RBDR	0.0849	0.0007	0.0006	1.3×10^{-10}	-
	OBDR	3.4×10^{-5}	0.0114	0.0139	0.9938	2.5×10^{-12}

Figure 5 records the computational time of algorithms on Dataset1~Dataset4 with the maximum number of clusters, i.e., Dataset1(6), which denotes Dataset1 with six subjects. It can be seen that SBDR achieves good performance but needs to take a long time to update the subspace structure matrix.

4.4. Parameter Analysis

Fixing the number of iterations of Algorithm 1, ρ will directly affect the clustering performance of OBDR. In this section, we search the optimal values of ρ in OBDR on different data sets. We use noise-free data with the maximum number of clusters in Dataset1~Dataset3 as the test sets, such as Dataset1(6). The maximum iteration times of OBDR is set to 500, and the values of λ and γ are set as Table 2. The effect of ρ on OBDR is shown in Figure 6.

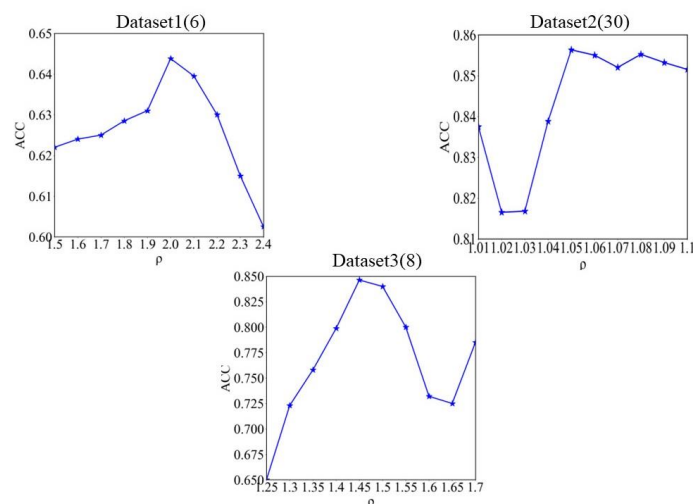


Figure 6. The effect of ρ .

Based on Figure 6, ρ is set to 2.0, 1.05, and 1.45 on Dataset1, Dataset2 (Dataset4), and Dataset3, respectively. After fixing ρ , we also conduct the experiments to test the sensitivity of λ and γ . The results are shown in Figure 7. From Figure 7, we can see that OBDR is more sensitive to λ than γ on all three datasets. It is observed that OBDR has steady clustering performance when λ is within [1, 100], and the parameter values in Table 2 are sound for the three datasets.

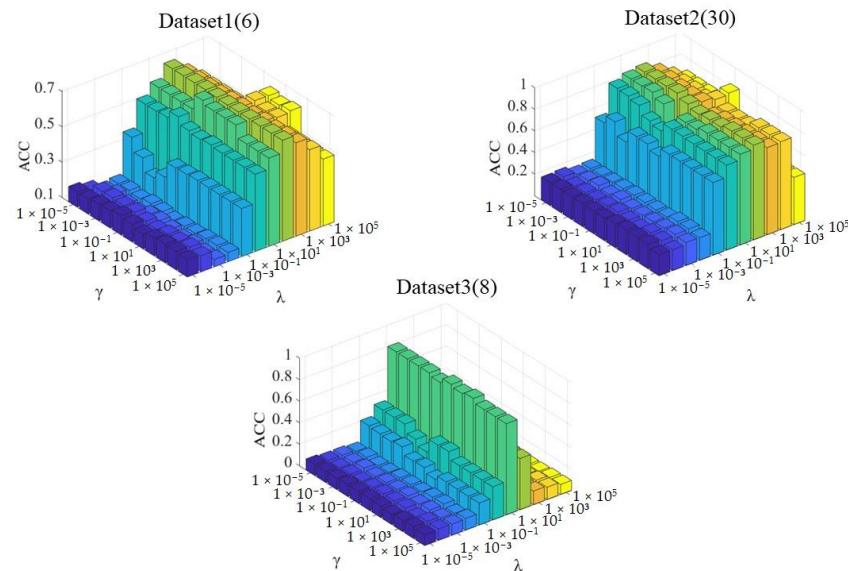


Figure 7. The ACCs of OBDR with different λ and γ .

5. Conclusions

In this paper, we proposed the robust subspace clustering with block diagonal representation algorithm (OBDR). OBDR reduces the influence of noises on the construction of the coefficient matrix via $l_{2,1}$ norm and F norm, recovers the block diagonal spatial structure distribution of data by the Laplacian rank constraint, and thus improves the clustering performance. Experiments on several artificial noisy datasets demonstrate the effectiveness of OBDR in dealing with complex noises. In this study, the OBDR is evaluated on the noisy face datasets and handwritten digit datasets, and the results show that the OBDR has advantages over the compared algorithms with regard to the evaluation metrics. Moreover, the OBDR can be applied in other subspace clustering tasks, such as motion segmentation, image segmentation, and object clustering. In the future, we plan to investigate subspace clustering with deep learning on more high dimensional data. Since the deep learning network can transform the input data into a low dimensional latent space [51–56], combining of OBDR and the deep learning network (e.g., AutoEncoder) is worthy of careful study.

Author Contributions: Conceptualization, Q.L.; methodology, Z.X. and L.W.; software, Z.X.; validation, L.W.; investigation, Q.L.; writing—original draft preparation, Z.X.; writing—review and editing, Z.X. and Q.L.; visualization, Z.X.; supervision, Q.L. and L.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by “National Natural Science Foundation of China”, grant number 62072391.

Data Availability Statement: The data used in this study are reported in the figures and tables of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Jain, A.K.; Murty, M.N.; Flynn, P.J. Data clustering: A review. *ACM Comput. Surv.* **1999**, *31*, 264–323. [\[CrossRef\]](#)
- Gear, C.W. Multibody grouping from motion images. *Int. J. Comput. Vis.* **1998**, *29*, 133–150. [\[CrossRef\]](#)
- Liu, Y.; Wang, K.; Liu, L.; Lan, H.; Lin, L. Tcgl: Temporal contrastive graph for self-supervised video representation learning. *IEEE Trans. Image Process.* **2022**, *31*, 1978–1993. [\[CrossRef\]](#)
- Ban, Y.; Liu, M.; Wu, P.; Yang, B.; Liu, S.; Yin, L.; Zheng, W. Depth estimation method for monocular camera defocus images in microscopic scenes. *Electronics* **2022**, *11*, 2012. [\[CrossRef\]](#)
- Jing, L.P.; Ng, M.K.; Huang, J.Z. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. *IEEE Trans. Knowl. Data Eng.* **2007**, *19*, 1026–1041. [\[CrossRef\]](#)
- Hong, W.; Wright, J.; Huang, K.; Ma, Y. Multiscale hybrid linear models for lossy image representation. *IEEE Trans. Image Process.* **2006**, *15*, 3655–3671. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhou, W.; Lv, Y.; Lei, J.; Yu, L. Global and local-contrast guides content-aware fusion for RGB-D saliency prediction. *IEEE Trans. Syst. Man Cybern.* **2021**, *51*, 3641–3649. [\[CrossRef\]](#)
- Basri, R.; Jacobs, D.W. Lambertian reflectance and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**, *25*, 218–233. [\[CrossRef\]](#)
- Jiang, S.; Zhao, C.; Zhu, Y.; Wang, C.; Du, Y.; Lei, W.; Wang, L. A practical and economical ultra-wideband base station placement approach for indoor autonomous driving systems. *J. Adv. Transp.* **2022**, *2022*, 3815306. [\[CrossRef\]](#)
- Johnson, S.C. Hierarchical clustering schemes. *Psychometrika* **1967**, *32*, 241–254. [\[CrossRef\]](#)
- Kriegel, H.P.; Kröger, P.; Sander, J.; Zimek, A. Density-based clustering. *Wiley Interdiscip. Rev.-Data Min. Knowl. Discov.* **2011**, *1*, 231–240. [\[CrossRef\]](#)
- Xiong, S.; Li, B.; Zhu, S. DCGNN: A single-stage 3D object detection network based on density clustering and graph neural network. *Complex Intell. Syst.* **2022**. [\[CrossRef\]](#)
- Elhamifar, E.; Vidal, R. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 2765–2781. [\[CrossRef\]](#) [\[PubMed\]](#)
- Qin, X.; Ban, Y.; Wu, P.; Yang, B.; Liu, S.; Yin, L.; Liu, M.; Zheng, W. Improved image fusion method based on sparse decomposition. *Electronics* **2022**, *11*, 2321. [\[CrossRef\]](#)
- Liu, G.C.; Lin, Z.C.; Yu, Y. Robust subspace segmentation by low-rank representation. In Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 1–8.
- Liu, M.S.; Wang, Y.; Sun, J.; Ji, Z.C. Structured block diagonal representation for subspace clustering. *Appl. Intell.* **2020**, *50*, 2523–2536. [\[CrossRef\]](#)
- Lu, C.Y.; Feng, J.S.; Lin, Z.C.; Mei, T.; Yan, S.C. Subspace clustering by block diagonal representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 487–501. [\[CrossRef\]](#) [\[PubMed\]](#)
- Vidal, R.; Ma, Y.; Sastry, S. *Generalized Principal Component Analysis*; Springer: New York, NY, USA, 2016.
- Liu, G.C.; Lin, Z.C.; Yan, S.C.; Sun, J.; Yu, Y.; Ma, Y. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 171–184. [\[CrossRef\]](#)
- Hu, J.; Wu, Y.; Li, T.; Ghosh, B.K. Consensus control of general linear multiagent systems with antagonistic interactions and communication noises. *IEEE Trans. Autom. Control* **2019**, *64*, 2122–2127. [\[CrossRef\]](#)
- Zhong, T.; Wang, W.; Lu, S.; Dong, X.; Yang, B. RMCHN: A residual modular cascaded heterogeneous network for noise suppression in DAS-VSP Records. *IEEE Geosci. Remote Sens. Lett.* **2022**, *20*, 7500205. [\[CrossRef\]](#)
- Yang, C.; Zhang, J.; Huang, Z. Numerical study on cavitation-vortex-noise correlation mechanism and dynamic mode decomposition of a hydrofoil. *Phys. Fluids* **2022**, *34*, 125105. [\[CrossRef\]](#)
- Huang, N.; Chen, Q.; Cai, G.; Xu, D.; Zhang, L.; Zhao, W. Fault diagnosis of bearing in wind turbine gearbox under actual operating conditions driven by limited data with noise labels. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 3502510. [\[CrossRef\]](#)
- Li, R.; Zhang, H.; Chen, Z.; Yu, N.; Kong, W.; Li, T.; Liu, Y. Denoising method of ground-penetrating radar signal based on independent component analysis with multifractal spectrum. *Measurement* **2022**, *192*, 110886. [\[CrossRef\]](#)
- Liu, F.; Zhao, X.; Zhu, Z.; Zhai, Z.; Liu, Y. Dual-microphone active noise cancellation paved with doppler assimilation for TADS. *Mech. Syst. Signal Process.* **2023**, *184*, 109727. [\[CrossRef\]](#)
- He, R.; Zheng, W.S.; Tan, T.N.; Sun, Z.N. Half-quadratic-based iterative minimization for robust sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *36*, 261–275.
- Favaro, P.; Vidal, R.; Ravichandran, A. A closed form solution to robust subspace estimation and clustering. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1801–1807.
- Qin, Y.; Zhang, X.; Shen, L.; Feng, G. Maximum block energy guided robust subspace clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 2652–2659. [\[CrossRef\]](#)
- Duy, H.; Dengy, Y.; Xueyz, J.; Mengyz, D.; Zhaoy, Q.; Xuy, Z. Robust online CSI estimation in a complex environment. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 8322–8336.
- Chen, J.H.; Yang, J. Robust subspace segmentation via low-rank representation. *IEEE Trans. Cybern.* **2014**, *44*, 1432–1445. [\[CrossRef\]](#) [\[PubMed\]](#)
- Liu, G.C.; Yan, S.C. Latent low-rank representation for subspace segmentation and feature extraction. In Proceedings of the 2011 IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 1615–1622.

32. Zhang, H.Y.; Lin, Z.C.; Zhang, C.; Gao, J.B. Robust latent low rank representation for subspace clustering. *Neurocomputing* **2014**, *145*, 369–373. [\[CrossRef\]](#)
33. Ji, P.; Salzmann, M.; Li, H.D. Efficient dense subspace clustering. In Proceedings of the 2014 IEEE Winter Conference on Applications of Computer Vision, Steamboat Springs, CO, USA, 24–26 March 2014; pp. 461–468.
34. Jing, L.P.; Ng, M.K.; Zeng, T.Y. Dictionary learning-based subspace structure identification in spectral clustering. *IEEE Trans. Neural Netw. Learn.* **2013**, *24*, 1188–1199. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Nie, F.; Chang, W.; Hu, Z.; Li, X. Robust subspace clustering with low-rank structure constraint. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 1404–1415. [\[CrossRef\]](#)
36. Guo, L.; Zhang, X.; Liu, Z.; Xue, X.; Wang, Q.; Zheng, S. Robust subspace clustering based on automatic weighted multiple kernel learning. *Inf. Sci.* **2021**, *573*, 453–474. [\[CrossRef\]](#)
37. Xue, X.; Zhang, X.; Feng, X.; Sun, H.; Chen, W.; Liu, Z. Robust subspace clustering based on non-convex low-rank approximation and adaptive kernel. *Inf. Sci.* **2019**, *513*, 190–205. [\[CrossRef\]](#)
38. He, R.; Zhang, Y.Y.; Sun, Z.N.; Yin, Q.Y. Robust subspace clustering with complex noise. *IEEE Trans. Image Process.* **2015**, *24*, 4001–4013. [\[PubMed\]](#)
39. Wang, L.J.; Huang, J.W.; Yin, M.; Cai, R.C.; Hao, Z.F. Block diagonal representation learning for robust subspace clustering. *Inf. Sci.* **2020**, *526*, 54–67. [\[CrossRef\]](#)
40. Nie, F.P.; Wang, H.; Cai, X.; Huang, H.; Ding, C. Robust matrix completion via joint Schatten p-norm and lp-norm minimization. In Proceedings of the 12th IEEE International Conference on Data Mining, Brussels, Belgium, 10–13 December 2012; pp. 566–574.
41. Rockafellar, R.T. Augmented Lagrange multiplier functions and duality in nonconvex programming. *SIAM J. Control Optim.* **1974**, *12*, 268–285. [\[CrossRef\]](#)
42. Khan, M.; Imtiaz, S.; Parvaiz, G.; Hussain, A.; Bae, J. Integration of Internet-of-Things with blockchain technology to enhance humanitarian logistics performance. *IEEE Access* **2021**, *9*, 25422–25436. [\[CrossRef\]](#)
43. Khan, M.; Parvaiz, G.; Dedahanov, A.; Abdurazzakov, O.; Rakhmonov, D. The Impact of technologies of traceability and transparency in supply chains. *Sustainability* **2022**, *14*, 16336. [\[CrossRef\]](#)
44. Huang, J.; Nie, F.P.; Huang, H.; Ding, C. Robust manifold nonnegative matrix factorization. *ACM Trans. Knowl. Discov. Data* **2014**, *8*, 1–21. [\[CrossRef\]](#)
45. Lin, Z.C.; Chen, M.M.; Ma, Y. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv* **2010**, arXiv:1009.5055.
46. Vincent, P.; Larochelle, H.; Lajoie, I.; Bengio, Y.; Manzagol, P.A. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* **2010**, *11*, 3371–3408.
47. Wang, Y.X.; Xu, H. Noisy sparse subspace clustering. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 89–97.
48. Vinh, N.X.; Epps, J.; Bailey, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.* **2010**, *11*, 2837–2854.
49. Friedman, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J. Am. Stat. Assoc.* **1937**, *32*, 675–701. [\[CrossRef\]](#)
50. Nemenyi, P.B. Distribution-Free Multiple Comparisons. Ph.D. Thesis, Princeton University, Princeton, NJ, USA, 1963.
51. Liu, Y.; Zhang, Z.; Liu, X.; Wang, L.; Xia, X. Efficient image segmentation based on deep learning for mineral image classification. *Adv. Powder Technol.* **2021**, *32*, 3885–3903. [\[CrossRef\]](#)
52. Liu, H.; Liu, M.; Li, D.; Zheng, W.; Yin, L.; Wang, R. Recent advances in pulse-coupled neural networks with applications in image processing. *Electronics* **2022**, *11*, 3264. [\[CrossRef\]](#)
53. Li, M.; Tian, Z.; Du, X.; Yuan, X.; Shan, C.; Guizani, M. Power normalized cepstral robust features of deep neural networks in a cloud computing data privacy protection scheme. *Neurocomputing* **2023**, *518*, 165–173. [\[CrossRef\]](#)
54. Zhou, W.; Wang, H.; Wan, Z. Ore image classification based on improved CNN. *Comput. Electr. Eng.* **2022**, *99*, 107819. [\[CrossRef\]](#)
55. Liu, R.; Wang, X.; Lu, H.; Wu, Z.; Fan, Q.; Li, S.; Jin, X. SCCGAN: Style and characters inpainting based on CGAN. *Mob. Netw. Appl.* **2021**, *26*, 3–12. [\[CrossRef\]](#)
56. Yang, M.; Wang, H.; Hu, K.; Yin, G.; Wei, Z. IA-Net: An inception-attention-module-based network for classifying underwater images from others. *IEEE J. Ocean. Eng.* **2022**, *47*, 704–717. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.