IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 August 2022, accepted 7 September 2022, date of publication 12 September 2022,
date of current version 22 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3205742

RESEARCH ARTICLE

A New Density Peak Clustering Algorithm Based on Cluster Fusion Strategy

FUXIANG LI[1], MING ZHOU[1], SHU LI[1,2], AND TIANHAO YANG[1]
[1]School of Science, Harbin University of Science and Technology, Harbin 150080, China
[2]Key Laboratory of Engineering Dielectric and Applications (Ministry of Education), School of Electrical and Electronic Engineering, Harbin University of Science and Technology, Harbin 150080, China

Corresponding authors: Fuxiang Li (lifx2013@163.com) and Shu Li (lishu@hrbust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 11871181, and in part by the Natural Science Foundation of Heilongjiang Province under Grant A2018008.

ABSTRACT When the density peak clustering algorithm deals with complex datasets and the problem of multiple density peaks in the same cluster, the subjectively selected cluster centers are not accurate enough, and the allocation of non-cluster centers is prone to joint and several errors. To solve the above problems, we propose a new density peak clustering algorithm based on cluster fusion strategy. First, the algorithm screens out the candidate cluster centers by setting two new thresholds to avoid the influence of noise points and outliers. Second, the remaining data points are allocated according to the density peak clustering algorithm to obtain the initial clusters. Third, considering the structural characteristics and spatial distribution of datasets, the new definitions of boundary points, inter-cluster intersection density and inter-cluster boundary density are provided. To correctly classify the clustering problems with multiple density peaks in the same cluster, a new cluster fusion strategy is proposed, which not only corrects the joint and several errors in the allocation of data points, but also correctly selects the cluster centers. Finally, to test the effectiveness of the proposed clustering algorithm, which is compared with DPC-KNN, DPC, K-means and DBSCAN on nine synthetic datasets and six real datasets. The experimental results demonstrate that the clustering performance of the proposed algorithm outperforms that of other algorithms.

INDEX TERMS Clustering, density peaks, candidate cluster center, cluster fusion strategy.

I. INTRODUCTION

Data mining is the mainstream technology of information industry and artificial intelligence neighborhood. Cluster analysis is one of the core technologies in data mining [1] field. Its purpose is to reveal the laws between data and mine the potential and valuable information between data. It provides auxiliary means for decision-making and preparation for technology. With the in-depth study of clustering algorithms, scholars have adopted different processing methods for data and successively proposed many excellent clustering algorithms.

(1) Partition-based clustering, which is divided into hard clustering and soft clustering. Among them, hard clustering is represented by K-means [2], which is mainly embodied in initializing the cluster center, establishing the objective function of the distance between each data point and the cluster center, and accurately classifying each data point into the cluster where the nearest cluster center is located. However, soft clustering is also known as fuzzy clustering, which introduces the idea of fuzzy mathematics into clustering analysis, represented by fuzzy C-means clustering (FCM) [3], [4], the data is classified by membership degree through optimizing the objective function. The FCM algorithm is more and more deeply studied by scholars, and the improved FCM algorithm is widely used in the field of image segmentation. Lei *et al*. [5] proposed a fast FCM clustering algorithm based on superpixels, the histogram was obtained by calculating the number of pixels in the superpixel image, then the FCM algorithm was implemented by combining the superpixel image and the histogram. The application of superpixel images is better adapted to irregular local spatial regions and helps

The associate editor coordinating the review of this manuscript and approving it for publication was Sotirios Goudos.

to improve color image segmentation. Tang *et al*. [6] first calculated the weighted sum distance of the image block to obtain the similarity measure and then obtained the new fuzzy coefficient, and finally proposed the patch-based fuzzy local similarity C-means (PFLSCM) algorithm according to the idea of the image block and the fuzzy coefficient. The algorithm better demonstrated the performance of image segmentation and revealed the practicability of FCM algorithm. FCM algorithm is increasingly becoming the research direction of clustering algorithms in the future.

(2) Hierarchical clustering [7], [8], using bottom-up agglomerative hierarchical clustering or top-down split hierarchical clustering to divide the dataset into layers, and the clustering is stopped until the stopping conditions are met, and finally, a cluster diagram with a tree structure is constructed.

(3) Density-based clustering is represented by DBSCAN [9]. In the algorithm, two parameters are set to describe the degree of closeness between datasets, starting from the core object, forming a cluster of data points whose density is reachable, and stopping the algorithm until all objects have been accessed.

(4) Grid-based clustering is represented by Statistical Information Grid (STING) [10], which describes the division of the data space into multiple rectangular data units with different resolutions. By judging whether the statistical information of each layer of grid cells satisfies the constraints, the layer-by-layer judgment is carried out, and finally, the data points in the same grid are grouped into a cluster.

(5) Model-based clustering [11] is represented by the Gaussian Mixture Model (GMM) [12], it is a probabilistic clustering algorithm that used multiple Gaussian distributions to characterize data, combined with the maximum likelihood method to calculate the parameters of the Gaussian distribution, and then estimate the probability that the data points belong to each cluster, and finally select the cluster with the highest probability as the final clustering result.

In recent years, density-based clustering algorithms have attracted more and more attention from scholars. In 2014, Rodriguez *et al*. [13] proposed the density peak clustering algorithm (Clustering by Fast Search and Find of Density Peaks, DPC) in *Science*. It is a new type of density-based clustering algorithm, which overcomes the problem that the two parameters in the DBSCAN algorithm are difficult to determine and avoids the poor clustering effect due to the large difference in the density of data points in the clustering. The DPC algorithm theory is simple and easy to understand, does not need to specify the number of clusters, has a unique experimental parameter, and can identify non-convex clusters. With the in-depth study of the DPC algorithm by scholars, the DPC algorithm and its variants have been widely used in the fields of fault diagnosis [14], face recognition [15], community personalized recommendation [16], medicine [17], etc. and play an increasingly important supporting role. However, the DPC algorithm still has some shortcomings, such as it is subjective to artificially select cluster centers through decision graphs, sensitive to the selection of parameters, easy occurrence of knock-on effects in the process of distributing the remaining data, and the poor clustering effect on the datasets with large distribution differences.

The main motivation of our research work is to avoid the inaccuracy of artificial subjective selection of cluster centers, reduce the allocation error of non-cluster centers, and improve the clustering performance when processing datasets with complex shapes and clusters with multiple density peaks. Therefore, we propose a new density peak clustering algorithm (CFDPC) based on Clustering Fusion Strategy. The main innovations and contributions are summarized below.

(1) By setting two new thresholds to filter out candidate clustering centers, initial cluster synthesis is performed based on density peak clustering, to avoid the influence of artificial selection of clustering centers.

(2) New definitions of boundary point, inter-cluster intersection density and inter-cluster boundary density are proposed, which are suitable for datasets with complex shapes, and lay the foundation for the proposed new clustering fusion strategy, which improves the joint and several errors in the process of data point allocation, and corrects the initial clustering results.

(3) The proposed cluster fusion strategy correctly selects the cluster centers, especially in the case of a cluster with multiple density peaks. However, some improved algorithms for DPC ignore this consideration.

(4) The clustering performance of the CFDPC algorithm is tested on artificial and real datasets and compared with other four advanced clustering algorithms, the experiments show that the CFDPC algorithm has higher clustering accuracy and robustness.

The structure of the rest of this paper is as follows: Section II reviews the related improvement research on DPC algorithms. Section III introduces the principle of the DPC algorithm. Section IV describes the work of the CFDPC algorithm in detail. Section V verifies the effectiveness of the algorithm, and gives the comparison of the clustering evaluation results of the CFDPC algorithm and the other four algorithms under the datasets of different shapes and structures. Section VI gives the conclusion of this paper.

II. RELATED WORKS

Scholars have improved the DPC algorithm from different research angles for different shortcomings.

The first aspect is the improvement of local density and cut-off distance. Du *et al*. [18] proposed a new way (DPC-KNN) to calculate local density, and used k-nearest neighbors instead of d_c to make the parameters easier to tune. Liu *et al*. [19] redefined the local density and relative distance based on the shared neighbor similarity, and considered the neighbor information of the data points in the density and distance calculation, overcoming the unicity of the DPC algorithm for data correlation calculation. Liu *et al*. [20] proposed a mixed density peak clustering algorithm (DDNFC) by

obtaining two results of data points density calculation using local spatial position deviation and reverse k-nearest neighbor technique. Chen *et al*. [21] proposed a Domain Adaptive Density Clustering Algorithm (DADC) and proposed a domain density calculation method using k-nearest neighbor method and a self-identification strategy of cluster centers for three specific distribution datasets. Finally, the integration method merges fragmented clusters. Jiang *et al*. [22] proposed an adaptive density peak clustering based on k-nearest neighbors and the Gini coefficient. Since the calculation of the local density is related to the choice of the cut-off distance d_c and has an important influence on the clustering results, the Gini coefficient is used to find the optimal cut-off distance d_c, which realizes the consideration of the overall situation of the datasets. The above scholars applied various deformation techniques of k-nearest neighbors to the DPC algorithm, which solved the sensitivity of the DPC algorithm to the parameter d_c and considered the neighborhood information of the data points. However, it is not easy to obtain the k-nearest neighbor parameters, and the above algorithms still have high complexity.

The second aspect is to reduce the amount of computation. Xu *et al*. [23] proposed a fast sparse search density peak clustering algorithm (FSDPC), which used random third-party data points to find nearest neighbors for sparse search, thereby improving the efficiency of the DPC algorithm. Xu *et al*. [24] proposed two density screening strategies, grid division (GDPC) and circle division (CDPC), which improved the efficiency of the DPC algorithm. Shan *et al*. [25] proposed a density peak clustering algorithm (SKTDPC) based on sparse search and k-d tree. The algorithm is based on the k-d tree theory to find the k nearest neighbors of data points and calculate the sparse distance matrix, realizing the double acceleration of local density and relative distance calculation, reducing the complexity of the algorithm and improving the efficiency of the DPC algorithm. Although these optimized algorithms improve the efficiency of clustering and reduce the complexity of the algorithm, they come at the expense of the stability of the algorithm. It can be clearly seen that some clustering results have large differences.

The third aspect is to determine the clustering centers. Flores *et al*. [26] automatically determined cluster centers by the spacing between data points in a one-dimensional decision graph. Lv *et al*. [27] calculated the difference change between the decision values, and automatically obtained the cluster centers according to the position of the inflection point. Shan *et al*. [25] used the second-order difference method to adaptively determine the cluster centers. The above improved algorithms of DPC avoid the inaccuracy and subjectivity of the artificial selection of clustering centers. However, some algorithms for large-scale datasets are more time-consuming.

The fourth aspect is the improvement of data points allocation. Xie *et al*. [28] proposed a clustering algorithm (FKNN-DPC) that uses breadth-first search and fuzzy weighted k-nearest neighbor search to assign non-cluster centers, avoiding misassignment of data points. Lotfi *et al*. [29] proposed a method density-based KNN graph labeling backbone (DPC-DBFN), which prevented the chain reaction caused by misassignment of data points. Yu *et al*. [30] proposed density peak clustering based on weighted local density sequence and nearest neighbor assignment (DPCSA). The error propagation of cluster labels is overcome by introducing weighted local density sequence and two-stage assignment strategy. Yang *et al*. [31] proposed a generalized density peak clustering algorithm (GDPC) based on the new order similarity, which used the Euclidean distance between samples to calculate the order similarity, and adopted two-step assignment to weaken the propagation of data errors. Yu *et al*. [32] proposed a three-way density peak clustering method (3W-DPET) based on evidence theory. Using evidence theory to construct and collect the information of k-nearest neighbors in order to assign ungrouped objects to the most suitable clusters, and can effectively solve the problem of cluster labels error propagation. Although the above algorithms overcome the error probability of data points assignment to a certain extent, it is not effective for datasets with close distribution and cross overlap.

The fifth aspect is the merging method of clustering. Fang *et al*. [33] proposed a density peak clustering (CFDPC) based on adaptive kernel fusion. The algorithm automatically found initial clusters based on density peaks, used an adaptive search method to find core points, and finally obtained the final clusters according to the core fusion strategy of intra-class similarity. Sun *et al*. [34] proposed density peak clustering (DPC-MC) based on k-nearest neighbors and self-recommendation, which selected initial cluster centers through a self-recommendation strategy, and aggregated the clusters by the degree of association between clusters. Liu *et al*. [35] proposed an adaptive density peak clustering and aggregation strategy based on k-nearest neighbors (ADPC-KNN), which merged clusters according to the density reachability strategy. Yuan *et al*. [36] used k-nearest neighbors to divide data points and used a cluster merging strategy to automatically aggregate over-segmented clusters. The above optimization algorithms are improvements of the DPC algorithm from the perspective of multi-cluster fusion, but the design requirements of each step of the fusion strategy are very strict, otherwise, there will be the phenomenon of excessive fusion or no fusion between clusters, resulting in the sacrifice of clustering precision.

The sixth aspect is algorithm fusion. Liu *et al*. [37] proposed a density gain rate peak algorithm (DGPC) based on spectral clustering. This algorithm integrated the idea of density gain rate into the DPC algorithm, and then used spectral clustering to extract the features of the similarity map of the samples, thereby clustering the samples. It overcomes the shortcomings of the two algorithms and takes into account the local structure of DPC algorithm to better handle the uneven datasets. Wang *et al*. [38] created the density attribute through the density peak algorithm and introduced it into the

AP algorithm, which provided a new similarity calculation method for the AP algorithm.

In short, it is difficult for all derivatives of DPC algorithms to simultaneously satisfy high clustering accuracy, stable algorithm performance, automatic identification of cluster centers, and effective avoidance of data points error propagation. These derivatives have some drawbacks more or less. In view of the consensus problem of DPC algorithm, this paper proposed a new density peak clustering algorithm based on cluster fusion strategy (CFDPC). Firstly, two new thresholds are set to screen out the candidate cluster centers, and the initial cluster synthesis is performed based on DPC, which avoids the influence of artificial selection of cluster centers. Secondly, new definitions of boundary points, inter-cluster intersection density, and inter-cluster boundary density are given for datasets with complex shapes to lay the foundation for the proposed new cluster fusion strategy. Finally, the accurate clustering centers and clusters are obtained by iterative fusion. It improves the joint and several errors in the data point assignment process and corrects the initial clustering results. And clusters with multiple density peaks are correctly classified.

III. PRINCIPLE OF DPC ALGORITHM

In this section, we review the principles of traditional DPC algorithms.

The density peak clustering algorithm (DPC) consists of two basic elements. First, the local density of the data points as the cluster centers are higher than that of other data points around; second, the relative distance between the two data points as the cluster center is far. Local density and relative distance are two important variables in the implementation of density peak clustering algorithm.

Local density ρ_i definition of data point i:

$$\rho_i = \sum_{j \neq i} \chi\left(d_{i,j} - d_c\right) \tag{1}$$

where $d_{i,j}$ represents the Euclidean distance between data points i and j; d_c is the cut-off distance, which is obtained by the percentage parameter p, and the value range of p generally makes the number of neighbors of each data point account for $1\% - 2\%$ of the total data volume. $\chi(u) = \begin{cases} 0, u \geq 0 \\ 1, u < 0 \end{cases}$ is a counting function. ρ_i means the number of data points in the circular neighborhood with data point i as the center and d_c as the radius. Equation (1) is generally applicable to a small number of datasets, and equation (2) is usually used to calculate the local density of datasets with a large number of data points, which is another definition of the local density.

$$\rho_i = \sum_{j \neq i} \exp\left(-\frac{{d_{i,j}}^2}{{d_c}^2}\right) \tag{2}$$

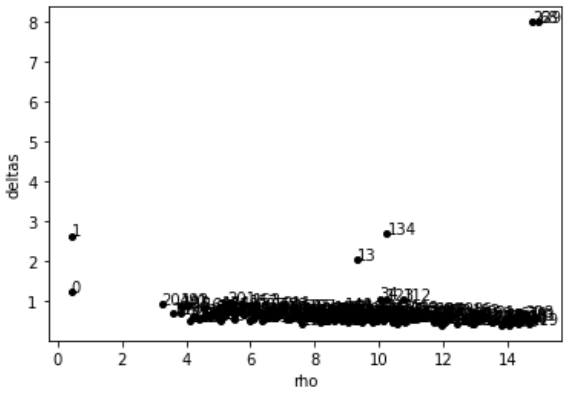


FIGURE 1. Decision graph of DPC algorithm.

The relative distance δ_i of data point i is defined as follows:

$$\delta_i = \begin{cases} \max\limits_{j}(d_{i,j}), & \rho_i = \max(\rho) \\ \min\limits_{j:\rho_j > \rho_i}(d_{i,j}), & otherwise \end{cases} \tag{3}$$

when the local density of data point i is the maximum, then data point i is the cluster center, the relative distance of data point i is the maximum distance from other data points to i; when the local density of data point i is not the maximum, the relative distance of data point i refers to the shortest distance from all data points j with greater density than data point i to data point i.

After calculating ρ_i and δ_i for each data point, construct a decision graph with local density as the horizontal axis and relative distance as the vertical axis as shown in Fig. 1.

Select data points with large local density and relative distance from the decision graph as cluster centers. Finally, the remaining data points are classified into the nearest data points whose density is greater than their own. The steps of the DPC algorithm are as follows:

Algorithm 1 DPC Algorithm

Input: Dataset $S = \{x_1, x_2, \ldots, x_n\}$, parameter p
Output: $C = \{C_1, C_2, \ldots, C_m\}$

1: Calculate distance matrix $D_{n \times n}$
2: Determine d_c value
3: Calculate ρ_i and δ_i for each data point according to (2) and (3)
4: Draw decision graph and select cluster centers
5: Assign non-cluster center points

IV. A NEW DENSITY PEAK CLUSTERING ALGORITHM BASED ON CLUSTER FUSION STRATEGY

To solve the problem of artificial selection of cluster centers in DPC algorithm, and to further correct the joint and several errors of data point allocation. This paper proposes a new density peak clustering algorithm based on cluster fusion strategy (CFDPC) and based on the local density and relative

distance of DPC algorithm, this paper defines a new method to select candidate cluster centers. Then initial clusters are obtained according to the allocation rules of DPC algorithm, and the number of initial clusters obtained is greater than the number of real clusters. Therefore, a new cluster fusion strategy is designed to aggregate initial clusters according to certain rules, so as to the final clustering is obtained. Experiments show that the CFDPC algorithm has better clustering performance.

A. CANDIDATE CLUSTER CENTERS AND INITIAL CLUSTERS

First, the local density ρ_i and relative distance δ_i are calculated according to (2) and (3), respectively. Second, two thresholds are adaptively set to automatically obtain the candidate cluster centers. The category labels of the remaining points are the same as those of the data points with the nearest distance and greater density to obtain the initial clustering.

Let $S=\{x_1, x_2, \ldots, x_n\}$ represents a dataset composed of n data points, and the data points as the cluster centers generally need to meet the conditions of large local density and relative distance at the same time. Therefore, in order to prevent the influence of noise points with small local density and large relative distance and outliers with large local density and small relative distance on clustering results, two thresholds are redefined to screen out candidate cluster centers.

$$OO_i=\{x_i|\rho_i \geq \mu(\rho)\} \tag{4}$$

where $\mu(\rho)$ is the mean of the local density of data points, and OO_i is the set that becomes the expected cluster centers excluding noise points:

$$CO_i=\{x_i|OO_i \geq \mu(\delta)\} \tag{5}$$

where $\mu(\delta)$ represents the mean value of the relative distance of data points, CO_i represents the set of real candidate cluster centers obtained by removing outliers on the basis of OO_i, and finally, the remaining data points are classified according to the distribution principle of the DPC algorithm to obtain initial clusters. However, the initial clustering results may be incorrect, so this paper proposes a cluster fusion strategy to improve the initial clusters.

B. CLUSTER FUSION STRATEGY

Definition 1 (Boundary Points of a Single Cluster): The set of boundary points of a single cluster C_a is represented by B_a, which is defined as

$$B_a = \{x_a|x_a = \arg sort(\rho_a)[:m]\} \tag{6}$$

$$m = int\left(|C_a| \times q\%\right), q \in [5, 20] \tag{7}$$

where ρ_a is the local density of cluster C_a, $|C_a|$ is the number of data points in cluster C_a, $\arg sort(\rho_a)$ represents the serial number of data points in cluster C_a sorted in an ascending order by local density, and m represents the number of boundary points, and $[:m]$ represents the first m data points corresponding to the sorted number. That is, the data points in cluster C_a are sorted in an ascending order according to the local density, and the data points with the first $5\% - 20\%$ of the local density are taken as the boundary points of cluster C_a.

Definition 2 (Inter-Cluster Boundary Density): Let the two clusters to be merged be C_u and C_v, and B_ρ represents the inter-cluster boundary density, which is defined as follows:

$$B_{u,v} = \frac{\overline{B_\rho^u} + \overline{B_\rho^v}}{2} \tag{8}$$

where $\overline{B_\rho^u}$ represents the mean value of the boundary point density of cluster C_u, and $\overline{B_\rho^v}$ represents the mean value of the boundary point density of cluster C_v.

Definition 3 (Inter-Cluster Intersection Density): Let the two clusters to be merged be C_u and C_v, and let h_u and h_v be the union of the set of data points in the neighborhood of circles with radius d_c for each data point in clusters C_u and C_v respectively, then inter-cluster intersection density is denoted by $A_{u,v}$:

$$A_{u,v} = \max\{\rho_i|i \in h_u \cap h_v\} \tag{9}$$

Definition 4 (Cluster Fusion Conditions): Let the two clusters to be merged be C_u and C_v. If the following conditions are met:

$$B_{u,v} \leq A_{u,v} \tag{10}$$

then cluster C_u and cluster C_v are fused.

The main ideas of cluster fusion strategy are as follows:

Step 1: Calculate the boundary points according to (6) and (7), and calculate the inter-cluster intersection density and inter-cluster boundary density to obtain inter-cluster intersection density matrix $A_{b\times b}$ and inter-cluster boundary density matrix $B_{b\times b}$ according to (8) and (9), where $B_{i,j}$ represents the boundary density between the i-th cluster and the j-th cluster; $A_{i,j}$ represents the intersection density between the i-th cluster and the j-th cluster, and $A_{b\times b}$ and $B_{b\times b}$ are symmetric matrix.

Step 2: The clusters are indexed in descending order of local density of cluster centers. That is: to index in the order of $i = 1, 2, \ldots, b, j = i+1, i+2, \ldots, b$, if $A_{i,j} > B_{i,j}$, merge the i-th cluster and the j-th cluster, update the cluster center set H: delete the j-th cluster center point from H, and the cluster center of the i-th cluster is taken as the cluster center of the new cluster; update the set N of the cluster: merge the data points of the j-th cluster into the i-th cluster, delete the j-th cluster from N, and sort the data points in the i-th cluster according to local density in an ascending order; update the label set M: update the label of the merged j-th cluster to the label of the i-th cluster, and delete the label of the j-th cluster from M; $b = b - 1$.

Step 3: Return to step 1 calculate the new inter-cluster intersection density matrix $A_{b\times b}$ and inter-cluster boundary density matrix $B_{b\times b}$, and then judge whether the cluster fusion conditions are met, and if so, return to step 2 until $A_{i,j} < B_{i,j}$, stop cluster fusion, and output the clustering result.

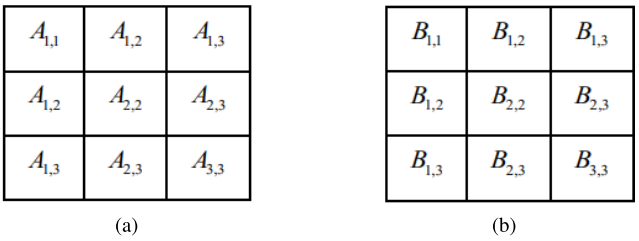


FIGURE 2. Inter-cluster intersection density matrix $A_{3\times3}$ and Inter-cluster boundary density matrix $B_{3\times3}$: (a)-(b).

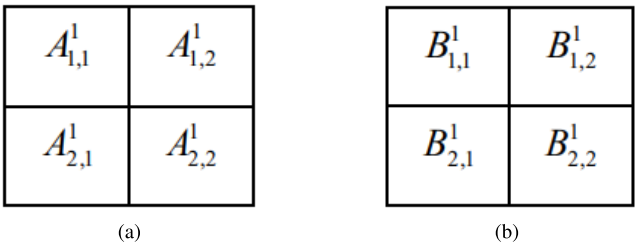


FIGURE 3. Inter-cluster intersection density matrix $A^1_{2\times2}$ and Inter-cluster boundary density matrix $B^1_{2\times2}$: (a)-(b).

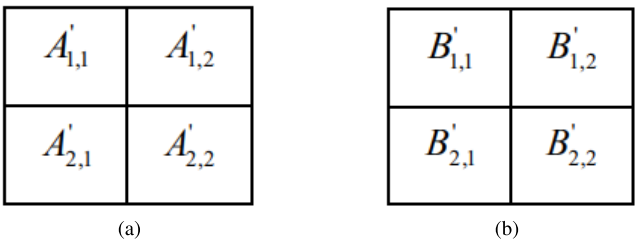


FIGURE 4. Inter-cluster intersection density matrix $A'_{2\times2}$ and Inter-cluster boundary density matrix $B'_{2\times2}$: (a)-(b).

Take three clusters as an example to describe the cluster fusion process:

By setting the threshold, the candidate cluster centers are screened out and the remaining data points are allocated in combination with the DPC algorithm. It is assumed that three initial clusters are obtained: the centers of the clusters are sorted according to the local density in a descending order to obtain $H = \{H_1, H_2, H_3\}$, and the set of labels is $M = \{M_1, M_2, M_3\}$, the cluster set is $N = \{N_1, N_2, N_3\}$, and the data points in the set $N_i (i = 1, 2, 3)$ are sorted according to the local density in an ascending order, these three sets are corresponding to each other.

(1) Calculate the boundary points according to (6) and (7), and calculate the inter-cluster intersection density and inter-cluster boundary density to obtain inter-cluster intersection density matrix $A_{3\times3}$ and inter-cluster boundary density matrix $B_{3\times3}$ according to (8) and (9), as shown in Fig. 2.

(2) Index the first cluster N_1 and the second cluster N_2 in the set N:

a) If the inter-cluster intersection density and the inter-cluster boundary density meet $A_{1,2} > B_{1,2}$, then H_1 and H_2 are merged and H_2 is deleted from H, H is updated to $H = \{H_1, H_3\}$, H_1 as the center of the new cluster; then N_2 is merged into N_1, the new cluster is still recorded as N_1, and $N = \{N_1, N_3\}$ is obtained, the data points in N_1 of the first cluster after merging are sorted according to local density in an ascending order; similarly, get $M = \{M_1, M_3\}$, update the label in M_1: update the merged cluster label M_2 to the label in M_1; recalculate the boundary points according to (6) and (7), and calculate the inter-cluster intersection density and inter-cluster boundary density to obtain inter-cluster intersection density matrix $A^1_{2\times2}$ and inter-cluster boundary density matrix $B^1_{2\times2}$ according to (8) and (9), as shown in Fig. 3.

According to experience, it is known that cluster fusion can not be continued, and there is no need to judge the cluster fusion conditions.

b) If $A_{1,2} < B_{1,2}$, index the first cluster N_1 and the third cluster N_3 in the set N, and judge whether $A_{1,3}$ and $B_{1,3}$ meet the merging conditions,

i) If the fusion condition is met, then H_1 and H_3 are merged and H_3 is deleted from H, H is updated to $H = \{H_1, H_2\}$, H_1 as the center of the new cluster; then N_3 is merged into N_1, the new cluster is still recorded as N_1, and $N = \{N_1, N_2\}$ is obtained, the data points in N_1 of the first cluster after merging are sorted according to the density in an ascending order; similarly, get $M = \{M_1, M_2\}$, update the label in M_1: update the merged cluster label M_3 to the label in M_1; recalculate the boundary points according to (6) and (7), and calculate the inter-cluster intersection density and inter-cluster boundary density to obtain inter-cluster intersection density matrix $A'_{2\times2}$ and inter-cluster boundary density matrix $B'_{2\times2}$ according to (8) and (9), as shown in Fig. 4.

According to experience, it is known that cluster fusion can not be continued, and there is no need to judge the cluster fusion conditions.

ii) If fusion condition is not met, index the second cluster N_2 and the third cluster N_3 in the set N, and judge whether $A_{2,3}$ and $B_{2,3}$ meet the merging conditions:

① If $A_{2,3} > B_{2,3}$, then H_2 and H_3 are merged and H_3 is deleted from H, H is updated to $H = \{H_1, H_2\}$, H_2 as the center of the new cluster; then N_3 is merged into N_2, the new cluster is still recorded as N_2, and $N = \{N_1, N_2\}$ is obtained, the data points in N_2 of the second cluster after merging are sorted according to the density in an ascending order; similarly, get $M = \{M_1, M_2\}$, update the label in M_2: update the merged cluster label M_3 to the label in M_2; recalculate the boundary points according to (6) and (7), and calculate the inter-cluster intersection density and inter-cluster boundary density to obtain inter-cluster intersection density matrix $\tilde{A}_{2\times2}$ and inter-cluster boundary density matrix $\tilde{B}_{2\times2}$ according to (8) and (9), as shown in Fig. 5.

According to experience, it is known that cluster fusion can not be continued, and there is no need to judge the cluster fusion conditions.

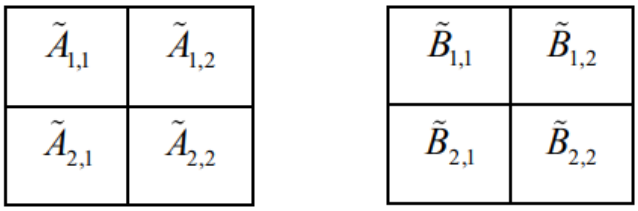


FIGURE 5. Inter-cluster intersection density matrix $\tilde{A}_{2\times2}$ and Inter-cluster boundary density matrix $\tilde{B}_{2\times2}$: (a)-(b).

② If $A_{2,3} < B_{2,3}$, without merging, there are still three clusters in the end.

C. CFDPC ALGORITHM FLOW

The steps of the CFDPC algorithm are as follows:

Algorithm 2 CFDPC Algorithm

Input: Dataset $S=\{x_1, x_2, \ldots, x_n\}$(where n is the total number of data points in the dataset S)

Output: $H = \{H_1, H_2, \ldots, H_l\}$, $N = \{N_1, N_2, \ldots, N_l\}$, $M = \{M_1, M_2, \ldots, M_l\}$ $(l < b)$(where b denotes number of initial clusters, and l denotes number of final clusters)

1: Calculate distance matrix $T_{n\times n}$
2: Selected percentage parameter p, determine cut-off distance d_c
3: Calculate d_c neighbors and calculate the local density and relative distance of each data point according to (2) and (3)
4: Determine candidate cluster centers according to (4) and (5)
5: Allocate the remaining data points according to DPC algorithm to obtain initial clusters: sort candidate cluster centers in descending order of density to obtain $H = \{H_1, H_2, \ldots, H_b\}$, get the initial cluster label $M = \{M_1, M_2, \ldots, M_b\}$ and the set of initial clusters $N = \{N_1, N_2, \ldots, N_b\}$, where N_i is sorted in ascending order of local density $(i = 1, 2, 3, \ldots, b)$. H, M and N correspond to each other
6: Selected percentage parameter q
7: Calculate the boundary points of clusters according to (6) and (7), get the inter-cluster border density according to (8), and finally get the border density matrix $B_{b\times b}$. If both the i-th cluster and the j-th cluster have no boundary points, then $B_{i,j} = 0$
8: The union of data points in the circle neighborhood with data point i in each cluster as the center and d_c as the radius is obtained, then according to (9), obtain inter-cluster intersection density. Finally, get the inter-cluster intersection density matrix $A_{b\times b}$. If the intersection of boundary points between the i-th cluster and j-th cluster is an empty set, then $A_{i,j} = 0$
9: **while** $A_{i,j} > B_{i,j}$ $(i < j, i = 1, 2, \ldots, b, j = i+1, i+2, \ldots, b)$ **do**
10: $H \leftarrow H - H[j]$ // Remove the j-th cluster center $H[j]$ from H, take the i-th cluster center $H[i]$ in H as the new cluster center of the merged two clusters
11: $N[i] \leftarrow N[i] \cup N[j]$ // Merge the data points in the j-th cluster $N[j]$ into the i-th cluster $N[i]$, and the i-th cluster $N[i]$ as a new cluster
$N \leftarrow N - N[j]$ // Remove the data point of the j-th cluster $N[j]$ from N, sort the data points in i-th cluster $N[i]$ by local density in descending order
12: $M[i] \leftarrow M[i] \cup M[j]$ // Merge the labels of the j-th cluster $M[j]$ into the labels of the i-th cluster $M[i]$, and the label of the merged cluster is updated to the label of the i-th cluster
$M \leftarrow M - M[j]$ // Remove the labels of the j-th cluster $M[j]$ from M
13: Update the set H, N, M respectively
14: $b = b - 1$
15: turn to 7-11// Perform recursion from 7 to 11
16: **end while**
17: **Return** $H = \{H_1, H_2, \ldots, H_l\}$, $N = \{N_1, N_2, \ldots, N_l\}$, $M = \{M_1, M_2, \ldots, M_l\}$ $(l < b)$

D. COMPLEXITY ANALYSIS OF CFDPC ALGORITHM

The computational complexity of the CFDPC algorithm in this paper is mainly composed of the following five parts: (1) the complexity of calculating the distance matrix $T_{n\times n}$ is $O(n^2)$; (2) the complexity of calculating the local density of data points is $O(n)$; (3) the complexity of calculating the relative distance is $O(n^2)$; (4) the complexity of selecting candidate cluster centers and assigning remaining points is $O(n)$; (5) perform cluster fusion: including (a) the number of boundary points of each cluster is less than n in theory, so the complexity of computing the boundary points of clusters is $O(n)$, (b) the number of samples in the intersection between clusters is much less than n, so the complexity of computing the inter-cluster intersection density matrix and the inter-cluster boundary density matrix is much less than $O(n^2)$, (c) the complexity of processing the cluster fusion is $O(n)$. To sum up, the computational complexity of CFDPC algorithm in this paper is $O(n^2)$, which is the same as that of DPC algorithm. The computational complexity of the four algorithms compared with the CFDPC algorithm in this paper is as follows: the computational complexity of DPC-KNN algorithm is $O(n^2)$; the computational complexity of DPC algorithm is $O(n^2)$; the computational complexity of K-means algorithm is $O(n)$; the computational complexity of DBSCAN algorithm is $O(n^2)$.

V. EXPERIMENTAL RESULTS AND ANALYSIS

All algorithms in this study are implemented using Python tools. To evaluate the clustering performance of the CFDPC algorithm proposed in this paper, it is compared with four more advanced algorithms DPC-KNN[13], DPC[12], K-means[2] and DBSCAN[5] algorithms under different datasets. These four algorithms are reproduced in Python by referring to the original literature.

TABLE 1. Artificial datasets.

Dataset	Sample	Dimension	Clusters
Flame	240	2	2
Spiral	321	2	2
R15	600	2	15
Jain	373	2	2
Skewed	1000	2	5
Asymmetric	1000	2	6
S1	5000	2	15
Compound	399	2	6
Aggregation	788	2	7

TABLE 2. Real datasets.

Dataset	Sample	Dimension	Clusters
Iris	150	4	3
Wdbc	569	30	2
Segment	2310	19	7
Yeast	1484	8	10
Vertebral	310	6	2
Ecoli	336	7	8

A. EXPERIMENTAL ENVIRONMENT

Hardware Environment: Personal computer with 2.4 GHz CPU, 12GB memory, and 500GB hard disk.

Software environment: Windows 11 Professional Edition 64-bit operating system, implemented in Python 3.9.7 environment.

B. CLUSTER EVALUATION INDEX

In order to compare the performance of different clustering algorithms, this paper selects four common clustering metrics in clustering: Accuracy Rate (ACC), Adjusted Mutual Information (AMI) [39], Adjusted Rand Index (ARI) [39], and Fowler-Males Index (FMI) [40] to evaluate the clustering performance of each algorithm. The maximum values of the four indicators are 1. The larger the indicator value, the better the clustering accuracy of the algorithm, and the stronger the clustering performance. The accuracy rate is defined as follows:

$$ACC = \frac{\sum_{i}^{m} l_i}{n} \tag{11}$$

where m represents the number of clusters, l_i represents the number of data points correctly classified into the corresponding cluster C_i, n represents the total number of data points, and ACC represents the proportion of the number of correctly classified data points to the total number of data points.

C. EXPERIMENTAL DATASETS

To verify the effectiveness of the CFDPC algorithm, we use 15 commonly used experimental datasets [41]-[44], and select nine artificial datasets from http://cs.joensuu.fi/sipu/datasets/ and six real datasets from http://archive.ics.uci.edu/ml, as listed in Table 1 and Table 2.

D. ALGORITHM PARAMETER SETTINGS

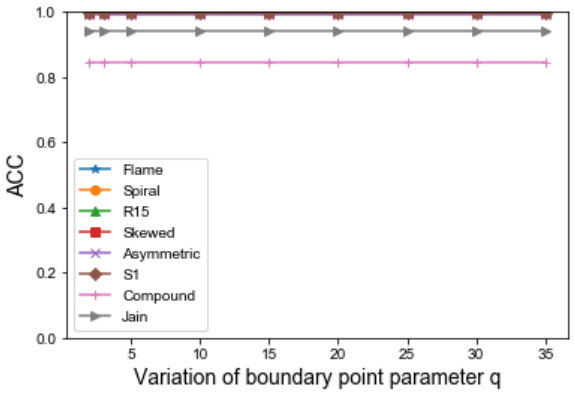
Table 3 lists the parameter settings of each algorithm for the 30 experiments on the above datasets to obtain the optimal clustering performance. Among them, the percentage parameter p of the CFDPC algorithm in this study is not specified, and the boundary point parameter q is generally selected within 5-20 according to experience. The optimal value of the parameter k in the DPC-KNN algorithm is selected in the range of 2-35; the value range of the parameter p in the DPC algorithm is not specified; the K-means algorithm iterates 100 times to obtain the optimal result on the premise of determining K clusters; the parameters required by the DBSCAN algorithm: neighborhood radius ε and the minimum number of samples in the neighborhood Minpts select the best value based on experience.

TABLE 3. Experimental parameter values.

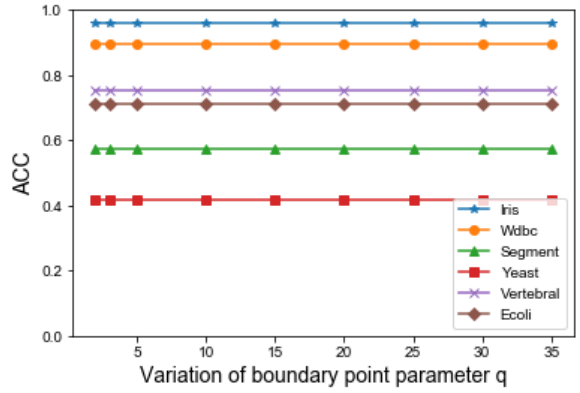
Data	CFDPC	DPC-KNN	DPC	K-means	DBSCAN
Flame	1.5/5	5	6	2	0.9/5
Spiral	2.4/5	7	2	3	1.3/3
R15	1.5/20	7	2	15	0.3/4
Jain	1.5/5	8	0.5	2	2.6/5
Skewed	1.9/10	3	0.5	6	70/40
Asymmetric	1.1/20	4	3	5	40/5
S1	0.95/10	6	1	15	40000/31
Compound	0.56/10	4	0.8	6	0.9/2
Aggregation	0.5/40	5	0.2	7	1/2
Iris	2.4/5	4	1.5	3	0.7/2
Wdbc	1.43/5	4	2	2	9.2/4
Segment	0.33/5	3	0.29	7	2/3
Yeast	0.081/20	30	1.3	10	6/6
Vertebral	0.5/5	13	0.29	2	8/5
Ecoli	0.28/10	17	0.4	8	0.11/4

E. ANALYSIS OF THE VALUE RANGE OF THE BOUNDARY POINT PARAMETER q

In this section, we will analyze and discuss the effect of the proposed CFDPC algorithm on ACC under different settings of the boundary point parameter q. The boundary point parameter q is a hyper-parameter in the CFDPC algorithm, and different settings of the parameter q have an important impact on the clustering effect. For different datasets, the size of the data volume is different. For a dataset with a small number of samples, when multiple initial clusters are generated, since the number of samples in the entire dataset is small, the number of samples in each initial cluster obtained will be less. In order to meet the cluster merging conditions, it is even possible to make cluster merging as meaningful as possible, so make each cluster have boundary points as far as possible. This paper conducts multiple experiments on the selected dataset to verify that it is more reasonable to set the lower limit of the boundary parameter q is 5; for datasets with a large number of samples, each initial cluster will contain more samples, and the number of new cluster samples and boundary points obtained by cluster merging will also increase. To ensure that cluster merging continues, the number of boundary points in each cluster should not exceed half of the number of clusters to which it belongs. Similarly, this paper conducts multiple experiments on the selected dataset to verify that the upper limit of the boundary point parameter q is set to 20 is reliable. But this is not absolute,



(a) ACC of CFDPC algorithm on synthetic datasets



(b) ACC of CFDPC algorithm on real datasets

FIGURE 6. Under the condition of fixed parameter p and for different parameter q, the ACC of CFDPC algorithm under synthetic and real datasets: (a)-(b).

TABLE 4. Comparison of clustering results on artificial datasets.

Data	Index	CFDPC	DPC-KNN	DPC	K-means	DBSCAN
Flame	ACC	**1**	**1**	**1**	0.854	0.967
	ARI	**1**	**1**	**1**	0.500	0.841
	AMI	**1**	**1**	**1**	0.460	0.755
	FMI	**1**	**1**	**1**	0.759	0.922
Spiral	ACC	**1**	**1**	**1**	0.346	**1**
	ARI	**1**	**1**	**1**	-0.006	**1**
	AMI	**1**	**1**	**1**	-0.005	**1**
	FMI	**1**	**1**	**1**	0.328	**1**
R15	ACC	**0.997**	**0.997**	**0.997**	0.807	0.923
	ARI	**0.993**	**0.993**	**0.993**	0.814	0.856
	AMI	**0.994**	**0.994**	**0.994**	0.923	0.901
	FMI	**0.993**	**0.993**	**0.993**	0.830	0.866
Jain	ACC	**0.944**	0.922	0.922	0.786	0.807
	ARI	**0.768**	0.710	0.710	0.324	0.256
	AMI	**0.685**	0.644	0.644	0.386	0.238
	FMI	**0.918**	0.878	0.878	0.710	0.803
Skewed	ACC	**0.995**	0.721	0.990	0.787	0.838
	ARI	**0.988**	0.572	0.978	0.664	0.630
	AMI	**0.984**	0.754	0.974	0.740	0.743
	FMI	**0.990**	0.672	0.981	0.720	0.690
Asymmetric	ACC	**0.994**	0.734	0.990	0.989	0.927
	ARI	**0.985**	0.642	0.975	0.973	0.896
	AMI	**0.981**	0.781	0.968	0.966	0.884
	FMI	**0.988**	0.742	0.980	0.978	0.917
S1	ACC	**0.995**	0.284	**0.995**	0.994	0.985
	ARI	**0.989**	0.212	**0.989**	0.986	0.975
	AMI	**0.989**	0.567	**0.989**	0.986	0.972
	FMI	**0.990**	0.403	**0.990**	0.987	0.968
Compound	ACC	**0.845**	0.747	0.642	0.789	0.832
	ARI	0.797	0.613	0.535	0.767	**0.872**
	AMI	**0.879**	0.746	0.730	0.792	0.860
	FMI	0.853	0.703	0.640	0.827	**0.907**
Aggregation	ACC	**0.999**	0.996	0.996	0.864	0.798
	ARI	**0.998**	0.994	0.994	0.735	0.775
	AMI	**0.996**	0.989	0.989	0.835	0.852
	FMI	**0.998**	0.995	0.995	0.792	0.836

not all datasets have boundary point parameters in the range of [5,20]. There are also some special cases: For example, for the Aggregation dataset, when $q = 40$, the algorithm takes to the optimal result. (This also satisfies that the number of boundary points does not exceed half the amount of data points in the cluster, which makes cluster merging meaningful and is allowed).

Therefore, we focus on analyzing the variation of the ACC of the CFDPC algorithm in the range of $q \in [5, 20]$, as shown in Fig. 6. In the case where the parameter p in Table 3 is set, we have respectively made a line chart of the change of ACC in the synthetic datasets and the real datasets when the boundary point parameter q takes different values. We can clearly see the ACC index values of the CFDPC algorithm in Figure. 6 tend to be a straight line. According to Fig. 6(a), we can see that the ACC index values of the Flame, Spiral, R15, Skewed, Asymmetric, and S1 datasets are almost highly coincident. When the appropriate parameter p is selected, it can be intuitively observed that with the change of the boundary point parameter q, the value of ACC is not affected, further illustrating the excellent robustness and stability of the algorithm. Although two parameters need to be set in this paper when the appropriate parameter p is selected, the parameter q has little influence on the clustering results, and most datasets obtain high ACC values under the CFDPC algorithm. If the parameters of the algorithm are searched in a large range, it will not only affect the search efficiency of the optimal parameters, but also makes the stability of the algorithm worse, and it is difficult to achieve the ideal clustering effect. If the algorithm can take parameter values in a small range and achieve the ideal clustering effect, this is what we pursue. Therefore, the CFDPC algorithm is adjusted in the range of $q \in [5, 20]$ to achieve the ideal clustering effect suitable for most datasets.

F. EXPERIMENTAL RESULTS ON ARTIFICIAL DATASETS

To evaluate the universality and robustness of the CFDPC algorithm proposed in this study, four clustering performances of the five algorithms are compared on nine artificial datasets with different data structures. The performance evaluation results are listed in Table 4 (bold data in the table represent the best clustering results), and the clustering effect is shown in Figs. 7-15.

Combined with Table 4 and Figs. 7-15, by analyzing the four index values of ACC, AMI, ARI, and FMI of the five algorithms on different datasets, it can be observed that except

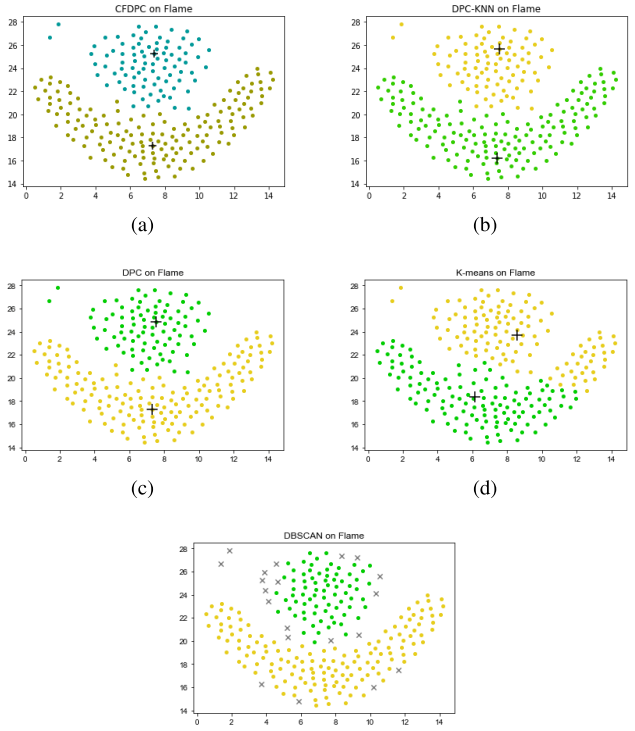


FIGURE 7. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Flame dataset: (a)-(e).

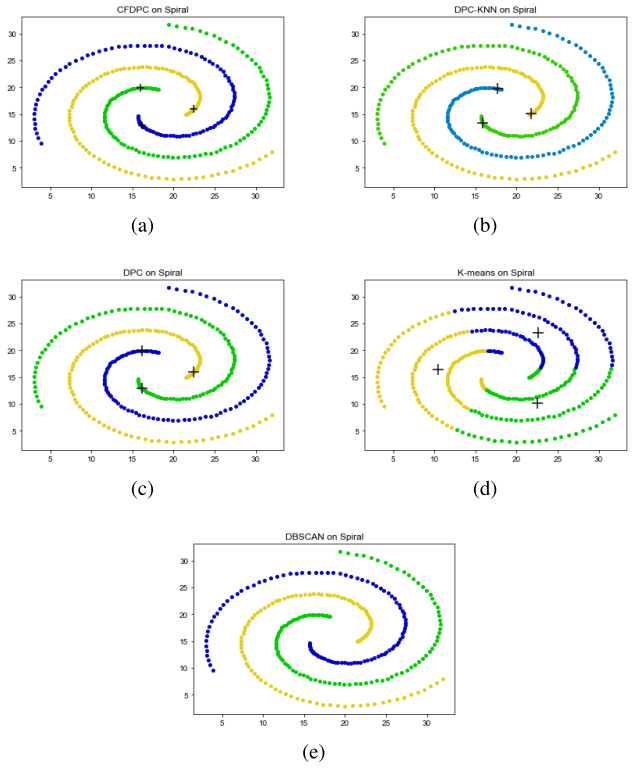


FIGURE 8. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Spiral dataset: (a)-(e).

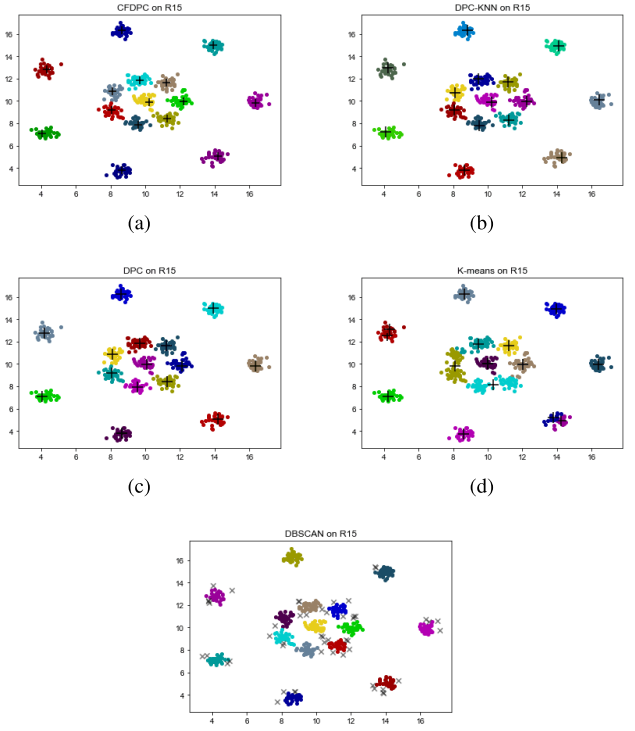


FIGURE 9. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on R15 dataset: (a)-(e).

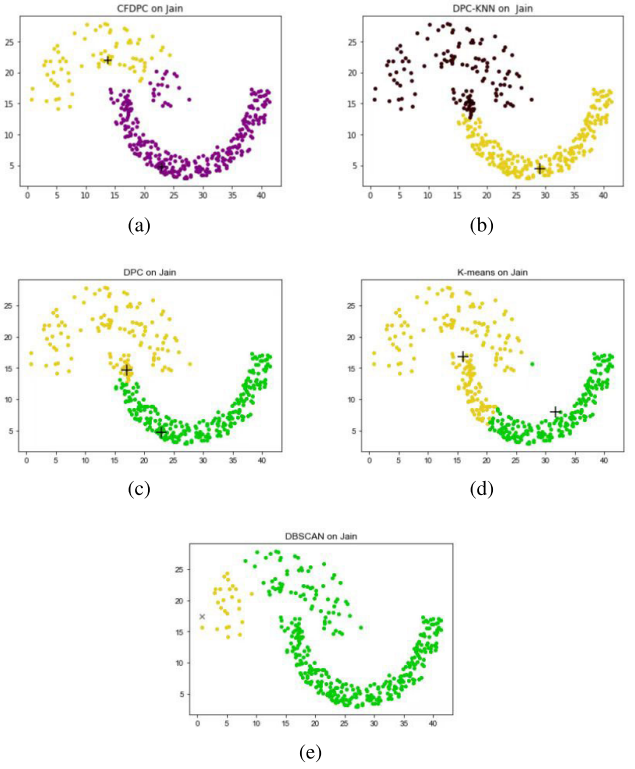


FIGURE 10. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Jain dataset: (a)-(e).

for the K-means algorithm, the other algorithms on the Spiral dataset can cluster accurately. This is because the K-means algorithm is significantly affected by the initial clustering centers when processing spiral data, and it is not suitable for non-convex datasets. On the Flame and R15 datasets, the performances of the CFDPC, DPC-KNN, and DPC

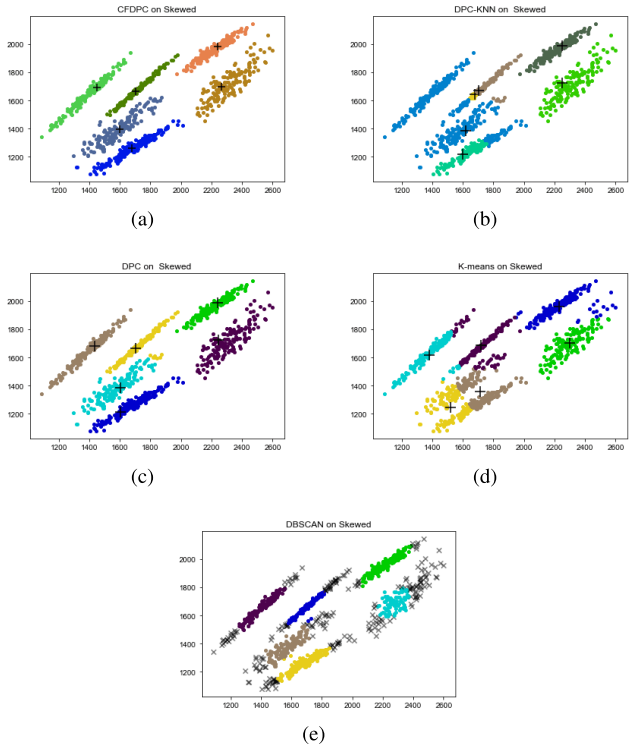


FIGURE 11. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Skewed dataset: (a)-(e).

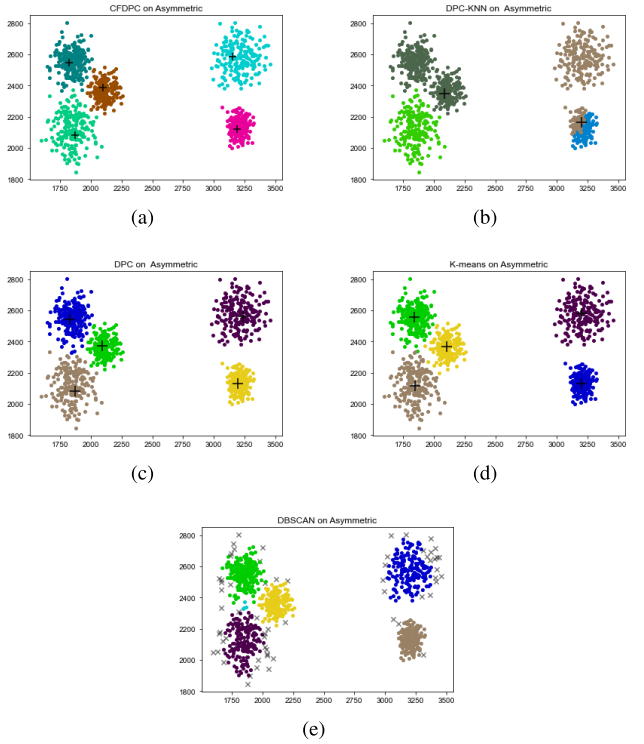


FIGURE 12. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Asymmetric dataset: (a)-(e).

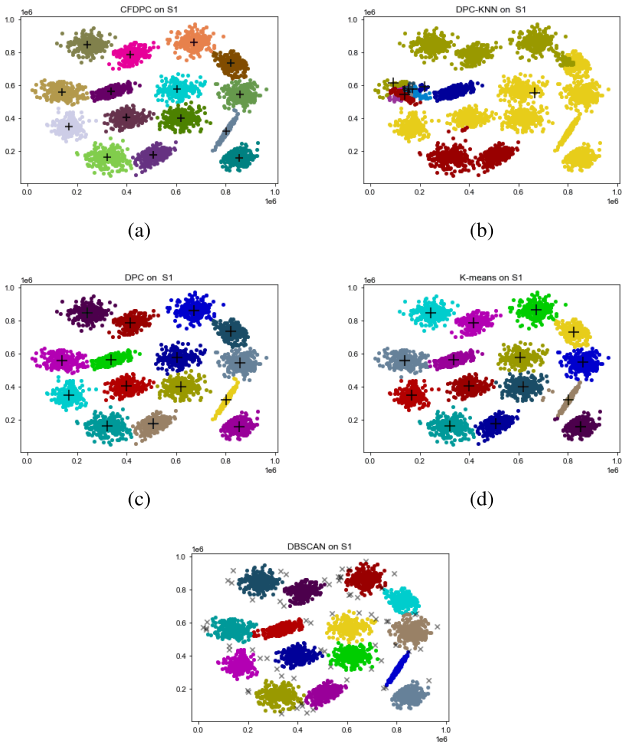


FIGURE 13. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on S1 dataset: (a)-(e).

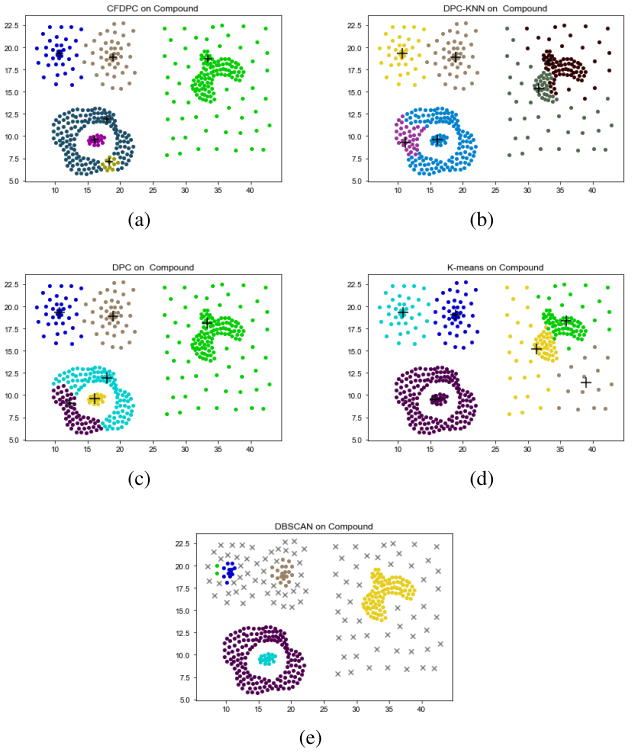


FIGURE 14. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Compound dataset: (a)-(e).

algorithms are better than those of the K-means and DBSCAN algorithms. The main reason for this is that the K-means algorithm has weak processing ability for non-spherical data, so the sample points can not be classified correctly. However, the DBSCAN algorithm can be seen from the rendering that identifying some data points as noise points leads to a low clustering performance. For the S1 dataset,

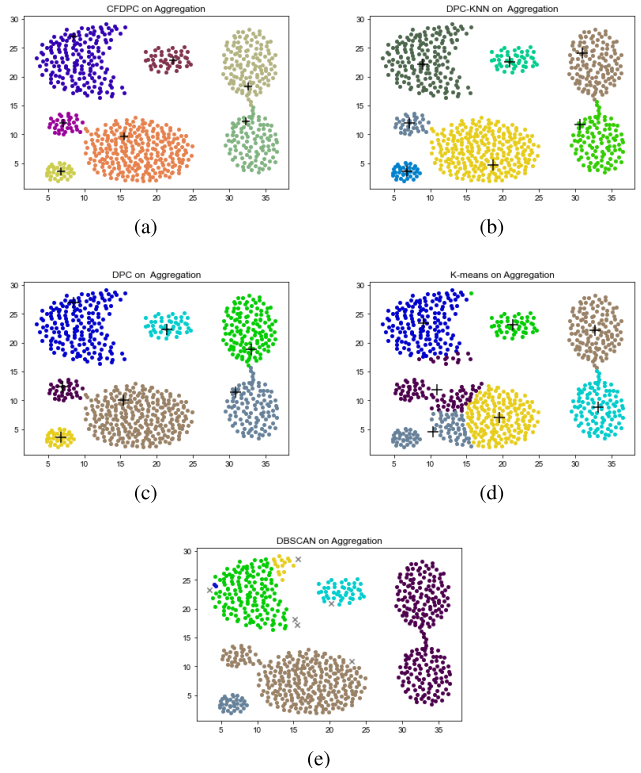


FIGURE 15. Rendering of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on Aggregation dataset: (a)-(e).

the CFDPC and DPC algorithms show the best clustering effect, and the DPC-KNN algorithm is the least effective. This is because the sample points are not normalized, and the data dimensions differed significantly. It can also be clearly observed from the rendering that almost all the cluster centers of the 15 clusters are clustered together, and even some single data points are regarded as a cluster. It is difficult for the DBSCAN algorithm to select parameters in the S1 dataset, and the clustering effect is inferior to those of the CFDPC and DPC algorithms. On the Compound dataset, only the CFDPC and DBSCAN algorithms achieve excellent clustering results. The CFDPC algorithm shows the best clustering performance on Aggregation, Skewed, Jain, and Asymmetric datasets.

The above analyses show that the CFDPC algorithm can achieve the same or even higher clustering accuracy as the DPC algorithm, and the overall performance of the CFDPC algorithm on artificial datasets is better than that of the other four algorithms.

G. EXPERIMENTAL RESULTS ON REAL DATASETS

To further test the clustering performance of the CFDPC algorithm, CFDPC algorithm is compared with four other algorithms on six real datasets with different structures and dimensions. The clustering evaluation results are shown in Table 5 (bold data in the table represents the best clustering results) and Fig.16.

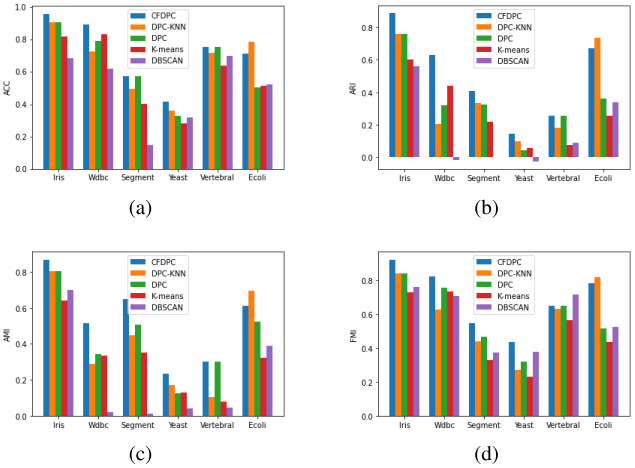


FIGURE 16. Cluster evaluation index comparison chart of CFDPC, DPC-KNN, DPC, K-means, DBSCAN algorithm on real datasets: (a)-(d).

TABLE 5. Comparison of clustering results on real datasets.

Data	Index	CFDPC	DPC-KNN	DPC	K-means	DBSCAN
Iris	ACC	**0.960**	0.907	0.907	0.820	0.687
	ARI	**0.886**	0.759	0.759	0.601	0.562
	AMI	**0.869**	0.803	0.803	0.642	0.701
	FMI	**0.923**	0.841	0.841	0.732	0.760
Wdbc	ACC	**0.896**	0.728	0.791	0.833	0.619
	ARI	**0.627**	0.203	0.318	0.441	-0.015
	AMI	**0.514**	0.290	0.342	0.334	0.019
	FMI	**0.822**	0.627	0.756	0.735	0.710
Segment	ACC	**0.573**	0.494	0.572	0.401	0.148
	ARI	**0.408**	0.335	0.324	0.219	0.000
	AMI	**0.650**	0.449	0.505	0.350	0.011
	FMI	**0.548**	0.441	0.465	0.331	0.374
Yeast	ACC	**0.416**	0.362	0.330	0.280	0.319
	ARI	**0.144**	0.098	0.044	0.058	-0.025
	AMI	**0.234**	0.170	0.124	0.131	0.043
	FMI	**0.435**	0.271	0.321	0.230	0.378
Vertebral	ACC	**0.755**	0.719	**0.755**	0.639	0.697
	ARI	**0.257**	0.181	**0.257**	0.074	0.091
	AMI	**0.302**	0.105	0.301	0.078	0.044
	FMI	0.652	0.633	0.652	0.565	**0.717**
Ecoli	ACC	0.711	**0.786**	0.503	0.515	0.524
	ARI	0.670	**0.736**	0.362	0.254	0.339
	AMI	0.612	**0.695**	0.524	0.322	0.391
	FMI	0.782	**0.821**	0.516	0.434	0.526

By analyzing the four evaluation index values of each algorithm in Table 5 on different datasets, it can be observed that both the CFDPC and DPC algorithms achieve the best results when processing the Vertebral dataset, the performance of CFDPC on the Ecoli dataset is slightly worse than that of the DPC-KNN algorithm. This is because the density distribution of the Ecoli dataset is not uniform, and there are cross-entanglement and overlapping among clusters, which results in a deviation between the data selected by the cut-off parameter p and the data selected by the k-nearest neighbor so that the CFDPC algorithm does not reach the optimum in these four evaluation index values. But its performance is still better than that of the DPC algorithm. Some index values of the DBSCAN algorithm on the Wdbc and Yeast datasets are negative, mainly because the distribution of Wdbc and Yeast data points is uneven and sparse, which makes it difficult for the DBSCAN algorithm to adjust the

parameters. Second, some data points are marked as noise points, resulting in a large deviation in the clustering results. The K-means algorithm is not suitable for dealing with non-linear separable datasets owing to its own property, which leads to poor clustering results. For processing the Iris, Wdbc, Segment, and Yeast datasets, the CFDPC algorithm shows the best clustering performance, outperforming the other four clustering algorithms. Combined with the comparison chart of the clustering evaluation indicators in Fig. 16, the overall strong clustering performance and universality of the CFDPC algorithm are displayed more clearly and intuitively.

VI. CONCLUSION

For a dataset with a more complex structure, the DPC algorithm can not automatically select accurate cluster centers through the decision diagram, and allocation errors are prone to occur in the process of allocating the remaining points. Based on the DPC algorithm, this study proposes a new density peak clustering algorithm based on cluster fusion strategy(CFDPC). First, to avoid the influence of human subjectivity, two new thresholds are set to obtain candidate clustering centers. Second, the initial clusters fusion strategy is adopted to improve the joint and several errors in the allocation process of the DPC algorithm and improve the clustering accuracy of the algorithm. Finally, the CFDPC algorithm proposed in this study is compared with the DPC-KNN, DPC, K-means, and DBSCAN algorithms on artificial and real datasets. The experimental results show that the CFDPC algorithm can not only accurately find the cluster centers and is suitable for processing arbitrarily shaped datasets, but also has optimal clustering performance.

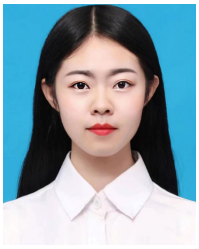
For future work, we will explore and solve the following problems. Firstly, the algorithm does not automatically provide two parameter settings. The focus of future research is to explore non-parametric algorithms to make the algorithm more intelligent. Secondly, CFDPC algorithm has high complexity in calculating the distance matrix. We will continue to explore a new distance calculation method for sparse search to reduce the computational complexity of the algorithm. Finally, we try to extend the CFDPC algorithm to large-scale datasets, high-dimensional datasets, and manifold datasets to improve the practical application ability of clustering algorithm.

REFERENCES

[1] A. N. Srivastava, "Data mining: Concepts, models, methods, and algorithms," *J. Comput. Inf. Sci. Eng.*, vol. 5, no. 4, pp. 394–395, Dec. 2005, doi: 10.1115/1.2123107.
[2] A. K. Jain, "Data clustering: 50 years beyond *K*-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, 2010, doi: 10.1016/j.patrec.2009.09.011.
[3] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *J. Cybern.*, vol. 3, no. 3, pp. 32–57, Sep. 1973, doi: 10.1080/01969727308546046.
[4] J. C. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algoritms*. New York, NY, USA: Plenum, 1981.
[5] T. Lei, X. Jia, Y. Zhang, S. Liu, H. Meng, and A. K. Nandi, "Superpixel-based fast fuzzy C-Means clustering for color image segmentation," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 9, pp. 1753–1766, Sep. 2019, doi: 10.1109/TFUZZ.2018.2889018.
[6] Y. Tang, F. Ren, and W. Pedrycz, "Fuzzy C-means clustering through SSIM and patch for image segmentation," *Appl. Soft Comput.*, vol. 87, Feb. 2020, Art. no. 105928, doi: 10.1016/j.asoc.2019.105928.
[7] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview," *Wiley Interdiscipl. Rev. Data Mining Knowl. Discovery*, vol. 2, no. 1, pp. 86–97, 2012, doi: 10.1002/widm.53.
[8] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview, II," *Wires Data Mining Knowl. Discovery*, vol. 7, no. 6, Sep. 2017, doi: 10.1002/widm.1219.
[9] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A desity-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining*, Portland, OR, USA, 1996, pp. 226–231.
[10] W. Wang, J. Yang, and R. Muntz, "STING: A statistical information grid approach to spatial data mining," in *Proc. 23rd Intl. Conf. Very Large Data Bases.*, Athens, Greece, 1997, pp. 186–195.
[11] T. Chen, N. L. Zhang, T. Liu, K. M. Poon, and Y. Wang, "Model-based multidimensional clustering of categorical data," *Artif. Intell.*, vol. 176, no. 1, pp. 2246–2269, 2012, doi: 10.1016/j.artint.2011.09.003.
[12] D. Peel and G. McLachlan, *Finite Mixture Models*. New York, NY, USA: Wiley, 2000.
[13] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014, doi: 10.1126/science.1242072.
[14] J. Gao, M. Kang, J. Tian, L. Wu, and M. Pecht, "Unsupervised locality-preserving robust latent low-rank recovery-based subspace clustering for fault diagnosis," *IEEE Access*, vol. 6, pp. 52345–52354, 2018, doi: 10.1109/ACCESS.2018.2869923.
[15] Y.-W. Chen, D.-H. Lai, H. Qi, J.-L. Wang, and J.-X. Du, "A new method to estimate ages of facial image for large database," *Multimed. Tools Appl.*, vol. 75, no. 5, pp. 2877–2895, Feb. 2016, doi: 10.1007/s11042-015-2485-9.
[16] J. Zheng, S. Wang, D. Li, and B. Zhang, "Personalized recommendation based on hierarchical interest overlapping community," *Inf. Sci.*, vol. 479, pp. 55–75, Apr. 2019, doi: 10.1016/j.ins.2018.11.054.
[17] X. Li and K.-C. Wong, "Evolutionary multiobjective clustering and its applications to patient stratification," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1680–1693, May 2019, doi: 10.1109/TCYB.2018.2817480.
[18] M. Du, S. Ding, and H. Jia, "Study on density peaks clustering based on *K*-nearest neighbors and principal component analysis," *Knowl.-Based Syst.*, vol. 99, pp. 135–145, May 2016, doi: 10.1016/j.knosys.2016.02.001.
[19] R. Liu, H. Wang, and X. Yu, "Shared-nearest neighbor based clustering by fast search and find of density peaks," *Inf. Sci.*, vol. 450, pp. 200–2226, Jun. 2018, doi: 10.1016/j.ins.2018.03.031.
[20] Y. Liu, D. Liu, F. Yu, and Z. Ma, "A double-density clustering method based on 'nearest to first in' strategy," *Symmetry*, vol. 12, no. 5, p. 747, May 2020, doi: 10.3390/sym12050747.
[21] J. Chen and P. S. Yu, "A domain adaptive density clustering algorithm for data with varying density distribution," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 6, pp. 2310–2321, Jun. 2021, doi: 10.1109/TKDE.2019.2954133.
[22] D. Jiang, W. Zang, R. Sun, Z. Wang, and X. Liu, "Adaptive density peaks clustering based on *K*-nearest neighbor and Gini coefficient," *IEEE Access*, vol. 8, pp. 113900–113917, 2020, doi: 10.1109/ACCESS.2020.3003057.
[23] X. Xu, S. Ding, Y. Wang, L. Wang, and W. Jia, "A fast density peaks clustering algorithm with sparse search," *Inf. Sci.*, vol. 554, pp. 61–83, Apr. 2021, doi: 10.1016/j.ins.2020.11.050.
[24] X. Xu, S. Ding, and Z. Shi, "An improved density peaks clustering algorithm with fast finding cluster centers," *Knowl.-Based Syst.*, vol. 158, pp. 65–74, Oct. 2018, doi: 10.1016/j.knosys.2018.05.034.
[25] Y. Shan, S. Li, F. Li, Y. Cui, S. Li, M. Zhou, and X. Li, "A density peaks clustering algorithm with sparse search and K-d tree," *IEEE Access*, vol. 10, pp. 74883–74901, 2022, doi: 10.1109/ACCESS.2022.3190958.
[26] K. G. Flores and S. E. Garza, "Density peaks clustering with gap-based automatic center detection," *Knowl.-Based Syst.*, vol. 206, Oct. 2020, Art. no. 106350, doi: 10.1016/j.knosys.2020.106350.
[27] Y. Lv, M. D. Liu, and Y. Xiang, "Fast searching density peak clustering algorithm based on shared nearest neighbor and adaptive clustering center," *Symmetry*, vol. 12, no. 12, pp. 394–395, Dec. 2020, doi: 10.3390/sym12122014.

[28] J. Xie, H. Gao, W. Xie, X. Liu, and P. W. Grant, "Robust clustering by detecting density peaks and assigning points based on fuzzy weighted K-nearest neighbors," *Inf. Sci.*, vol. 354, pp. 19–40, Aug. 2016, doi: 10.1016/j.ins.2016.03.011.
[29] A. Lotfi, P. Moradi, and H. Beigy, "Density peaks clustering based on density backbone and fuzzy neighborhood," *Pattern Recognit.*, vol. 107, Nov. 2020, Art. no. 107449, doi: 10.1016/j.patcog.2020.107449.
[30] D. Yu, G. Liu, M. Guo, X. Liu, and S. Yao, "Density peaks clustering based on weighted local density sequence and nearest neighbor assignment," *IEEE Access*, vol. 7, pp. 34301–34317, 2019, doi: 10.1109/ACCESS.2019.2904254.
[31] X. Yang, Z. Cai, R. Li, and W. Zhu, "GDPC: Generalized density peaks clustering algorithm based on order similarity," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 3, pp. 719–731, Mar. 2021, doi: 10.1007/s13042-020-01198-0.
[32] H. Yu, L. Chen, and J. Yao, "A three-way density peak clustering method based on evidence theory," *Knowl.-Based Syst.*, vol. 211, Jan. 2021, Art. no. 106532, doi: 10.1016/j.knosys.2020.106532.
[33] F. Fang, L. Qiu, and S. Yuan, "Adaptive core fusion-based density peak clustering for complex data with arbitrary shapes and densities," *Pattern Recognit.*, vol. 107, Nov. 2020, Art. no. 107452, doi: 10.1016/j.patcog.2020.107452.
[34] L. Sun, X. Qin, W. Ding, J. Xu, and S. Zhang, "Density peaks clustering based on k-nearest neighbors and self-recommendation," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 7, pp. 1913–1938, Mar. 2021, doi: 10.1007/s13042-021-01284-x.
[35] L. Yaohui, M. Zhengming, and Y. Fang, "Adaptive density peak clustering based on K-nearest neighbors with aggregating strategy," *Knowl.-Based Syst.*, vol. 133, pp. 208–220, Oct. 2017, doi: 10.1016/j.knosys.2017.07.010.
[36] X. Yuan, H. Yu, J. Liang, and B. Xu, "A novel density peaks clustering algorithm based on k nearest neighbors with adaptive merging strategy," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 10, pp. 2825–2841, Aug. 2021, doi: 10.1007/s13042-021-01369-7.
[37] J. Liu and C. Zhao, "Density gain-rate peaks for spectral clustering," *IEEE Access*, vol. 9, pp. 46000–46010, 2021, doi: 10.1109/ACCESS.2021.3066498.
[38] L. Wang, Z. Hao, and W. Sun, "A novel self-adaptive affinity propagation clustering algorithm based on density peak theory and weighted similarity," *IEEE Access*, vol. 7, pp. 175106–175115, 2019, doi: 10.1109/ACCESS.2019.2956963.
[39] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance," *J. Mach. Learn. Res.*, vol. 11, pp. 2837–2854, Jan. 2010.
[40] E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *J. Amer. Statist. Assoc.*, vol. 78, no. 383, pp. 553–569, 1983, doi: 10.2307/2288117.
[41] C. Ren, L. Sun, Y. Yu, and Q. Wu, "Effective density peaks clustering algorithm based on the layered K-Nearest neighbors and subcluster merging," *IEEE Access*, vol. 8, pp. 123449–123468, 2020, doi: 10.1109/ACCESS.2020.3006069.
[42] Z. Bian, F.-L. Chung, and S. Wang, "Fuzzy density peaks clustering," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 7, pp. 1725–1738, Jul. 2021, doi: 10.1109/TFUZZ.2020.2985004.
[43] Z. Zhang, Q. Zhu, F. Zhu, J. Li, D. Cheng, Y. Liu, and J. Luo, "Density decay graph-based density peak clustering," *Knowl.-Based Syst.*, vol. 224, Jul. 2021, Art. no. 107075, doi: 10.1016/j.knosys.2021.107075.
[44] L. Sun, R. Liu, J. Xu, and S. Zhang, "An adaptive density peaks clustering method with Fisher linear discriminant," *IEEE Access*, vol. 7, pp. 72936–72955, 2019, doi: 10.1109/ACCESS.2019.2918952.

FUXIANG LI was born in 1972. He received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China. He is currently a Professor with the School of Science, Harbin University of Science and Technology, China. He has published multiple SCI indexed papers in the international high-level journals. His research interests include nonlinear numerical analysis, computational mathematics, and machine learning.



MING ZHOU was born in 1997. She is currently pursuing the M.S. degree with the School of Science, Harbin University of Science and Technology, Harbin, China. Her current research interests include machine learning, mathematics of computation, and nonlinear numerical analysis.

SHU LI was born in 1980. He received the Ph.D. degree from Tianjin University, Tianjin, China. He is currently a Professor with the School of Electrical and Electronic Engineering, Harbin University of Science and Technology, China. He has published over 20 SCI indexed articles in the internationally renowned journals. He also has obtained six software copyrights. His research interests include machine learning, data mining, and modeling and calculation for phase transition theory.

TIANHAO YANG was born in 1998. He is currently pursuing the M.S. degree with the School of Science, Harbin University of Science and Technology, Harbin, China. His current research interests include inverse problems and mathematics of computation.