Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 6393652, 15 pages
https://doi.org/10.1155/2017/6393652

Research Article

Fast Density Clustering Algorithm for Numerical Data and Categorical Data

Chen Jinyin,[1] He Huihao,[1] Chen Jungan,[2] Yu Shanqing,[1] and Shi Zhaoxia[1]

[1]*Zhejiang University of Technology, Zhejiang 310023, China*
[2]*Electrical Engineering Department, Ningbo Wanli University, Ningbo 310023, China*

Correspondence should be addressed to Chen Jinyin; chenjinyin@zjut.edu.cn

Received 20 August 2016; Revised 2 January 2017; Accepted 15 January 2017; Published 26 March 2017

Academic Editor: Erik Cuevas

Data objects with mixed numerical and categorical attributes are often dealt with in the real world. Most existing algorithms have limitations such as low clustering quality, cluster center determination difficulty, and initial parameter sensibility. A fast density clustering algorithm (FDCA) is put forward based on one-time scan with cluster centers automatically determined by center set algorithm (CSA). A novel data similarity metric is designed for clustering data including numerical attributes and categorical attributes. CSA is designed to choose cluster centers from data object automatically which overcome the cluster centers setting difficulty in most clustering algorithms. The performance of the proposed method is verified through a series of experiments on ten mixed data sets in comparison with several other clustering algorithms in terms of the clustering purity, the efficiency, and the time complexity.

1. Introduction

As one of the most important techniques in data mining, clustering is to partition a set of unlabeled objects into clusters, where the objects which fall into the same cluster have more similarities than others [1]. Clustering algorithms have been developed and applied to various fields including text analysis, customer segmentation, and image recognition. They are also useful in our daily life, since massive data with mixed attributes are now emerging. Typically, these data contain both numeric and categorical attributes [2, 3]. For example, the analysis of an applicant for a credit card would involve data of age (integers), income (float), marital status (categorical), and so forth, forming a typical example of data with mixed attributes.

Up to now, most research on data clustering has been focusing on either numeric or categorical data instead of both types of attributes. K-means [4], BIRCH [5], DBSCAN [6], K-modes [7], fuzzy K-modes [8], BFCM [9], COOLCAT [10], TCGA [11], AS′ fuzzy k-modes [12], and k-means based method [13] are classic clustering algorithms. K-means clustering algorithm [4] is put forward based on partition, where k cluster centers need to be initialized by users or experience. Initialized cluster centers number k could decide the clustering purity and efficiency. BIRCH [5] is short for balanced iterative reducing and clustering using hierarchies. Clustering feature and clustering feature trees are adopted to describe cluster specifically. Two stages are defined to implement BIRCH, including database scanning to build a clustering feature tree and global clustering to improve purity and efficiency. DBSCAN [6] (Density-Based Spatial Clustering of Applications with Noise) is a classic density-based clustering algorithm, which is capable of dealing with data with noise. Compared with K-means, DBSCAN does not need to set cluster numbers priorly. However, two sensitive parameters are essential for DBSCAN, which are eps and minPts. Until now, various revised DBSCANs are brought up to improve the performance of DBSCAN algorithm. However, parameter sensitivity is still a challenge for DBSCAN for its further applications. K-modes [7] is an upgraded version of K-means by introducing categorical attributes clustering capability. Fuzzy K-modes [8] is a modified K-modes clustering algorithm with fuzzy mechanism to improve its robustness for various types of data sets. BFCM [9] is short

for bias-correction fuzzy clustering algorithm which is an extension of hard clustering and it is based on fuzzy membership partitions. COOLCAT [10] is an entropy-based algorithm for categorical clustering which brought up a novel idea of clustering on basis of entropy. Data clusters are generated by their entropy values. TCGA [11] is a two-stage genetic algorithm for automatic clustering. Bioinspired clustering algorithm summarizes clustering process as an optimization problem and genetic algorithm is adopted for convergence to the global optima. These above-mentioned methods face difficulties when dealing with data with mixed attributes, while the latter is emerging very quickly [14–23]. Fast density clustering algorithm is put forward to solve clustering center determination problem [24]. However, its mixed similarity calculation method is based on relationship of all attributes which has high computation complexity. And its cluster center determination method is mainly dependent on parameter dc which is difficult to set priorly.

For example, distance measure functions for numerical values cannot capture the similarity among data with mixed attributes. Moreover, the representation of a cluster with numerical values is often defined as the mean value of all data objects in the cluster, which, however, is illogical for other attributes. Algorithms have been proposed [14, 15, 17, 21, 22] to cluster hybrid data, most of which are based on partition. First, a set of disjoint clusters are obtained and refined to minimize a predefined criterion function. The objective is maximizing the intracluster connectivity or compactness while minimizing intercluster connectivity [25]. However, most partition clustering algorithms are sensitive to the initial cluster centers which are yet difficult to determine. They are also suitable for spherical distribution data without outliers handling capacity.

The main contributions of our work include four aspects. A novel mixed data similarity metric is come up for mixed data clustering. Clustering center self-set algorithm (CSA) is applied to determine center automatically. Bisection method is adopted to calculate parameter for clustering to overcome parameter sensibility problem. Fast one-time scan density clustering algorithm (FDCA) is brought up to implement fast and efficient clustering for mixed data.

The rest of this paper is organized as follows. Section 2 introduces related works of mixed data clustering. In Section 3, the similarity metric for data with mixed attributes and how FDCA works are presented. In Section 4, the abundant simulations are carried out to testify FDCA's performance compared with other classic algorithms. Section 5 is a practical application for handwriting number image recognition based on FDCA. And finally Section 6 concludes the paper.

2. Related Works

2.1. Mixed Data Clustering Algorithms Overview. As stated above, mixed data clustering algorithm is designed for data set of mixed attributes including numerical and categorical attributes. Numerical attributes of mixed data are evaluated by real values, while categorical attributes of mixed data represent the fact that those attributes are ordinal. It is still a challenge to cluster data with both numerical and categorical attributes. Lots of novel clustering algorithms are put forward to deal with mixed data. Huang proposed a k-prototypes [14] algorithm which combines K-means and k-mode algorithms. K-prototypes algorithm is an updated version of K-means and k-mode algorithm, especially designed for dealing with mixed data. It is a very early stage mixed data clustering algorithm. When the data set is uncertain, most clustering algorithm could not achieve purity and efficiency as expected. KL-FCM-GM [15] algorithm is an extended algorithm of k-prototypes proposed by Chatzis. It is a fuzzy c-means-type algorithm for clustering data with mixed numeric and categorical attributes by employing a probabilistic dissimilarity functional. It is designed for the Guss-multinormal distributed data. When the data set is large, the data similarity metric processing costs much more time than expected. So it is not quite suitable for big data objects. Zheng et al. developed a new algorithm called EKP [17], which is an improved k-prototypes algorithm to overcome its flaws. EKP algorithm has global search capability by introducing an evolutionary algorithm. Later, Li and Biswas proposed the Similarity-Based Agglomerative Clustering (SBAC) algorithm [18], which adopts the similarity measure defined by Goodall [19] to evaluate the similarity. It is an unsupervised analysis method for identifying critical samples in large populations, so the efficiency of the similarity metric is not stable. Hsu and Chen proposed a clustering algorithm based on the variance and entropy (CAVE) [20] for clustering mixed data. However, the CAVE algorithm needs to build the distance hierarchy for every categorical attribute and the determination of distance hierarchy requires the domain expertise.

Besides the above-mentioned unsupervised similarity metric for clustering, there are further researches on mixed data similarity calculation methods proposed. Ahmad and Dey proposed a k-means type algorithm [21] to deal with mixed data. Cooccurrence of categorical attribute values is used to evaluate the significance of each attribute. For mixed data attributes, Ji et al. proposed IWKM algorithm [22], in which distribution centroid is applied to represent the prototypes clusters. And the significance of different attributes is taken into account towards the clustering process. Besides, Ji et al. proposed WFK-prototypes [23] by introducing fuzzy centroid to represent the cluster prototypes. The significance concepts proposed by Ahmad and Dey [21] are adopted to extend k-prototypes algorithm in WFK-prototypes algorithm. WFK-prototypes algorithm is a classic mixed data clustering algorithm until now. David and Averbuch proposed a categorical spectral clustering algorithm for numerical and nominal data, called SpectralCAT [26]. Cheung and Jia [27] proposed a mixed data clustering algorithm based on a unified similarity metric without knowing clusters number. The embedded competition and penalization mechanisms are used to determine the number of clusters automatically by gradually eliminating the redundant clusters.

In a word, there are a lot of mixed data similarity metrics and clustering algorithms designed for different applications. We still want to develop a universal numerical and categorical

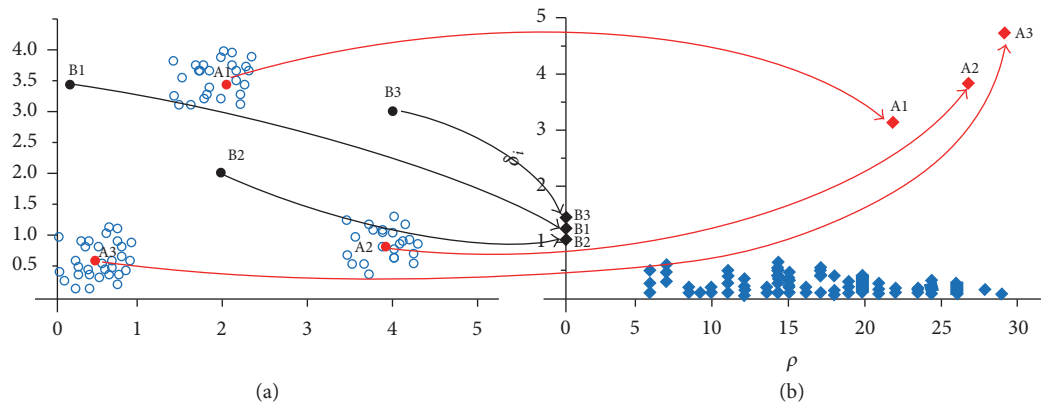


Figure 1: The algorithm in two dimensions. (a) Point distribution. (b) ρ and δ distribution of (a).

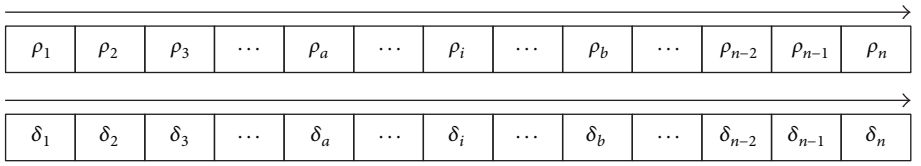


Figure 2: The descending order of ρ and δ.

data similarity metric and clustering algorithm that could be applied to most cases and practical data sets.

2.2. Fast Data Clustering Algorithm. Rodriguez and Laio had got their novel paper "Clustering by Fast Search and Fine of Density Peaks" published on Science in June 2014 [28]. In their algorithm, clustering centers could be observed from density-distance relationship graph. Inspired by their method, we conclude their method as follows: the cluster centers are surrounded by neighbors with lower density and they are at a relatively large distance from any points with a higher density. Noise points have comparatively larger distance and smaller density.

The density ρ_i of data point i is defined as follows:

$$\rho_i = \sum_j f\left(d_{ij} - d_c\right), \tag{1}$$

$$f(x) = \begin{cases} 1, & x = d_{ij} - d_c < 0, \\ 0, & \text{else}, \end{cases} \tag{2}$$

where ρ_i denotes data x_i's density, d_{ij} represents distance between data x_i and data x_j, and dc is the threshold distance of each cluster defined priorly. According to (2), if the distance between data x_i and data x_j is less than dc, then density of data x_i is $\rho_i = \rho_i + 1$. In other words, ρ_i is equal to the number of points that are closer than dc to point i.

δ_i is measured by computing the minimum distance between the point i and any other point with higher density:

$$\delta_i = \min_{j:\rho_j > \rho_i} \left(d_{ij}\right). \tag{3}$$

For the point with highest density, we conventionally take $\delta_i = \max_j(d_{ij})$. Note that δ_i is much larger than the typical nearest neighbor distance only for points that are local or global maxima in the density. Thus, cluster centers are recognized as points for which the value of δ_i is anomalously large.

This observation, which is the core of the algorithm, is illustrated by the simple example in Figure 1(a). Then the density and distance of every point are computed. ρ and δ distribution is shown in Figure 1(b).

There is a mapping between point distribution and ρ and δ distribution. For example, there are three red points A1, A2, and A3 in Figure 1(a) and they are cluster centers in original point distribution; the corresponding points A1, A2, and A3 in Figure 1(b) have larger distance and larger density than other points. In addition, there are three black points B1, B2, and B3 in Figure 1(a) and they are isolated and called the noise points. The corresponding points B1, B2, and B3 in Figure 1(b) have larger distance and smaller density than other points. Other points belong to one cluster and are called border points.

For all the data objects, we sort the density in descending order, as shown in Figure 2.

For any data point i, there are some qualitative relationships as follows:

(1) If $\rho_i \in (\rho_b, \rho_n)$ and $\delta_i \in (\delta_b, \delta_n)$, the data point i is the cluster center.

(2) If $\rho_i \in (\rho_1, \rho_a)$ and $\delta_i \in (\delta_b, \delta_n)$, the data point i is a noise point.

If the data point does not meet situations 1 and 2, then the data point i is a border point. Because cluster center has relatively larger density and larger distance compared to

Table 1: Six distance measures of partition-based clustering algorithms.

Algorithm	Distance measure	Numerical part	Categorical part	Type
k-means [8]	$d(X_i, X_j) = \sqrt{\sum_{p=1}^{m} (X_i^p - X_j^p)^2}$	$d(X_i^p, X_j^p) = (X_i^p - X_j^p)^2$	none	Numerical
k-modes [14]	$d(X_i, X_j) = \sum_{p=1}^{m} \delta(X_i^p, X_j^p)$	none	$\delta(X_i^p, X_j^p) = \begin{cases} 0, & X_i^p = X_j^p, \\ 1, & X_i^p \neq X_j^p \end{cases}$	Categorical
k-prototypes [17]	$d(X_i, Q_l) = \sum_{j=1}^{p} (X_{ij}^r - q_{ij}^r)^2 + \mu_l \sum_{j=p+1}^{m} \delta(X_{ij}^c, q_{ij}^c)$	$d(X_i, Q_l) = (X_i^p - Q_j^p)^2$	$\delta(X_i^p, Q_j^p) = \begin{cases} 0, & X_i^p = Q_j^p, \\ 1, & X_i^p \neq Q_j^p \end{cases}$	Mixed
EKP [20]	$d(X_i, Q_l) = \sum_{j=1}^{p} (X_{ij}^r - q_{ij}^r)^2 + r \sum_{j=p+1}^{m} \delta(X_{ij}^c, q_{ij}^c)$	$d(X_i, Q_l) = (X_i^p - Q_j^p)^2$	$\delta(X_i^p, Q_j^p) = \begin{cases} 0, & X_i^p = Q_j^p, \\ 1, & X_i^p \neq Q_j^p \end{cases}$	Mixed
WFK-prototypes [25]	$d(X_i, Q_l) = \sum_{l=1}^{p} (s_l (X_{ij}^r - q_{ij}^r)^2) + \sum_{l=p+1}^{m} \varphi(X_{ij}^c, v_{ij}^c)^2$	$d(X_i, Q_l) = s_l (X_i^p - Q_j^p)^2$	$\varphi(X_i^p, Q_j^p)^2$	Mixed
FPC-MDACC	$d(X_i, X_j) = d(X_i, X_j)_n + d(X_i, X_j)_c$	$d(X_i, X_j)_n$ depends on data type (including three types)	$d(X_i, X_j)_c$ depends on data type (including three types)	Mixed

other centers, while noise data only has relatively larger distance from cluster centers and much less density, both cluster centers number and noise amount are relatively small compared with other data objects. The average density value and distance value are mainly dependent on majority of data objects besides centers and noise. So the specific value of a and b for different data set could be self-determined during the finding cluster center process. For instance, if the data size is 1000, the cluster center is selected from ρ_n. For one data object x_i, if its density is ρ_n, then we check if its δ is δ_n or a little bit less than δ_n. If so, then data object x_i is one of the cluster centers. And if its density is more like ρ_1 while its distance is more like δ_n, then data object x_i is noise data. By checking those data objects according to CSA in Section 3.2.2, we could get all those cluster centers one by one.

In summary, the only points of high ρ and relatively high δ are the cluster centers. The points have relatively high δ and low ρ because they are isolated; they can be considered as noise points.

3. Fast Density Clustering Algorithm for Numerical Data and Categorical Data

3.1. Numerical Data and Categorical Data Unified Similarity Metric

3.1.1. Main Idea. Similarity metric is important for a meaningful cluster analysis. Table 1 lists typical similarity metrics for current clustering algorithms.

As shown in Table 1, six classic mixed data similarity metrics are listed and compared. According to each algorithm, different distance measure equations are developed including numerical attributes calculation part and categorical attributes calculation part. For instance, K-means algorithm is only suitable for numerical attributed data only, so there is no definition for measuring categorical attribute part for data set. And k-modes algorithm is designed for dealing with categorical attributed data which has no numerical attributes similarity metric. The other four similarity metrics are applied to mixed data, so all of them have both numerical attribute and categorical attribute parts distance metrics.

The Euclidean distance is adopted by K-means algorithm to deal with the pure numerical data. The simple matching distance is adopted by K-modes algorithm to deal with the pure categorical data. K-prototypes algorithm integrates K-means and K-modes to deal with mixed data. Algorithms EKP and WFK-prototypes improved k-prototypes algorithm by introducing fuzzy factor or weight coefficient in original distance measure, so that it can more accurately measure the similarity between objects. FPC-MDACC algorithm [29] adopts three different distance measure methods for mixed data depending on their types which need prior work to determine which type the current mixed data is, and this represents extra time cost and extra algorithm complexity.

Until now, we still need an efficient similarity metric for calculating distance of data objects of mixed data. We believe that one unified similarity metric for both numerical and categorical data is more efficient and reasonable for mixed data instead of independent calculation for each of the other attributes.

3.1.2. Unified Similarity Metric for Numerical and Categorical Data. A unified similarity metric is presented in this section for mixed data, which is applicable for any type of mixed data which has numerical attributes or categorical attributes or both.

Definition 1. Given the data set $D = \{X_1, X_2, \ldots, X_i, \ldots, X_n\}$, each data object X_i has d dimensions. The distance $D(X_i, X_j)$ between two data objects X_i and X_j is defined as

$$D\left(X_i, X_j\right) = \frac{\sum_{p=1}^{n} \omega_{i,j}^{p} d_{i,j}^{p}}{\sum_{p=1}^{n} \omega_{i,j}^{p}}, \tag{4}$$

where $\omega_{i,j}^{p}$ denotes weight of pth attribute and n is the number of attributes. If the attribute value of pth is missing, then $\omega_{i,j}^{p} = 0$; else $\omega_{i,j}^{p} = 1$. $d_{i,j}^{p}$ denotes distance of pth attribute for data objects X_i and X_j.

(1) If pth attribute is numerical, then $d_{i,j}^{p}$ is defined as follows:

$$d_{i,j}^{p} = \frac{\left|x_i^p - x_j^p\right|}{\max_h x_h^p - \min x_h^p}, \tag{5}$$

where h goes through every possible attribute value of data objects X_i and X_j.

Since the numerical attribute for different data could be quite different, in case the value is quite large or small, we have to balance its contribution to the final distance. So numerical attributes need to be normalized into [0, 1].

(2) If pth attribute is categorical or binary, then $d_{i,j}^{p}$ is defined as follows:

$$d_{i,j}^{p} = \begin{cases} 0, & x_i^p = x_j^p, \\ 1, & \text{otherwise}, \end{cases} \tag{6}$$

where p goes through every possible attribute value of data objects X_i and X_j.

The categorical attribute is defined to evaluate whether the data objects i and j are the same or not on this attribute. If they have the same attribute, then the distance defined equals 0; otherwise the distance is 1.

(3) If pth attribute is order, then $d_{i,j}^{p}$ is defined as follows:

$$d_{i,j}^{p} = \frac{\left|z_i^p - z_j^p\right|}{\max_h z_h^p - \min z_h^p}, \tag{7}$$

where p goes through every possible attribute value of data objects X_i and X_j. z_i^p is defined as follows:

$$z_k^p = \frac{r_k^p - 1}{M_p - 1}, \tag{8}$$

where r_k^p denotes order of each x_k^p and M_p is the total number of values x_k^p has among all data objects.

In this paper, ordinal attributes are defined different from categorical attributes. Ordinal attributes are ordered by their values from big to small. For instance, pth attribute of data object i is represented as $x_i^p = 1$, pth attribute of data object j is represented as $x_j^p = 2$, and attribute of data object k is represented as $x_i^p = 3$. If the pth attribute is categorical, then the distance between i and j equals 1; distance between i and k equals 1 as well. However, in our case, the pth attribute is ordinal, so these two distances should be distinguished. We calculated their pth attribute distance according to (7) and (8).

In this way, similarity for all the data objects could be calculated based on (5) to (8). In order to demonstrate how these three types of attribute are defined and measured according to the above proposed methods, we take data set Heart from UCI as an example.

Table 2: Attributes information of three data samples.

Data sample	Numerical attributes	Categorical attributes	Ordinal attributes
Data sample 1	70, 130, 322, 109, 2.4	1, 0, 2, 0, 2, 3	4, 3
Data sample 2	67, 115, 564, 160, 1.6	0, 0, 2, 0, 2, 0	3, 0
Data sample 3	57, 124, 261, 141, 0.3	1, 0, 0, 0, 1, 0	2, 0

3.1.3. Illustration for Unified Similarity Metric. As the unified similarity metric is put forward in Section 3.1.2, we would like to take data set Heart from UCI as an example to testify how it works.

Data in Heart has 13 attributes including the following:

(1) Age
(2) Sex
(3) Chest pain type (4 values)
(4) Resting blood pressure
(5) Serum cholesterol in mg/dL
(6) Fasting blood sugar > 120 mg/dL
(7) Resting electrocardiographic results (values 0, 1, and 2)
(8) Maximum heart rate achieved
(9) Exercise induced angina
(10) Oldpeak = ST depression induced by exercise relative to rest
(11) The slope of the peak exercise ST segment
(12) Number of major vessels (0–3) colored by fluoroscopy
(13) Thal: 3 = normal; 6 = fixed defect; 7 = reversible defect

According to their practical meanings, five attributes are defined as numerical attributes (1, 4, 5, 8, 10), two attributes are defined as ordinal attributes (3, 12), and the remaining six attributes are defined as categorical attributes (2, 6, 7, 9, 11, 13). Based on (1) to (8), the data similarity of Heart could be measured according to their attribute type. For instance, three data samples data sample 1, data sample 2, and data sample 3 are listed in Table 2 for calculating and explaining how brought up similarity calculation metric works.

Table 3: Distance between two data samples.

Distance donation	Numerical distance	Categorical distance	Ordinal distance	Distance
d(data sample 1, data sample 2)	0.71	2	1	3.71
d(data sample 1, data sample 3)	0.46	3	1.25	4.71
d(data sample 2, data sample 3)	0.77	3	0.25	4.02

According to the unified similarity metric, the distance of each data sample can be measured as in Table 3.

From Table 2, we can conclude that data sample 1 and data sample 3 are more likely to be clustered into one cluster because their distance is less, while data sample 2 has less similarity with data sample 1 and data sample 3. From Heart data set from UCI, original label information of data samples is given. And data sample 1 and data sample 3 are labeled as the same class, while data sample 2 belongs to another class. Our similarity results are correct, and the unified metric for mixed data set is efficient from this illustration.

3.2. Fast Density Clustering Algorithm (FDCA) for Mixed Data

3.2.1. Main Idea. Based on analysis of Figure 3, the only points of relatively larger ρ and larger δ are the cluster centers. The points which have relatively larger δ and less ρ can be considered as noise points because they are isolated. In order to realize cluster centers self-determination, more information from all data objects in the descending order of ρ and δ is explored.

First of all, all data objects are sorted in descending order of their ρ and δ values each. And a fast center set algorithm (CSA) is adopted to choose the clustering centers automatically. After the cluster centers have been found, each remaining point is assigned to the same cluster as its nearest neighbor of higher density. The cluster assignment is executed through one-time scan. Different from other partitioned clustering algorithms, FDCA can deal with arbitrary shape cluster. Each remaining point is assigned to the same cluster as its nearest neighbor of higher density. As shown in Figure 3, the number means the level of density: the bigger the number, the larger the density. Data object "3" is a cluster center and the cluster label is CENTER-1. The cluster label of data object "4" should be the same as the nearest neighbor of higher density, so the cluster label should be the same as data object "5," which is CENTER-1.

For the noise point, FDCA does not introduce a noise-signal cutoff. Instead, we first find for each cluster a border region, defined as the set of points assigned to that cluster but being within a distance dc from data points belonging to other clusters. We then find, for each cluster, the point with highest density within its border region. Its density is denoted by ρ_b, and only keep the points that have density larger than or equal to ρ_b.

The main idea of how CSA algorithm is applied for FDCA is shown as chart in Figure 4.

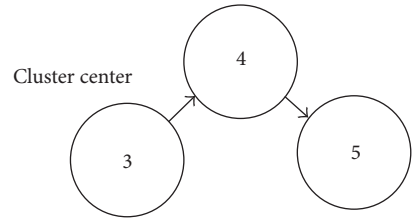


Figure 3: Data clustering rules based on portioned clustering.

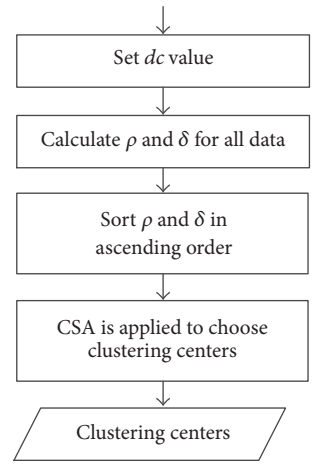


Figure 4: The chart for main idea of how CSA is applied for FDCA.

3.2.2. Clustering Center Set Algorithm (CSA). CSA algorithm is brought up to find out clustering centers for data clustering automatically based on ρ and δ descending order of all data objects. The process of CSA algorithm is shown in Algorithm 1.

3.2.3. Parameter dc Optimization. CSA algorithm is sensitive only to the choice of dc; proper selection of dc could help CSA to find the correct clustering centers which would lead to high-efficient FDCA. This section would focus on how to get proper value of dc.

In Alex algorithm [28], as a rule of thumb, proper value for dc is in the scale of 1% to 2% of data objects number in data set. For example, if the total number of data objects is 1000, then $dc \in [10, 20]$. Since our designed FDCA aims to cluster mixed data, the target data set is different from Alex algorithm. Therefore, mixed data set is observed from UCI Machine Learning Repository. Because mixed data has more complicated similarity metric, the distances between cluster center and its data objects are more likely to be of wider scale. We can choose dc in the scale of 1% to 20% of data objects number in data set for all possibilities. For example, if the total number of data objects is 1000, then $dc \in [10, 200]$. However, in this way, we could only confirm the value scale of dc but could not achieve the optimal value.

Suppose that the data set has N data samples; the scale for dc could be defined as $d_{c\,\mathrm{low}} = N * 1\%$ and $d_{c\,\mathrm{high}} = N * 20\%$. For one dc from $[d_{c\_\mathrm{low}}, d_{c\_\mathrm{high}}]$, density and distance for

Step 1. Suppose $D = \{X_1, X_2, \ldots, X_i, \ldots, X_n\}$ denotes a set of n data objects. Calculate the corresponding density set $\overrightarrow{\rho} = \{\rho_1, \rho_2, \ldots, \rho_i, \ldots \rho_n\}$ and distance set $\overrightarrow{d} = \{d_1, d_2, \ldots, d_i, \ldots, d_n\}$. Initial set the center set $C = \emptyset$ and sort the density in descending order as shown:

ρ_n	$\cdots$	ρ_i	$\cdots$	ρ_3	ρ_2	ρ_1

max $\longrightarrow$ min

d_n	$\cdots$	d_i	$\cdots$	d_3	d_2	d_1

Step 2. Select the maximum density value ρ_s, $\rho_s \in \overline{\rho}$ and $\rho_s \geq \rho_i$, $\rho_i \in \overline{\rho}$.
Step 3. Find the corresponding data object X_s according to ρ_s.
Step 4. Get the distance value d_s of data object X_s, and

If $(d_s \geq d_j,\ d_j \in \overrightarrow{d})$
then
$C = C \cup X_s$ and turn to step 2;
else if $(d_s \geq \sum_{i=1}^{n} d_i/n,\ d_i \in \overrightarrow{d})$
then
$C = C \cup X_s$ and turn to step 2;
else
Turn to step 5.

Step 5. Output cluster centers set $C = \{c_1, c_2, \ldots, c_k\}$.

Algorithm 1

all data objects could be calculated. From the corresponding relationship graph of density and distance for each data object, CSA algorithm is adopted to determine cluster centers. After the cluster centers have been found, each remaining point is assigned to the same cluster as its nearest neighbor of higher density. The rest of data objects are divided into those clusters based on FDCA (described in Section 3.2). This whole process is called one iteration for one dc. Because clustering is an unsupervised method, whether dc is an optimal value of distance threshold or not could not be evaluated by data samples' original label class. Another performance evaluation index is designed.

Suppose that there are c clusters; each cluster center could be represented as c_i. Data objects clustered into c_i are denoted as c_j^i, $j \in [0, N_i]$, where N_i represents the number of data objects belonging to cluster c_i. Then the performance evaluation index for each dc is defined as

$$Z = \frac{\sum_{j=1}^{N_i} d\left(c_j^i, c_i\right)}{c}, \tag{9}$$

where $d(c_j^i, c_i)$ is short for distance between data object c_j^i and its cluster center c_i.

The value of Z could reflect the closeness of clusters. So we would optimize dc value with the minimum of Z. So finding proper value of dc could be summarized as an optimization problem. Optimization algorithm is applied for selecting optimal parameters for clustering algorithms such as PSO [29]. PSO based parameter self-adaptive method is proven useful by comprehensive simulations. However, PSO is a bioinspired optimization algorithm based on iterations, which results in high algorithm complexity and time complexity. In order to realize fast data clustering, dichotomy [30] is adopted instead of bioinspired algorithms to search for optimal dc.

Table 4: dc value influences clustering centers number on Iris data set.

d_c	ρ	δ	k	d_c	ρ	δ	k
0.1	1	4.1	2	0.55	17	1.1	4
0.15	2	3.0	2	0.6	22	3.3	4
0.2	**3**	**2.9**	**3**	0.65	28	3.1	4
0.25	**4**	**2.8**	**3**	0.7	32	3.1	4
0.3	**6**	**2.8**	**3**	0.75	38	3.4	4
0.35	7	3.1	5	0.8	41	3.4	4
0.4	8	1.29	5	0.85	44	3.4	4
0.45	9	1.2	5	0.9	45	3.6	4
0.5	12	1.1	4	0.95	46	3.4	4

According to this rule, for each data set, we can get an initial range for dc as $[d_{c\_\text{low}}, d_{c\_\text{high}}]$. The only problem is how to get the optimal value of dc. We already know that proper dc could make CSA get the optima clustering centers, so we have to get how dc influences clustering efficiency. We take Iris data set as an example. dc is set from 0.1 to 0.9 with 0.05 as a step. CSA is adopted to get clustering centers number k as in Table 4.

From Table 4, we can conclude that sequential value of dc from minimum value of 0.1 to maximum value of 0.95 with each 0.05 step could get the optima value of dc as 0.2, 0.25, or 0.3 whose clustering centers number is 3. So we could use a fast searching algorithm to find the best value of dc to get the optima value of clustering center.

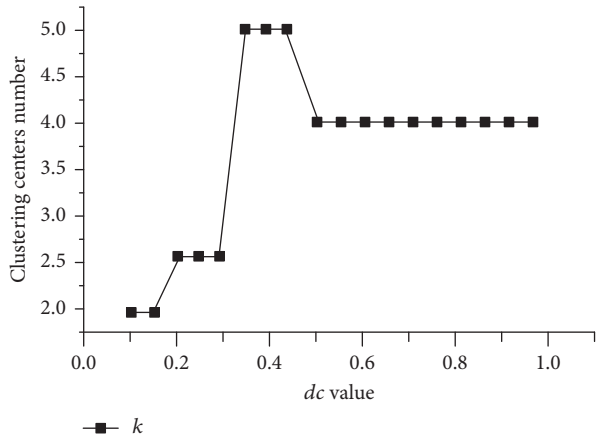


Figure 5: Relationship of dc and clustering centers number k on Iris data set.

Table 5: Twelve data sets from UCI.

Data sets	Attributes	r	q	k	Number of instances
Aggregation	2	2	0	7	788
Spiral	2	2	0	3	312
Jain	2	2	0	2	373
Flame	2	2	0	2	240
Iris	4	4	0	3	150
Breast	9	9	0	2	699
Soybean	35	0	35	4	47
Zoo	15	1	14	7	101
Acute	7	1	6	2	120
Statlog (Heart)	13	5	8	2	270
Credit	15	9	6	2	690
KDD CUP-99	41	34	7	/	1000

r is the number of numerical attributes, q is the number of categorical attributes, "/" is for unknown parameters, and k is the number of clusters.

We apply the self-adaptive strategy of dc value on Iris data shown in Figure 5 to testify the efficiency of clustering centers numbers of dc value.

Dichotomy algorithm is applied to search the optima value of dc for clustering algorithm. We define the value scale of dc as $[d_{c\_\text{low}}, d_{c\_\text{high}}]$, where dc is from 1% to 20% of total data samples number. For fixed value ξ, dichotomy algorithm uses function $f(x)$ to find the approximate zero by the following steps.

Step 1. Fix value range $[d_{c\_\text{low}}, d_{c\_\text{high}}]$, verify and make sure that $f(d_{c\_\text{low}}) \cdot f(d_{c\_\text{high}}) < 0$, and set definition ξ.

Step 2. Calculate midpoint for range $[d_{c\_\text{low}}, d_{c\_\text{high}}]$, which is denoted as c.

Step 3. Calculate $f(c)$ according to CSA to get specific clustering centers number k based on $dc = c$.

Step 4. If $f(c) = 0$, then c is the optima value of dc. Else if $f(d_{c\_\text{low}}) \cdot f(c) < 0$, then $d_{c\_\text{high}} = c$. Else if $f(c) \cdot f(d_{c\_\text{high}}) < 0$, then $d_{c\_\text{low}} = c$.

Step 5. If the definition ξ is achieved, in other words, $|d_{c\_\text{high}} - d_{c\_\text{low}}| < \xi$, then the optima value of dc is current d_{c_high} or d_{c_low}; end the algorithm; else go to step 2.

Therefore, initial dc is selected randomly in scale of $[d_{c\_\text{low}}, d_{c\_\text{high}}]$, and CSA algorithm is executed to determine cluster centers automatically. According to the current result of clustering, compute Z defined as (9) to evaluate whether current dc is good enough for clustering. If it is, then we fix current dc as the optimum and calculate the purity and efficiency of FDCA. Otherwise, dichotomy searching algorithm is applied to find another dc and repeat CSA and FDCA. The brought up optimal dc self-adaptive algorithm is faster than PSO based algorithm. For PSO or other bioinspired optimization algorithms, from Table 4, we can conclude that proper value of dc could help CSA find the correct cluster center. However, with the slight difference of dc value from 0.1 to 0.5, the clusters number k is the same, which means we only have to find the proper scale of dc from its initial scales instead of finding the optimal value. Dichotomy searching algorithm is a fast searching algorithm to find the proper half area for dc. In Section 4, abundant simulations and the real-life application testify its efficiency in finding proper dc.

4. Simulations and Analysis

4.1. Data Settings. Ten data sets from UCI Machine Learning Repository are used for clustering algorithm simulations, as shown in Table 5.

4.2. Performance Analysis. (1) In clustering analysis, the clustering accuracy (r) [11] is one of the most commonly used criteria to evaluate the quality of clustering results, defined as follows:

$$r = \frac{\sum_{i=1}^{k} a_i}{n}, \tag{10}$$

where a_i is the number of data objects occurring in both ith cluster and its corresponding true class and n is the number of data objects in the data sets. According to this measure, the larger r is, the better the clustering results are, and for perfect clustering $r = 1.0$.

(2) Another clustering quality measure is the average purity of clusters defined as follows:

$$\text{Pur} = \frac{\sum_{i=1}^{k} \left(\left| C_i^d \right| / |C_i| \right)}{k}, \tag{11}$$

where k denotes the number of clusters. $|C_i^d|$ denotes the number of points with the dominant class label in cluster i. $|C_i|$ denotes the number of points in cluster i. Intuitively, the purity measures the purity of the clusters with respect to the true cluster (class) labels that are known for our data sets.

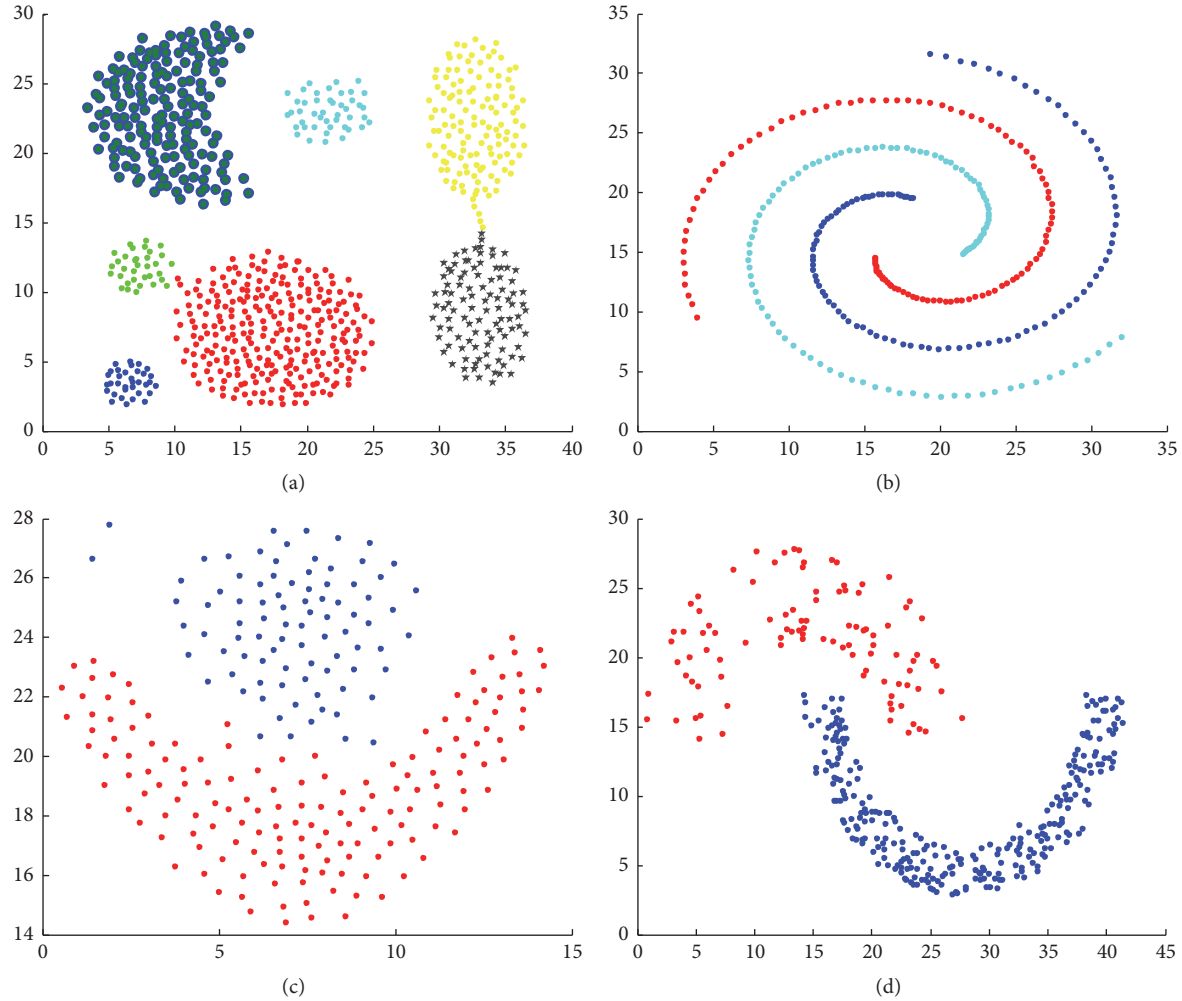


Figure 6: Example data sets. (a) Aggregation, (b) Spiral, (c) Flame, and (d) Jain.

4.3. Result Analysis

4.3.1. Clustering Efficiency. There are four 2-dimensional data sets (Aggregation, Jain, Spiral, and Flame) with various shapes of clusters (circular, elongated, spiral, etc.). The results are presented in Figure 6.

The results in Table 6 show that the algorithm is capable of clustering arbitrary shape, variable density clusters and has a good clustering quality.

The performance of FDCA is compared with K-prototypes, SBAC, KL-FCM-GM, IWKM, DBSCAN, BIRCH, SpectralCAT, TGCA, and FPC-MDACC algorithms. The experiments results on different data sets show that FDCA algorithm is able to find optimal solution after a small number of iterations. The following reasons contribute to the better performance of our proposed algorithm. FDCA needs to analyse the density and distance of each point, and we then adopt dichotomy analysis techniques to fit the functional relationship $\delta_i^* = f(\rho_i)$. Afterwards, by analysis, the residuals distribution finds the cluster centers automatically. It conforms with the original data distribution of mixed data, which leads to a good clustering result.

4.3.2. Clustering Algorithm Time Complexity. Figure 7 lists the average execution time of our proposed algorithm and other algorithms on the eight data sets.

Because the number of data records in Iris, Soybean, Zoo, and Acute data sets is small, the execution is fast. The KDD CUP sample data sets and Breast data have a relatively large number of data records, and thus the execution time is longer. Since the balanced data, like Heart and Credit, adopt the probability and statistics method in the pretreatment stage, therefore they need more time than others.

4.3.3. Complexity Analysis. Assume that the data set has n data objects; the time complexity of FDCA algorithm mainly consists of the computation of the distance and density of each data object, and the computational costs are $O(n^2)$ and $O((n^2 - n)/2)$. After the cluster centers are found, the cluster assignment is performed in a single step, and the corresponding computational cost is $O(n - k)$, where k denotes the number of cluster centers.

The time complexity of partition-based clustering algorithms and hierarchical clustering algorithms is $O(\text{iter} * k * n)$

Table 6: Clustering quality evaluation on all data sets.

Data sets	Algorithms	AC(r)	Purity
Iris	K-prototypes	0.819	0.842
	SBAC	0.426	0.46
	KL-FCM-GM	0.335	0.382
	IWKM	0.822	0.84
	DBSCAN	0.695	0.72
	BIRCH	0.86	0.89
	SpectralCAT	0.96	0.98
	TGCA	1.00	1.00
	FPC-MDACC	0.96	0.964
	FDCA	**0.98**	**0.976**
KDD CUP-99	K-prototypes	0.84	0.862
	KL-FCM-GM	0.673	0.724
	IWKM	0.882	0.893
	DBSCAN	0.655	0.71
	BIRCH	0.72	0.763
	k-modes	0.84	0.863
	Fuzzy k-modes	0.87	0.887
	OCIL	0.916	0.928
	FPC-MDACC	0.938	0.945
	FDCA	**0.95**	**0.961**
Zoo	K-prototypes	0.806	0.83
	KL-FCM-GM	0.426	0.485
	IWKM	0.864	0.843
	DBSCAN	0.629	0.694
	BIRCH	0.908	0.876
	k-modes	0.42	0.53
	Fuzzy k-modes	0.732	0.764
	OCIL	0.881	0.91
	FPC-MDACC	0.892	0.849
	FDCA	**0.93**	**0.921**
Heart	K-prototypes	0.577	0.644
	SBAC	0.752	0778
	KL-FCM-GM	0.758	0.802
	EKP	0.545	0.589
	WFK-prototypes	0.835	0.826
	SpectralCAT	0.82	0.824
	OCIL	0.827	0.831
	FPC-MDACC	0.848	0.833
	FDCA	**0.912**	**0.903**
Breast cancer	K-prototypes	0.84	0.862
	KL-FCM-GM	0.673	0.724
	IWKM	0.882	0.893
	DBSCAN	0.655	0.71
	BIRCH	0.72	0.763
	k-modes	0.84	0.863
	Fuzzy k-modes	0.87	0.887
	OCIL	0.916	0.928
	FPC-MDACC	0.938	0.945
	FDCA	**0.97**	**0.97**
Soybean	K-prototypes	0.856	0.877
	KL-FCM-GM	0.617	0.642
	IWKM	0.903	0.895
	DBSCAN	0.908	0.922
	BIRCH	0.915	0.9
	k-modes	0.957	0.985
	Fuzzy k-modes	0.898	0.902
	OCIL	0.894	0.895
	FPC-MDACC	0.957	0.985
	FDCA	**0.978**	**0.988**
Acute	K-prototypes	0.610	0.72
	SBAC	0.508	0556
	KL-FCM-GM	0.682	0.749
	EKP	0.508	0.586
	WFK-prototypes	0.710	0.765
	SpectralCAT	0.867	0.824
	OCIL	0.763	0.786
	FPC-MDACC	0.917	0.918
	FDCA	**0.92**	**0.933**
Credit	K-prototypes	0.562	0.624
	SBAC	0.555	0.627
	KL-FCM-GM	0.574	0.632
	EKP	0.682	0.749
	IWKM	0.779	0.806
	WFK-prototypes	0.838	0.826
	SpectralCAT	0.77	0.794
	OCIL	0.713	0761
	FPC-MDACC	0.796	0.833
	FDCA	**0.90**	**0.912**

and $O(n^2)$. So the time complexity of our proposed algorithm is higher than the partition-based clustering algorithms and hierarchical clustering algorithms. The advantages of our proposed algorithm are that the algorithm can determine the cluster centers automatically, can deal with arbitrary shape clusters, and is not sensitive to parameters.

5. Unsupervised Number Image Recognition Based on FDCA

5.1. Problem Description. Unsupervised number image recognition is defined as recognizing the number automatically from images without any label information in advance. Currently, there are three types of number image recognition methods including statistics, logic decision, and syntax analysis. Based on template matching algorithm and geometry feature extraction algorithm, most recognition algorithms are suitable for printer image recognition. In case of handwriting number images, most unsupervised recognition algorithms are confronted with low recognition rate because of different handwriting styles. For those supervised recognition methods, the most important premise is that there are enough

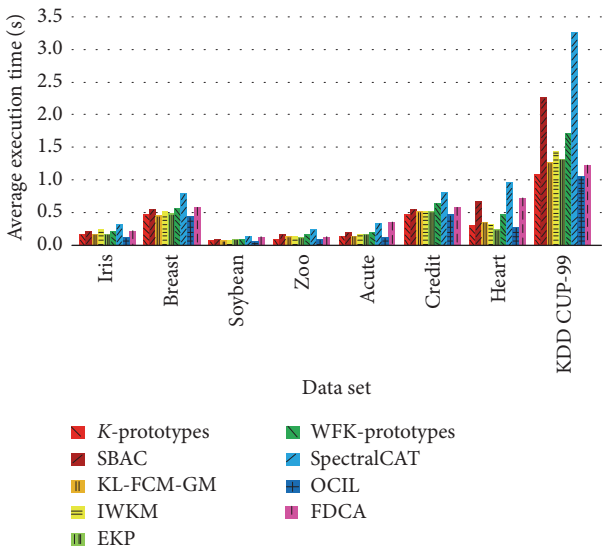


Figure 7: Average execution time comparison.

labeled examples for classifiers to train, while in practical cases, it is not always suitable. Aiming at those problems, an unsupervised number image recognition method based FDCA is brought up to improve the recognition rate of handwriting number images without any labeled samples in advance. First of all, number images are clustered based on FDCA. And then a strict filter is designed to extract cluster centers and typical cluster members automatically for classifier to guarantee that those training samples have pure cluster features. Finally, traditional classifiers BP artificial neural network (ANN) [31] is adopted to classify number images based on those selected cluster centers and typical images as training sample instead of label known images in advance to realize unsupervised method. MNIST data set is recognized to testify our designed unsupervised image recognition method based on FDCA.

5.2. Handwriting Images Clustering Based on FDCA

5.2.1. Similarity Metric for Handwriting Number Images. CW-SSIM (Complex Wavelet Structural Similarity) is applied to evaluate the similarity of number handwriting images. Assume that mother wavelet of symmetric complex wavelet is

$$w(u) = g(u)\, e^{jw_c u}, \tag{12}$$

where ω_c is central frequency of modulation band pass filter; $g(u)$ is a progressive function with symmetry. After stretching and shifting transformation, we can obtain corresponding wavelet clusters:

$$\omega_{s,p}(u) = \frac{1}{\sqrt{s}}\omega\left(\frac{u-p}{s}\right) = \frac{1}{\sqrt{s}}g\left(\frac{u-p}{s}\right)e^{j\omega_c(u-p)/s}, \tag{13}$$

where $s \in R^+$ is stretch factor and $p \in R$ is shift factor. Continuous complex wavelet transform of real signal $x(u)$ is

$$X(s,p) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} X(\omega)\sqrt{s}G(s\omega - \omega_c)\, e^{j\omega p}d\omega, \tag{14}$$

where $X(\omega)$ and $G(\omega)$, respectively, represent Fourier transform of $x(u)$ and $g(u)$.

In the process of complex wavelet transform, assume that $C_x = \{C_{x,i} \mid i = 1,\ldots,N\}$ and $C_y = \{C_{y,i} \mid i = 1,\ldots,N\}$, respectively, represent two coefficient sets of different images to be compared, which are extracted from same wavelet subband and same spatial location:

$$\widetilde{S}(C_x, C_y) = \frac{2\left|\sum_{i=1}^{N} C_{x,i}C_{y,i}^*\right| + K}{\sum_i^N |C_{x,i}|^2 + \sum_{i=1}^N |C_{y,i}|^2 + K}, \tag{15}$$

where C^* and C are complex conjugates; K is positive constant with small value, which is used to improve robustness of $\widetilde{S}$ at low signal-to-noise ratio.

In order to better understand CW-SSIM, right part of the equation is multiplied by an equivalent factor, whose value is 1:

$$\widetilde{S}(C_x, C_y) = \frac{2\sum_{i=1}^{N}\left|C_{x,j}\right|\left|C_{y,j}\right| + K}{\sum_i^N |C_{x,i}|^2 + \sum_{i=1}^N |C_{y,i}|^2 + K} \cdot \frac{2\left|\sum_{i=1}^{N} C_{x,i}C_{y,i}^*\right| + K}{2\sum_{i=1}^{N}\left|C_{x,i}C_{y,i}^*\right| + K}. \tag{16}$$

In the first part of right-hand side, each factor is constant or mode of complex wavelet coefficient. For two given images, complex wavelet coefficient corresponds to a certain value. If the condition of $|C_{x,i}| = |C_{y,i}|$ for all i is met, then the first part of right-hand side has maximum value of 1, and the value of second part is related to phase change of C_x and C_y. If the condition that phase change of $C_{x,i}$ and $C_{y,i}$ is constant for all i is met, then the second part has maximum value of 1. The reason for taking this part as image structural similarity index is mainly based on the following two points:

(1) The structural information of local image features is all included in phase pattern related to wavelet coefficients.

(2) The constant phase change of all coefficients does not change structure of local image features.

With dual-tree complex wavelet transform with shift invariance and good direction selectivity, CW-SSIM index based on dual-tree complex wavelet transform is given. Firstly, the image is decomposed into 6 levels through dual-tree complex wavelet decomposition, which can avoid serrated subband. And then calculate local CW-SSIM index of each wavelet subband by moving sliding window on subband, whose size is $7 * 7$. In the experiment, we found that performance of CW-SSIM will not be obviously affected by slight perturbation of parameter K, so that take $K = 0$. However, the value of K must be adjusted to obtain the best results

```
Initialize h_i = 0, ρ_i^b = 0;
For i = 1, 2, ..., N − 1
        For j = i + 1, i + 2, ..., N
                IF (c_i = c_j and dist(x_i, x_j) < R)
                        ρ̄ = (1/2)(ρ_i + ρ_j);
                        IF ρ̄ > ρ_{c_i}^b  ρ_{c_i}^b = ρ̄;
                        IF ρ̄ > ρ_{c_j}^b  ρ_{c_j}^b = ρ̄;
        END FOR
END FOR
FOR i = 1, 2, ..., N
        IF ρ_i < ρ_{c_i}^b
                h_i = 1;
END FOR
```

Algorithm 2

under noisy environment. Finally, CW-SSIM of whole image is obtained by weighted sum of each subband. The weight function is obtained by Gauss distribution, whose standard deviation is quarter of best layer image size from controllable pyramid.

The range of CW-SSIM is $[0, 1]$. The larger the value is, the higher the image similarity is.

5.2.2. Strict Filter Design. In order to guarantee the purity of each cluster, strict filter is designed to kick out the members which lie on the edge of cluster. Therefore, after the cluster centers are determined and the remaining points are assigned to appropriate cluster, boundary region of fixed cluster is set. Data points within the region have the following characteristics: the data points are belonging to the cluster, but within a distance of R (R is adjustable) there are objects belonging to the other cluster. By means of the objects in the boundary region, we can determine a local average density of the cluster; the object with density which is larger than the local density will be divided into the cluster, whereas the other objects are rejected, in order to ensure the cluster's purity. The implementation process is as Algorithm 2 shows.

5.3. Unsupervised Number Image Recognition Based on FDCA

5.3.1. Data Set and Evaluation Index. MNIST data set is applied to testify the performance of image recognition method based on FDCA, which consists of 60000 number handwriting images. Numbers from 0 to 9 are all collected for classifier stored as binary file, each of which is $28 * 28$, shown in Figure 8.

In this paper, we adopt the consistent indicators and recognition rate to evaluate the results as follows.

(1) Specific equation of recognition rate is defined as follows:

$$\gamma_{\text{recognition}} = \frac{\text{Number of face recognized correctly}}{\text{Total number of face attending recognition}} \times 100\%. \quad (17)$$

Figure 8: Examples of each number in MNIST.

(2) r_{true} represents the fraction of pair of images of the same subject correctly associated with the same cluster. r_{false} represents the fraction of pair of images of different subjects erroneously assigned to the same cluster. We define them as follows:

$$\gamma_{\text{true}} = \frac{TP + TN}{N(N-1)/2} \times 100\%,$$
$$\gamma_{\text{false}} = \frac{FP}{N(N-1)/2} \times 100\%, \quad (18)$$

where N represents the number of objects in data sets, $N(N-1)/2$ represents the pair number of the data sets, TP represents the same type of objects assigned to the same cluster, TN represents objects of different classes assigned to different clusters, and FP represents objects of different classes assigned to the same cluster.

5.3.2. Application Results and Analysis. The recognition algorithm is processed as follows.

Step 1. Original images are input to calculate their similarity based on CW-SSIM.

Step 2. FDCA is applied to cluster images to get training samples for BP ANN. Those cluster centers and typical members are selected by strict filters.

Step 3. Train BP ANN with cluster label information images.

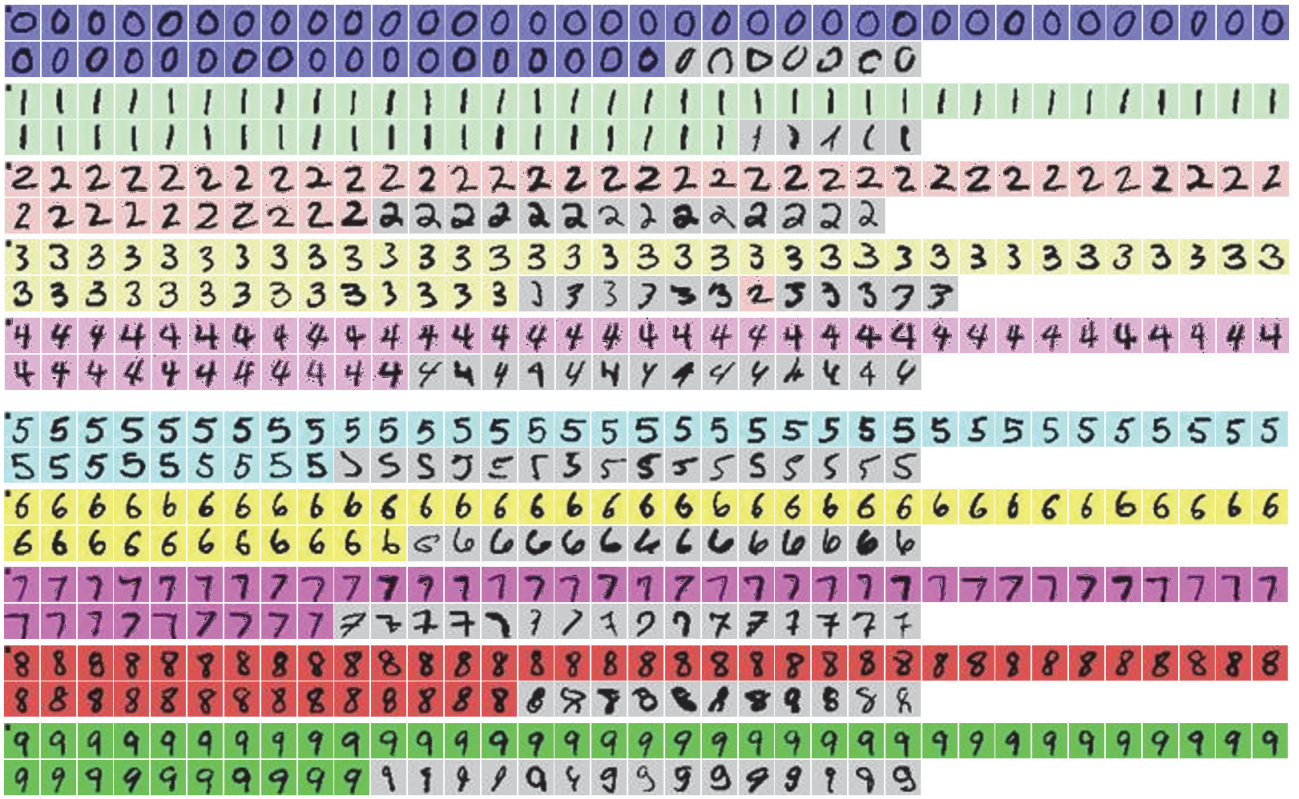
Step 4. Recognition process is carried out based on BP ANN. In our method, BP ANN is adopted according to paper [31].

First of all, we select 600 images for clustering to get cluster centers and other typical images for classifier to train. Those images contain numbers from 0 to 9, and each number has 60 images. Figure 9(a) is the cluster center self-determination process based on FDCA based on density and distance values. Different color is used to denote the different cluster centers. Figure 9(a) is the distribution of ρ_{sort} and δ_{sort}. Figure 9(b) is the result of number image cluster based on FDCA for all 600 images.

Before the strict filter is added into the method, the cluster results consist of two situations. (1) For image x, it is clustered into cluster A, while its true label is X_label, and A_label = X_label; then image x has been clustered correctly. (2) If case (1) is not established, then image x has been wrongly clustered. In order to make sure that training samples for classifier

	max ⟶								min	
ρ_{sort}	ρ_{600}	⋯	ρ_{582}	⋯	ρ_{574}	⋯	ρ_{571}	ρ_{570}	⋯	⋯
ρ_{val}	1.0000	⋯	0.7024	⋯	0.6369	⋯	0.6196	0.6166	⋯	⋯
ρ_{sort}	ρ_{558}	⋯	ρ_{527}	⋯	ρ_{517}	⋯	ρ_{512}	⋯	ρ_{495}	⋯
ρ_{val}	0.5763	⋯	0.5136	⋯	0.4984	⋯	0.4876	⋯	0.4603	⋯
δ_{sort}	δ_{600}	δ_{599}	⋯	⋯	δ_{597}	δ_{596}	⋯	⋯	δ_{594}	δ_{574}
δ_{val}	0.4887	0.4887	⋯	⋯	0.2921	0.2862	⋯	⋯	0.2689	0.2142
δ_{sort}	⋯	⋯	δ_{569}	⋯	δ_{564}	⋯	δ_{562}	⋯	δ_{533}	⋯
δ_{val}	⋯	⋯	0.2093	⋯	0.2050	⋯	0.2035	⋯	0.1853	⋯
img_lab	53	107	172	195	206	244	311	371	378	400
num	1	8	4	7	9	3	0	5	6	2

(a) Cluster center self-determination process based on FDCA



(b) The same color numbers belong to the same cluster, while those grey images do not belong to any cluster. For each cluster, those images with tiny black circle are cluster centers for each cluster

Figure 9: MNIST recognition results based on FDCA.

have been clustered as correctly as possible, we adopt strict filter to keep cluster pure through deleting cluster edge members.

As shown in Table 7, R is the radius parameter of strict filter denoted as the distance from cluster center. In other words, for strict filter with radius R, if the distance between cluster member and center is larger than R, then this member would be removed from the cluster to guarantee the purity of the cluster. With different filter R, we could achieve different clustering efficiency as shown in Table 7. $\overline{\gamma_{true}}$ denotes clustering accuracy, while $\overline{\gamma_{false}}$ denotes error rates. We can conclude from Table 7 that, without filters, recognition based on FDCA could achieve $\overline{\gamma_{true}} = 89.8\%$ and $\overline{\gamma_{false}} = 4.6\%$. The higher $\overline{\gamma_{true}}$ is, the higher $\overline{\gamma_{false}}$ is at the same time. On the contrary, the lower $\overline{\gamma_{true}}$ is, the lower $\overline{\gamma_{false}}$ is. The reason for this result is that the more strict filter is, more cluster members would be

Table 7: Performances comparison of different strict filters for MNIST data set.

Efficiency	R									
	0	$0.1 * dc$	$0.2 * dc$	$0.3 * dc$	$0.4 * dc$	$0.5 * dc$	$0.6 * dc$	$0.7 * dc$	$0.8 * dc$	$0.9 * dc$
$\overline{\gamma_{\text{true}}}$ (%)	89.8	88.6	88.0	85.5	85.0	72.0	61.2	53.3	43.2	39.2
$\overline{\gamma_{\text{false}}}$ (%)	4.6	4.5	4.5	4.0	3.0	2.1	1.3	0.2	0.0	0.0
Efficiency	R									
	$1.0 * dc$	$1.1 * dc$	$1.2 * dc$	$1.3 * dc$	$1.4 * dc$	$1.5 * dc$	$1.6 * dc$	$1.7 * dc$	$1.8 * dc$	$1.9 * dc$
$\overline{\gamma_{\text{true}}}$ (%)	38.8	35.8	32.7	31.6	26.1	23.3	17.9	17.6	17.6	16.5
$\overline{\gamma_{\text{false}}}$ (%)	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

excluded from cluster and the purer cluster would be, so $\overline{\gamma_{\text{true}}}$ would be low, with lower $\overline{\gamma_{\text{false}}}$ at the same time.

6. Conclusion

A novel fast density clustering algorithm (FDCA) for mixed data which determines cluster centers automatically is proposed. A unified mixed data similarity metric is defined to calculate data distances. Moreover, the CSA is used to fit the relationship of density and distance of every data object, and residual analysis is used to determine the centers automatically, which conforms to the original mixed data distribution. Finally, dichotomy analysis is adopted to eliminate parameter sensitivity problem. The experiments validated the feasibility and effectiveness of our proposed algorithm. Furthermore, our proposed FDCA is applied to number image recognition as an unsupervised method. MNIST data set is adopted to testify the high recognition rate with low false rate of our FDCA based method as a typical application. The future research will focus on the clustering data stream to achieve high clustering quality based on this work.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by a grant from the National Natural Science Foundation of China (no. 61502423), Zhejiang Provincial Natural Science Foundation (Y14F020092), and Zhejiang Natural Science Foundation (LY17F040004).

References

[1] J. Han and M. Kamber, *Data Mining Concepts and Techniques*, Morgan Kaufmann, San Francisco, Calif, USA, 2001.

[2] C.-C. Hsu, C.-L. Chen, and Y.-W. Su, "Hierarchical clustering of mixed data based on distance hierarchy," *Information Sciences*, vol. 177, no. 20, pp. 4474–4492, 2007.

[3] C.-C. Hsu and Y.-P. Huang, "Incremental clustering of mixed data based on distance hierarchy," *Expert Systems with Applications*, vol. 35, no. 3, pp. 1177–1185, 2008.

[4] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[5] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: an efficient data clustering method for very large databases," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 103–114, ACM, Montreal, Canada, June 1996.

[6] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD '96)*, Portland, Ore, USA, August 1996.

[7] Z. Huang, "A fast clustering algorithm to cluster very large categorical data sets in data mining," in *Research Issues on Data Mining and Knowledge Discovery*, pp. 1–8, ACM Press, Tuscon, Ariz, USA, 1997.

[8] Z. Huang and M. K. Ng, "A fuzzy k-modes algorithm for clustering categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 4, pp. 446–452, 1999.

[9] M.-S. Yang and Y.-C. Tian, "Bias-correction fuzzy clustering algorithms," *Information Sciences*, vol. 309, pp. 138–162, 2015.

[10] D. Barbara, J. Couto, and Y. Li, "COOLCAT: an entropy-based algorithm for categorical clustering," in *Proceedings of the 11th International Conference on Information and Knowledge Management*, pp. 582–589, ACM Press, McLean, Va, USA, November 2002.

[11] H. He and Y. Tan, "A two-stage genetic algorithm for automatic clustering," *Neurocomputing*, vol. 81, no. 1, pp. 49–59, 2012.

[12] A. Saha and S. Das, "Categorical fuzzy k-modes clustering with automated feature weight learning," *Neurocomputing*, vol. 166, pp. 422–435, 2015.

[13] S. Zahra, M. A. Ghazanfar, A. Khalid, M. A. Azam, U. Naeem, and A. Prugel-Bennett, "Novel centroid selection approaches for KMeans-clustering based recommender systems," *Information Sciences*, vol. 320, pp. 156–189, 2015.

[14] Z. Huang, "Clustering large data sets with mixed numeric and categorical values," in *Proceedings of the the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 21–34, World Scientific Publishing, Singapore, 1997.

[15] S. P. Chatzis, "A fuzzy c-means-type algorithm for clustering of data with mixed numeric and categorical attributes employing a probabilistic dissimilarity functional," *Expert Systems with Applications*, vol. 38, no. 7, pp. 8684–8689, 2011.

[16] I. Gath and A. B. Geva, "Unsupervised optimal fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 773–780, 1989.

[17] Z. Zheng, M. Gong, J. Ma, L. Jiao, and Q. Wu, "Unsupervised evolutionary clustering algorithm for mixed type data," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1–8, Barcelona, Spain, 2010.

[18] C. Li and G. Biswas, "Unsupervised learning with mixed numeric and nominal data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 4, pp. 673–690, 2002.

[19] D. W. Goodall, "A new similarity index based on probability," *Biometrics*, vol. 22, no. 4, pp. 882–907, 1966.

[20] C.-C. Hsu and Y.-C. Chen, "Mining of mixed data with application to catalog marketing," *Expert Systems with Applications*, vol. 32, no. 1, pp. 12–23, 2007.

[21] A. Ahmad and L. Dey, "A k-mean clustering algorithm for mixed numeric and categorical data," *Data & Knowledge Engineering*, vol. 63, no. 2, pp. 503–527, 2007.

[22] J. Ji, T. Bai, C. Zhou, C. Ma, and Z. Wang, "An improved k-prototypes clustering algorithm for mixed numeric and categorical data," *Neurocomputing*, vol. 120, pp. 590–596, 2013.

[23] J. Ji, W. Pang, C. Zhou, X. Han, and Z. Wang, "A fuzzy k-prototype clustering algorithm for mixed numeric and categorical data," *Knowledge-Based Systems*, vol. 30, no. 1, pp. 129–135, 2012.

[24] J.-Y. Chen and H.-H. He, "A fast density-based data stream clustering algorithm with cluster centers self-determined for mixed data," *Information Sciences*, vol. 345, no. 1, pp. 271–293, 2016.

[25] B. Everitt, S. Landau, and M. Leese, *Cluster Analysis*, Arnold, London, UK, 2001.

[26] G. David and A. Averbuch, "SpectralCAT: categorical spectral clustering of numerical and nominal data," *Pattern Recognition*, vol. 45, no. 1, pp. 416–433, 2012.

[27] Y.-M. Cheung and H. Jia, "Categorical-and-numerical-attribute data clustering based on a unified similarity metric without knowing cluster number," *Pattern Recognition*, vol. 46, no. 8, pp. 2228–2238, 2013.

[28] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[29] J.-Y. Chen and H.-H. He, "Research on density-based clustering algorithm for mixed data with determine cluster centers automatically," *Acta Automatica Sinica*, vol. 41, no. 10, pp. 1798–1813, 2015.

[30] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining*, Morgan Kaufmann, 2011.

[31] Z. Xiao, S.-J. Ye, B. Zhong, and C.-X. Sun, "BP neural network with rough set for short term load forecasting," *Expert Systems with Applications*, vol. 36, no. 1, pp. 273–279, 2009.