

Research Article

Structure Identification-Based Clustering According to Density Consistency

Chunzhong Li^{1,2} and Zongben Xu^{1,2}

¹ *Institute for Information and System Science, Xi'an Jiaotong University, Xi'an 710049, China*

² *Ministry of Education Key Lab for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049, China*

Correspondence should be addressed to Chunzhong Li, lichunzhongli@gmail.com

Received 22 March 2011; Revised 31 July 2011; Accepted 31 July 2011

Academic Editor: Wei-Chiang Hong

Copyright © 2011 C. Li and Z. Xu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Structure of data set is of critical importance in identifying clusters, especially the density difference feature. In this paper, we present a clustering algorithm based on density consistency, which is a filtering process to identify same structure feature and classify them into same cluster. This method is not restricted by the shapes and high dimension data set, and meanwhile it is robust to noises and outliers. Extensive experiments on synthetic and real world data sets validate the proposed the new clustering algorithm.

1. Introduction

As one important stage of the data mining process, data clustering is a division of data into groups of similar objects. Each group, called cluster, consists of objects similar among themselves and dissimilar to other groups. Representing data by fewer clusters necessarily loses certain fine details, but achieves compact representation. Clustering is the basis of pattern recognition and knowledge discovery. Tremendous literatures on clustering techniques have appeared since the last century. These techniques are widely applied in image segmentation [1], information retrieval [2], text mining [3], image quantization [4], and so on.

Clustering algorithms can be basically divided into two categories of approaches, the partitional approach [5, 6] and the hierarchical approach [7–9]. The partitional approach generally obtains a partition of the data set by minimizing a cost function, which is commonly modeled to prefer the clusters with maximal intracluster similarities and minimal inter-cluster similarities. The existing algorithms that belong to this kind of approach include FORGY [10], ISODATA [11], WISH [5, 6], and so forth.

Hierarchical clustering algorithms are the most commonly used clustering techniques specially useful in biology and medicine. The basic idea of such algorithms is to successively cluster a data set into clusters in a hierarchical fashion. Hierarchical clustering algorithms are able to produce multilevel clusters, which is necessary for the applications in biology and other applications. Examples of hierarchical clustering algorithms are CURE [7], ROCK [8], and Chameleon [9].

Density-based algorithm is a kind of classical clustering algorithm. One method of density clustering algorithm is based on the assumption for the whole data set, and the other method is based on the local density feature to mine the structure of data set.

In order to assume the distribution of the whole data set to clustering, the first thing is to suppose different distribution functions on different density zones of data set, and which commonly uses the nonparametric density estimation. Using the nonparametric density estimation with a kernel function to estimate the distribution, the clusters of a data set are obtained by repeatedly moving the data to the local maximal points of the estimated density. There exist density-based clustering algorithms such as famous Mean Shift [12–15], CSSF [16], and so on. The computation of the algorithm is simple and the cluster number is not required to be known in advance. However, they tightly depend on the density estimation quality of the data set. Without any prior knowledge on structure of the data set, it is difficult to provide a good density estimation to fit the data set, so the clustering results might not be desirable for real data sets or images.

Describing structure of data set by mining the local density feature, the clustering is to agglomerate the data points with same density feature. For a data set, the density difference of each data point can be measured with k -NN neighborhood, in which the larger k -NN neighborhood represents that the local density is low or the data points in this region are sparse. The data points classified into same cluster should have same local density value or same radius of k -NN neighborhood. The work in [17] and DBScan [18] are of this kind. This kind of method need not to assume the distribution of the whole data set and are not be restricted the shape of data set. However, the algorithms need to compare the density difference to agglomerate the data points.

The present research is a new clustering based on density consistency to cluster data set with filtering process. Through describing the local structure feature of data set, the data points with same feature can be classified into same cluster. This new clustering algorithm is not restricted by the shapes and high dimension data set, meanwhile it is robust to noises and outliers.

The reminder of this paper is organized as follows. In Section 2 we introduce some useful structure features of data set, which is important to clustering problems. Section 3 introduces the clustering process, includes the modeling of filters, the feature extraction process and feature integration process following with top-down process. The experiments and conclusion are given in Sections 4 and 5.

2. Density Feature of Data Sets for Clustering

Structures of data sets and clustering are closely related. It is commonly considered that the clusters are made of some discrete data points according to some structure features, and these structure features contain local and large scale structure features. Local structure is mainly described with the spatial relationship between near data points, which is also called neighborhood relationship. Local spatial relationship is the most original and elementary for clustering problem. As shown in Figure 1(a), there are lots of tiny clusters, which may be

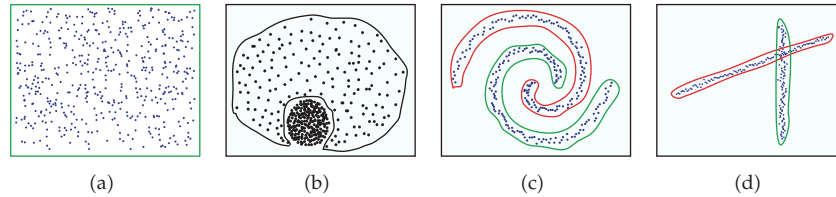


Figure 1: Structure features of data sets. (a) A data set with no structure. (b) A data set with density feature: two clusters of data with different densities. (c) A data set with connectedness feature: it contains two connected manifolds. (d) A data set with orientation feature.

extracted at local scale though the data set follows uniform distribution and has only one cluster at large scale.

The large scale structure features mainly contain density, connectedness, and direction, and we will give explanation to these structure features in the following section.

Density Feature

Density characterizes the distribution feature of a data set, which can be measured with the number of data in a certain volume of pattern space. According to the proximity and similarity laws in psychology, a region of data set with higher density is prone to be recognized as a cluster by human eyes. See Figure 1(b) for example, where two regions of data with different densities are very naturally perceived to be two different clusters, despite the fact that they are very close to each other. A uniformly distributed data set is naturally accepted as no feature because no visual difference is observed (see Figure 1(a)).

Connectedness Feature

The neighbor regions with local structure feature may connect each other and form curve or manifold structure as Figure 1(c). By the Gestalt continuity law, the connected data are easily perceived to be a same cluster, but the disconnected data are prone to be perceived as different clusters.

Direction Feature

If the local regions connect each other and have same principal direction, then the principal direction is called the direction feature of the data set, see Figure 1(d) as an example, which has two different direction features.

The local and large scale structure features are the most typical features for valid clustering of data set. Local structure features can form large scale structure features with bottom-up process while the large scale features provides feedback information with top-down process. In next section, we will introduce clustering process to extract and integrate these features into clusters.

3. Clustering Process

Since the data set to be clustered has many native structure features, in this section we propose feature extraction and feature integration methods to obtain valid clustering results.

Let $X = [x_1, x_2, \dots, x_N]_{N \times d}$ be a pattern matrix and $x_i^T \in R^d$ be a pattern. The data set to be clustered can be viewed as a kind of imaginary image with its native structure features, and then the clustering problem is viewed as a cognition problem. To extract the structure features, we employ filtering methods in the theory of vision research. Given a stimuli, the response of a neuron in primary visual system can be measured

$$f(x; \Theta) = I(x) * K(x; \Theta) = \int I(y - x) \cdot K(y; \Theta) dy, \quad (3.1)$$

where $*$ is convolution operation, $I(x)$ is stimuli (an image or imaginary image), $K(x; \Theta)$ is a series of filters (or kernel function), and Θ is a set of parameters. In (3.1), $f(x; \Theta)$ is the response of the neuron with stimuli $I(x)$, which is the filtering response and is called as a *feature* of $I(x)$.

Data sets may have many native structure features, which need to be extracted based on (3.1) with different filters (or kernels). These filters should be determined automatically to deal with different structure features of different data sets. Next we introduce how to construct filters automatically based on these salient structural features.

3.1. Modeling of Filters to Extract Local Structure Features

Like image processing, each data point is corresponding to the center of a filter, and for clustering problem here we model a new type of low-pass filter

$$K(x_i; x_{(j|i)}, k) = \exp_{x_{(j|i)} \in \Omega(x_i; k)} \left(-(x_i - x_{(j|i)}) A (x_i - x_{(j|i)})^T \right), \quad (3.2)$$

where $\Omega(x_i; k)$ is k -NN of x_i with new special method, and $x_{(j|i)}$ is the j th neighbor of data point x_i . The matrix A is a covariance which can control similarity measurement according to the neighborhood relationship and structure of data set.

To determine the special neighborhood for each data point is very important for the new clustering algorithm, and the aim is to depict $\Omega(x_i; k)$ according to density difference. Then we will introduce how to determine the data set $\Omega(x_i; k)$ for each data point x_i and how to scale the neighborhood $\Omega(x_i; k)$ according to matrix A .

3.1.1. Determination of Neighborhood

Unlike classical k -NN strategy, we use a new method to find nearest neighborhood for each data point. First, we initialize k nearest neighborhood data points [19] in which each new adding data point is the nearest to origin data set, then we select proper neighbors among the initial k data points.

Definition 3.1 (acceptable neighborhood radius). Each data point has k initial neighbors, and $d_i(x)$ is the distance between i th new adding nearest neighbor x_i while data set containing origin i data points. Then the neighborhood radius of the data point x is defined as

$$r(x; k) = \frac{1}{k} \sum_{i=1}^k d_i(x). \quad (3.3)$$

Take Figure 2(a) as an example, in which we find the nearest 4 data points from a . First the data point b is the nearest to a and its distance is d_1 , then a and b can be viewed as a whole, and the second nearest data point to a is c rather than d , its distance is d_2 . So in this way the *acceptable neighborhood radius* of data point a is $(d_1 + d_2 + d_3 + d_4)/4$.

With the definition, each data point is corresponding to a neighborhood radius. In order to select true neighbors for a data point, the neighbors should have similar features as the data point. So the consistency criterion is built up.

Definition 3.2 (internal consistency criterion (ICC)). If data point y is neighbor of data point x according to Euclidean distance, they must satisfy

$$\begin{aligned} \max\{r(x; k), r(y; k)\} &\leq \alpha \cdot \min\{r(x; k), r(y; k)\}, \\ d_i &\leq \alpha \cdot r(x; k), \end{aligned} \quad (3.4)$$

where d_i is the i th adding distance in (3.3) and α is a parameter.

The aim of the ICC is to wipe density different data point out. With the criterion, the neighbors of each data point can be determined by the structures of data set, meanwhile the selection mechanism for neighbors can also explore the structures of data set. In Figure 2(b), the neighbors of data point a are b, c, d , and e according to classical k -NN. However, the data point a and b are not satisfied with the two-criterion (3.4), so the data point a has no neighbor and it is an outlier. The two criteria can also detect the density structure as shown in Figure 2(c), and the initial 6 nearest neighbors are b, c, d, e, f , and g . The first three data points b, c and d are the neighbors of a because they satisfy the criterions. However, the data points d, e and distance d_4 are not satisfied the criterions, so the number of nearest neighbors are 3, $\{b, c, d\}$. Thus, the data points e, f , and g are not its neighbors. Note that the number of nearest neighbors may not be the initial number k . Note that the method to select neighbors according to ICC is not related to sequence of nearest neighbors.

3.1.2. Local Scale and Direction of Neighborhoods

Combined with neighbors, data point x has a neighborhood $\Omega(x_i; x, k)$ which is important to depict the interaction between x and each of its neighbors. The matrix A in (3.2) cannot only provide the scale restriction, but endow each data point with the direction feature. Then the matrix A can be concreted by employing covariance matrix

$$B(x) = \frac{\sum_{x_i \in \Omega(x_i; x, k)} (x - x_i)^T (x - x_i)}{|\Omega(x_i; x, k)|}, \quad (3.5)$$

where $|\cdot|$ is the number of neighbors in the neighborhood. Then in (3.2), the local scale matrix can be represented with

$$A(x) = B^{-1}(x). \quad (3.6)$$

Whenever B is irreversible, we add each diagonal element of B by an any small positive constant.

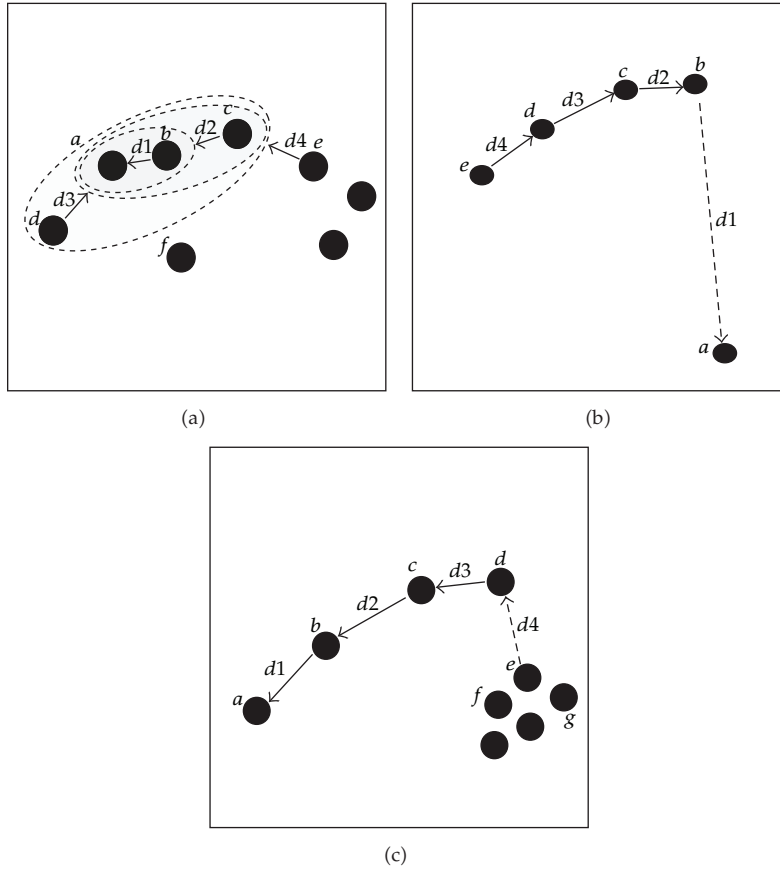


Figure 2: Criteria to determine nearest neighbors for each data point. (a) Initial 4 nearest neighbors for data point a . (b) An example to determine neighbors for data point e according to the two criteria. (c) An example to determine neighbors for data point a according to the two criteria.

With the local scale matrix A in (3.2), the neighborhood of each data point has a local scale which can restrict the similarity (or the relationship) between each pair of data points in a proper scope. Meanwhile, the biggest eigenvector of $A(x)$ is the local direction at the point x which can provide other agglomerate criterion and feedback information to refine clusters.

3.2. Agglomeration Process of Clusters

In this subsection, we introduce the agglomerate process of clusters, in which different clustering feature can be extracted, from local features into large scale features and then clusters. The main process contains three steps: feature extraction with local similarity, feature cognition with large scale structure features and the reidentification with top-down process. In the procedure of feature extraction, the data set is successively filtered (convolved) with multiscale filters, yielding various feature representation of the data set. In the procedure of feature cognition, the clustering result is obtained by integrating various large scale features. In top-down process, the clusters are rechecked to form satisfied ones, such as noisy data points, manifold clusters, clusters with principle direction, and so on.

3.2.1. Extraction of Local Scale Features with Filtering Process

Local scale features embody the local structures of data sets which are based on the local similarity. Neighbor data points with close relationship can form local structure features, such as the density feature of the neighborhood, shape, and principal direction of the neighborhood and so on. In order to extract these local scale features automatically, we use a series of self-tuning filters to agglomerate local neighborhood data points.

Let $X(t)$ be the feature of data set extracted at t -layer, and $X(0)$ simply corresponds to X . Then $X(t)$ can be expressed as

$$X(t+1) = U(X)X(t), \quad X(0) = X, \quad t = 1, 2, \dots, \quad (3.7)$$

where U is a data-driven filtering matrix according to (3.2)

$$U(X) = [u_{ij}]_{1 \leq i, j \leq N} = \left[\frac{K(x_i; x_j, k)}{\sum_{s=1}^N K(x_i; x_s, k)} \right]_{i, j=1, \dots, N}. \quad (3.8)$$

The filtering matrix is a sparse matrix, in which each element u_{ij} represents the relationship between i and j . If they are not neighbors, the element is zero.

The representation (3.7) defines a scale space $\{X(t+1) = U(X)^t X(t) \mid t \geq 0\}$, which we call the *discrete scale space (DSS)* of the data set deduced from its feature. For definiteness, we assume that the feature extraction procedure is accomplished within T layers of processing. Furthermore, it can be proved that $X(t)$ defined by (3.7) will be finally stabilized as $t \rightarrow \infty$. So we can also assume that T is so large that $X(t)$ reaches to a stationary state when $t \geq T$, when the data points with strong relationship will shrink into a same destination to form a cluster.

3.2.2. Integration of Large Scale Feature Clusters

At local scales, the data points with close relationship will be grouped together to form local structure features, and these structure features are rudiments of clusters. These rudiments of clusters then can be grouped according to some aggregative properties to final clusters. Next we introduce the integration of large scale features.

The two similar regions (clusters) merge together, with the precondition that the two regions (clusters) are close, and the average distances of each one are almost equal. The distance of sets is represented with $d(C_i, C_j) = \min\{d(x, y) \mid x \in C_i, y \in C_j\}$. As for the distance between the two clusters, we use the modified distance of sets. m minimal distances of clusters are selected and the average is set to be the intracluster distance. In this paper, the value is set as 3 to make the algorithm more robust. Then the average distance of cluster is defined as follows.

Definition 3.3 (average distance of cluster). C_i is the i th cluster obtained with filtering process and $x \in C_i$, the neighborhood of x is $\Omega(x'; x, k)$, then the average distance $d(C_i)$ of cluster C_i is defined as

$$d(C_i) = \sum_{x \in C_i} \sum_{j=1}^{|\Omega(x'; x, k)|} \frac{d_j(x)}{|\Omega(x'; x, k)|}. \quad (3.9)$$

As the discussion above, the two clusters C_i and C_j merge together when the two criteria are satisfied as

$$\begin{aligned} \max\{d(C_i), d(C_j)\} &\leq \beta \cdot \min\{d(C_i), d(C_j)\}, \\ d(C_i, C_j) &\leq \beta \cdot \min\{d(C_i), d(C_j)\}. \end{aligned} \quad (3.10)$$

With the definitions of average distance and the intradistance between two close clusters, also with the agglomerating criterions (3.10) the local structure features will integrate into large scale structure features, such as connectedness and direction features. This procedure is a rough agglomerating and a bottom-up procedure, in which different local structure features may merge together. Then the Top-down process should be considered to analyze whether these local features should merge or not.

3.2.3. Top-Down Procedure

The identification of noisy data points and large scale structure features of data set are difficult for clustering problem and are parallel with clustering algorithm. Here the identification process embeds Top-down procedures because the agglomerating process is only a bottom-up procedure without any large scale feature information. In this paper, the identification of noisy data points, connectedness, and direction features are considered.

Identification of Noisy Data Points or Outlier

Lots of noisy data points or outliers may appear with the local features agglomeration in the filtering process. These data points strongly depend on the local similarity and may not be true noises, and whether they are noisy data points or not depends on the large scale structure features after agglomeration.

Some outliers appear in the seeking of neighbors for each data point, and then some of these outliers may agglomerate in clusters in the filtering process. To avoid this, the outliers should be identified whether they are noisy or not, or whether they should be classified into close clusters according to the statistical properties (such as mean and variance of Gaussian). An example is given in Section 3.2.4 to demonstrate the identification procedures.

Identification of Connectedness and Direction Feature

With the whole agglomeration procedure, fine clusters merge into coarse clusters, and the number of clusters decrease. The coarse clusters are made of small fine clusters which have their own local principal directions. The fine clusters are regions with various shapes, and the region of a fine cluster C_i can be measured with covariance matrix

$$\Sigma_i = \sum_{x \in C_i} (x - \hat{x})^T (x - \hat{x}), \quad (3.11)$$

where \hat{x} is the mean of the local cluster. The principal direction of Σ_i is the local direction of the cluster. In agglomerating procedure into coarse clusters, if the directions of subagglomerates are almost the same, the coarse cluster has a direction feature.

Input: Data set X , in which each row is a pattern; parameters: k , α , β .

Output: Label vector L , in which each element is cluster label.

Step (1): Find k nearest neighbors for each data point x according to Euclidean distance.

Step (2): Find nearest neighbors for each data point x according to Definition 3.2.

Step (3): Construct filtering matrix U based on (3.2) and (3.8).

Step (4): Filtering process $X = UX$, obtain elementary clusters, which are the element with various kind of structure features. Top-down process is included to identify connectedness and direction.

Step (5): Integrate elementary clusters with same structure features into meaningful clusters according to (3.9).

Step (6): Top-down process to identify noisy data points and outliers.

Algorithm 1: The process of new clustering algorithm.

3.2.4. Algorithm Outline

The clustering algorithm includes three procedures, the local structure feature extraction with filtering process, integration of large scale features, and the top-down process to refine clustering results. The clustering algorithm is summarized by an outline in Algorithm 1.

An example is shown in Figure 3, where two Gaussian data sets have different mean and variance. From left to right, and from the top to down are the agglomerating procedure of clustering. The second subfigure is the result of local features integration under filtering procedure, and then the second subfigure is the result after the integration of large scale structure features. And the last is the result with top-down process, where some noisy data points attribute to the two final clusters according to the variance of the two main clusters.

3.2.5. Complexity of the Algorithm

The high complexity of clustering algorithm is difficulty to deal with large datasets, and meanwhile the preprocessing origin datasets is expensive. So in the scope of this paper, we do not intend to solve efficiently the preprocessing. The KNN procedure (typically $k = 40$) is used in step (1) of the new algorithm. In low dimension, KNN has complexity of $O(N \log N)$, while in high dimension the complexity is $O(N^2)$, where N is the number of data points.

The complexity of our clustering algorithm is $O(N)$. In the filtering process, the complexity is not more than $O(N \cdot k \cdot d)$ because the filtering matrix is a sparse matrix, in which the number of each nonzero element in each row is not more than k (typically $k = 40$), and d is the dimension of each pattern. So the complexity of the filtering process is $O(N)$. In the integration procedure, the complexity is $O(N_c^2)$ and N_c is the number of elementary clusters. In practice $N_c \ll N$, and the total complexity of the algorithm is linear in the size of the data set, which is much lower than CSSF [20] and VClust [21] ($O(N^3)$).

4. Experiment

We provide a series of experiments and applications in this section to demonstrate the effectiveness of the proposed clustering algorithm. The test data sets include synthetic data sets

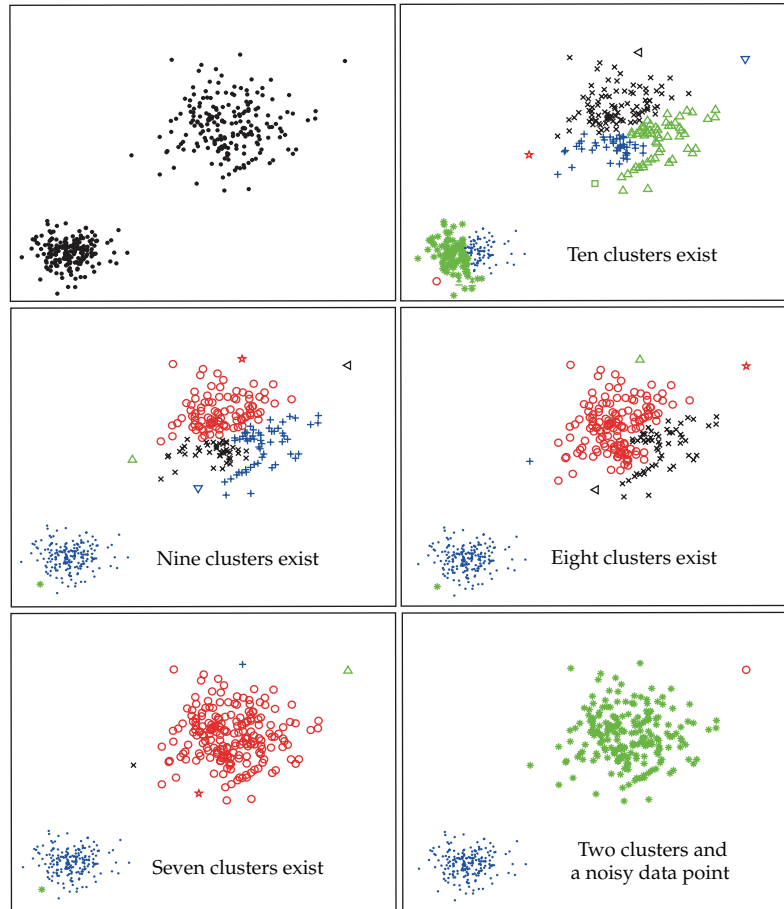


Figure 3: An example to show the clustering procedure.

and a set of real world data sets from public domains that are extensively used in testing the performance of classical clustering algorithms. The synthetic data sets are specially generated to exhibit complex structure features, such as the unbalanced and complex manifold shaped. The real world data sets are those derived from image segmentation and a runway detection task from remote sensing image.

We have applied the new algorithm to the above data sets with comparison to several well-known clustering algorithms. These algorithms are known to be representatives of the existing clustering approaches. The method of clustering by scale space filtering-(CSSF-) [20] in the vision-simulation-based approach, the Chameleon [9] method in the graph-based approach, the spectral-Ng method [22] in the spectrum-based approach, Mean Shift algorithm [15] in the density-based algorithm, and the latest LEGClust algorithm [23] from the information entropy-based approach.

In experiments, we have used the normalized mutual information (NMI) [24], as the criterion of measuring the performance of a clustering algorithm, whenever the data sets are of high dimension. There are three parameters in the algorithm: initial neighbor parameter k , agglomerating criterion parameters α and β . These parameters are very stable for the algorithm, k (usually $10 \leq k \leq 40$) is only an initial number of neighbors for each data point

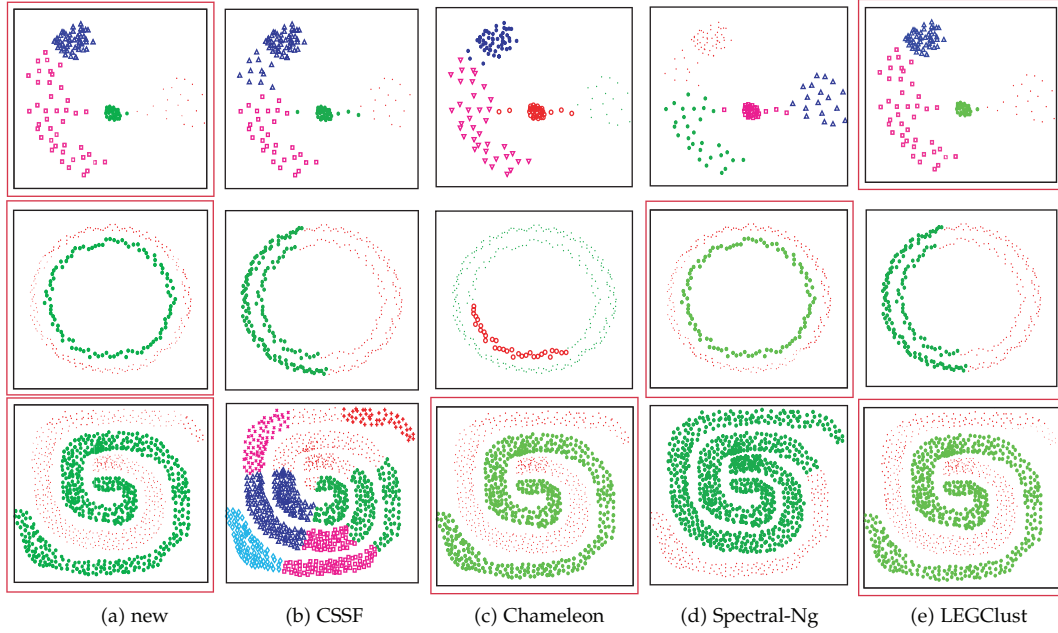


Figure 4: Comparing results with different classical algorithms on three toy data sets. (a) Clustering results with the new algorithm on three data sets. (b) Clustering results with the CSSF. (c) Clustering results with Chameleon. (d) Clustering results with Spectral Ng. (e) Clustering results with LEGClust.

and proper neighbors are less than k , as the same, the other two parameters are very stable. Usually in this paper, $k = 40$, $\alpha = 4$, and $\beta = 2$.

The three toy data sets and the results of different clustering algorithms are shown in Figure 4, where each row is a synthetic data set, and each column contains the results of three data sets with a same algorithm. We have employed 50 people who are familiar with clustering problem to evaluate these results with the figures, and most acceptable results are labeled with red box. From the results with each cluster algorithm (each column), the new algorithm and VClust are better than the others. The only advantage of the new algorithm is that the computation load is lower ($O(N^2)$) than VClust ($O(N^3)$).

We further compare the new algorithm with other clustering algorithms by applying to real world problems. The real data sets are from UCI and USPS, where the noisy data points follow uniform distribution. In Iris data set, the 30 noisy data points are set while in USPS-01 the 100 uniform noisy ones are set. Due to the uniform unsatisfying performance of the density-based algorithms in afore conducted simulations, we compare the the algorithm only with the graph-based algorithms whose results are more better. Thus, as representatives, Chameleon, spectral-Ng, noise robust spectral clustering (NRSC) [24], and self-tuning spectral clustering (STSC) [25] are selected to apply, as compared with the new algorithm. All the algorithms are applied to the four real data sets, shown in Table 1.

The clustering algorithms are applied in two natural images, which are shown in Figure 5. The images are composed of regions with different textures and gradually changed colors. Two well-known algorithms, which are very successful in image segmentation, are normalize cut [26] and mean shift [15]. We apply the new algorithm and CSSF in the chosen images, and compare the results with normalized cut and mean shift. The yielded segmentation results are shown in Figure 5. Note that the second column shows one kind

Table 1: Clustering results when different clustering algorithms, applied to four real world data sets.

Data-set	New algorithm	NMI			
		NRSC	Spectral-Ng	STSC	Chameleon
Iris	0.8623	0.7612	0.8058	0.7612	0.7652
Noisy Iris	0.8623	0.7779	0.5759	0.5881	0.7652
Pendigit	0.6429	0.5899	0.4902	0.5107	0.7706
Noisy USPS-01	1.0	1.0	0.6234	0.7439	0.8972

**Figure 5:** Image segmentation using different clustering algorithms. From left to right: original image, manual drawing, the new algorithm, CSSF and Normalized Cut.

of acceptable segmentation results through manual labeling, and the result of mean shift is similar as normalized cut.

We also apply the new algorithm to the following identification problem to detect/locate the airport runway from remotely sensed image shown as in Figure 6(a), which comes from LANDSAT thematic mapper (TM) data acquired over a suburb area in Hangzhou, China. The region contains the runway and parking apron of a certain civilian aerodrome. The image consists of a finite rectangular 95×60 lattice of pixels. Similar to [27], we selected Band 5 as a feature variable. A feature subset of data shown in Figure 6(b) can then be extracted by using a simple technique which selects a pixel point when its gray-level value is above a given threshold (e.g., 250).

In [27], regression class mixture decomposition (RCMD) method was proposed to mine line objects in the remotely sensed image. Using a feature subset of data in Figure 6(b), RCMD method successfully detected two runways, which can be viewed as two regression classes. At $t = 0.05$ level, two line equations identified by RCMD method are $y = 0.774x + 34.874$ and $y = 0.341x + 22.717$. The detected two runways are depicted in Figure 6(c). Here we take this detection problem as a clustering problem, and then apply the new clustering algorithm. The obtained clustering results are shown in Figure 6(d), where each cluster corresponds to a runway. In the procedure of clustering with the new clustering algorithm, two principal directions have been obtained by finding the mean principle directions from the anisotropy matrices in each cluster. Therefore the detected two runways with the new algorithm can be described by $y = 0.332x + 23.04$ and $y = 0.714x + 37.362$, as shown in Figure 6(e). It is seen that the new clustering algorithm has been applied to yield an almost similar result with RCMD. Nevertheless, we observe that the new clustering algorithm provides a much more simpler approach to the identification problem than RCMD does.

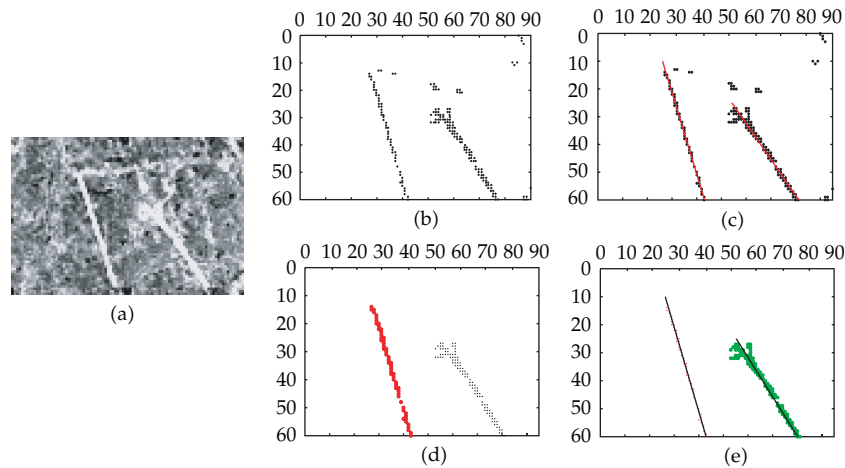


Figure 6: Runway detection in remote sensory image. (a) Remote sensory image. (b) Pixels extracted from image. (c) Detected result with RCMD. (d) Clustering result with the new algorithm. (e) Two detected runway with the new algorithm.

5. Conclusion

We have suggested a new clustering by identifying structure feature based on density consistency. Through viewing a data set as an (imaginary) image and explaining the data clustering as a cognition problem, it solves the problem via mimicking the mechanism of visual information processing. First, the results with the new algorithm are most close to the cognition of people for low dimension complex structure data sets. Second, it can deal with various structures of subspace clusters and is not restricted by shapes and high dimension. Third, almost no parameters has to be considered and it is easily to use. Fourth, the computation load is much lower than CSSF and VClust.

References

- [1] A. K. Jian and P. J. Flynn, "Image segmentation using clustering," in *Advances in Image Understanding: A Festschrift for Azriel Rosenfeld*, pp. 65–83, Wiley-IEEE Computer Society, 1996.
- [2] N. Jardine and C. J. van Rijsbergen, "The use of hierarchic clustering in information retrieval," *Information Storage and Retrieval*, vol. 7, no. 5, pp. 217–240, 1971.
- [3] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *Proceedings of the 6th ACM SIGKDD World Text Mining Conference*, Boston, Mass, USA, 2000.
- [4] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publications, Norwell, Mass, USA, 1992.
- [5] R. Dubes and A. K. Jain, "Clustering techniques: the user's dilemma," *Pattern Recognition*, vol. 8, no. 4, pp. 247–260, 1976.
- [6] M. R. Anderberg, *Cluster Analysis for Applications*, Academic Press, New York, NY, USA, 1973.
- [7] S. Guha, R. Rastogi, and K. Shim, "CURE: an efficient clustering algorithm for large databases," in *Proceedings of the ACM International Conference on Management of Data*, pp. 73–84, 1998.
- [8] S. Guha, R. Rastogi, and K. Shim, "Rock: a robust clustering algorithm for categorical attributes," *Information Systems*, vol. 25, no. 5, pp. 345–366, 2000.
- [9] G. Karypis, E. H. Han, and V. Kumar, "Chameleon: hierarchical clustering using dynamic modeling," *Computer*, vol. 32, no. 8, pp. 68–75, 1999.
- [10] E. Forgy, "Cluster analysis of multivariate data: efficiency versus interpretability of classifications," *Biometrics*, vol. 21, pp. 768–769, 1965.

- [11] G. Ball and D. Hall, "A clustering technique for summarizing multivariate data," *Neural Computation*, vol. 12, pp. 153–155, 1976.
- [12] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with application in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, no. 1, pp. 32–40, 1975.
- [13] W. Tao, H. Jin, and Y. Zhang, "Color image segmentation based on mean shift and normalized cuts," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 37, no. 5, pp. 1382–1389, 2007.
- [14] D. Comaniciu and P. Meer, "Mean shift analysis and applications," in *Proceedings of the 7th IEEE International Conference on Computer Vision (ICCV'99)*, pp. 1197–1203, September 2000.
- [15] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.
- [16] Y. Leung, J. S. Zhang, and Z. B. Xu, "Clustering by scale-space filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1396–1410, 2000.
- [17] D. Kushnir, M. Galun, and A. Brandt, "Fast multiscale clustering and manifold identification," *Pattern Recognition*, vol. 39, no. 10, pp. 1876–1891, 2006.
- [18] M. T. Ester, H. P. Kriegel, J. Scander, and X. W. Wu, "DBScan: 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*, 1996.
- [19] E. Gokcay and J. C. Principe, "Information theoretic clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 158–171, 2002.
- [20] Y. Leung, J. S. Zhang, and Z. B. Xu, "Clustering by scale-space filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1396–1410, 2000.
- [21] C. Z. Li, Z. B. Xu, and J. Sun, "Visual clustering: an approach inspired from visual perception," submitted to *IEEE Transactions on Systems, Man, and Cybernetics, Part B*.
- [22] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering analysis and algorithm," in *Advances in Neural Information Processing Systems*, pp. 849–856, MIT Press, 2001.
- [23] J. M. Santos, J. Marques, and L. A. Alexandre, "LEGClust—a clustering algorithm based on layered entropic subgraphs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 1, pp. 62–75, 2008.
- [24] Z. Li, J. Liu, S. Chen, and X. Tang, "Noise robust spectral clustering," in *Proceedings of the IEEE 11th International Conference on Computer Vision*, October 2007.
- [25] M. L. Zelnik and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems*, pp. 1601–1608, MIT Press, 2005.
- [26] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.
- [27] Y. Leung, J. H. Ma, and W. X. Zhang, "A new method for mining regression classes in large data sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 5–21, 2001.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

