QCC:A novel cluster algorithm based on Quasi-Cluster Centers

Jinlong Huang[1], Qingsheng Zhu[1], Lijun Yang[1], Dongdong Cheng[1]

[1]Chongqing Key Lab. of Software Theory and Technology, College of Computer Science,
Chongqing University, Chongqing 400044, China
Email: qszhu@cqu.edu.cn; 352720950@qq.com

Abstract—**Cluster analysis is aimed at classifying elements into categories on the basis of their similarity. And cluster analysis has been widely used in many areas such as pattern recognition, and image processing. In this paper, we propose an approach based on the idea that the density of cluster centers are highest in its k nearest neighborhood or reverse k nearest neighborhood, and clusters is divided by sparse region. We firstly define the similarity between clusters. Based on this idea, no matter non-spherical data or complex manifold data, the proposed algorithm is applicable. And the proposed algorithm has a certain capacity on outliers detection. We demonstrate the power of the proposed algorithm on several test cases. Its clustering performance is better than DBSCAN, DP and K-AP clustering algorithms.**

Keywords—Cluster; Center; Similarity; Neighbor; manifold

I. Introduction

Clustering is a primary method of data mining and data analysis. The aim of clustering is to classify elements into categories, or clusters, on the basis of their similarity. Clusters are collections of objects whose intra-class similarity is high and inter-class similarity is low. Now the study on clustering algorithm has been very active. Several different clustering methods have been proposed[1]. They can be roughly divided into Partitioning Methods[2-4], Hierarchical clustering[5-7], Density-Based Clustering[8-9], Grid-Based Clustering[10-11], Model-Based Method[12-13].

For many clustering algorithm, it is important that finding cluster center to clustering. In K-means[2] and K-medoids[3] methods, the data was classified to a cluster by a small distance to the cluster center. An objective function, typically the sum of the distance to a set of putative cluster centers, is optimized until the best cluster centers candidates are found. In 2007, Brendan and Delbert proposed a new clustering algorithm by passing messages between data points, called "affinity propagation"(AP)[14]. AP takes as input measures of similarity between pairs of data points. Real-valued messages are exchanged between data points until a high-quality set of exemplars and corresponding clusters gradually emerges. However, AP clustering algorithm con not directly specify the final class number. In order to generate K clusters, Zhang et al. proposed K_AP clustering algorithm[15].

However, these above center-based methods are not able to detect non-spherical clusters[16], since data points are always assigned to the nearest center. In 2014, Rodriguez et al. proposed a new clustering algorithm in Science, called DP[17]. The DP algorithm has its basis in the assumptions that cluster centers are surrounded by neighbors with lower local density and that they are at a relatively large distance from any points with a higher local density. And the non-spherical shape clusters can be easily detected by DP[17] clustering algorithm. But DP is not application to complex manifold data sets. In 2014, Hongjie et al. proposed the DAAP clustering algorithm[18] that can solve the complex manifold problem by computing the particular similarity that defined in paper[18]. However, the time complexity of DAAP is higher than AP, K_AP and DP for computing the particular similarity that the sum of the Edge-Weight in shortest path. And the clustering result of DAAP is effected by many parameters such as the number of neighbors and clusters, damping coefficient, the maximum iteration. Moreover, generally, the clustering effect of DAAP is bad on datasets with noise points. Detailed description see paper[18].

In this paper, we propose a new clustering method. The proposed algorithm has its basis in the assumptions that the density of cluster centers is the maximum of its neighbors or reverse neighbors and clusters are divided by sparse area. Similar to the above methods, it based on the cluster centers. However, unlike these above methods, the cluster centers of the proposed algorithm are not 'real' cluster centers that one center corresponds to one final cluster, but the Quasi-Cluster Center that one Quasi-Cluster Center corresponds to one initial cluster. Then we obtain the final cluster by merging the clusters that the similarity is greater than alpha. The proposed clustering algorithm and it's related definitions will be detailed descripted in section 3.

II. Related Work

Most of density based clustering algorithms, such as DBSCAN, DP, AP and K-AP, define the number of neighbors that distance is smaller than d_c as the density of each point. And the density of points as the follows formula.

$$\rho_i = \sum_j^n X(d_{ij} - d_c) \tag{1}$$

Where $X(d_{ij} - d_c) = 1$ if $d_{ij} - d_c < 0$ and $X(d_{ij} - d_c) = 0$ otherwise, and d_c is a cutoff distance. Basically, ρ_i is equal to the number of points that are closer than d_c to point

DOI reference number: 10.18293/SEKE2016-028

i. However, once the intra-class density variations is great, the value of d_c is hard to set. For example, as shown in Figure 1, if the value of d_c is set inappropriately, then there are no neighbors in the neighborhood of point a and b within d_c, but the neighbors of the center of C_1 include all of the points of C_1. Moreover, although point a and b are normal point, they will be regard as the noise in density-based algorithms. And point a and b will be regard as the cluster center in DP algorithm, since the local density of point a and b is the biggest in its neighborhood.

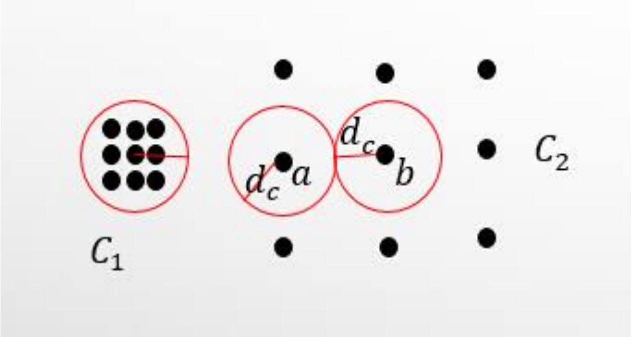

Figure 1. the density variations

As the most of exist density based clustering algorithm, the proposed method need to compute the density of every point so that we can get the cluster centers that the density peaks. In order to avoid the above question, we introduce the follow definitions.

Let D be a database, p and q be some objects in D, and k be a positive integer. We use d(p,q) to denote the Euclidean distance between objects p and q.

Definition1 (K-distance and Density): The k-distance of p, denoted as $K_{dist}(p)$, is the distance d(p,o) between p and o in D, such that:

- (1) For at least K objects o' ∈ D/{p} is holds that d(p,o')<=d(p,o), and
- (2) For at most (K-1) objects o' ∈ D/{p} it holds that d(o,o')<d(p,o)

The $K_{dist}(p)$ can represent the density of the object p. The smaller $K_{dist}(p)$ is, the much denser the area around p is. So, like paper [19], we define the density of p, denoted as Den(p), as the follows equation:

$$\mathrm{Den(p)} = {}^{1}/_{K_{dist}(p)} \tag{2}$$

Definition2 (K Nearest Neighbor and Reverse K Nearest Neighbor) If $\mathrm{d(p,q)} \le K_{dist}(p)$, then call the object q as the K Nearest Neighbor of p. All of the K Nearest Neighbor compose the K Nearest Neighborhood, denote as KNN(p). Conversely, call the object p as the Reverse K Nearest Neighbor of q, and all of the Reverse K Nearest Neighbor compose the Reverse K Nearest Neighborhood, denote as RKNN(q). The formulation of KNN(p) and RKNN(p) as following:

$$\mathrm{KNN(p)} = \{\mathrm{q}|\mathrm{d(p,q)} \le K_{dist}(\mathrm{p})\}$$

$$\mathrm{RKNN(q)} = \{\mathrm{p}|\mathrm{d(p,q)} \le K_{dist}(\mathrm{p})\}$$

III. The Proposed Algorithm

In this paper we divide the neighbors of every point into Dense Neighbors and Sparse Neighbor, defined as the **Definition3.**

Definition3 (Dense and Sparse Neighbor): If the density of q is greater than p and $\mathrm{q} \in \mathrm{KNN}(p)$, then call the object q as the Dense Neighbor of p, denote as DN(p). On the contrary, if the density of q is smaller than p and $\mathrm{q} \in \mathrm{KNN}(p)$, then q is called as the Sparse Neighbor, denote as SN(p).

Definition4 (Exemplar) If the density of q is the maximum in the neighbors of p and p≠q, then call the object q is the Exemplar of p.

From the Definition of Exemplar, we can know that each point of dataset possess at most one Exemplar. If the density of p is greater than the density of all k nearest neighbors or reverse k nearest neighbors of p, then p is the Exemplar of itself. And we call p is the **Quasi-Cluster Center(QCC)**.

Definition5 (Quasi-Cluster Center) If object p satisfied one of the follows two conditions, then we call p as QCC.

- (1) $\forall \mathrm{q} \in \mathrm{KNN(p)}, \mathrm{Den(p)} \ge \mathrm{Den(q)}$ or
- (2) $\forall \mathrm{q} \in \mathrm{RKNN(p)}, \mathrm{Den(p)} \ge \mathrm{Den(q)}$

Figure 2 is the Exemplar Graph(EG) which can be comprised by connecting each point p to its Exemplar. As shown in Figure 2, the parameter k=30, points $c_1, c_2, \ldots, c_7$ etc. is QCC that marked in red. Other red points will be treated as outliers that will be explained in **Algorithm1**. And through many experiments and analysis, we find that the number of **Quasi-Cluster Center** appears to get smaller as the parameter k becomes bigger.

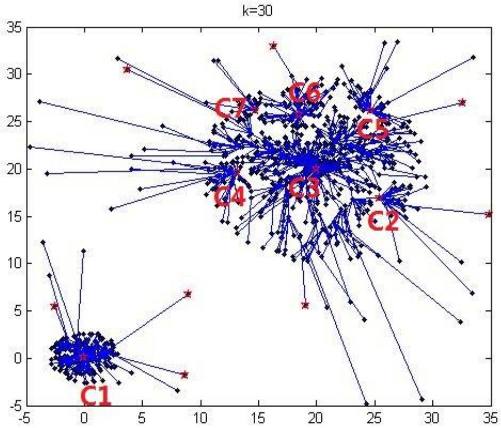

Figure2: Exemplar Graph and Quasi-Cluster Center

Definition6 (Similarity between clusters): Similarity between clusters C_i and C_j, denote as $\mathrm{Sim}(C_i, C_j)$, is defined as the ratios of the number of objects q that $\mathrm{q} \in C_i \cap q \in C_j$ and K. The formulation of similarity between clusters as follows:

$$\mathrm{Sim}(C_i, C_j) = \mathrm{Num}(\{\mathrm{q}|\mathrm{q} \in C_i \cap q \in C_j\})/\mathrm{K} \tag{3}$$

As shown in Figure 3, We consider the ratios of the number of these points that between two adjacent marked in red and K as the similarity of two adjacent initial clusters. If these two adjacent initial clusters are divided by sparse area, then the similarity of these two cluster is small. In other words, these two clusters are two individual cluster. On the contrary, if these two adjacent initial clusters are connected by density area, the similarity of these two adjacent clusters will be great. And these two clusters will be merged to one cluster. In this way, even if one big cluster was divided to many small clusters because the value of k is small, like Figure1 shows, these small clusters C1, C2, …, C7 will be merged into one cluster finally.

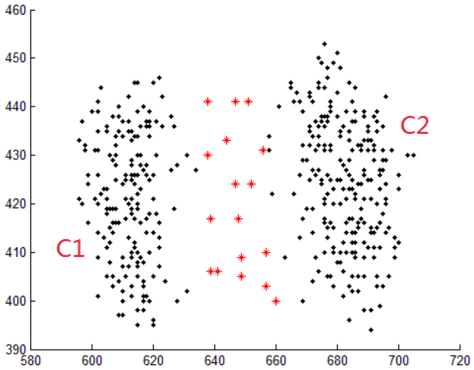

Figure3: The similarity between C1 and C2

Based on the above definitions, we proposed a novel clustering algorithm, named QCC, with the capacity that outlier detecting. The procedure of QCC algorithm is minutely described in **Algorithm1.**

Firstly, the proposed clustering algorithm QCC use the **KNN-Searching** to obtain the KNN and RKNN of each point of dataset D, so we need the parameter k that the number of the neighbors of every points. Then compute the density of each point. Secondly, step 5 in **Algorithm1**, QCC find the Exemplar of each point using **Definition4**, and obtain all of the Quasi-Cluster Center using **Definition5**. After that, QCC obtain the initial clusters.

- (1) QCC arbitrarily find a Quasi-Cluster Center, and classify it and it's Sparse Neighbors to the same cluster C_i.
- (2) Then QCC arbitrarily find a point p in this cluster and classify the Sparse Neighbors of p to cluster C_i, until all points of this cluster have been visited.
- (3) Then QCC find an other Quasi-Cluster Center and repeat the above steps, until all Quasi-Cluster Center have been visited.

In this way, the clusters extended from dense area to the sparse area. As shown in Figure(3), the points marked in red are classified to C1 and C2 at the same time. Then QCC merge all the clusters that similarity greater than alpha into one cluster. If the similarity of cluster C1 and C2 is smaller than alpha, then the red points will classified into the cluster that it's Exemplar belongs to. alpha is a artificial parameter. Generally, the value of alpha is 0.2 to 0.5. The higher the alpha, more clusters can be obtained.

Algorithm1: QCC(D,K,alpha) //alpha is the similarity between clusters.
Output: C={$c_1, c_2, \ldots, c_M$}

```
(1)  Initializing Variables: r=0, K_dis(i)=0, Den(i)=0, KNN($i$) = $\emptyset$, $RKNN(i) = \emptyset$,
     SN(i)= $\emptyset$, Exemplar(i)=i, Sim$(c_i, c_j)$ = 0, $Q_{CC}$ = $\emptyset$;
(2)  [KNN($i$), RKNN($i$)]= KNN-Searching(D,K)
     //use the KNN-Searching algorithm to obtain the KNN and RKNN of every point.
(3)  For $\forall$x $\in$ D  // compute the K-distance and the Density of the objects in D.
     a.  Find y that the K-th nearest neighbor of x
     b.  $K\_dis(x)$=$||x - y||_2$;
     c.  Den(x)=$1/K\_dis(x)$;
(4)  For $\forall$x $\in$ D find the SN(x).
(5)  For $\forall$x $\in$ D  // find the Exemplar of x and obtain the Quasi-Cluster Center.
     a.  y=max(Den($KNN(x)$));
     b.  if y $\neq$ x then $Exemplar(x)$ = y;
     c.  if y==x then r=r+1 and $Q_{CC}$(r)=x;
     d.  z=max(Den(RKNN(x)));
     e.  if x==z then r=r+1 and $Q_{CC}$(r)=x;
(6)  For i=1 to r  //obtain the initial cluster.
     a.  $c_i$ = {$Q_{CC}$(i)}$\cup$ $SN$($Q_{CC}$(i));
     b.  For $\forall$x $\in$ $c_i$
     c.      If visited(x)$\neq$true then
         visited(x)=true and $c_i = c_i \cup SN(x)$;
(7)  Compute the similarity matrix Sim$(c_i, c_j)$ between the clusters.
(8)  While max(Sim$(c_i, c_j)$)>=alpha
     //merge the initial cluster
     a.  (i,j)=max(Sim$(c_i, c_j)$);
     b.  Merge cluster $c_i$ and $c_j$;
     c.  Update the similarity matrix;
(9)  If $\exists$(0 < Sim$(c_i, c_j)$ < alpha)
     a.  If x $\in$ $c_i$ & x $\in$ $c_j$ then x is classified to the cluster that it's exemplar belongs to.
(10) For i=1:length(C)
     a.  If number of $c_i$ < k;
     b.  Then $\forall$x $\in$ $c_i$ x is marked as outlier and delete $c_i$ from C;
(11) Output the final clusters
     C={$c_1, c_2, \ldots, c_M$}
```

After the above steps, QCC regard the clusters that the number of points smaller than k as outlier clusters. In other words, the points in these clusters is marked as outliers. So the red points in Figure2 will be regarded as outliers except C1, C2, …, C7. So QCC will obtain the accurate clustering results as long as the value of k is smaller than the number of points in smallest cluster of dataset. Finally, QCC output the ultimate clusters. So QCC not only cluster the dataset, but also has certain ability of outlier detection.

IV. Experimental Analysis

A. *Cluster on Artificial Data Set*

We chose four challenging artificial data sets. Data1, taken from [8], consist of two spherical data and two manifold data that one is simple, another is complex ,and a few outliers, a total of 582 points. Data2, taken from [20], composed of three spherical data, one complex manifold data and some noise points, a total of 1400 points. Data3, taken from [21], composed of six high density manifold data and some noise points, a total of 8000 points. Data4, taken from [22], composed of one dense spherical cluster and one sparse manifold cluster, a total of 159 points.

In all results, we don't show the decision graph, decide the number of the clusters ,of DP. We decide the right number of clusters to Data1, Data3 and Data4. For Data2, we show the best cluster result in repeated test. For DAAP, we set the density factor is assigned as ρ=2, the maximum iteration maxits=1000, convergence of iteration coefficient convits=100.

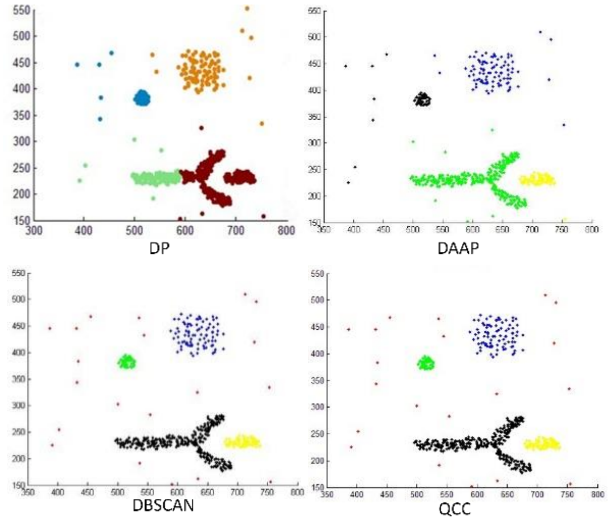

Figure 4: The cluster results of DP, DAAP, DBSCAN and QCC algorithm on Data1.

Figure 4 shows the DP, DAAP, DBSCAN and QCC algorithm's clustering results on Data1. For DAAP, the value of the number(k) of neighbors that used for construct the adjacency matrix is set 6, the value damping coefficient(lam) is 0.9. From this figure, we can see that, as analyses in section1, DP algorithm can correctly cluster the spherical data and simple manifolds data, but can't correctly cluster the complex manifolds data. Data1 is correctly clustered by DAAP, DBSCAN(eps=15, minpoints=5) and QCC(k=20, alpha=0.3). Moreover, DBSCAN and QCC algorithm detect out the noise points in Data1, but DAAP can't.

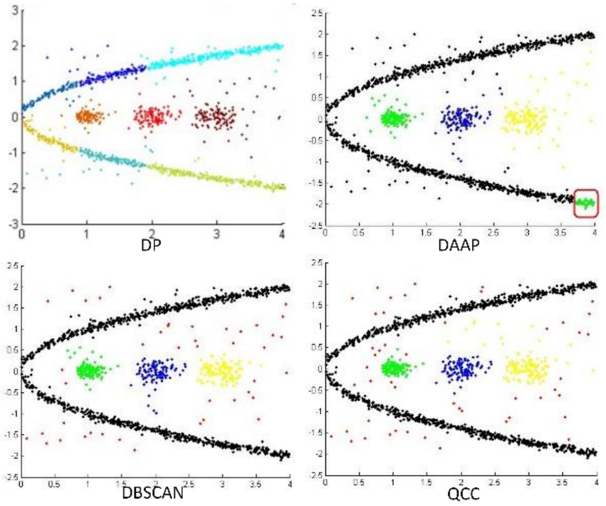

Figure 5: The cluster results of DP, DAAP, DBSCAN and QCC algorithm on Data2.

Figure 5 shows the four algorithm's clustering results on Data2. For DAAP, the value of the number(k) of neighbors that used for construct the adjacency matrix is set 6, the value damping coefficient(lam) is 0.9. DP failed to cluster the complex manifolds data that grouped into 6 clusters. Owing to particular similarity that the sum of the weight of the edge in shortest path, DAAP has some capacity of cluster to complex manifold datasets. However, DAAP failed to cluster the manifolds cluster in Data2. Since the shortest path is too long, so the end region(marked by red square) of the manifold cluster is classified the wrong cluster. Data2 is correctly clustered by DBSCAN(eps=0.2, minpoints=40) and QCC(k=20, alpha=0.3) algorithm. And most of the noise points in Data2 is detected out by DBSCAN and QCC.

Figure 6 shows the four algorithm's clustering results on Data3. Although DP obtain the right number of clusters in Data3 by artificially select the cluster centers in decision graph, three clusters are wrongly clustered among these clusters. DAAP obtain the right number of clusters, but some clusters are wrongly clustered too. Moreover, DAAP mistakenly regard a part of noise as a small normal cluster. The cluster result of DBSCAN is obviously superior to DP and DAAP, and DBSCAN detect out the noise points in Data3. However, some points in normal clusters are treated as noise points. Although QCC(k=80, alpha=6) failed to detect out the noise in Data3, QCC obtain the right number of cluster without the number of clusters that artificial set, and correctly cluster the normal points.

Figure 7 shows the four algorithm's clustering results on Data4. Same as the results on Data4, although DP and DAAP obtained the right number of clusters by artificially select or set. For the density variations of the two clusters in Data4 is great, DBSCAN(eps=2, minpoints=5) failed to correctly cluster Data4. DBSCAN don't obtain the right number of clusters in Data4, and mistakenly treat some normal points as noise. The performance of QCC(k=5, alpha=0.2) is obviously superior to DP, DAAP and DBSCAN on Data4.

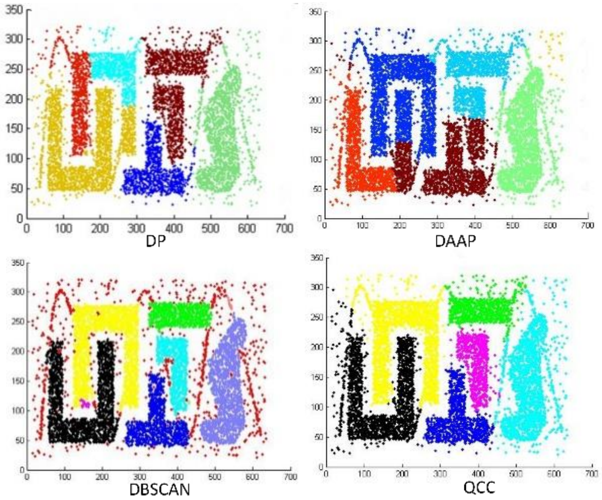

Figure 6: The cluster results of DP, DAAP, DBSCAN and QCC algorithm on Data3.

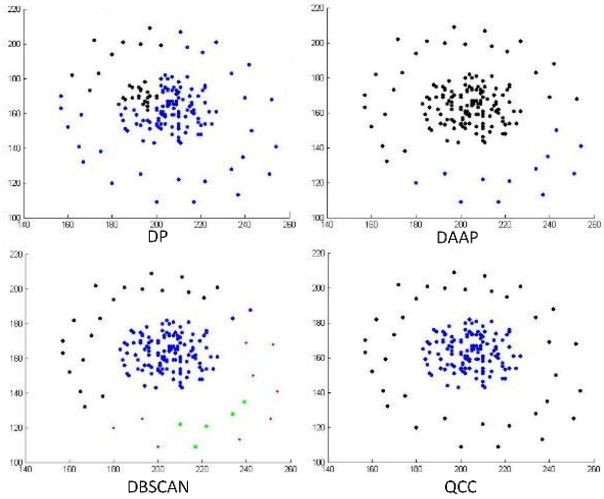

Figure 7: The cluster results of DP, DAAP, DBSCAN and QCC algorithm on Data4.

From the above results and analysis, we can see that, DP algorithm has a certain capacity to cluster non-spherical data(correctly cluster Data2). But, as shown in the above results, DP algorithm can hardly correctly cluster the complex manifold datasets. DAAP has a certain capacity of cluster to complex manifold datasets. However, DAAP failed to cluster those datasets that include long manifolds data(Data2), lots of noise(Data3) or great density variations clusters(Data4). Although the performance of DBSCAN is superior to DP and DAAP, DBSCAN failed to correctly cluster on Data3 and Data4. So, from the results of artificial datasets, we can see that QCC that proposed in this paper can get the right number of final clusters without human intervention. And the scope of QCC's application is wider than other cluster algorithms. No matter complex manifold datasets or density variations is great, QCC can get satisfactory clustering results.

It should be noted that the value of k should smaller than the number of points of minimum cluster, and the value of alpha is between 0.2 and 0.5 for most data sets. When the bulk density is high, the value of alpha should be greater accordingly, such as Data4. In order to demonstrate the effectiveness of QCC, we also experiment on real datasets as the follows section.

B. *Cluster on Olivetti Face Database*

Like the paper[17], we also applied the QCC algorithm to the Olivetti Face Database[23], a widespread benchmark for machine learning algorithms, with the aim of identifying, without any previous training, the number of subjects in the database. For this experiment, we used 10 clusters of Olivetti face database. And each cluster is composed of 10 face picture. The size of each picture is <112x92 nint8>. The similarity between two images, denote as S(A,B), was computed by the follows equation.

$$\mathrm{S(A,B)} = \frac{\sum_m \sum_n (A_{mn}-\bar{A})(B_{mn}-\bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn}-\bar{A})^2)(\sum_m \sum_n (B_{mn}-\bar{B})^2)}} \qquad (4)$$

Here A and B are the subjects of Olivetti Face Database. A_{mn} and B_{mn} represent the pixels of the two subjects picture. The value of S is scaled between 0 and 1. Bigger the value of S is, more similar the two picture is. So we define the distance of two picture, denote as d(A,B), as following equation.

$$\mathrm{d(A,B)} = 1 - \mathrm{S(A,B)} \qquad (5)$$

The density is estimated as **Definition1**.

The results is shown in Figure 8. In the results, faces with same color belong to the same cluster.

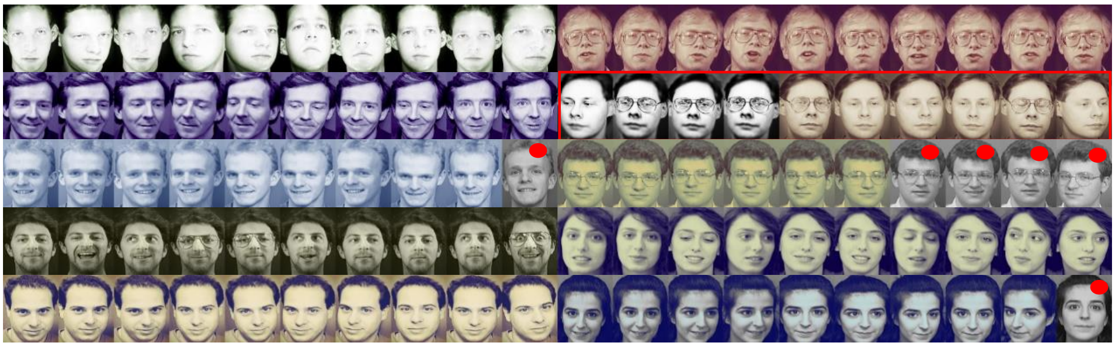

Figure 8: QCC(k=4, alpha=0.2) algorithm's clustering results on Olivetti

In order to intuitively descript the efficiency of QCC, we use two criteria(Purity, Recall) to evaluate the clustering performance. And the calculation formula is as follows:

$$\text{Purity} = {\sum_{i=1}^{k}\left(\max_{tc\in TC}\left({tc \cap c_i}/{N_{C_i}}\right)\right)}\Big/{k} \qquad (6)$$

$$\text{Recall} = \frac{1}{N}\sum_{i}^{k} N_{C_i} \qquad (7)$$

Here, Let D be a database and contains Tk clusters $\mathrm{TC} = \{tc_1, tc_2, \dots, tc_{Tk}\}$. The result of clustering algorithm is $\mathrm{C} = \{c_1, c_2, \dots, c_k\}$. N_{C_i} is the number of points of c_i. N is the number of points of whole dataset. The value of Purity and Recall is [0,1], the larger the value of ACC, means the better the clustering performance of the algorithm.

TABLE I. ACC AND EECALL OF THE FOUR ALGORITHM

	DP	DAAP	DBSCAN	QCC
Purity	0.88	0.61	0.98	1
Recall	1	1	0.64	0.94

The QCC algorithm's clustering results on Olivetti Face Database is shown in Figure8. The results show that Olivetti Face Database was grouped into 11 clusters, because one of the real 10 cluster that within the red border was divided into two clusters. And 6 images that marked with red spot was considered as outliers by QCC algorithm. However, among the 11 clusters, 6 clusters is really correct, 2 clusters is identified 9 face images, one cluster is identified 6 faces in all 10 faces. Moreover, all of the 11 clusters remain pure, namely include only images of the same cluster. So the Purity of QCC is 1(the best one). The value of Purity of DP and DAAP is 0.88 and 0.61. Although the value of Purity of DBSCAN is 0.98 that closest to QCC, the value of Recall of DBSCAN is the lowest(0.64). Moreover, the Recall of QCC is 0.94 that close to 1.

Through above analysis to results that cluster on artificial data and Olivetti Face Database, it is obvious that the results of QCC outperform the DP, DAAP and DBSCAN algorithm. And we can get the conclusion that QCC algorithm has a more broad application than AP and DP algorithm. QCC algorithm has a certain ability that outliers detecting and can cluster on complex manifold data sets. Furthermore, QCC is not likely to omit any cluster center as DP. So QCC cluster algorithm superior to AP, DP and DBSCAN algorithm.

V. CONCLUSION

In this study, we propose a new cluster algorithm(QCC). The core idea of QCC is that clusters is divided by sparse region. Based on this idea, we define the Quasi-Cluster Centers. Remarkably, the real cluster centers must be included by Quasi-Cluster Centers. So QCC is not likely to omit any clusters. Then QCC obtain the initial clusters by step5 of Algoritm1. After this, we define the Similarity between initial clusters. Therefore, QCC applies to complex manifold data sets. Through the experiments on the four artificial datasets, we confirmed that the proposed cluster algorithm(QCC) can correctly cluster on complex manifold data sets that DP, DAAP and DBSCAN can't. The results from the Olivetti Face Database also demonstrated that QCC is more effective than DP, DAAP and DBSCAN. Furthermore the scope of application of QCC is more extensive.

ACKNOWLEDGMENT

This research was supported by the National Nature Science Foundation of China (No. 61272194).

REFERENCES

[1] Xu, R. and D. Wunsch, Survey of clustering algorithms. Neural Networks, IEEE Transactions on, 2005. 16(3): p. 645-678.

[2] Han, J. and M. Kamber, Data mining: concepts and techniques, Morgan Kaufmann San Francisco, Calif, USA. 2001.

[3] Kaufman, L. and P.J. Rousseeuw, Finding groups in data: an introduction to cluster analysis: John Wiley & Sons. Vol. 344. 2009.

[4] Ng, R.T. and J. Han, Clarans: A method for clustering objects for spatial data mining. Knowledge and Data Engineering, IEEE Transactions on, 2002. 14(5): p. 1003-1016.

[5] Zhang, T., R. Ramakrishnan, and M. Livny. BIRCH: an efficient data clustering method for very large databases. in ACM SIGMOD Record. 1996. ACM.

[6] Guha, S., R. Rastogi, and K. Shim. CURE: an efficient clustering algorithm for large databases. in ACM SIGMOD Record. 1998. ACM.

[7] Guha, S., R. Rastogi, and K. Shim. ROCK: A robust clustering algorithm for categorical attributes. in Data Engineering, 1999. Proceedings., 15th International Conference on. 1999. IEEE.

[8] Ester, M., et al. A density-based algorithm for discovering clusters in large spatial databases with noise. in Kdd. 1996.

[9] Hinneburg, A. and D.A. Keim. An efficient approach to clustering in large multimedia databases with noise. in KDD. 1998.

[10] Wang, W., J. Yang, and R. Muntz. STING: A statistical information grid approach to spatial data mining. in VLDB. 1997.

[11] Wang, W., J. Yang, and R. Muntz. STING+: An approach to active spatial data mining. in Data Engineering, 1999. Proceedings., 15th International Conference on. 1999. IEEE.

[12] Moore, A.W., Very fast EM-based mixture model clustering using multiresolution kd-trees. Advances in Neural information processing systems, 1999: p. 543-549.

[13] Smith, A., et al., Sequential Monte Carlo methods in practice: Springer Science & Business Media. 2013.

[14] Frey, B.J. and D. Dueck, Clustering by passing messages between data points. science, 2007. 315(5814): p. 972-976.

[15] Zhang, X., et al. K-AP: generating specified K clusters by efficient affinity propagation. in Data Mining (ICDM), 2010 IEEE 10th International Conference on. 2010. IEEE.

[16] Jain, A.K., Data clustering: 50 years beyond K-means. Pattern recognition letters, 2010. 31(8): p. 651-666.

[17] Rodriguez, A. and A. Laio, Clustering by fast search and find of density peaks. Science, 2014. 344(6191): p. 1492-1496.

[18] Jia, H., et al., A density-adaptive affinity propagation clustering algorithm based on spectral dimension reduction. Neural Computing and Applications, 2014. 25(7-8): p. 1557-1567.

[19] Jin, W., et al., Ranking outliers using symmetric neighborhood relationship, in Advances in Knowledge Discovery and Data Mining, Springer. 2006. p. 577-593.

[20] Ha, J., S. Seok, and J.-S. Lee, Robust outlier detection using the instability factor. Knowledge-Based Systems, 2014. 63: p. 15-23.

[21] Cassisi, C., et al., Enhancing density-based clustering: Parameter reduction and outlier detection. Information Systems, 2013. 38(3): p. 317-330.

[22] Zhu, Q., et al., A clustering algorithm based on natural nearest neighbor. Journal of Computational Information Systems, 2014. 10(13): p. 5473-5480.

[23] Samaria, F.S. and A.C. Harter. Parameterisation of a stochastic model for human face identification. in Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on. 1994. IEEE.