A New Fuzzy Cover Approach to Clustering

Jung-Hsien Chiang, *Member, IEEE*, Shihong Yue, and Zong-Xian Yin

***Abstract*—This paper presents a new fuzzy cover-based clustering algorithm. In the proposed algorithm, the concept of fuzzy cover and objective function are employed to identify holding points in the dataset, and we associate these holding points together to build up the backbones of the final clusters. Three specific objectives underlie the presentation of the proposed approach in this paper. The first is to describe mathematical formulation of the fuzzy covers, and the second is to summarize the detailed procedure of constructing fuzzy covers and splicing them into clusters. The third goal is to demonstrate that this approach is able to find out reasonable representative patterns in the final clusters. We illustrate this approach with four examples in order to verify the clustering effectiveness.**

***Index Terms*—Binary fuzzy relation, cluster analysis, fuzzy clustering, fuzzy cover.**

I. INTRODUCTION

CLUSTERING is one of the fundamental human cognitive activities, and it has been used extensively in computer vision and pattern recognition [5], [2], [12], [14], [17], [28], [13], [19]. Of these methods, fuzzy clustering has been shown to be effective in solving the problem that each data point has a membership grade indicating its belongingness degree to each cluster, rather than assigning it to only one of the clusters [25], [27], [15], [11]. The *fuzzy c-means* (FCM) algorithm is one of the best known fuzzy clustering approaches, and its use for various applications is well described and analyzed in a previous paper by Bezdek (1992). This clustering method relies on the optimization of a specific cost function, and it works well when the clusters are compact or isotropic.

Another fuzzy clustering method is to determine the partitions from binary fuzzy relations between pairs of samples in a dataset by using the *transitive closure* technique [6], [16], [9], [7]. In this approach, a cluster can be defined as a fuzzy set whose elements are similar to each other, and the fuzzy relations between pairs of these elements are not less than a certain level. The advantages of this method are that every fuzzy equivalence relation (a relation that is reflexive, symmetric, and transitive) induces a partition in each of its α*-cut* and the number of clusters does not have to be specified in advance. However, the drawback is that these fuzzy equivalence relations cannot easily be found since the constraints are so restrictive that very few relation functions exist. Moreover, the computation of transitive closure is complicated [20].

The research described here presents a formulation of generalized fuzzy covers and outlines a new view of the problem of clustering based on the binary fuzzy relation. In this approach, we employ the concept of fuzzy cover [33] and objective function to group the data points into appropriate clusters. The fuzzy cover is a fuzzy set in which the fuzzy relation of any enclosed data point to the centroid data point is larger than certain level. The binary fuzzy relation in our method is not restricted to reflexive, symmetric and transitive constraints of the equivalence relation, which allows more suitable fuzzy similarity relations to utilize in different engineering applications. We incorporate the objective function to search for optimal fuzzy covers that can spread over the whole dataset. The objective function reflects the natural grouping of fuzzy covers, and it makes the clustering results more reasonable. The proposed algorithm then identifies those fuzzy covers and the centroids of the fuzzy covers build up supports of overall dataset. These derived centroids form the backbones of the final clusters after splicing the fuzzy covers into rational groups. The number of final clusters is determined by the appointed fuzzy relation and it does not have to be specified in advance. In this paper, we discuss the theoretical framework of the *fuzzy covers clustering* (FCC) algorithm and analyze the clustering effectiveness of representative patterns in each cluster.

In what follows, our attention will be focused primarily on defining the basic concept of fuzzy cover. In Section III, we explain how to use the fuzzy covers to construct the clusters, and then describe the procedures of the proposed clustering algorithm. Experiments are then discussed in Section IV.

II. FUZZY COVER

Let $X = \{x_j | 1 \leq j \leq n\}$ be a set of samples with a finite number of members in the domain, D. Zadeh defined a fuzzy equivalence relation R for each sample in X, and a fuzzy subset based on the relation was denoted as $R(x_j) = R(x_j, y)$, where $y \in X$ [33]. For λ in [0, 1], the λ*-level-set* based on the fuzzy relation R is denoted as

$$R_\lambda(x_j) = \{(x_j, y) | R(x_j, y) \geq \lambda\} \tag{1}$$

where x_j is the center of this subset. In this formulation, $R_\lambda(x_j)$ is a subset of the dataset on R, defined as a *fuzzy cover* of x_j by Zadeh or referred to as Ovchnnikov's fuzzy partition in [24]. Assume that p_j represents a cover with the center x_j as

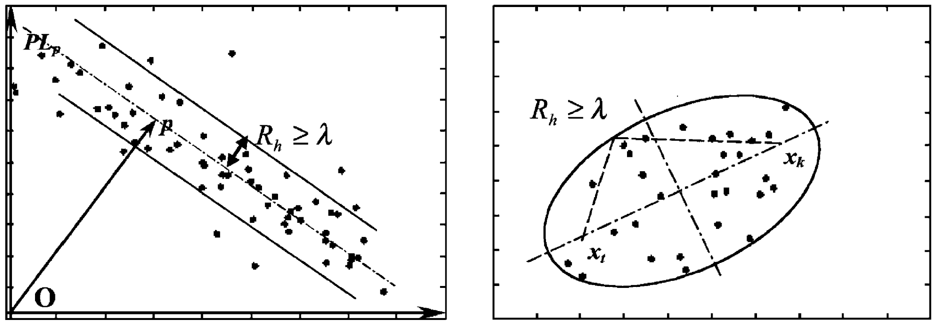
$$\begin{cases} p_j(x_k) = 1 & \text{if } R(x_j, x_k) \geq \lambda \\ p_j(x_k) = 0 & \text{otherwise} \end{cases} \quad j = 1, 2, \ldots, n \quad k = 1, 2, \ldots, n \tag{2}$$

The family $\sum$ is the power set of $\{p_j\}$, denoted as $\sum = P\{p_j\}$. It can be seen that the similarity of this partition does

Manuscript received November 18, 2002; revised July 22, 2003.

The authors are with the Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan 70101, Taiwan (e-mail: jchiang@mail.ncku.edu.tw).

Digital Object Identifier 10.1109/TFUZZ.2004.825076



(a) (b)

Fig. 1. Two kinds of fuzzy covers each with different fuzzy proximate function. (a) Hyperplane fuzzy cover. (b) Ellipsoid fuzzy cover.

not determine the domain but determines only a cover. In a traditional definition of fuzzy cover, the relation R must satisfy the reflexive, symmetric and transitive of equivalence relation [33].

In this paper, we generalize the definition of the fuzzy cover, which means that more relations can be used as the proximity measures. The following definition states the concept and motivation.

Definition 1: (Fuzzy cover) Given a subset ρ_j with the center x_j in X, and

$$\rho_j = \{y \,|\, r(x_j, y) = h(y \,|\, x_j) \geq \lambda \; y \in X\} \tag{3}$$

where λ is a threshold value, r is the fuzzy relation between pairs of data, h is a fuzzy proximity function, and both their values are between [0, 1]. The family Σ is the power set of $\{\rho_j\}$ and $X = \bigcup_{j \in |\sum|} \rho_j$, then Σ is a set of fuzzy covers in X.

In this fuzzy cover, one can easily choose a specific fuzzy proximate function h for a given dataset where h does not need to satisfy the equivalence relation. We now illustrate the basic idea using the two examples with different fuzzy proximate relation functions in Fig. 1. In Fig. 1(a), we define a fuzzy proximate relation function as

$$h(y \,|\, p) = \exp - \left(\frac{d(PL_p, y)}{\xi} \right) y \in X, \qquad \xi > 0$$

where $d(\cdot)$ is the Euclidean distance measure and ξ is a constant. The cover shape formed by this fuzzy proximate function is hyperplane. This hyperplane PL_p passes through the sample p and is perpendicular to $\overrightarrow{op}$. The dashed line is the hyperplane PL_p with the normal vector $\overrightarrow{op}$. If a sample y belongs to this cover, then it must satisfy $\exp -(d(PL_p, y))/(\xi) \geq \lambda$; otherwise, the sample does not belong to this cover. Two solid lines are the margins of the cover with respect to threshold λ, and the fuzzy relation degrees of the samples to the hyperplane within these two solid lines are greater than λ. In Fig. 1(b), we define another form of fuzzy similarity relation function as

$$h(y \,|\, x_t, x_k) = \exp - \left[\frac{d(y, x_t) + d(y, x_k)}{2\xi} \right] y \in X, \qquad \xi > 0.$$

This kind of fuzzy proximate relation is in inverse proportion to the sum distance from an arbitrary sample to x_t and x_k. The ellipse margin is the fuzzy relation degree equal to λ.

These two fuzzy covers are constructed by two different fuzzy proximity relations. It can be seen that the advantage of proposed fuzzy cover is that it does not need to be restricted to the equivalence relations. Even more appropriate relations can be used to construct fuzzy covers, and we can utilize these fuzzy covers to construct final clusters. In following section, we describe the theoretical framework and algorithm in detail.

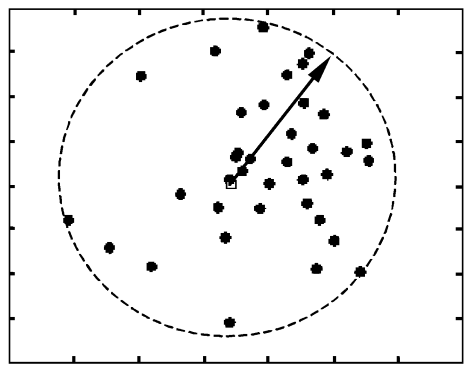


Fig. 2. Example of one-cluster dataset.

III. Cluster Construction Using Fuzzy Covers

In this section, we describe several important properties of the fuzzy cover and optimization procedures that are used in the proposed fuzzy clustering algorithm. In Fig. 2, we first give a simple one-cluster example to illustrate the clustering procedure. In this example, we can find a minimal spherical cover that can enclose all data points, and whose centroid "□" stands for the center of the cluster. This concept can easily determine a reasonable cluster center even if the distribution of the cluster is skewed, and the centroid of the cover is the *holding point* of this cluster in this example. We define the holding point in next section.

However, if a cluster does not have a spherical distribution, then a single holding point does not exist. In this example, the holding points of a cluster are obtained from multiple covers, as shown in Fig. 3. It can be seen that we are unable to use a spherical cover, e.g., the cover C_1, to enclose the right cluster. Several samples that do not belong to this cluster are enclosed by C_1. It is reasonable to apply three spherical covers (P_1^1, P_2^1, P_3^1) to overspread all samples. The centroids of these covers are the holding points of the cluster.

In this example, the holding points of a cluster in the dataset can be obtained by the fuzzy covers. In other words, if we can

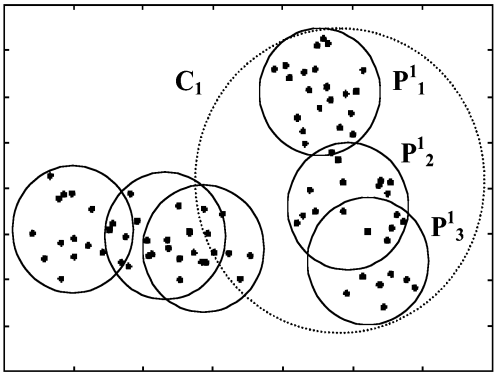


Fig. 3. Two oblong shape clusters.

TABLE I
PSEUDOCODE OF THE FCC ALGORITHM

Fuzzy Covers Clustering (FCC) Algorithm

Assign confidence margin $[\lambda_{\min}, \lambda_{\max}]$, and $\Delta\lambda$.
Determine the fuzzy proximate function *h*.
DO
  Construct Fuzzy Covers for samples (**CFC algorithm**).
  Find Minimal number of Covers to enclose the Samples in X (**MCS algorithm**).
  Splicing Covers into Cluster (**SCC algorithm**).
  Compute validity value T_t and $\lambda_{t+1} = \lambda_t + \Delta\lambda$.
WHILE ($\lambda_{t+1} \le \lambda_{\max}$)
Select the final clustering result that has the most optimal validity value.

identify the samples that are nearest to the holding points of a cluster, then we can use these samples to symbolize the same cluster. The rest of the samples will automatically be grouped into these clusters according to distance. Therefore, in the proposed clustering algorithm, we first search for the samples that can serve as the holding points using the fuzzy covers.

Table I summarizes the pseudo-code of the proposed **FCC clustering algorithm**. We describe each procedure in the following sections.

A. Construct Fuzzy Covers for Samples

In the first step, the fuzzy proximity function is used to construct fuzzy covers. The threshold λ determines how many samples are enclosed in one cover. In this paper, we utilize two fuzzy proximity functions. The first one is a common spherical fuzzy proximity function

$$h(x_j \,|\, x_k) = 1 - \frac{d_{\mathrm{jk}}}{d_{\max}}, \qquad \text{for } j = 1, 2, \ldots, n \qquad (4)$$

where x_k is the centroid of cover; $d_{\max}$ is the maximum pairwise distance in X, and d_{jk} is the distance between x_j and x_k. This fuzzy proximity function is spherical shape, and it satisfies symmetric but not transitive properties. The other function is the hyperplane relation

$$h(x_j \,|\, x_p) = \frac{1}{1 + \exp\left(-\frac{d(PL_p, x_j)}{\eta}\right)}, \qquad \text{for } j = 1, 2, \ldots, n \qquad (5)$$

TABLE II
CONSTRUCT FUZZY COVERS FOR SAMPLES ALGORITHM

Construct the Fuzzy Covers for samples (CFC) Algorithm

Given a set of sample $X = \{x_1, x_2, \ldots, x_n\}$.
Determine the fuzzy proximity function h.
Determine the threshold λ.
FOR all samples in X
  Construct the cover correspond to each sample $\mathrm{P} = \{\rho_j\}$ where $j = 1, 2, \ldots, n$.
  Find out all members for each cover $\rho_j = \{y \,|\, h(y|x_j) \ge \lambda, \ y \in X\}$.
END FOR
FOR all covers, $\{\rho_j\}$
  Find the covers that contain only one member, and construct as the set A_1.
  $A_1 = \{\rho_j \,|\, |\rho_j| = 1\} \quad j = 1, 2, \ldots, n$
  Find the covers that are subsets of another covers, and construct the set A_2.
  $A_2 = \{\rho_j \,|\, \rho_j \subseteq \rho_k, \ j, k = 1, 2, \ldots, n, \ j \ne k\}$
END FOR
$A' = A_1 \cup A_2$.

The set $\Sigma = \{\mathrm{P} - A'\}$ contains the candidate covers for the algorithm in next section.

where PL_p is the hyperplane that passes through sample x_p and that has normal vector $\overrightarrow{ox_p}$; $d(PL_p, x_j)$ is the distance between the sample x_j and the plane PL_p; and η is a constant. This relation satisfies neither symmetric nor transitive properties.

Since the fuzzy proximity relations do not need to satisfy the three constraints of fuzzy equivalence relation, we can select desired fuzzy proximity functions for different datasets. Here, the threshold λ is an important parameter since if it is too small, a single fuzzy cover contains most of the samples in the dataset; or if it is too large, it may cause many redundant clusters. We will discuss this in detail in the experiments. The **CFC *algorithm*** is shown in Table II.

In the **CFC *algorithm***, there are two sets of covers, i.e., A_1 and A_2, which will be removed in order to reduce the complexity in the clustering algorithm. The shaded regions shown in Fig. 4(a) and Fig. 4(b) show A_1 and A_2 in the **CFC** procedure, respectively.

There are many other fuzzy relations described by classical fuzzy relations. However, most of those fuzzy equivalence relations have to obey many constraints and restrict their extension. In this paper, we define and prove the extended fuzzy equivalence relation based on the *t-norm* operator, and show that this relation can be also serve as one kind of fuzzy proximity function for constructing fuzzy cover. These proofs are in Appendix.

B. Find Minimal Number of Covers to Enclose the Samples

In this section, we discuss how to find the holding points of possible clusters using fuzzy covers. Since holding points are always located in dense regions, we mark out those regions as the *backbones* or *cores*. In fact, we identify the samples that are close to the backbones and consider them as the holding points for clusters. The definition of the holding point in our algorithm is as follows.

Definition 2: (Holding point) Given a set of covers with the threshold λ that enclose the dataset, we assign the centroids of the covers as the holding points of the possible clusters if they satisfy

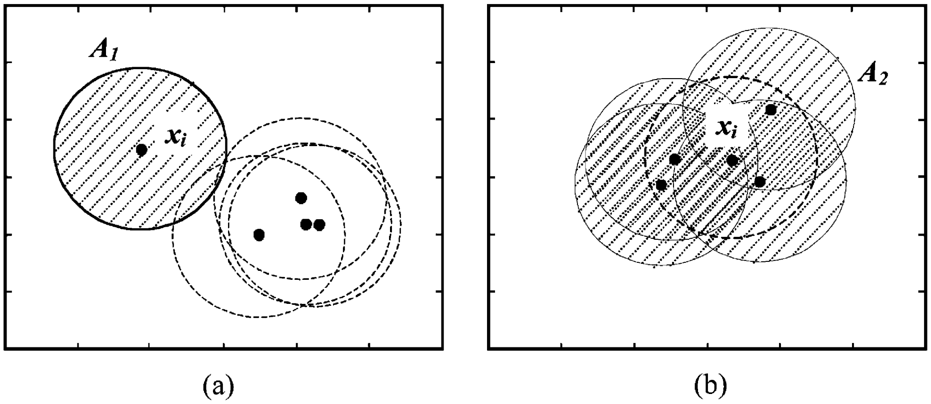


Fig. 4. Two sets A_1 and A_2 in the **CFC** algorithm. (a) Isolated cover. (b) Fully overlapped cover.

1) the number of covers is minimal;
2) the number of joint samples of two covers is minimal.

The first condition confirms these centroids that are located around the backbone in X. The second condition reduces the ambiguity of a sample to the clusters.

As stated in the definition, the holding points are always located at the dense regions in X. Assuming we can find a set of finite covers that can enclose all the samples in X, and we define the average number of samples within a cover as

$$\mathrm{mean}_\rho = \frac{n}{n_\rho} \tag{6}$$

where n_ρ is the number of covers. Obviously, smaller n_ρ makes larger mean_ρ. We can easily find the holding points by "maximizing the mean_ρ." This idea is analogous to finding the minimal number of covers that enclose all samples, and this is a classical *vertex cover* (VC) problem [29]. Since this is a NP-hard problem, there are several optimization methods which can solve this problem.

Solving the VC problem [30] is the same as searching for the optimal covers in our algorithm. The original formulation of VC problem is as

$$\min\left\{\sum_{i\in\sum} w_i s_i \,\middle|\, \mathrm{As} \geq b, s \in \{0,1\}^{|\sum|}\right\} \tag{7}$$

where $\sum$ is a set of covers and s_i is the characteristic vector; w_i is a (nonnegative) cost associated with $\sum$ and which is always 1.0 in our algorithm; A is a 0-1 matrix and $b = \vec{1}$. This problem seeks to find the minimum cost of a set of covers. Now, we formulate our objective function of finding the optimal covers as

$$\min\left\{\sum_{i\in\sum} w_i s_i + \mu \sum_{\substack{\rho_t,\rho_l\in\Sigma \\ t\neq l}} |\rho_t \cap \rho_l| \,\middle|\, s \in \{0,1\}^{|\sum|}, \mathrm{As} \geq b\right\} \tag{8}$$

where ρ_t and ρ_l are two covers from cover set Σ. There is an additional term in (8), which states that the objective function should include the minimal joint samples between covers. This additional term is related to the second condition in Definition 2. Equation (8) is the objective function for **MCS *algorithm***, and the final centroids which satisfy this object function are the holding points of possible clusters. In this work, we use the greedy algorithm [20] to search for the optimal $\{\rho_i\}$, as shown in Table III. The worst-case time complexity is $O((|\Sigma| + \sum_{j=1}^{K} |\rho'_j|) \times \log |\Sigma|)$ where $\rho'_j \in \Sigma, j = 1, 2, \ldots, K$.

TABLE III
SEARCHING MINIMAL NUMBER OF COVER ALGORITHM

Finding Minimal number of Covers to enclose the Samples (MCS) Algorithm

Label the covers in Σ from ***CFC algorithm*** with ρ'_j, ***for*** $j = 1,2,\ldots,K$.
Mark all ρ'_j as un-visited.
Initialize the characteristic vector $s = \{0\}^K$,
Let $|\rho'_{anchor}|$ is maximal.
$s[anchor] = 1$;
Mark ρ'_{anchor} as visited;
WHILE existed Φ is a set of un-visited covers that have the joint element with ρ'_{anchor},
  FOR all the elements in Φ,
    IF $\rho'_{pioneer} \in \Phi$ and $|\rho'_{pioneer} \cap \rho'_{anchor}|$ is minimal;
      $s[pioneer] = 1$;
      Mark $\rho'_{pioneer}$ as visited;
      Replace the index of ρ'_{anchor} as $\rho'_{pioneer}$;
    END IF
  END FOR
END WHILE

These centroids of the covers obtained from the **MCS *algorithm*** are the holding points of possible clusters. But several of those possible clusters need to be spliced into one final cluster, as described as follows.

C. Splicing Covers Into a Cluster

We obtain m covers $\rho_1, \rho_2, \ldots, \rho_m$ using the **MCS *algorithm***, and each cover represents one possible cluster. In this section, we discuss how to splice these possible clusters into final clusters. The splicing criterion can be used to determine how many possible clusters are spliced into one final cluster, and this is based on the fuzzy proximate function that used in the **CFC *algorithm***. We apply two splicing criterion functions for the two proximate functions in this paper.

TABLE IV
SPLICING COVERS INTO FINAL CLUSTERS

Splicing Covers into Clusters (SCC) Algorithm

Let $\{\pi\}$ be a set of final clusters,
Initialize $\{\pi\}=\{\varnothing\}$ and $c=0$.
$\{\rho'_k, k=1,2,\ldots,m\}$ is the set of covers obtained from the ***MCS*** algorithm.
Rank the covers $\{\rho'_k\}$ into $\{\rho''_j, j=1,2,\ldots,m\}$ where $|\rho''_m| \ge |\rho''_{m-1}| \ge \ldots \ge |\rho''_1|$.
$\rho''_m \to \pi_1$
FOR all ρ''_j, $j=m-1$ *to* 1
  IF there is a $\rho''_w \in \pi_u$, ρ''_j and ρ''_w satisfy Eq. (9) or Eq. (10),
    $\rho''_j \to \pi_u$
  ELSE
    $c=c+1$
    $\rho'_k \to \pi_c$
  END IF
END FOR

For the spherical proximate function in (4), ρ_i and ρ_j are two spherical covers. If these two covers belong to one final cluster, they must have joint elements. In addition to this, they must satisfy the following:

$$\max|\rho_{\rho_i \cap \rho_j}| \ge \psi \min\{|\rho_i|, |\rho_j|\} \tag{9}$$

where $\rho_{\rho_i \cap \rho_j}$ is a cover, and its centroid is one of the joint elements of ρ_i and ρ_j; $|\rho_{\rho_i \cap \rho_j}|$ is the number of samples enclosed by $\rho_{\rho_i \cap \rho_j}$; ψ is a constant and it is smaller than and close to 1.0. Equation (9) estimates the density of the cover. If the number of elements in $\rho_{\rho_i \cap \rho_j}$ is similar to the numbers of elements in ρ_i and ρ_j, then they are parts of the final cluster. Otherwise, they are independent. All jointed spherical covers in X that satisfy the criterion function in (9) can be spliced into one final cluster.

The hyperplane proximate function in (5) is another fuzzy proximate function used in this paper. This criterion function is composed of the included angle and the densities of two jointed hyperplane covers ρ_i and ρ_j as

$$\cos(\angle(\rho_i, \rho_j)) \times \frac{|\rho_i \cap \rho_j|}{|\rho_i| + |\rho_j|} \ge t \tag{10}$$

where $\angle(\rho_i, \rho_j)$ is the included angle of ρ_i and ρ_j. If the value in (10) is larger than a certain threshold, t, we splice two covers into one final cluster. Otherwise, they are independent clusters. The procedure of this splicing algorithm is shown in Table IV. The worst-case time complexity of this algorithm is $O(m^2)$, where m is the number of covers obtained from the MCS algorithm. Although the time complexity depends on the square of size m, it does not need too much execution time on this procedure. The amount of the remainder covers after the MCS algorithm is much smaller than that in the initial dataset, as $n \ge K \gg m$.

After the splicing procedure, the final clusters are obtained, i.e., c clusters. We need to further evaluate the cluster validity for this result, as follows.

D. Goodness of λ

There are many validity functions that can help to evaluate the final cluster performance [32], [25]. In this approach, a different λ value affects the clustering performance. We propose a scatter-based validity function T to evaluate the goodness of λ as

$$\sum_{i=1}^{c} \frac{S_w^i + d_m}{S_b^i + d_m} \tag{11}$$

where S_w is the within-class scatter, which computes the mean of distances pairs of centroids that belong to the same final cluster; S_b is the between-class scatter, which computes the mean of distances pairs of centroids that belong to different final clusters; and d_m is the average distances between samples.

The range of T is in $[1/2, 2]$. Two extremes obtained when each individual sample takes a single cluster or all samples belonging to one cluster. In our experiments, we use this validity function to evaluate the overall clustering results.

E. Membership Degree Computation

Now, consider the membership degree of the sample x_j to the cluster π_i in the FCC algorithm. The μ_{ij} value is given by

$$\mu_{ij} = \mu(x_j, \pi_i) = \max_{\{\rho_k^i \in \pi_i\}}\{h(x_j, x_k)\}, \quad i = 1, 2, \ldots, c \quad j = 1, 2, \ldots, n \tag{12}$$

where x_k is the centroid of ρ_k^i. We calculate all the membership degrees in $U = \{u_{ij}\}$ using (12). The holding points in X are the samples whose membership degrees are 1.0. Several experiments are provided to explain how to construct the clusters in a dataset using the proposed clustering algorithm.

IV. EXPERIMENTAL RESULTS

In this section, we provide four examples to illustrate the proposed clustering method. We first present two simple examples to show how the FCC algorithm works. Then, two realistic examples are used to demonstrate the effectiveness of the FCC algorithm.

Example 1 Gaussian Distributed Data: In first experiment we have two classes, each a 150-point Gaussian distribution centered at (0, 0) and (6, 0), respectively. The variances of both distributions are 1.0. The dataset is shown in Fig. 5(a). We now group the samples by the proposed clustering algorithm based on spherical proximity function, and the clustering result is Fig. 5(b). In this experiment, we obtain five covers in left cluster and six covers in right cluster, respectively, as we set $\lambda = 0.85$. The symbols "•" are the centroids of those covers, and they form cores of the two clusters. It can be seen that the proposed algorithm identifies those centroid samples that can serve as the holding points. x_f is the final cluster center among five covers on the left. The coordinate of x_f is $(-0.1608, -0.098)$, and it is fairly close to the mean of the left distribution, i.e., (0, 0). Furthermore, x_r is the final center on the right covers and its coordinate is $(5.9925, -0.1806)$. It is close to the mean of the right distribution, i.e., (6, 0). From this experiment, it can be seen that we identify holding points by the FCC algorithm and link those holding points to form the cores of final clusters. The final cluster centers can be easily associated with the mean of the Gaussian distribution.

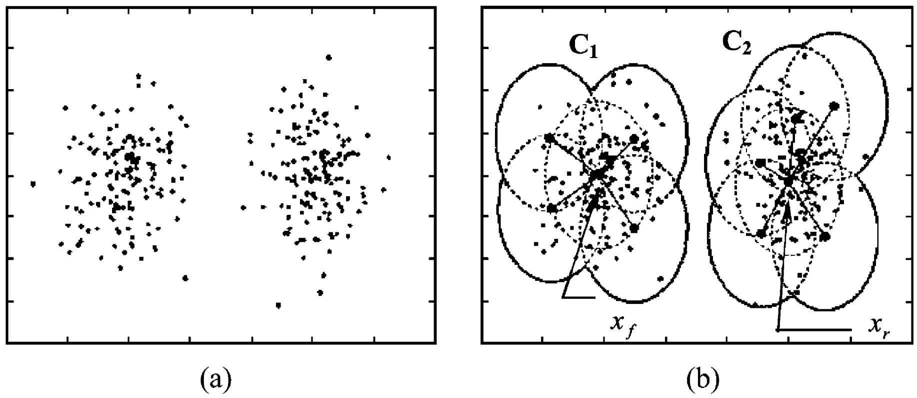


Fig. 5. Two Gaussian distributions dataset. (a) Source dataset. (b) Clustering result obtained from the FCC algorithm.

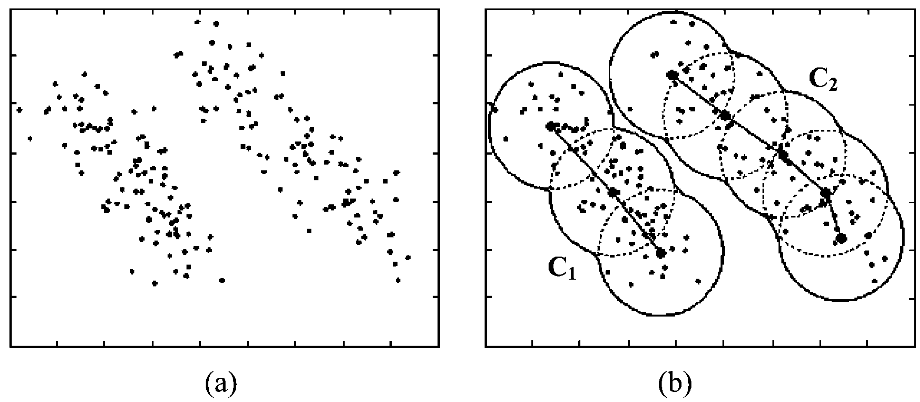


Fig. 6. Two nonspherical clusters example. (a) Source dataset. (b) Clustering result obtained from the FCC algorithm.

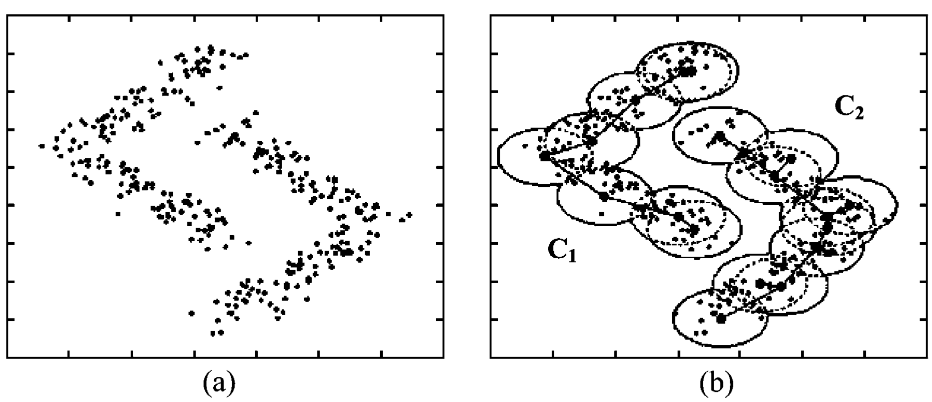


Fig. 7. Two-hooked-cluster dataset. (a) Source dataset. (b) Clustering result obtained from the FCC algorithm.

Example 2 Nonspherical Problem: The second example involves two nonspherical datasets. Fig. 6(a) is an example of two elongated clusters.

The clustering result obtained from the FCC algorithm with the spherical proximate function is shown in Fig. 6(b). In this experiment, we use the spherical cover with $\lambda = 0.85$ to construct the clusters. It can be seen that we obtain three covers for the left cluster and five covers for the right cluster. As we noted before, the centroids of covers form the backbones of the two clusters.

Fig. 7(a) illustrates a two-hooked-cluster dataset, and it is similar to the nonspherical clusters proposed in [23] and [18]. In this example, we use the spherical proximate function with $\lambda = 0.9$ as our fuzzy proximate function. The clustering result is shown in Fig. 7(b), the cover centroids form the backbones of the two hooks. It can be seen that the FCC algorithm identifies the samples that can serve as the holding points and produces two reasonable hook-shaped clusters.

Example 3 Breast Cancer Databases From the University of Wisconsin: We now adopt the Wisconsin breast cancer database obtained from the Machine Learning Repository[1] as a more realistic clustering example. The number of instances is 699 (as of 15 July 1992), and each instance has 10 attributes plus the class attribute. There are 16 missing instances in the original

[1]UCI Machine Learning Repository http://www.ics.uci.edu/~mlearn/ML-Repository.html

TABLE V
CONFUSION MATRIX OBTAINED FROM THE WISCONSIN BREAST CANCER DATABASE.

		Assigned class		
		Benign	Malignant	Accuracy
True class	Benign	437	7	98.42 %
	Malignant	18	221	92.47 %
	Total			96.34%

dataset, which we removed. We then apply the rest of the samples to the FCC algorithm with the hyperplane proximate relation function. Two clusters are obtained in this experiment, 437 samples correctly grouped into the cluster "Benign" and 221 samples grouped to another cluster "Malignant." Based on the original class label, the overall accuracy of these two classes is 96.34%, as shown in Table V. Followings are several publications reported in the original repository: 93.7% from the 1-nearest neighbor method [34], 93.5% from the two pairs of parallel hyperplanes method, and 95.9% from the three pairs of parallel hyperplanes method [31]. In this experiment, the unsupervised FCC algorithm provides good clustering results, encouraging the study below on cluster analysis for handwritten digits.

Example 4 Allograph Analysis in Handwritten Digits: Traditional classification schemes may consider all of the patterns which belong to the same class as one category. However, this may not be a suitable approach for handwritten digit patterns since many digit classes consist of subclasses due to different styles of writing. Those specific subclass prototypes are referred to as allographs [26]. In order to recognize the distinct allographs, clustering-based scheme can help to find typical prototypes to construct valid training sets for classification. The dataset for this example consists of un-normalized binary handwritten digits extracted from the BR digit set of the standard CEDAR CDROM-1 [10] and the ITRI database [3]. To construct the prototypes, we used two classes of patterns, class "2" and class "5", where there are 600 samples per class. We utilized the 60-dimensional transition feature set [8] as the input vector for each sample during the FCC clustering.

The first digit class we employ in this experiment is "2." We compare the clustering performance of the FCC algorithm with the FCM approach in order to analyze the writing variation within individual clusters. In this experiment, we utilize the spherical proximate function with $\lambda = 0.875$, and five clusters are obtained from the FCC algorithm, as shown by Fig. 8(a). The samples in each row are the top ten representative samples according to membership values for each corresponding cluster. The expressions of the representative samples in each row are neat and tidy. The samples of first two clusters do not have a distinct loop on the bottom. However, the samples in ***cluster B*** have larger cambers than in ***cluster A***. In ***cluster C***, the loop is more distinct than the first two clusters. The last two clusters are much different from the first two clusters, and the loop becomes

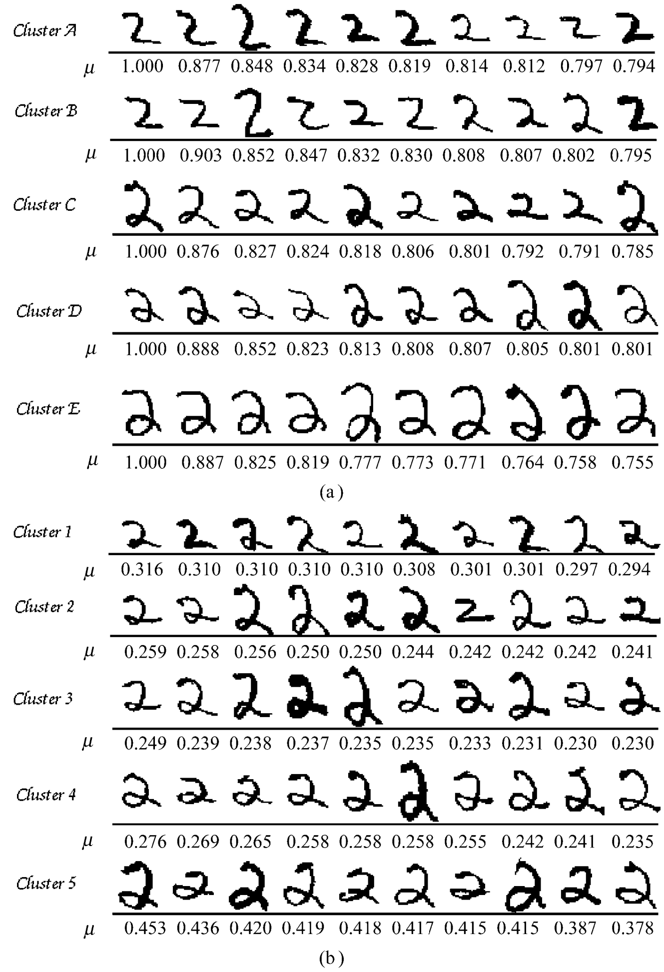


Fig. 8. (a) Clustering results obtained from the FCC. (b) Clustering results obtained from the FCM.

a recognizable feature. In addition to this, the samples in ***cluster E*** have the larger cambers than in ***cluster D***. Clearly, these five clusters can be treated as typical allographs in digit class "2." Now, we apply the FCM to the same examples with number of clusters $c = 5$. The clustering result obtained from the FCM is shown in the Fig. 8(b), which shows the top ten samples that are closest to the cluster centers according to membership values. It can be seen that the cluster seems unclearly expressed since the FCM equally divides the patterns into five groups. This result does not mean that FCM is not useful, only that FCM unable to detect holding points for partition. In other words, we can not use it to find typical prototypes since the cluster center does not properly stand for its corresponding partition.

The second experiment involves another handwritten digit "5." The writing style of this class "5" has more strokes than other numerals, and its strokes include a right angle "⅃" and an arc "⊃." Presumably it has several differentallographs in this class. We use spherical proximity function with $\lambda = 0.85$, and sixclusters are obtained from the FCC algorithm, as shown by Fig. 9(a). The samples in first three clusters are similar to the printed style of digit "5," and they are neat. The degrees of slants are increasing from the first cluster to third one. The following three clusters are cursive writing styles. Clearly, the styles of ***cluster D*** and ***cluster F*** are tilted forward whereas ***cluster E*** is

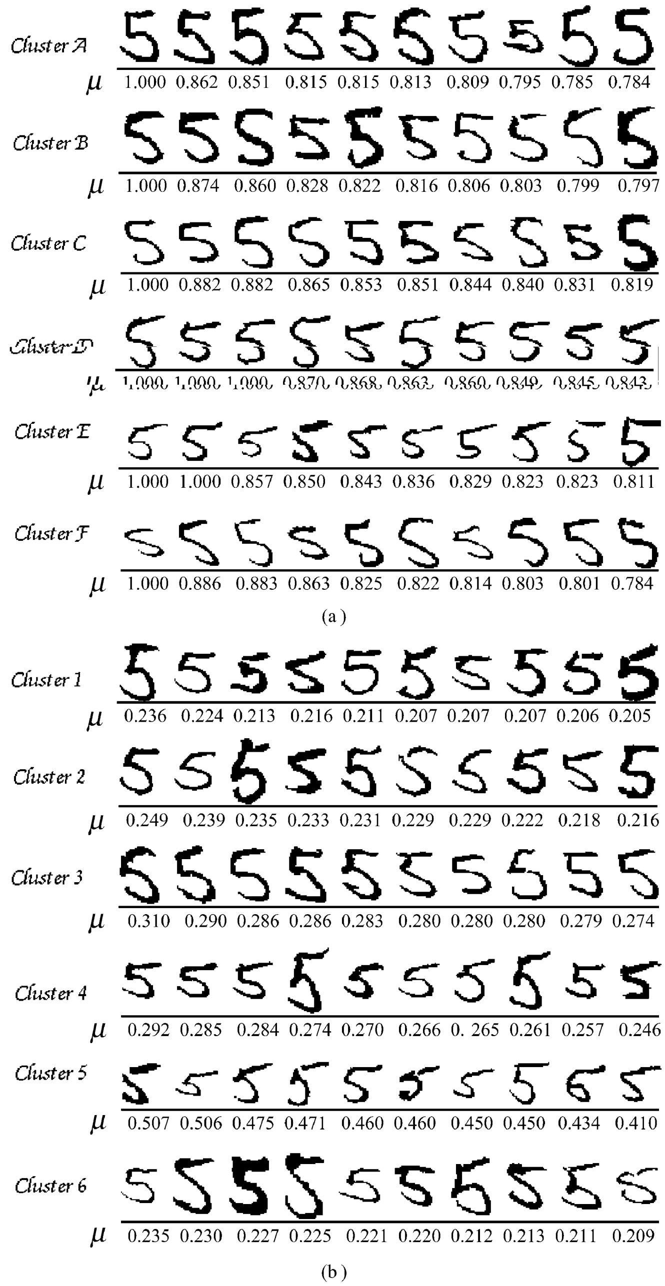


Fig. 9. (a) Clustering results obtained from the **FCC**. (b) Clustering results obtained from the **FCM**.

backward tilt. ***Cluster F*** is an important allograph since it can easily be confused with the letter "S". There are three and two exemplars in ***cluster D*** and ***cluster E*** with membership degrees of 1.0, respectively. This is because each of the two clusters do not overspread by a single spherical cover. The clustering results obtained from the FCM algorithm with $c = 6$ are shown in Fig. 9(b), showing that the expression of each cluster is confused and disordered. It is clear from both experiments that the FCC approach can be a good way to identify particular allographs in individual digit class involving different writing styles.

V. Conclusion

In this paper, we provide a new fuzzy clustering algorithm based on binary fuzzy relations in dataset. The fuzzy relations in our algorithm do not need to satisfy the reflexive, symmetric and transitive properties of the fuzzy equivalent relation. This advantage provides more practical clustering application. In this work, we utilize the fuzzy covers to find the holding points which support the structure of clusters in the original sample space, and then associate those holding points together to form final clusters. Two synthesis examples and two real problems are applied to show how the proposed algorithm works, and we also analyze the effectiveness of the representative data points in the clusters. Future study could focus on investigating different kinds of fuzzy relations to enrich the grouping capability of this approach, and express the clustering results linguistically into mathematical forms by using the fuzzy rule-based approach.

Appendix

Lemma 1: Given a *t-norm* $i\ :\ [0,1] \times [0,1] \rightarrow [0,1]$ in X, then a binary relation $r(x,y) = \sup\{u \,|\, u \in U, i(u, x \vee y) \leq x \wedge y x, y \in X\}$ must be a fuzzy relation.

Proof: The reason comes from the facts: the upper bound of the *t-norm* operator is $\min(x,y)$. Now, we assume that $u = (x \wedge y)$ then

$$\begin{aligned} i(u, x \vee y) &= i(x \wedge y, x \vee y) \leq \min[x \wedge y, x \vee y] \\ &= (x \wedge y) \wedge (x \vee y) = x \wedge y \end{aligned}$$

as this inequality shows that the minimal value of the u is $x \wedge y$, because if $u < x \wedge y$, we can not find a *t-norm* operator that can satisfy $i(u, x \vee y) = x \wedge y$. Consequently, the fuzzy relation $r(x,y) \geq x \wedge y = u$.

This lemma makes it possible to obtain fuzzy relation from an arbitrary *t-norm*. When we define a set of fuzzy relation in our algorithm, it must satisfy the condition of *Lemma 1*. According to this, we have following definitions.

Definition A1: Given a fuzzy relation $R(x,y)\ :\ [0,1] \times [0,1] \rightarrow [0,1]$. We modify the transitive property as

$$\forall x, y, z \in X, \qquad R(x,y) \geq i(R(x,z) \quad R(z,y)) \qquad \text{(A1)}$$

and it satisfies reflexive and transitive properties. Then, $R(x,y)$ is an *extended fuzzy equivalence relation*.

According to basic fuzzy theory, the fuzzy equivalence relation must satisfy $i(x,y) = x \wedge y$. Unfortunately, this constraint fails for most cases. Now, we define a new extended fuzzy equivalent relation, which does not satisfy the constraints such as fuzzy equivalent relation. For this extended definition, it is easier to find a suitable operator based on need. According to this definition, many kinds of fuzzy relations can be selected as candidates for the extended fuzzy equivalence relation.

Definition A2: A set of covers $H = \{\gamma_1, \gamma_2, \ldots, \gamma_n\}$ in X satisfies the following condition:

$$\forall x, y \in X \quad R(\gamma_j(x), \gamma_j(y)) \geq r(x,y), \qquad j = 1, 2, \ldots, n \qquad \text{(A2)}$$

and is called $r/R-T$ covers. This is the extended definition of classical fuzzy cover.

Assume that x_j is the center of cover γ_j. When we choose a fuzzy equivalence relation $R(x,y)$, it must satisfy that the fuzzy relation of aggregating $r(x_j, x)$ and $r(x_j, y)$ should not be less than the fuzzy relation $r(x, y)$ of x and y. That is, if two elements x and y have higher binary fuzzy relation, then they both have high membership degrees to one cluster. Now, we define the condition that $R(x, y)$ must satisfy.

Theorem 1: Given any fuzzy relation $r(x,y)$ and $G = \{g_z | z \in X, \forall y \in X, g_z(y) = r(z,y)\}$ where $r(x,y) = \sup\{u | u \in U, i(u, x \vee y) \leq x \wedge y\}$. G is a $r/R-T$ cover for X if and only if $r(x, y)$ is an extended fuzzy equivalence relation.

Proof: According to Definition A2, we need to check whether the inequality (A2) satisfies Definition A1. The necessary condition of Theorem 1 is proved as follows.

From the definition of the $r/R - T$ cover, we obtain $R(g_z(x), g_z(y)) = R(r(z,x), r(z,y))$. Since $R(x, y)$ is an extended fuzzy equivalence relation, the following inequality:

$$\begin{aligned} i(r(z,x), r(z,y)) &\leq i(R(r(z,s), r(s,x)), R(r(z,s), r(s,y))) \\ &\leq R(r(x,s), r(y,s)) \quad \text{(A3)} \end{aligned}$$

holds. If $s \in X$, then $R(r(x,s), r(y,s)) = r(x,y)$ holds. Consequentially, (A3) can be formulated as $r(x,y) \geq i(r(x,z), r(z,y))$ that satisfies Definition A1, and the necessary condition is proof. Besides, the sufficient condition of Theorem 1 is derived as follows.

If $r(x,y)$ is an extended fuzzy equivalence relation, we have

$$\forall x, y, z \in X \qquad i(r(x,z), r(z,y)) \leq r(x,y). \quad \text{(A4)}$$

Now, our goal is to obtain the following equations:

$$\begin{aligned} &\forall x, y, z \in X r(g_z(x), g_z(y)) \\ &\quad = r(r(x,z), r(z,y)) \\ &\quad = \sup\{u \,|, i(u, r(x,z) \vee r(z,y)) \\ &\qquad \leq r(x,z) \wedge r(z,y)\} \\ &\quad = \sup\{u \,|\, i(u, r(x,z) \vee r(z,y)) \\ &\qquad \leq r(x,z) \wedge r(z,y)\} \geq r(x,z) \wedge r(z,y \\ &\quad \because\ r(x,z) \wedge r(z,y) \geq r(x,y) \\ &\quad \Rightarrow \sup\{u \,|\, i(u, r(x,z) \vee r(z,y)) \\ &\qquad \leq r(x,z) \wedge r(z,y)\} \geq r(x,y) \\ &\quad \text{then}, r(g_z(x), g_z(y)) \geq r(x,y). \quad \text{(A5)} \end{aligned}$$

First, we assume that $r(x,z) \leq r(z,y)$ and need to check whether it satisfy the inequality (A5). We have the following inequality:

$$\begin{aligned} r(g_z(x), g_z(y)) &= \sup\{u | i(u, r(x,z) \vee r(z,y)) \\ &\qquad \leq r(x,z) \wedge r(z,y)\} \\ &= \sup\{u \,|\, i(u, r(z,y)) \\ &\qquad \leq r(x,z)\} \geq r(x,z) \geq r(x,y) \quad \text{(A6)} \end{aligned}$$

and, thus, proving the conclusion. On the other hand, assuming $r(z,x) > r(z,y)$, the proof is similar. This completes the proof.

REFERENCES

[1] K. P. Bennett and O. L. Mangasarian, "Robust linear programming discrimination of two linearly inseparable sets," in *Optimization Methods and Software 1*. New York: Gordon and Breach, 1992, pp. 23–34.

[2] J. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

[3] J.-H. Chiang and P. Gader, "Hybrid fuzzy-neural systems in handwritten word recognition," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 497–510, Aug. 1997.

[4] R. N. Dave and R. Krishnapuram, "Robust clustering methods: A unified view," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 270–293, Apr. 1997.

[5] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *J. Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.

[6] ——, "A graph theoretic analysis for pattern classification via Tamura's fuzzy relation," *IEEE Trans. Syst., Man, Cybern.*, vol. 4, pp. 310–313, May 1974.

[7] G. Fu, "Optimization methods for fuzzy clustering," *Fuzzy Sets Syst.*, vol. 98, pp. 301–309, 1998.

[8] P. Gader, M. Mohamed, and J.-H. Chiang, "Handwritten word recognition with character and inter-character neural networks," *IEEE Trans. Syst., Man, Cybern.*, vol. 27, pp. 158–164, Jan. 1997.

[9] L. Gonzalez and A. Marin, "Level-based extension of relations," *Fuzzy Sets Syst.*, vol. 84, pp. 95–96, 1996.

[10] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, pp. 550–554, June 1994.

[11] H. Imai, A. Tanaka, and M. Miyakoshi, "A method of identifying influential data in fuzzy clustering," *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 90–101, Feb. 1998.

[12] M. A. Ismail and S. A. Selim, "On the local optimality of the fuzzy ISODATA clustering algorithm," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, pp. 284–288, Apr. 1986.

[13] M. S. Kamel and S. Z. Selim, "New algorithms for solving the fuzzy clustering problem," *Pattern Recogn.*, vol. 27, no. 3, pp. 421–428, 1994.

[14] A. Kandel, *Fuzzy Mathematical Techniques with Applications*. Reading, MA: Addison-Wesley, 1986.

[15] N. B. Karayiannis and J. C. Bezdek, "An integrated approach to fuzzy learning vector quantization and fuzzy C-means clustering," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 622–628, Aug. 1997.

[16] G. J. Klir and Y. Bo, *Fuzzy Sets and Fuzzy Logic*. New York: Prentice-Hall, 1995.

[17] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 98–110, Apr. 1993.

[18] R. Lippmann, "An introduction to computing with neural network," *IEEE Acoust., Speech, Signal Processing Mag.*, vol. 61, pp. 4–22, Jan. 1987.

[19] Y. Man and I. Gath, "Detection and separation of ring-shaped clusters using fuzzy clustering," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, pp. 855–861, Aug. 1994.

[20] U. Manber, *Introduction to Algorithms—A Creative Approach*. Reading, MA: Addison-Wesley, 1989, pp. 208–212.

[21] O. L. Mangasarian, R. Setiono, and W. H. Wolberg, "Pattern recognition via linear programming: Theory and application to medical diagnosis," in *Large-Scale Numerical Optimization*. Philadelphia, PA: SIAM, 1990, pp. 22–30.

[22] O. L. Mangasarian and W. H. Wolberg, "Cancer diagnosis via linear programming," *SIAM News*, vol. 23, no. 5, pp. 1–18, Sept. 1990.

[23] G. Nagy, "State of the art in pattern recognition," in *Proc. IEEE*, vol. 56, May 1968, pp. 836–882.

[24] S. Ovchinnikov, "Similarity relations, fuzzy partitions and fuzzy orderings," *Fuzzy Sets Syst.*, vol. 40, pp. 107–126, 1991.

[25] N. R. Pal and J. C. Bezdek, "On cluster validity for the fuzzy C-means model," *IEEE Trans. Fuzzy Syst.*, vol. 3, pp. 370–379, June 1995.

[26] M. Parizeau and S.-W. Lee, "A fuzzy-syntactic approach to allograph modeling for cursive script recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 702–712, July 1995.

[27] W. Pedrycz, "Fuzzy multimodels," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 139–148, Apr. 1996.

[28] K. Rose, E. Gurewitz, and G. C. Foz, "Constrained clustering as an optimization method," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 785–794, Aug. 1993.

[29] S. Skiena, "Minimum vertex cover," in *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Reading, MA: Addison-Wesley, 1990, p. 218.

[30] F. Toshihiro, "Approximation algorithms for submodular set cover with applications," *IEICE Trans. INF. SYST.*, vol. E83-D, no. 3, Mar. 2000.

[31] H. W. Wolberg and O. L. Mangasarian, "Multisurface method of pattern separation for medical diagnosis applied to breast cytology," *Proc. Natl. Acad. Sci.*, vol. 87, pp. 9193–9196, Dec. 1990.

[32] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 841–846, Aug. 1991.

[33] L. A. Zadeh, "Similarity relations and fuzzy orderings," *Inform. Sci.*, vol. 3, pp. 177–200, 1971.

[34] J. Zhang, "Selecting typical instances in instance-based learning," in *Proc. 9th Int. Machine Learning Conf.*, Aberdeen, U.K., 1992, pp. 470–479.

Jung-Hsien Chiang (M'02) received the B.Sc. degree in electrical engineering from National Taiwan Institute of Technology, Taiwan, and the M.S. and Ph.D. degrees in computer engineering from the University of Missouri, Columbia, in 1991 and 1995, respectively.

He is an Associate Professor in the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan. Previously, he worked as a Researcher at the Computer and Communication Laboratory, Industrial Technology Research Institute, the largest information technology research institute in Taiwan. His current research interests include fuzzy modeling, neural networks, pattern clustering, and modeling of bioinformatics problems.

Shihong Yue received the B.Sc. degree in mathematics from Yili Normal University, Xinjiang, China, the M.S. degree in application mathematics from Xi'An University of Technology, China, and the Ph.D. degree in application mathematics from Xi'An Jiaotong University, in 1997 and 2000, respectively.

Since 2000, he has been with the Institute of Industrial Process Control from Zhejiang University, China, as a Postdoctoral Researcher. He became a Deputy Professor in July 2001.

Zong-Xian Yin received the B.S. and M.S. degrees in industrial design from National Cheng Kung University, Taiwan, in 1997 and 2000, respectively. He is currently working toward the Ph.D. degree in the Department of Computer Science and Information Engineering, National Cheng Kung University.

His research interests include machine intelligence, fuzzy theory, and pattern clustering.