Pattern Recognition 48 (2015) 2710–2724

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

# NCM: Neutrosophic *c*-means clustering algorithm

Yanhui Guo [a,*], Abdulkadir Sengur [b]

[a] School of Science, St. Thomas University, Miami Gardens, FL 33054, USA
[b] Department of Electrical and Electronics Engineering, Firat University, 23119 Elazig, Turkey

ARTICLE INFO

*Article history:*
Received 20 July 2013
Received in revised form
20 December 2014
Accepted 19 February 2015
Available online 5 March 2015

*Keywords:*
Data clustering
Fuzzy *c*-means clustering
Neutrosophic set
Neutrosophic clustering
Image segmentation

ABSTRACT

In this paper, a new clustering algorithm, neutrosophic *c*-means (NCM), is introduced for uncertain data clustering, which is inspired from fuzzy *c*-means and the neutrosophic set framework. To derive such a structure, a novel suitable objective function is defined and minimized, and the clustering problem is formulated as a constrained minimization problem, whose solution depends on the objective function. In the objective function, two new types of rejection have been introduced: the ambiguity rejection which concerns the patterns lying near the cluster boundaries, and the distance rejection dealing with patterns that are far away from all the clusters. These measures are able to manage uncertainty due to imprecise and/or incomplete definition of the clusters. We conducted several experiments with synthetic and real data sets. The results are encouraging and compared favorably with results from other methods as FCM, PCM and FPCM algorithms on the same data sets. Finally, the proposed method was applied into image segmentation algorithm. The experimental results show that the proposed algorithm can be considered as a promising tool for data clustering and image processing.



## 1. Introduction

Data clustering or cluster analysis is an important field in pattern recognition, machine intelligence and computer vision community, which has a numerous of applications in the last three decades [1–4]. Generally, clustering term is known as grouping a set of $N$ samples into $C$ clusters whose members are similar in some sense. This similarity between different samples is either a suitable distance based on numeric attributes, or directly in the form of pair-wise similarity or dissimilarity measurements.

Clustering can classify similar samples into the same group. The clustering process could be described as follows. Let $X=\{x_i,\ i=1,2,\ldots,N\}$ be a data set, and $x_i$ be a sample in a $d$-dimensional space. The problem of hard clustering is to find a partition $P=\{p_1,p_2,\ldots,p_C\}$, which satisfies $X=\cup_{i=1}^{C} p_i$, $p_i\neq\Phi$ for $i=1,2,\ldots,C$, $p_i\cap p_j=\Phi$ for $i,j=1,2,\ldots,C; i\neq j$.

In literatures, most of the clustering algorithms can be roughly classified into two types [5,6]: hard and fuzzy clustering methods. In hard clustering methods, data are grouped so that if a certain data point belongs to a cluster, then it cannot be included in other clusters. On the contrary, with fuzzy partitioning, each object may belong to several clusters with different degrees of membership.

The $k$-means type algorithms are one of the hard clustering algorithms that widely used in real applications [7–9]. The main advantages of these algorithms are its simplicity and speed, which allow it to run on large datasets. However, it depends on the initial assignments and might not yield the same result with each run. Moreover, it tries to minimize intra-cluster variance, but does not ensure that the result has a global minimum of variance. The $k$-means++ is proposed as an approximation algorithm for a NP-hard $k$-means problem, and used for choosing the initial values for the $k$-means clustering algorithm [10]. The $k$-means++ algorithm improves the poor clustering sometimes in the standard $k$-means algorithm. Kang et al. [11] presented a variation of $k$-means clustering, $k$-medians clustering, where it calculates the median for each cluster to determine its centroid, instead of the mean value. It has the effect of minimizing error over all clusters with respect to the 1-norm distance metric, in contrast to the square of the 2-norm distance metric. The $k$-medoids algorithm is also related to the $k$-means algorithm [12]. It attempts to minimize squared error, the distance between points labeled to be in a cluster and a point designated as the center of that cluster. In contrast to the $k$-means algorithm, $k$-medoids chooses data points as centers.

The fuzzy *c*-means algorithm (FCM) is one of the most popular fuzzy clustering algorithms where the membership degrees of the data are obtained through iterative minimization of a cost function, subject to the constraint that the sum of membership degrees over the clusters for each data be equal to 1. The FCM algorithm suffers from several drawbacks: it also tries to minimize the intra-cluster variance as well, and has the same problems as $k$-means algorithm;

* Corresponding author at: School of Science, Technology, and Engineering Management, St. Thomas University, Miami Gardens, Florida 33054. Tel.: +305 474 6015 (Office); fax: +305 628 6706.
*E-mail addresses:* yanhui.guo@aggiemail.usu.edu, yguo@stu.edu (Y. Guo).
*URL:* http://www.stu.edu/science (Y. Guo).

http://dx.doi.org/10.1016/j.patcog.2015.02.018
0031-3203/

the minimum is a local minimum, and the results depend greatly on the initializations [2]. In addition, the FCM algorithm is very sensitive to the presence of noise. The membership of noise points might be significantly high. The FCM algorithm cannot distinguish between equally highly likely and equally highly unlikely [2,13,14], and is sensitive to the selection of distance metric. To overcome these problems, Krishnapuram and Keller [15] proposed a clustering model named possibilistic *c*-means (PCM), where the constraint is relaxed. In PCM, the membership is interpreted as the compatibilities of the datum to the class typicality which corresponds to the intuitive concept of degree of belonging or compatibility. However, PCM is sensitive to initializations, and sometimes generates coincident clusters. Moreover, the membership values to the clusters are also sensitive to the choice of the additional parameters in PCM. To solve the PCM coincident cluster problem, Pal et al. [16] combine PCM and FCM, and took into account both relative and absolute resemblance to cluster centers. Gustafson and Kessel [17] proposed a G–K algorithm using the Mahalanobis distance as the metric in FCM. They reported that the G–K algorithm is better than Euclidean distance based algorithms when the shape of data is considered. A robust clustering algorithm called noise clustering (NC) was proposed by Dave [18]. The algorithm modified the objective of FCM to make the parameter estimation more resistant to noise. So, the inability of FCM algorithm to detect atypical data points can be solved. NC has two major drawbacks. First, a single parameter is used to describe the scale or resolution parameter. This is clearly insufficient in the general clustering problem where clusters are not guaranteed to be of the same size or to have the same inlier bounds. Second, the scale parameter needs to be known in advance, or pre-estimated from the data. There also exist several fuzzy based clustering approaches such as Roubens' fuzzy non-metric model (FNM) model [19], the relational fuzzy *c*-means (RFCM) model [20] and the assignment-prototype (AP) model [21]. The non-Euclidean relational fuzzy *c*-means (NERFCM) model was later extended by Hathaway and Bezdek [22] to cope with non-Euclidean dissimilarity data. The robust versions of the FNM and RFCM algorithms were then proposed by Dave [23]. More recently, various evidential clustering algorithms, including ECM and RECM were proposed by Masson and Denoeux [24,25]. The evidential clustering term generalizes existing concepts of hard, fuzzy (probabilistic) or possibilistic partitions by allowing an object to belong to several subsets of classes. An extension of the conventional FCM, type-2 FCM was proposed by Rhee and Hwang [26]. In type-2 FCM, the membership values for each sample are extended as type-2 fuzzy memberships by assigning membership grades to the classical memberships. Another type-2 FCM algorithm was proposed by Linda and Manic for uncertain fuzzy clustering [27], which can be regarded as an extension of the work in [26]. It focused on managing of uncertainty associated with the parameters of fuzzy clustering algorithms. Rhee introduced the uncertain fuzzy clustering and proposed several insights and recommendations about it [28].

Neutrosophic set (NS), was proposed as a new branch of philosophy dealing with the origin, nature and scope of neutralities, and their interactions with different ideational spectra [29]. An element ⟨E⟩ in neutrosophic set is considered in relation to its opposite, ⟨Anti-E⟩ and its neutrality ⟨Neut-E⟩, which is neither ⟨E⟩ nor ⟨Anti-E⟩, and three memberships are employed to measure the degree of truth, indeterminacy and falsity of ⟨E⟩. Based on this character, neutrosophic theory provides a powerful tool to deal with the indeterminacy, and has found practical applications in a variety of different fields, such as relational database systems, semantic web services [30], financial dataset detection [31] and new economies growth and decline analysis [32]. Moreover, several image processing applications such as image de-noising [33], thresholding [34], segmentation [35], and color texture image segmentation [36] can be seen in the literature.

In this paper, based on neutrosophic set, we propose a new clustering algorithm, neutrosophic *c*-means (NCM) clustering. NCM calculates the degrees belonging to the determinant and indeterminate clusters at the same time for each of the data points. While the membership $T$ can be considered as the membership degree to determinant clusters, and two memberships $I$ and $F$ can be used to determine two kinds of indeterminate clusters: an ambiguity cluster and an outlier cluster for each data point, respectively. Ambiguity cluster allows us to consider about the data points that are laying near the clusters boundaries and outlier cluster allows us to reject individual data points when they are very far from the centers of each cluster. Both ambiguity and outlier clusters are introduced in the clustering iterations and not in the decision processing. The membership degrees to the ambiguity and outlier class of a data point are explicit, and these values are learned in the iterative clustering problem. So, the membership functions are more immune to noise and they correspond more closely to the notion of compatibility. We proposed a new objective function to make the parameter estimation more resistant to noise and outliers, and derived the membership and prototype update equations from the conditions for minimization of our cost function. Therefore, the inability of FCM algorithm to detect atypical data points can be solved.

The rest of the paper is organized as follows. Section 2 recalls the necessary background about the FCM algorithm. Section 3 gives the definition of the NCM algorithm and explains how to compute a partition from data. The interpretation of an NCM is illustrated using synthetic and real data sets in Section 4. In addition, the proposed method was applied to image segmentation, and several experiments were taken. Finally, Section 5 concludes the paper.

## 2. Fuzzy *c*-means algorithm

A typical FCM clustering can be described as follows. Let $X=\{x_i, i=1,2,\ldots,N\}$ be a data set, and $x_i$ be a sample in a $d$-dimensional space. Let $C$ $(2\le C\le N)$ be the desired number of classes. Each cluster is represented by a center $c_k$. FCM finds a partition of data by minimizing the objective function.

$$J_m=\sum_{i=1}^{N}\sum_{j=1}^{C}(\mu_{ij})^m\|x_i-c_j\|^2 \tag{1}$$

where $m$ is any real number greater than 1, $\mu_{ij}$ is the degree of membership of $x_i$ in the cluster $j$, $x_i$ is the $i$th of $d$-dimensional measured data, $c_j$ is the $d$-dimension center of the cluster, and $\|\bullet\|$ is a norm.

Fuzzy partitioning is carried out through an iterative optimization of the objective function, and the membership $\mu_{ij}$ and the cluster centers $c_j$ are updated by

$$\mu_{ij}=\frac{1}{\sum_{k=1}^{C}\left(\|x_i-c_j\|/\|x_i-c_k\|\right)^{2/m-1}} \tag{2}$$

$$c_j=\frac{\sum_{i=1}^{N}(\mu_{ij})^m\cdot x_i}{\sum_{i=1}^{N}(\mu_{ij})^m} \tag{3}$$

The iteration will not stop until $\max_{ij}\{\left|\mu_{ij}^{(k+1)}-\mu_{ij}^{(k)}\right|\}<\varepsilon$, where $\varepsilon$ is a termination criterion between 0 and 1, and $k$ is the iteration step. This procedure converges to a local minimum or a saddle point of $J_m$. Finally, each data is assigned into different class according to the membership value.

## 3. Proposed method

In clustering analysis, the traditional methods only describe the degree to every group. In fact, for some samples, especial for the

samples on the boundary region between different groups, it is difficult to determine which group they are belonged to and their partitions are indeterminate. If a hard partition is made, it will make the centers of different clusters inaccurate. The disadvantage of traditional clustering algorithm, such as FCM, can be illustrated using an example inspired from a classical diamond data set [21]. It is composed of 12 objects, which are represented in Fig. 1. Objects 1–5, 7–11 are in two different diamond sets, whereas object 6 is in the boundary region between two diamond sets, and object 12 regards as an outlier or noise data.

The 12 objects in Fig. 1 are clustered into 2 clusters using FCM algorithm. Each object is required to assign to an appropriate cluster according to its membership value in Table 1. $T_{c1}$ and $T_{c2}$ indicate the memberships of each data point to $c1$ and $c2$ clusters. In the FCM result, objects 1–6 are assigned to cluster 1, and objects 7–12 are in cluster 2. The cluster centers of $c1$ and $c2$ are $(-2.63, 0.17)$ and (4.13, 1.17), respectively. Because of the objects 6 and 12, the cluster centers are not exactly as same as the centers diamond sets, which locate at $(-3.34, 0)$ and (3.34 0). If we remove the object 6 from cluster $c1$ and the object 12 from $c2$, the clusters become compact and their centers are as same as the centers diamond set.

In the proposed clustering algorithm, inspired by the neutrosophic set, we consider not only the degree belonging to determinate clusters, but also the degree belonging to the indeterminate clusters. A new unique set $A$ has been proposed, which regards as the union of the determinant clusters and indeterminate clusters. Let $A = C_j \cup B \cup R$, $j=1, \ldots, C$. where $C_j$ is an indeterminate cluster, $B$ regards the clusters in boundary regions, $R$ is associated with noisy data and $\cup$ is the union operation. $B$ and $R$ are two kinds of indeterminate clusters. $T$ is defined as the degree to determinant clusters, $I$ is the degree to the boundary clusters, and $F$ is the degree belonging to the noisy data set.

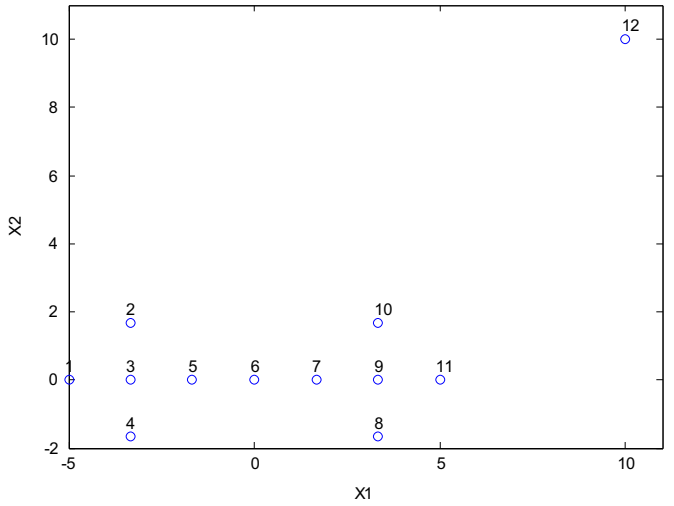


**Fig. 1.** Diamond data set.

**Table 1**
Clustering result using FCM on the diamond set.

| Object | $T_{c1}$ | $T_{c2}$ | arg max $\{T_{c1}, T_{c2}\}$ | Partition |
|---|---|---|---|---|
| 1 | 0.9376 | 0.0624 | $T_{c1}$ | $c_1$ |
| 2 | 0.9532 | 0.0468 | $T_{c1}$ | $c_1$ |
| 3 | 0.9908 | 0.0092 | $T_{c1}$ | $c_1$ |
| 4 | 0.9427 | 0.0573 | $T_{c1}$ | $c_1$ |
| 5 | 0.9735 | 0.0265 | $T_{c1}$ | $c_1$ |
| 6 | 0.7264 | 0.2736 | $T_{c1}$ | $c_1$ |
| 7 | 0.2864 | 0.7136 | $T_{c2}$ | $c_2$ |
| 8 | 0.1823 | 0.8177 | $T_{c2}$ | $c_2$ |
| 9 | 0.0531 | 0.9469 | $T_{c2}$ | $c_2$ |
| 10 | 0.0226 | 0.9774 | $T_{c2}$ | $c_2$ |
| 11 | 0.0352 | 0.9648 | $T_{c2}$ | $c_2$ |
| 12 | 0.3049 | 0.6951 | $T_{c2}$ | $c_2$ |

Considering clustering with indeterminacy, a new objective function and membership are defined as

$$J(T,I,F,C) = \sum_{i=1}^{N}\sum_{j=1}^{C} \varpi_1 T_{ij})^m ||x_i - c_j||^2 + \sum_{i=1}^{N}\sum_{j=1}^{\binom{C}{2}} \varpi_2 I_{2ij})^m ||x_i - \bar{c}_{2j}||^2 + \sum_{i=1}^{N}\sum_{j=1}^{\binom{C}{3}} \varpi_3 I_{3ij})^m ||x_i - \bar{c}_{3j}||^2 + \sum_{i=1}^{N}\sum_{j=1}^{\binom{C}{4}} \varpi_4 I_{4ij})^m ||x_i - \bar{c}_{4j}||^2 + \ldots + \sum_{i=1}^{N}\sum_{j=1}^{\binom{C}{C}} \varpi_C I_{Cij})^m ||x_i - \bar{c}_{Cj}||^2 + \sum_{i=1}^{N} \delta^2 (\varpi_{C+1} F_i)^2 \quad (4)$$

where $\bar{c}_{2j}$ is the mean of any two classes, $\bar{c}_{nj}$ is the mean of any $n$ clusters, and $\bar{c}_{Cj}$ is the mean of all clusters. $\varpi_i$ is the weight factor. $\delta$ is used to control the number of objects considered as outliers. $I_{2ij}$ is the degree to the data $i$ to any two classes, and $I_{Cij}$ is the indeterminate degree to any $C$ classes.

When the clustering number $C$ is greater than 3, the objective function in Eq. (4) is very complex and time consuming. In fact, the indeterminate degree of each data greatly depends on the determinate clusters close to it. In this situation, if we only consider the two closest determinate clusters which have the biggest and the second biggest membership values, the objective function will be simplified, and computation cost will be reduced while the clustering accuracy is not decreased greatly. This assumption will be justified in Section 4.1. After this simplification, the objective function is rewritten as

$$J(T,I,F,C) = \sum_{i=1}^{N}\sum_{j=1}^{C} (\varpi_1 T_{ij})^m ||x_i - c_j||^2 + \sum_{i=1}^{N} (\varpi_2 I_i)^m ||x_i - \bar{c}_{i\,max}||^2 + \sum_{i=1}^{N} \delta^2 (\varpi_3 F_i)^m \quad (5)$$

$$\bar{c}_{i\,max} = \frac{c_{pi} + c_{qi}}{2} \quad (6)$$

$$p_i = \arg\max_{j=1,2,\cdots C} (T_{ij}) \quad (7)$$

$$q_i = \arg\max_{j \neq p_i \cap j = 1,2,\cdots C} (T_{ij}) \quad (8)$$

where $m$ is a constant. $p_i$ and $q_i$ are the cluster numbers with the biggest and second biggest value of $T$. When the $p_i$ and $q_i$ are identified, the $\bar{c}_{i\,max}$ is calculated and its value is a constant number for each data point $i$, and will not change anymore. T. $T_{ij}$, $I_i$ and $F_i$ is the membership values belonging to the determinate clusters, boundary regions and noisy data set, $0 < T_{ij}, \ I_i, \ F_i < 1$, which satisfy with the following formula:

$$\sum_{j=1}^{C} T_{ij} + I_i + F_i = 1 \quad (9)$$

The objective function in Eq. (5) is derived from Eq. (1) whose convergence has been discussed and proved [37]. In this condition,

the convergence of Eq. (5) is achievable. According to the above formula, the Lagrange objective function is constructed as

$$L(T,I,F,C,\lambda)=\sum_{i=1}^{N}\sum_{j=1}^{C}(\varpi_1 T_{ij})^m\|x_i-c_j\|^2+\sum_{i=1}^{N}(\varpi_2 I_i)^m\|x_i-\bar{c}_{i\ max}\|^2 + \sum_{i=1}^{N}\delta^2(\varpi_3 F_i)^m-\sum_{i=1}^{N}\lambda_i\Big(\sum_{j=1}^{C}T_{ij}+I_i+F_i-1\Big) \tag{10}$$

For each point i, the $\bar{c}_{i\ max}$ is computed according to indexes of the largest and second largest value of $T_{ij}$ which are obtained using a comparison process.

To minimize the Lagrange objective function, we use the following operations:

$$\frac{\partial L}{\partial T_{ij}}=m(\varpi_1 T_{ij})^{m-1}\|x_i-c_j\|^2-\lambda_i \tag{11}$$

$$\frac{\partial L}{\partial I_i}=m(\varpi_2 I_{ij})^{m-1}\|x_i-\bar{c}_{i\ max}\|^2-\lambda_i \tag{12}$$

$$\frac{\partial L}{\partial F_i}=\delta^2 m(\varpi_3 F_i)^{m-1}-\lambda_i \tag{13}$$

$$\frac{\partial L}{\partial c_j}=-2\sum_{i=1}^{N}(\varpi_1 T_{ij})^m(x_i-c_j) \tag{14}$$

The norm is specified as the Euclidean norm. Let $(\partial L/\partial T_{ij})=0$, $(\partial L/\partial I_i)=0$, $(\partial L/\partial F_i)=0$, and $(\partial L/\partial c_i)=0$, then

$$T_{ij}=\frac{1}{\varpi_1}\left(\frac{\lambda_i}{m}\right)^{1/m-1}(x_i-c_j)^{-(2/m-1)} \tag{15}$$

$$I_i=\frac{1}{\varpi_2}\left(\frac{\lambda_i}{m}\right)^{1/m-1}(x_i-\bar{c}_{i\ max})^{-(2/m-1)} \tag{16}$$

$$F_i=\frac{1}{\varpi_3}\left(\frac{\lambda_i}{m}\right)^{1/m-1}\delta^{-(2/m-1)} \tag{17}$$

$$c_j=\frac{\sum_{i=1}^{N}(\varpi_1 T_{ij})^m x_i}{\sum_{i=1}^{N}(\varpi_1 T_{ij})^m} \tag{18}$$

Let $\left(\frac{\lambda_i}{m}\right)^{1/m-1}=K$,

$$\sum_{j=1}^{C}T_{ij}+I_i+F_i=1 \tag{19}$$

$$\sum_{j=1}^{C}\frac{K}{\varpi_1}(x_i-c_j)^{-(2/m-1)}+\frac{K}{\varpi_2}(x_i-\bar{c}_{i\ max})^{-(2/m-1)}+\frac{K}{\varpi_3}\delta^{-(2/m-1)}=1 \tag{20}$$

$$K=\left[\frac{1}{\varpi_1}\sum_{j=1}^{C}(x_i-c_j)^{-(2/m-1)}+\frac{1}{\varpi_2}(x_i-\bar{c}_{i\ max})^{-(2/m-1)}+\frac{1}{\varpi_3}\delta^{-(2/m-1)}\right]^{-1} \tag{21}$$

Therefore,

$$T_{ij}=\frac{K}{\varpi_1}(x_i-c_j)^{-(2/m-1)} \tag{22}$$

$$I_i=\frac{K}{\varpi_2}(x_i-\bar{c}_{i\ max})^{-(2/m-1)} \tag{23}$$

$$F_i=\frac{K}{\varpi_3}\delta^{-(2/m-1)} \tag{24}$$

The partitioning is carried out through an iterative optimization of the objective function, and the membership $T_{ij}$, $I_i$, $F_i$ and the cluster centers $c_j$ are updated by Eqs. (18), (22), (23) and (24) at each iteration. The $\bar{c}_{i\ max}$ is calculated according to indexes of the largest and second largest value of $T_{ij}$ at each iteration. The iteration will not stop until $\left|T_{ij}^{(k+1)}-T_{ij}^{(k)}\right|<\varepsilon$, where $\varepsilon$ is a termination criterion between 0 and 1, and $k$ is the iteration step.

The above equations allow the formulation of NCM algorithm. It can be summarized in the following steps:

**Step 1** Initialize $T^{(0)}$, $I^{(0)}$, and $F^{(0)}$;
**Step 2** Initialize the $C$, $m$, $\delta$, $\varepsilon$, $\varpi_1$, $\varpi_2$, $\varpi_3$ parameters
**Step 3** Calculate the centers vectors $c^{(k)}$ at $k$ step using Eq. (18);
**Step 4** Compute the $\bar{c}_{i\ max}$ according to indexes of the largest and second largest value of $T$ by a comparison process;
**Step 5** Update $T^{(k)}$ to $T^{(k+1)}$ using Eq. (22), $I^{(k)}$ to $I^{(k+1)}$ using Eq. (23), and $F^{(k)}$ to $F^{(k+1)}$ using Eq. (24);
**Step 6** If $|T^{(k+1)}-T^{(k)}|<\varepsilon$ then stop; otherwise return to **Step 3**.
**Step 7** Assign each data into the class with the biggest TM= [$T$, $I$, $F$] value: $x(i)\in k$th class if $k=\underset{j=1,2,\ldots,C+2}{\arg\max}\ (TM_{ij})$

## 4. Experimental results

In this section, we compare the proposed method with other approaches by FCM [3], PCM [15] and FPCM [38] on Iris data [39] and several simulated datasets to demonstrate their performance in clustering.

In the experiments, the parameter $m$ has the same meaning as the fuzzification constant in the fuzzy clustering algorithms, and is usually selected as 2. We assume that the number of cluster $C$ is known in advance. We selected $\varepsilon=10-5$, $\delta=1.4$, and $\varpi_1=0.8$, $\varpi_2=0.1$ and $\varpi_3=0.1$ for the experiments in Section 4.1, which are tuned in our experiments. We will discuss the influence of these parameters in Section 4.4.

In the experiments, no ambiguity has arisen in the experiments in computing $\bar{c}_{i\ max}$, i.e., for each i, the second largest value among $T_{i1},\ldots,T_{iC}$ is always different from the third largest value. When the largest and the second largest values are same in some cases, to make sure that $\bar{c}_{i\ max}$ not change too much, the second largest must be selected differently from the third largest one.

### 4.1. Experiment on the diamond data

We illustrate the performance of NCM using a first example inspired from a classical diamond data set [29]. It is composed of 12 objects, which are represented in Fig. 1. Objects 1–11 are normal data, whereas object 12 is an outlier. A 2-class partition was imposed so that three membership set elements ($T$, $I$ and $F$) have been considered in the optimization process: $c_1$, $c_2$, indeterminacy set and the outlier set. The $T$, $I$ and $F$ are represented in Fig. 2(d) which $T$, $I$ and $F$ are plotted against the point number.

It can be seen that the two natural clusters are correctly clustered for points 1–11 where the high true set ($T$) values obtained. Point 6 is assigned to indeterminacy set, which reveals that this point is ambiguous, and it could be assigned either to $c_1$ or $c_2$. Point 12, which can be considered as an outlier, is logically assigned to the falsity set.

Table 2 shows the $T$, $I$ and $F$ membership values and the last column shows the neutrosophic cluster assignments for the diamond data set points. Note that both classes are completely indeterminate for data point 6.

We compared the results obtained with the FCM, PCM and FPCM algorithms. The optimal parameters were used in

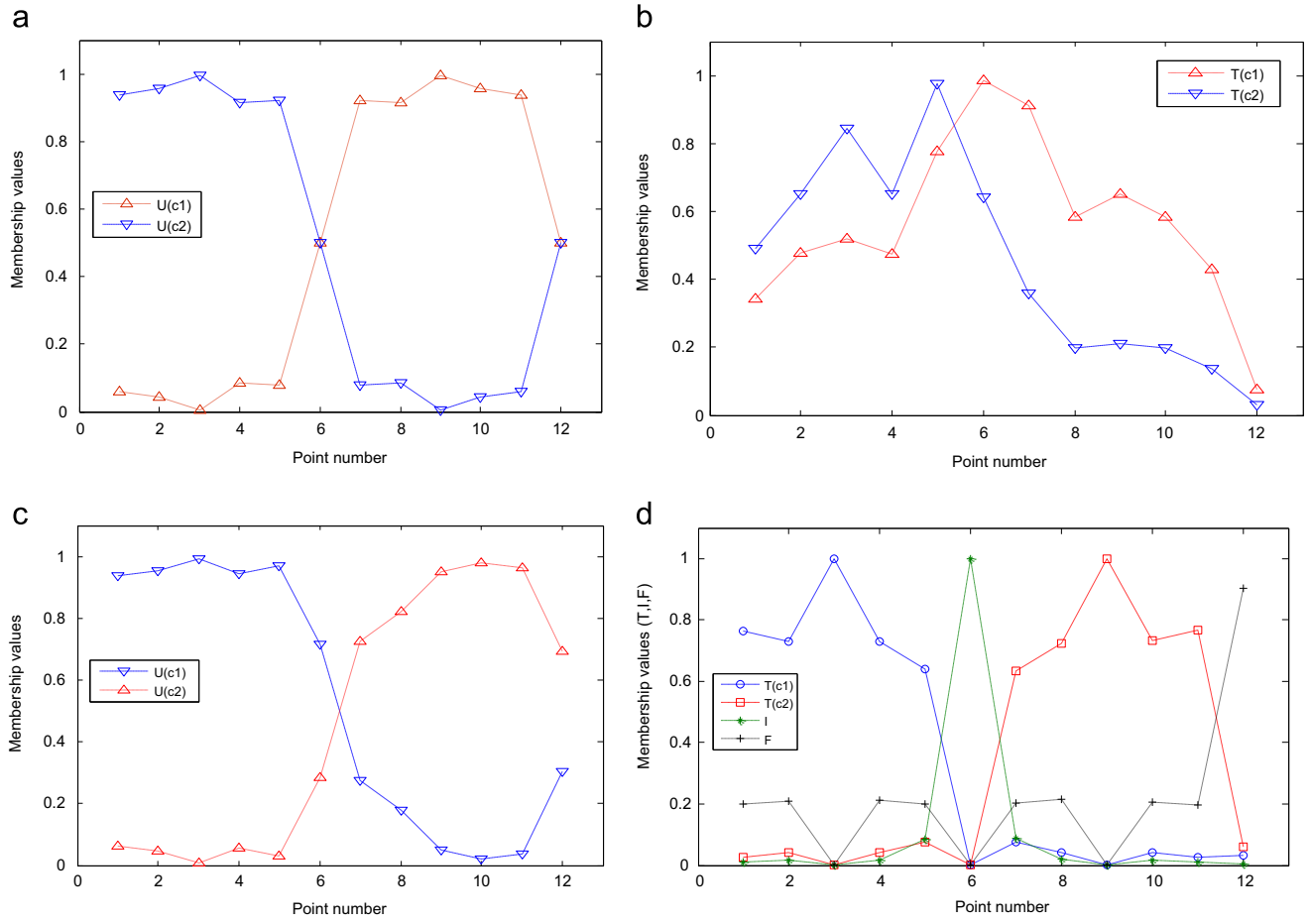


**Fig. 2.** Clustering results on the diamond data set with two classes: (a) result of FCM; (b) result of PCM; (c) result of FPCM and (d) result of NCM.

experimental comparison for FCM, PCM and FPCM. These results are given in Fig. 2(a), (b) and (c) respectively, which shows the membership values obtained by the algorithms. Every algorithm finds a reasonable partition of the data, but the ambiguous and the outlier data points are assigned to the clusters. On the other hand, NCM is the only algorithm able to detect atypical data point like object 6 and object 12.

We further illustrate the performance of NCM using a more complicated diamond data set, where three diamond class points are used, and one diamond class is between the other diamond two diamond clusters. In other words, the new diamond cluster's center is located on the mean location of the rest diamond clusters. It is composed of 18 objects, shown in Fig. 3. So there are two ambiguous data points 6 and 12, and an outlier data point 18. Clustering results are given in Fig. 4. Again, the proposed NCM algorithm correctly grouped the three natural clusters.

Points 6 and 12 are assigned to indeterminacy set, which reveals that these points are ambiguous and finally the point 18, which can be considered as an outlier, is logically assigned to the falsity set. We again compared the results obtained with the FCM, PCM and FPCM algorithms. The clustering capabilities of the compared algorithms are given in Fig. 4(a), (b) and (c) respectively, which shows the membership values by different algorithms. One more time every algorithm finds a reasonable partition of the data, but the ambiguous and the outlier data points are assigned to the clusters by all compared methods. NCM is the only algorithm that can detect atypical data point like object 6, 12 and object 18.

As it is indicated in Ref. [16], typicality is an important means for alleviating the undesirable effects of outliers. For a fairly comparison with the FPCM method, we also considered the typicality in the FPCM [16]. We carried out an experiment on X12 database that was used in [16] for evaluating FPCM performance based on the typicality. The data points of the X12 database are given in Fig. 5.

**Table 2**
Various partitions obtained using NCM on the diamond data set with two classes.

| Point | $T_{c1}$ | $T_{c2}$ | $I$ | $F$ | arg max $\{T_{c1}, T_{c2}, I, F\}$ | Neutrosophic partition |
|---|---|---|---|---|---|---|
| 1 | 0.8262 | 0.0294 | 0.0104 | 0.1339 | $T_{c1}$ | $c_1$ |
| 2 | 0.7952 | 0.0451 | 0.0196 | 0.1401 | $T_{c1}$ | $c_1$ |
| 3 | 0.9996 | 0.0001 | 0.0000 | 0.0003 | $T_{c1}$ | $c_1$ |
| 4 | 0.7920 | 0.0456 | 0.0197 | 0.1426 | $T_{c1}$ | $c_1$ |
| 5 | 0.6950 | 0.0799 | 0.0915 | 0.1336 | $T_{c1}$ | $c_1$ |
| 6 | 0.0007 | 0.0007 | 0.9982 | 0.0005 | $I$ | Ambiguous |
| 7 | 0.0835 | 0.6802 | 0.0990 | 0.1373 | $T_{c2}$ | $c_2$ |
| 8 | 0.0475 | 0.7854 | 0.0207 | 0.1464 | $T_{c2}$ | $c_2$ |
| 9 | 0.0003 | 0.9987 | 0.0001 | 0.0008 | $T_{c2}$ | $c_2$ |
| 10 | 0.0444 | 0.7994 | 0.0195 | 0.1367 | $T_{c2}$ | $c_2$ |
| 11 | 0.0284 | 0.8334 | 0.0101 | 0.1280 | $T_{c2}$ | $c_2$ |
| 12 | 0.0477 | 0.0938 | 0.0084 | 0.8502 | $F$ | Outlier |

In the comparison, the FPCM model was run using the parameters in [40] and the obtained typicality degrees ($T_1^T$ and $T_2^T$) are tabulated in Table 3. The NCM results ($T_{c1}$, $T_{c2}$ and $I$) are listed in first four columns of Table 3. In the NCM results, it is obvious that the data points 1, 2, 3, 4 and 5 were assigned to $c_1$ cluster, data point 6 is ambiguous, data points 7, 8, 9, 10 and 11 were assigned $c_2$ cluster and finally data point 12 was an outlier. In the FPCM results, according to the typicality degrees, it is obvious that points 1–5 belong to the left cluster and 7–11 in the right cluster.

However, it is difficult to assign the data point 6 and 12 to proper classes because of $T_1^T = T_2^T$.

We also conducted another experiment to compare NCM and FPCM using a three class dataset, which is illustrated in Fig. 6. In Fig. 6, the data points 6 and 12 are ambiguous and data points 18 and 19 are outliers. The results of FPCM and NCM are tabulated in Table 4.

In Table 4, the belonging of each data point can be easily made according the results of the NCM. The first five data points belong to the first class because of their higher $Tc_1$ values. Similar observation can be inferred for the other classes ($C_2$ and $C_3$). Data points 6 and 12 were ambiguous because of the high $I$ values. Finally, last two data points (18 and 19) were deduced as outlier.

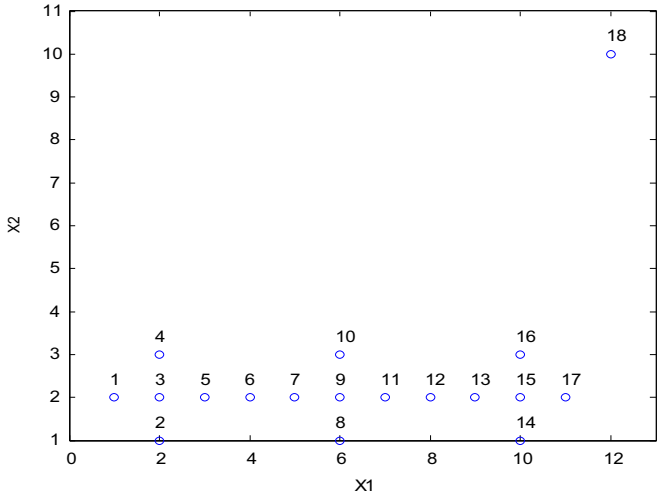


**Fig. 3.** Diamond dataset with three classes.

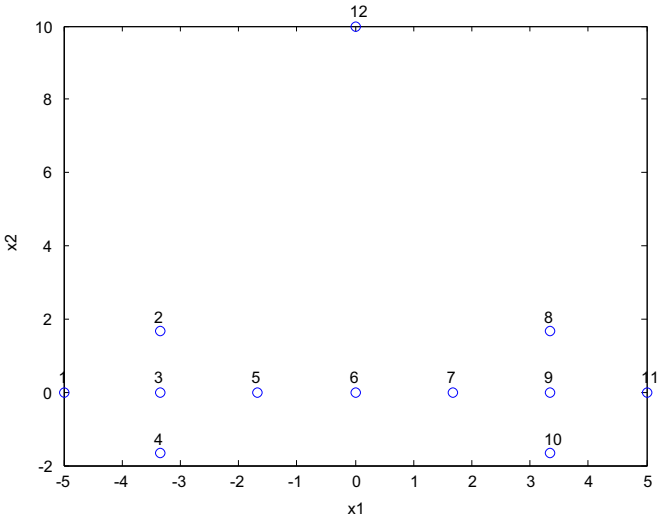


**Fig. 5.** X12 dataset.

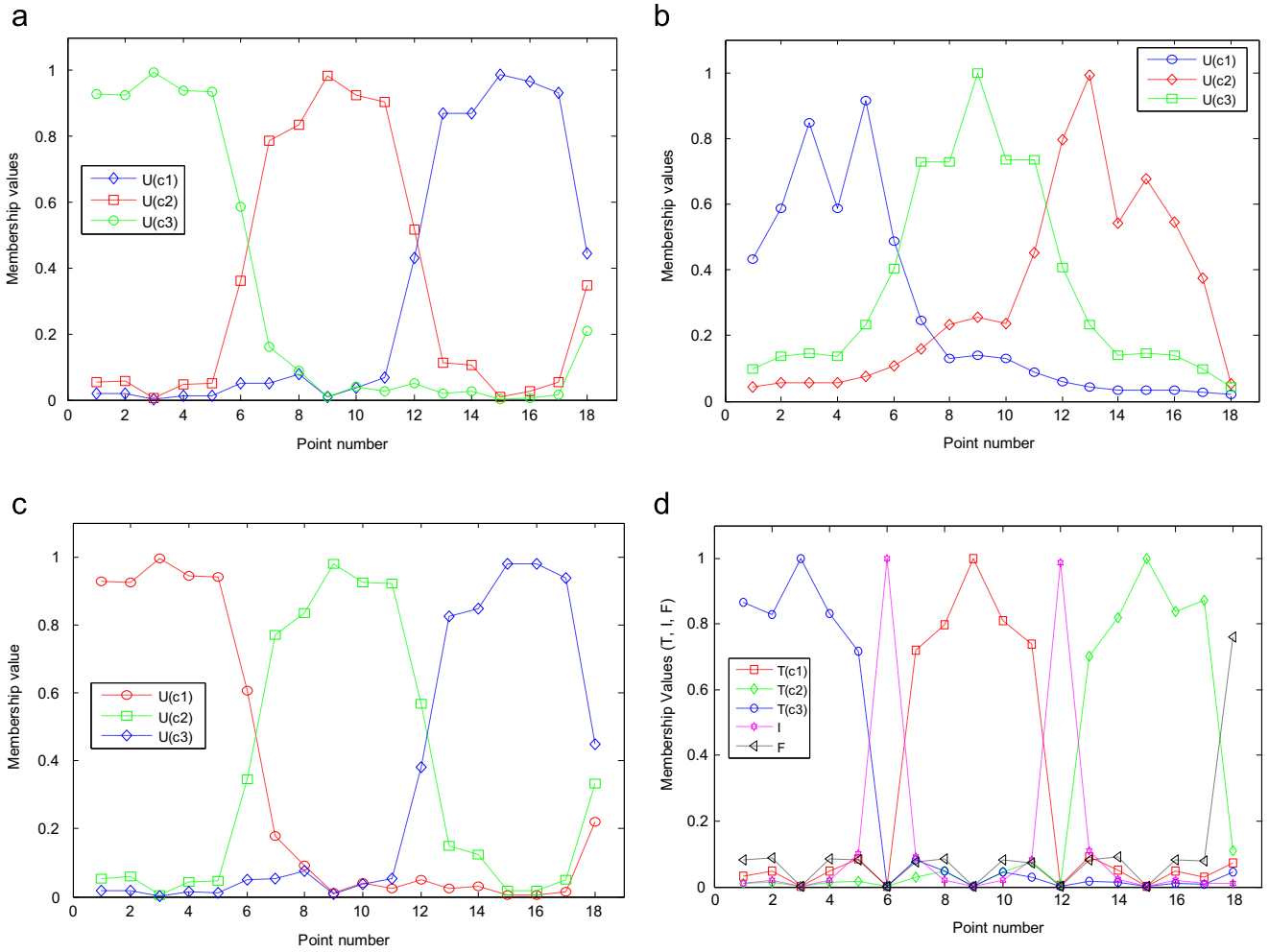


**Fig. 4.** Clustering results on the diamond data set with three classes: (a) result of FCM; (b) result of PCM; (c) result of FPCM and (d) result of NCM.

However, when the belonging is determined by the typicality degrees in the FPCM, there is not an obvious judgment to determine the true clusters, ambiguous data points and the outliers. The experiment with the three class diamond dataset shows the clustering ability of NCM clearly against the FPCM.

**Table 3**
*T*, *I* and *F* values of NCM and typicality values for FPCM for two classes diamond dataset.

| Point | NCM | | | | FPCM | |
|---|---|---|---|---|---|---|
| | $T_{c1}$ | $T_{c2}$ | $I$ | $F$ | $T_1^T$ | $T_2^T$ |
| 1 | 0.8256 | 0.0298 | 0.0105 | 0.1341 | 0.0515 | 0.0034 |
| 2 | 0.8022 | 0.0439 | 0.0192 | 0.1348 | 0.1485 | 0.0052 |
| 3 | 0.9994 | 0.0001 | 0.0001 | 0.0004 | 0.5957 | 0.0053 |
| 4 | 0.7838 | 0.0478 | 0.0205 | 0.1478 | 0.0448 | 0.0048 |
| 5 | 0.6926 | 0.0810 | 0.0934 | 0.1330 | 0.1055 | 0.0098 |
| 6 | 0.0016 | 0.0016 | 0.9956 | 0.0012 | 0.0233 | 0.0233 |
| 7 | 0.0810 | 0.6926 | 0.0934 | 0.1330 | 0.0098 | 0.1055 |
| 8 | 0.0439 | 0.8022 | 0.0192 | 0.1348 | 0.0052 | 0.1485 |
| 9 | 0.0001 | 0.9994 | 0.0001 | 0.0004 | 0.0053 | 0.5957 |
| 10 | 0.0478 | 0.7838 | 0.0205 | 0.1478 | 0.0048 | 0.0448 |
| 11 | 0.0298 | 0.8256 | 0.0105 | 0.1341 | 0.0034 | 0.0515 |
| 12 | 0.1088 | 0.1088 | 0.0152 | 0.7673 | 0.0022 | 0.0022 |

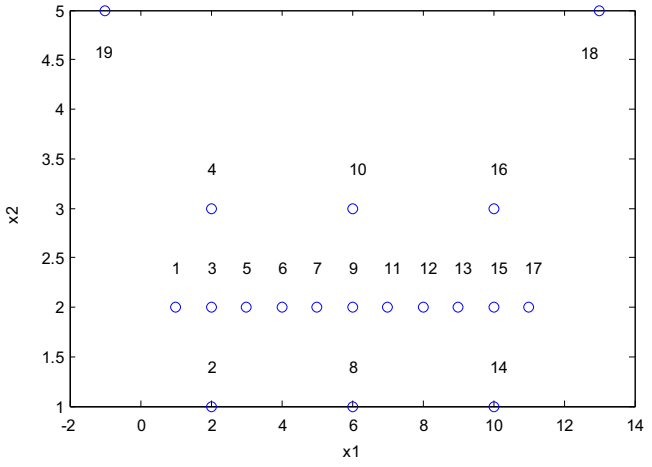


**Fig. 6.** Three classes diamond dataset.

**Table 4**
*T*, *I* and *F* values for NCM and typicality values for FPCM for three class diamond dataset.

| Point | NCM | | | | | FPCM | | |
|---|---|---|---|---|---|---|---|---|
| | $T_{c1}$ | $T_{c2}$ | $T_{c3}$ | $I$ | $F$ | $T_1^T$ | $T_2^T$ | $T_3^T$ |
| 1 | 0.8900 | 0.0236 | 0.0070 | 0.0150 | 0.0645 | 0.1463 | 0.0011 | 0.0000 |
| 2 | 0.7759 | 0.0578 | 0.0144 | 0.0444 | 0.1076 | 0.0506 | 0.0014 | 0.0001 |
| 3 | 0.9880 | 0.0030 | 0.0007 | 0.0029 | 0.0053 | 0.4965 | 0.0014 | 0.0001 |
| 4 | 0.8393 | 0.0411 | 0.0103 | 0.0332 | 0.0762 | 0.1895 | 0.0014 | 0.0001 |
| 5 | 0.5816 | 0.0928 | 0.0161 | 0.2182 | 0.0913 | 0.0550 | 0.0018 | 0.0001 |
| 6 | 0.0124 | 0.0149 | 0.0016 | 0.9646 | 0.0065 | 0.0181 | 0.0024 | 0.0003 |
| 7 | 0.0689 | 0.7032 | 0.0261 | 0.1249 | 0.0769 | 0.0088 | 0.0034 | 0.0011 |
| 8 | 0.0434 | 0.7920 | 0.0434 | 0.0317 | 0.0894 | 0.0048 | 0.0048 | 0.0010 |
| 9 | 0.0000 | 0.9999 | 0.0000 | 0.0000 | 0.0000 | 0.0052 | 0.0052 | 0.9943 |
| 10 | 0.0421 | 0.7996 | 0.0421 | 0.0315 | 0.0847 | 0.0051 | 0.0051 | 0.0012 |
| 11 | 0.0261 | 0.7032 | 0.0689 | 0.1249 | 0.0769 | 0.0034 | 0.0088 | 0.0011 |
| 12 | 0.0016 | 0.0149 | 0.0124 | 0.9646 | 0.0065 | 0.0024 | 0.0181 | 0.0003 |
| 13 | 0.0161 | 0.0928 | 0.5816 | 0.2182 | 0.0913 | 0.0018 | 0.0550 | 0.0001 |
| 14 | 0.0144 | 0.0578 | 0.7759 | 0.0444 | 0.1076 | 0.0014 | 0.0506 | 0.0001 |
| 15 | 0.0007 | 0.0030 | 0.9880 | 0.0029 | 0.0053 | 0.0014 | 0.4965 | 0.0001 |
| 16 | 0.0103 | 0.0411 | 0.8393 | 0.0332 | 0.0762 | 0.0014 | 0.1895 | 0.0001 |
| 17 | 0.0070 | 0.0236 | 0.8900 | 0.0150 | 0.0645 | 0.0011 | 0.1463 | 0.0000 |
| 18 | 0.0370 | 0.0854 | 0.3046 | 0.0324 | 0.5406 | 0.0007 | 0.0066 | 0.0000 |
| 19 | 0.3046 | 0.0854 | 0.0370 | 0.0324 | 0.5406 | 0.0066 | 0.0007 | 0.0000 |

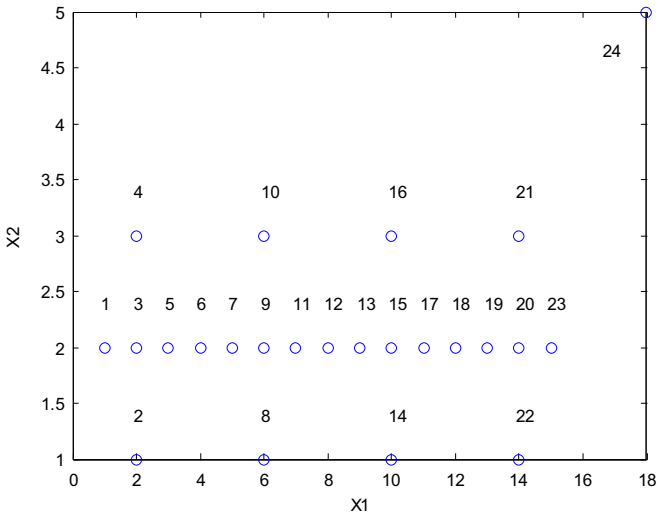


**Fig. 7.** Diamond dataset with four classes.

Finally, we conducted an experiment with a four clusters diamond data set to justify the assumption about the simplification from Eqs. (4) and (5). This dataset, shown in Fig. 7, is composed of 24 data points with three ambiguous data points: 6, 12 and 18, and an outlier data point 24. There are four diamond clusters ($C=4$), whose centers are located on the data point: 3, 9, 15 and 20. We have $\binom{C}{2}=6$ combinations to compute the centers of different two clusters ($\bar{c}_{2j}$), $\binom{C}{3}=4$ combinations to compute the centers of different three clusters ($\bar{c}_{3j}$), and $\binom{C}{4}=1$ combination for all four clusters. In order to reduce the computation, we simplify the Eq. (4) into Eq. (5) considering only $C$-1 pairs, which is based on the fact that the indeterminacy greatly depends on the closest determinate clusters with high truth values. We conducted a new experiment to justify this assumption, and the results are listed in Fig. 8 and Table 5.

In this experiment, NCM was run with the same parameters using objective functions of Eqs. (4) and (5). The clustering results are shown in Fig. 8, and the cluster centers for both objective functions are listed in Table 5.

The membership values for each data point are given in Fig. 8 for both equations. Fig. 8(a) shows *T* membership values of Eq. (4), Fig. 8(b) shows the *I* (red line) and *F* (black line) membership values of Eq. (4), and Fig. 8(c) shows all *T*, *I* and *F* membership values of Eq. (5). As we are using the maximum value of [*T*, *I*, *F*] for assigning the cluster label to the data point, from Fig. 8(c), we can see that the simplified objective function achieve the correct clustering result. Four classes are grouped correctly, and the ambiguous data points and the outlier data point are correctly determined. The clustering results were obtained according to the membership values in Fig. 8 (a) and (b). The results show that the points on the centers of (2, 2), (6, 2) and (14, 2) are clustered correctly. There are 2 data points (13 and 15) wrongly clustered in the diamond cluster that is located on (10, 2). Moreover, the indeterminate data point 18 is assigned to the diamond cluster that is located on (10, 2). So, totally 3 data points have false labels according to this clustering. All *I* and *F* memberships are shown in Fig. 8(b). $I(1)$ shows the membership of $\bar{c}_{4j}$, $I(2)$–$I(7)$ shows the membership of different $\bar{c}_{2j}$, and $I(8)$–$I(11)$ shows the membership of different $\bar{c}_{3j}$. Moreover, for the clustering result centers in

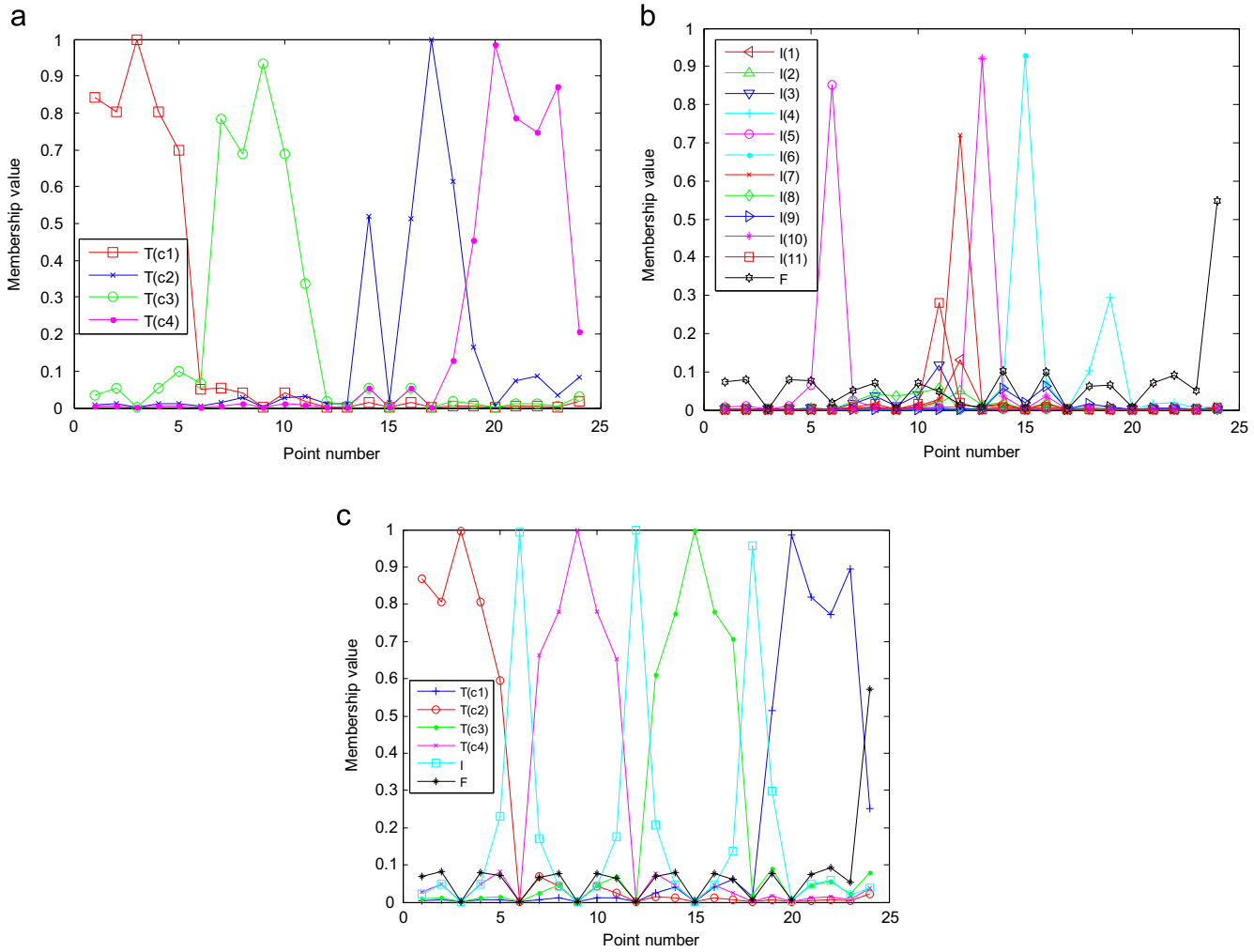


**Fig. 8.** Clustering results on the diamond data set with four classes: (a) and (b) result of NCM using Eq. (4) and (c) result of NCM using Eq. (5). (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

**Table 5**
Center of clusters for diamond dataset with four classes.

| Clustering centers using Eq. (4) | | Clustering centers using Eq. (5) | |
|---|---|---|---|
| $x_1$ | $x_2$ | $x_1$ | $x_2$ |
| 1.9489 | 2.0005 | 1.8975 | 2.0005 |
| 5.8058 | 2.0008 | 6.0035 | 2.0018 |
| 11.0010 | 2.0087 | 10.0611 | 2.0083 |
| 14.2491 | 2.0838 | 14.2248 | 2.0788 |

Table 5, both objective functions produced almost the same values, and few deviations can be seen in the obtained cluster centers. This experiment clearly shows that the simplified objective function given in Eq. (5) can achieve the same cluster ability as the objective function in Eq. (4) while the cluster performance is not affected.

We also investigate the effect of simplification regards to total number of clusters around the boundary region. For this purpose, we construct a database where 3 clusters are overlapped. There are 300 data points (each cluster have equal number of data) and their centroids are located on the (10, 10), (15, 15) and (20, 10). Fig. 9 (a) shows the data points and the corresponding cluster centers. Fig. 9(b) shows the obtained clustering results with the following parameters: $\varepsilon=10-5$, $\delta=15$, and $\varpi_1=0.7$ , $\varpi_2=0.2$ and $\varpi_3=0.1$. Cyan, green and blue colors show the clustered data points and the black color show indeterminate data points. Visually, the indeterminate data points are all in the overlapped region. The proposed simplification produces reasonable clustering results. The center difference error between the ground truth and the clustered centers is calculated based on the Euclidean distance. The total error is 2.18.

### 4.2. Experiment on IRIS data

We have also tested our method using the famous IRIS dataset [39], which has extensively been used in clustering algorithms' testing. The IRIS dataset contains three classes, i.e., three varieties of Iris flowers, namely, Iris Setosa, Iris Versicolor and Iris Virginica consisting of 50 samples each. Each sample has four features, namely, sepal length (SL), sepal width (SW), petal length (PL) and petal width (PW). One of the three clusters is clearly separated from the other two, while these two classes admit some overlap. The following parameters $\varepsilon=10-3$, $\delta=50$, and $\varpi_1=0.7$, $\varpi_2=0.15$ and $\varpi_3=0.15$ are chosen for IRIS dataset. The results of NCM are given in Table 6. Iris Setosa is clustered with a 100% correct clustering rate, and in other clusters (Versicolor and Virginica), there are 10 misclassified data points. 6 data points determined as indeterminate. The data points in the indeterminate set are the 78th, 124th, 127th, 128th, 134th and 147th points, which are in the overlap region between the Iris Versicolor and Iris Virginica classes.

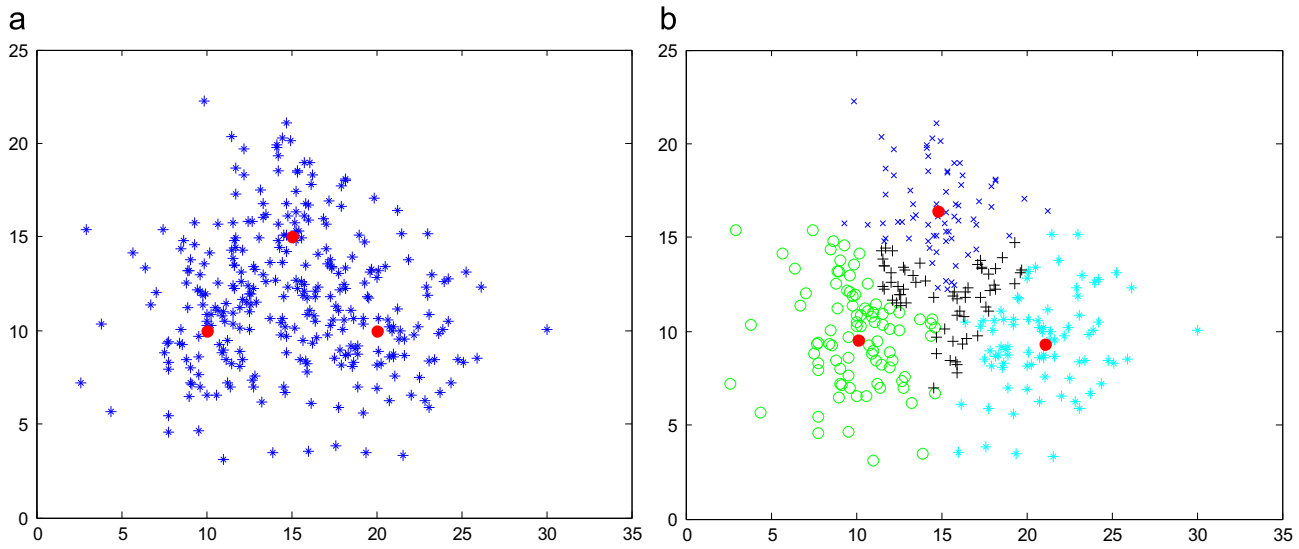


**Fig. 9.** NCM performance on three clusters overlap: (a) original data and (b) NCM clustering result. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

**Table 6**
NCM clustering results of the IRIS dataset.

| Clustered | Actual | | |
|---|---|---|---|
| | Setosa | Versicolor | Virginica |
| Setosa | 50 | 0 | 0 |
| Versicolor | 0 | 47 | 8 |
| Virginica | 0 | 2 | 37 |

**Table 7**
Resubstitution errors on the IRIS data using FPcM and FcM.

| $E(H_{MT}(T))$ FPCM | $E(H_{MM}(U))$ FPCM | $E(H_{MM}(U))$ FCM | $E(H_{MM}(T))$ PCM | $E(H_{MM}(T))$ NCM |
|---|---|---|---|---|
| 14 | 13 | 16 | 50 | 10 |

**Table 8**
$3 \times 3$ Spatial neighborhoods.

| | | |
|---|---|---|
| $T(i-1,j-1)$ | $T(i-1,j)$ | $T(i-1,j+1)$ |
| $I(i-1,j-1)$ | $I(i-1,j)$ | $I(i-1,j+1)$ |
| $F(i-1,j-1)$ | $F(i-1,j)$ | $F(i-1,j+1)$ |
| $T(i,j-1)$ | $T(i,1)$ | $T(i,j+1)$ |
| $I(i,j-1)$ | $I(i,1)$ | $I(i,j+1)$ |
| $F(i,j-1)$ | $F(i,1)$ | $F(i,j+1)$ |
| $T(i+1,j-1)$ | $T(i+1,j)$ | $T(i+1,j+1)$ |
| $I(i+1,j-1)$ | $I(i+1,j)$ | $I(i+1,j+1)$ |
| $F(i+1,j-1)$ | $F(i+1,j)$ | $F(i+1,j+1)$ |

The resubstitution error is an important metric to measure the clustering results [24,38]. This error notated as$E(H(U))$ hardened label matrices, where $U$ is the membership values, and $H$ shows the hardening of the matrix argument by either maximum memberships (MM) or maximum typicalities (MT).

We evaluated the clustering results using the resubstitution error. Table 7 shows the evaluation results of FPCM, FCM and PCM [9,38] on IRIS dataset. The fuzziness parameter $m$ and the $\eta$ constant are set to 2 for the FPCM, FCM and PCM algorithms. As shown in Table 7, NCM yielded the minimum resubstitution error compared with other clustering methods, and achieved better clustering performance on IRIS dataset. The improvement of classification ability in NCM is that it uses I and F subsets to describe the indeterminacy and the outliers in clustering. In IRIS data, the third cluster has four outliers [40]. NCM is quite effective to deal with indeterminacy and the outliers in clustering. The experimental results also show that the NCM found several outliers in the Iris dataset and yielded less error.

### 4.3. Application to image segmentation

In image segmentation process, each pixel's intensity can be employed as the input for clustering algorithm, and the clustering result can be labeled as the segmentation result. In this situation, image segmentation problem is transformed into data clustering problem, which can be solved using NCM. In this section, we applied NCM into image segmentation, and compared with the existing image segmentation algorithm. A number of images with difference noise levels were employed to test the NCM's ability to cluster the fuzzy and indistinct data.

In NCM, because the membership functions ($T$, $I$ and $F$) do not contain any spatial information, it is not proper to be directly applied into the image segmentation applications [6]. In addition, the correct determination of the pixel labels derived from membership functions might yield segmentation errors. To reduce the influence of undesired factors on the final determination of membership functions, the spatial neighborhood is taken into account.

Let $p(i,j)$ be the pixel at the spatial position $(i,j)$ in the image, and $S$ be its neighborhood pixels. $g(i,j)$ is the gray level of $p(i,j)$. For $p(i,j)$, its membership degree is associated to the membership degrees of all its neighboring pixels in $S$ such that the resulting membership function contains spatial information. Table 8 shows a nine-neighborhood. The membership functions, designated by $T(i,j)$, $I(i,j)$ and $F(i,j)$ of the central pixel $p(i,j)$, is obtained by combining the membership degrees of all the pixels belonging to this neighborhood.

In this application, the average of its neighboring pixels is used as the central pixel's final membership value, and all pixels are assigned into different determinate clusters with the value of T membership. In the simple hypothesis cases, the calculation of the averaged membership function of the gray level at the current

position $(i,j)$ is done according to the following formula [41]:

$$\begin{aligned}\overline{T}(i,j) &= \tfrac{1}{z^2}\sum_{m,n\in S} T(m,n)\\ \overline{I}(i,j) &= \tfrac{1}{z^2}\sum_{m,n\in S} I(m,n)\\ \overline{F}(i,j) &= \tfrac{1}{z^2}\sum_{m,n\in S} F(m,n)\end{aligned} \tag{25}$$

where $z$ is the size of $S$, and it is equal to 3 in the present case.

After associating with the spatial information, the NCM algorithm was tested on two simulated images (Fig. 10(a) and (b)) to evaluate the proposed approach. Fig. 10(1) shows a synthesized image with four classes, and the corresponding gray values are 50 (upper left, UL), 100 (upper right, UR), 150 (low left, LL) and 200 (low right, LR) respectively. Each cluster (sub-image) contains $64\times 64$ pixels. The image is degraded by the Gaussian noise ($\mu=0$, $\sigma=25.5$). Fig. 10(2) shows another synthesized image that contains three regions: two equal sized rectangular on a uniform background and the corresponding gray values 20 (upper step), 100 (lower step) and 255 (background). Gaussian noise with 0 mean and 25.5 variance is also added to the synthesized image.

The segmentation results are presented in column (c) of Fig. 10. The NCM algorithm archived good homogeneity in the segmented regions. The boundaries are smooth and only several misclassified pixels can be seen in the boundary of the homogenous regions. While 53 misclassified pixels were observed for Fig. 10(1), 15 misclassified pixels were observed for Fig. 10(2).

This experiment also compared the NCM algorithm with the FCM–AWA algorithm [11] on image segmentation. FCM–AWA algorithm performs quite well in segmenting the images that are degraded by noise. The FCM–AWA was realized by modifying the objective function in the conventional FCM algorithm, and incorporating the spatial neighborhood information. An adaptive weighted averaging (AWA) filter was given to indicate the spatial influence of the neighboring pixels on the central pixel. The parameters of control template are automatically determined in the implementation of the weighted averaging image by a predefined nonlinear function.

The FCM–AWA was tested on same images with the default settings: $m=2$, $\alpha=50$, $\varepsilon=10^{-5}$, $r=2$, $k_0=0.45$ and $k_1=0.65$. The results can be seen in column (b) of Fig. 10. FCM–AWA also produced good homogeneity in the segmented regions but there are a lot of misclassified pixels observed in the boundary transition regions. While 418 misclassified pixels are produced by the FCM–AWA for Fig. 10(1), 235 misclassified pixels are countered for Fig. 10(2). Table 9 is also given to present the misclassified pixels for both NCM and FCM–AWA methods. The results indicate the superiority of the proposed NCM approach.

We also compare the NCM method with FCM–AWA on real images. Fig. 11 shows four images: “rice”, “eight”, “hand” and “woman” with various sizes. In order to evaluate the methods' performance on noisy image, the images were degraded by the Gaussian noise ($\mu=0$, $\sigma=2.25$) as shown in Fig. 11(a). Fig. 11(b) and (c) is the segmentation results of the degraded images by FCM–AWA and NCM algorithms respectively. FCM–AWA algorithm is run with parameters $m=2$, $\alpha=50$, $\varepsilon=10^{-3}$, $r=1$, $k_0=0.45$ and $k_1=0.65$. Visually, Fig. 11(b) and (c) indicates that NCM performs more efficient than FCM–AWA in removing of Gaussian noise. NCM produced more homogenous segmented regions than FCM–AWA does. Moreover, Fig. 11(c) shows that the NCM performs more efficient in preserving the edge information in the image than FCM–AWA does, as shown in Fig. 11(b).

With the aim of achieving robust and accurate segmentation in case of mixed noise, [40] incorporated spatial information with the clustering algorithm, which is called adaptive spatial information theoretic clustering algorithm (ASIC). ASIC's objective function has a new dissimilarity measure, and the weighting factor for neighborhood effect is fully adaptive to the image content. It enhances

**Table 9**
Number of misclassified pixels.

| | NCM | FCM–AWA |
|---|---|---|
| Fig. 10(1) | 70 | 418 |
| Fig. 10(2) | 15 | 235 |

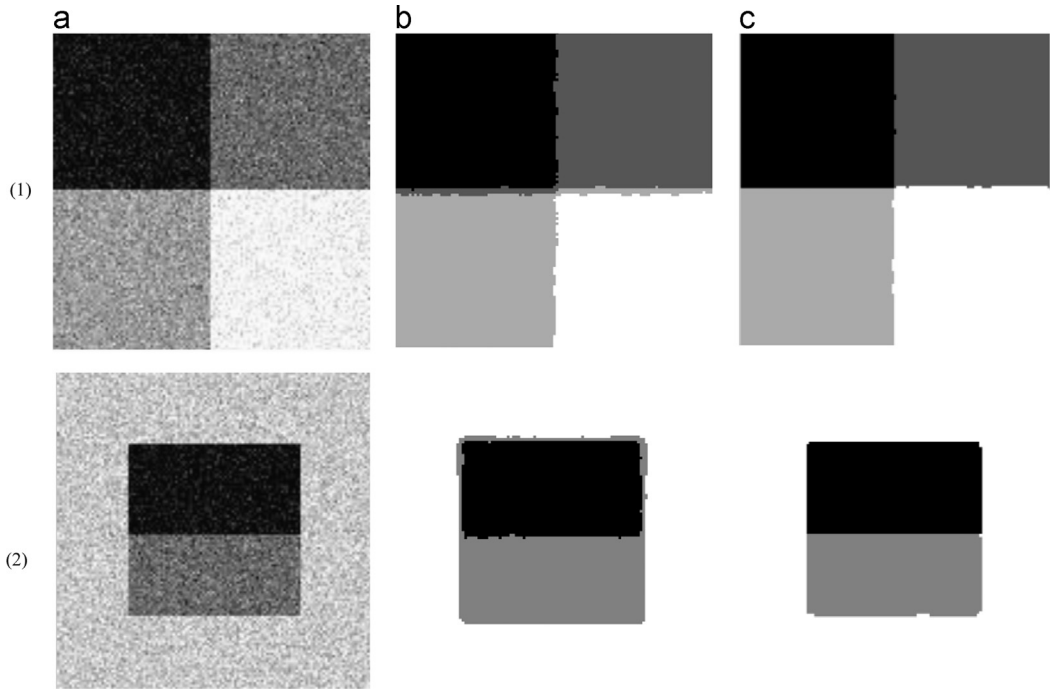


**Fig. 10.** Comparison of NCM and FCM–AWA segmentation results on synthetic images degraded by Gaussian noise: (a) original images; (b) FCM–AWA segmentation results and (c) NCM segmentation results.

the smoothness towards piecewise-homogeneous segmentation and reduces the edge blurring effect. Furthermore, a unique characteristic of the segmentation algorithm is that it has the capabilities to eliminate outliers at different stages of the ASIC algorithm.

We also compare the performance of our algorithm through the same simulated images and real images with the ASIC algorithm [40].

**Table 10**
Number of misclassified pixels.

| | NCM | ASIC |
|---|---|---|
| Fig. 12(1) | 70 | 144 |
| Fig. 12(2) | 15 | 47 |

In the experiments, the cooling factor $\alpha$ is selected as 0.95 [42] for the ASIC algorithm. We use the same synthetic test images that are degraded by the Gaussian noise ($\mu=0$, $\sigma=2.25$) as shown in Fig. 12(a). The results can be seen in Fig. 12(b). ASIC produced good homogeneity in the segmented regions but there are a lot of misclassified pixels observed in the boundary regions. While 144 misclassified pixels are produced by the ASIC for Fig. 12(1), 47 misclassified pixels are counted for Fig. 12(2). Table 10 presents the misclassified pixels for both NCM and ASIC methods. The results indicate the better performance of the proposed NCM approach on synthetic images.

We further compared the performance of our algorithm with ASIC on the real images that are degraded by Gaussian noise. The contaminated images are shown in Fig. 13(a). From Fig. 13(b) and (c), we can see that our proposal gives a much better segmentation

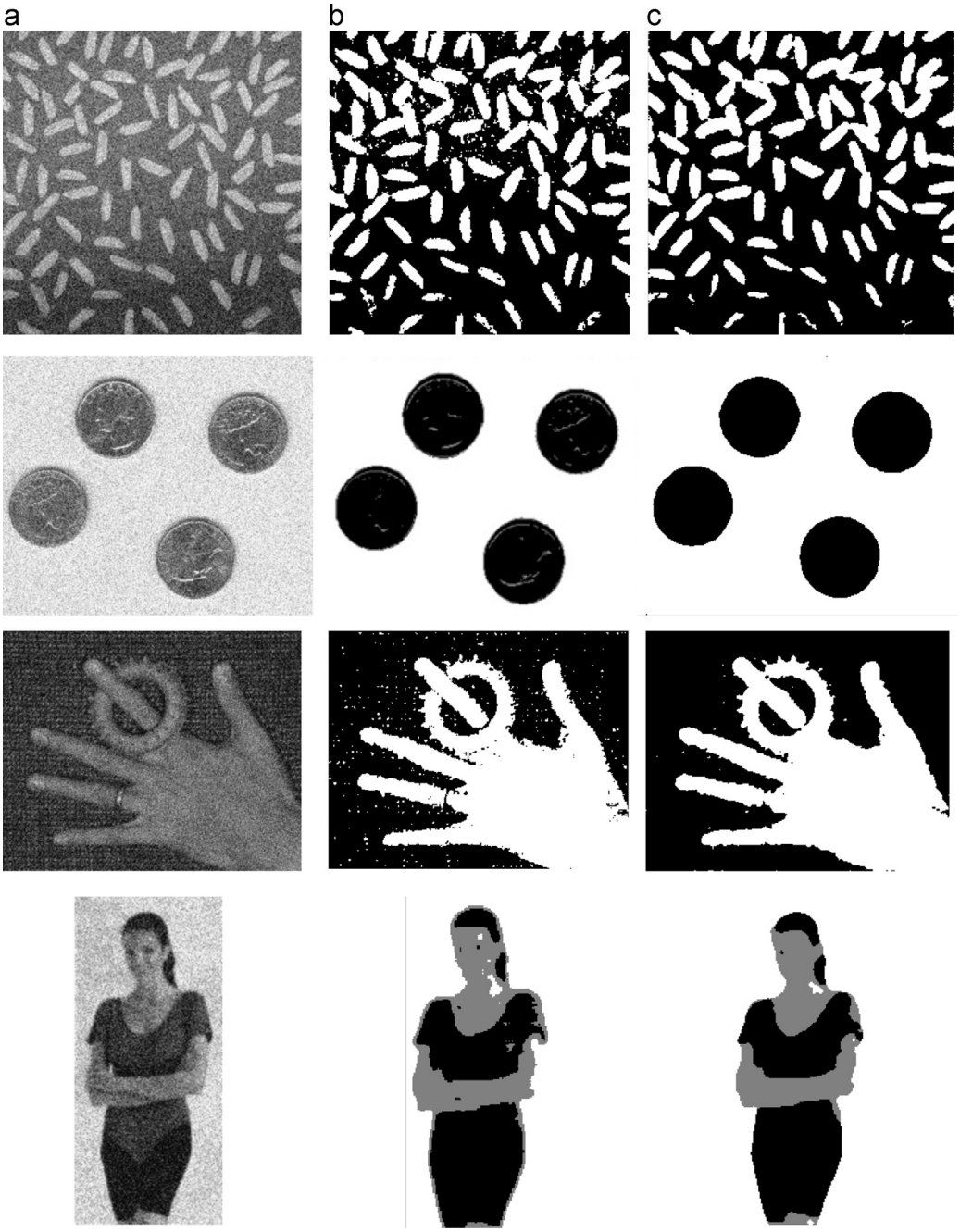


**Fig. 11.** Comparison of segmentation results on “rice”, “eight”, “hand” and “woman” images degraded by Gaussian noise: (a) the images degraded by Gaussian noise; (b) using FCM–AWA and (c) using NCM.

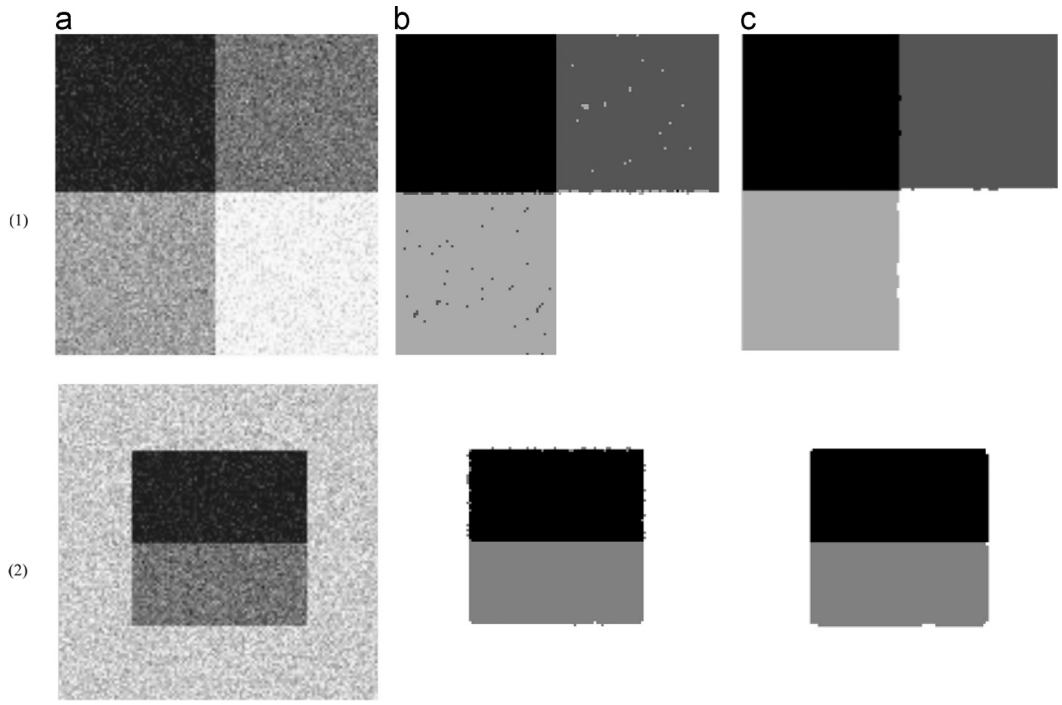


**Fig. 12.** Comparison of NCM and ASIC segmentation results on synthetic images degraded by Gaussian noise: (a) original images; (b) ASIC segmentation results and (c) NCM segmentation results.

than the ASIC. ASIC did not segment the background exactly for "rice" and "hand" images, many background pixels classified as foreground wrongly. For "eight" image, the segmented coin patterns are not homogenous. There are also many misclassified regions in the "woman" image. On the other hand, for the results of NCM, the background and the foreground objects are segmented well. The segmented regions are homogenous and the object boundaries are smooth. The superiority of the NCM algorithm is apparent in the comparison.

For further evaluation of the NCM's performance in image segmentation quantitatively, a measurement, namely *F*-measure [40] is used to evaluate the performance of different methods. The *F*-measure is a measure to evaluate the segmentation accuracy. It considers both the precision *P* and the recall *R* of the segmentation result, and is defined as follows [43]:

$$F=\frac{P\cdot R}{\xi\cdot P+(1-\xi)\cdot R} \tag{26}$$

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{27}$$

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{28}$$

where $\xi$ is a constant number and selected as 0.5 in [40]. *P* is precision, and *R* is recall rate. TP is the number of correct results, FP is the number of false segmented pixels, and FN is the number of the missed pixels in the result. The *F*-measure value is in the range of [0, 1], and a larger *F*-Measure value indicates a higher segmentation accuracy.

In Table 11, we tabulate the *F*-measure values for each method on the images in Fig. 13. The NCM produces higher *F*-measure values than the other methods. The *F*-measure values demonstrate the better performance of NCM method than the FCM–AWA and ASIC methods. The experimental results show the NCM yields more reasonable segmentations than the compared methods on both the visual and quantitative results.

It is also worth to mentioning that the NCM method produces smoother segmentation than the compared methods. It is because the mean value of *T*, *F* and *I* are calculated within a $3\times 3$ local neighborhood in Eq. (25). This process can be considered as a kind of mean filtering, which yields smoother results. Although, the mean filtering process may cause to lose some details in the final segmentation results, the qualitative and quantitative evaluation results indicate that the NCM's segmentation results are still more reasonable than other methods.

In summary, from the comparisons with the existing image segmentation algorithm using clustering method, we can see that the segmentation method based on NCM yielded better segmentation results both on the synthesized noisy images and the real images with the noise, and NCM demonstrates better ability to cluster the obscure data. We also found the NCM improved the classification and clustering ability in the real data application. The reason of improving the classification ability is that NCM can detect and describe outliers in real data. Real data sets generally contain outliers and ambiguous data, such as noise on the real images. Those noise data might make the performance of classifiers worse. To improve the performance of a classifier, these outliers should be detected and described. NCM improves the classification and clustering ability because of its ability to determine the outliers and noise data points, which has been proved using the experiments on many datasets.

### 4.4. Initialization of parameters

In this section, the influence of various parameters such as $\delta$,$\overline{w}_1$, $\overline{w}_2$ and $\overline{w}_3$ on the performance of the NCM algorithm are investigated. In all experiments, the values of $T(0)$, $I(0)$, $F(0)$ are initialized randomly. This section will also employ experiments to show the influence of the parameters' initialization. Two diamond dataset as shown in Fig. 14(a) and (b) were used. While Fig. 14 (a) shows the X10 dataset Fig. 14(b) illustrates the X12 dataset. It is obvious that X12 dataset is obtained from X10 dataset by adding

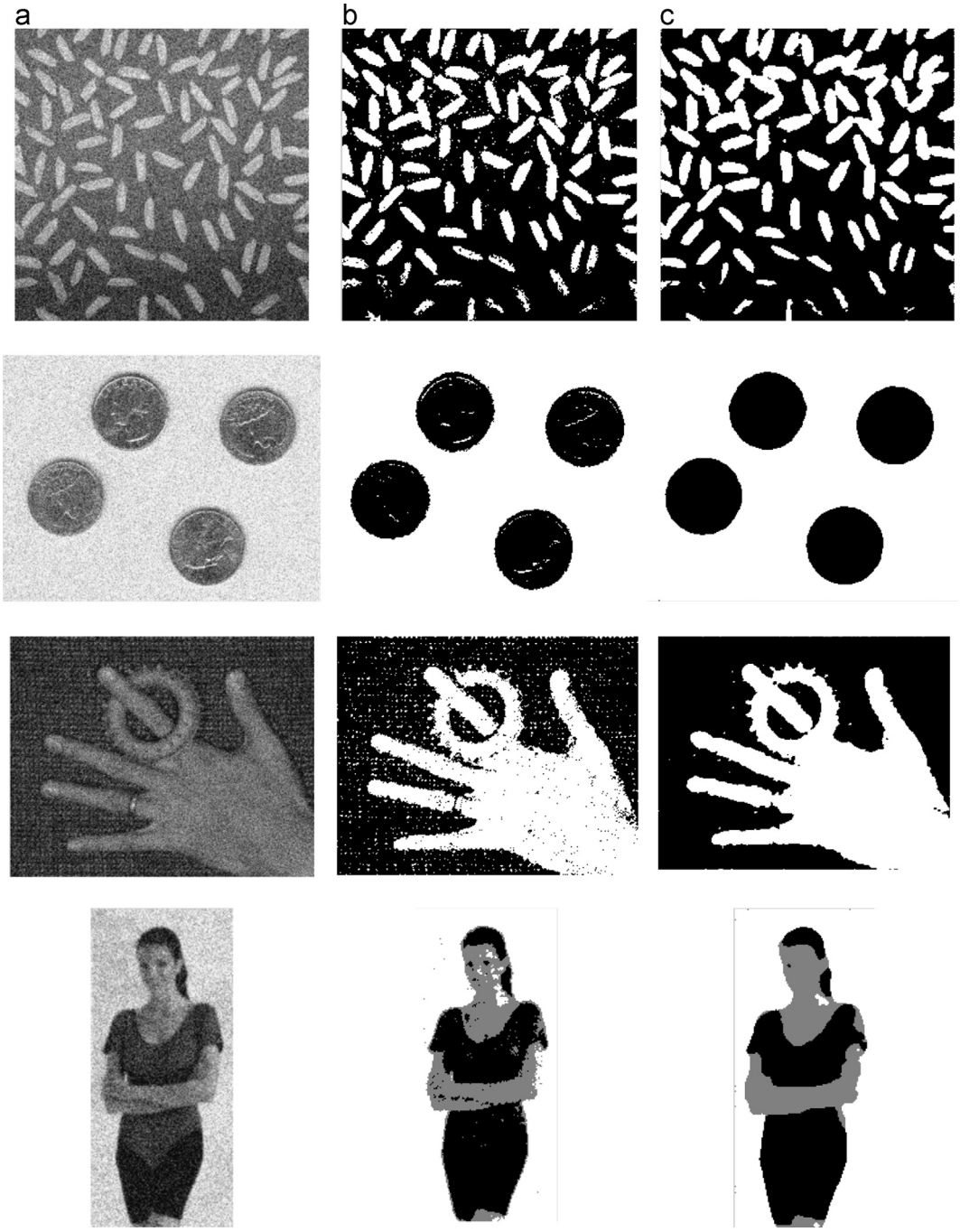


**Fig. 13.** Comparison of segmentation results on different images with Gaussian noise: (a) images degraded by Gaussian noise; (b) result of ASIC and (c) result of NCM.

**Table 11**
*F*-measure values for NCM, FCM–AWA and ASIC methods.

| Image | NCM | FCM–AWA | ASIC |
|---|---|---|---|
| "rice" | **0.8166** | 0.7802 | 0.7312 |
| "eight" | **0.8627** | 0.8307 | 0.7988 |
| "hand" | **0.6801** | 0.5642 | 0.4936 |
| "woman" | **0.8384** | 0.7978 | 0.7345 |

two new data points: the point 6 at (0,0) and the point 12 at (0,10), respectively.

Table 12 shows the results of the NCM on the X10 dataset for different values of $\delta$,$\overline{w}_1$, $\overline{w}_2$ and $\overline{w}_3$. The ground truth centroid of two clusters $V_1$ and $V_2$ are $(-3.34,\ 0)$ and $(3.34,\ 0)$. The experimental results in Table 12 demonstrate that if we vary $\delta$ from 0.5 to 50, keeping all other parameters fixed ($\overline{w}_1 = 0.8$, $\overline{w}_2 = 0.1$ and $\overline{w}_3 = 0.1$), the clustering results of NCM are not much changed. When the $\delta$ is fixed as 0.5 and assign the 0.33 to all weighting parameters in the row 7, and in the row 8, the $\delta$ is still 0.5 and the weighting parameters are assign as 0.3, 0.4 and 0.4, the performance of the clustering are degraded. The cluster centers move toward to grand mean of the data plane, and are close to the 5th and 7th points. The typicality values are high for only a few points close to the 5th and 7th objects. The $T$ membership reveals that there is only one point in each cluster: the 5th in the left cluster and the 7th in the right cluster, and their typicality is close to 1.0. For other points in the same clusters their typicality values are very small. In the row 9, the $\delta$ is increased from 0.5 to 1.2, and

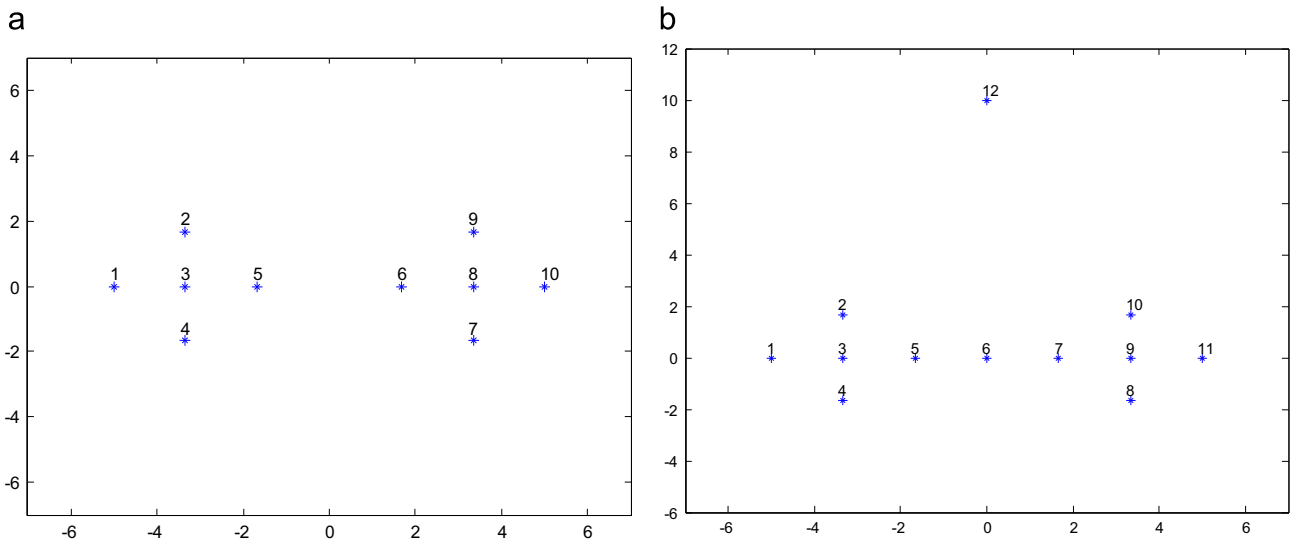


**Fig. 14.** (a) X10 dataset and (b) X12 dataset.

**Table 12**
Results produced by NCM for different values of the parameters on X10 dataset.

| $\delta$ | | | | $V_1$ | $V_2$ |
|---|---|---|---|---|---|
| 0.5 | 0.8 | 0.1 | 0.1 | −3.34 | 3.34 |
| | | | | 0.00 | 0.00 |
| 0.8 | 0.8 | 0.1 | 0.1 | −3.40 | 3.40 |
| | | | | 0.00 | 0.00 |
| 1.4 | 0.8 | 0.1 | 0.1 | −3.42 | 3.42 |
| | | | | 0.00 | 0.00 |
| 2 | 0.8 | 0.1 | 0.1 | −3.42 | 3.42 |
| | | | | 0.00 | 0.00 |
| 10 | 0.8 | 0.1 | 0.1 | −3.42 | 3.42 |
| | | | | 0.00 | 0.00 |
| 50 | 0.8 | 0.1 | 0.1 | −3.42 | 3.42 |
| | | | | 0.00 | 0.00 |
| 0.5 | 0.33 | 0.33 | 0.33 | −1.68 | 1.68 |
| | | | | 0.00 | 0.00 |
| 0.5 | 0.3 | 0.4 | 0.4 | −1.68 | 1.68 |
| | | | | 0.00 | 0.00 |
| 1.2 | 0.3 | 0.4 | 0.4 | −3.39 | 3.39 |
| | | | | 0.00 | 0.00 |
| 1.2 | 1 | 1 | 1 | −3.41 | 3.41 |
| | | | | 0.00 | 0.00 |

**Table 13**
Results produced by NCM for different values of the parameters with X10 dataset.

| $\delta$ | $\overline{w}_1$ | $\overline{w}_2$ | $\overline{w}_3$ | $V_1$ | $V_2$ |
|---|---|---|---|---|---|
| 0.5 | 0.8 | 0.1 | 0.1 | −3.36 | 3.36 |
| | | | | 0.00 | 0.00 |
| 0.8 | 0.8 | 0.1 | 0.1 | −3.40 | 3.40 |
| | | | | 0.00 | 0.00 |
| 1.4 | 0.8 | 0.1 | 0.1 | −3.40 | 3.40 |
| | | | | 0.00 | 0.00 |
| 2 | 0.8 | 0.1 | 0.1 | −3.38 | 3.38 |
| | | | | 0.10 | 0.10 |
| 10 | 0.8 | 0.1 | 0.1 | −3.23 | 3.23 |
| | | | | 0.49 | 0.49 |
| 50 | 0.8 | 0.1 | 0.1 | −3.21 | 3.21 |
| | | | | 0.54 | 0.54 |
| 0.5 | 0.33 | 0.33 | 0.33 | −1.69 | 0.01 |
| | | | | 0.00 | 0.00 |
| 0.5 | 0.3 | 0.4 | 0.4 | −1.67 | 1.67 |
| | | | | 0.00 | 0.00 |
| 1.2 | 0.3 | 0.4 | 0.4 | 0.21 | 3.40 |
| | | | | 0.00 | 0.00 |
| 1.2 | 1 | 1 | 1 | −3.40 | 3.40 |
| | | | | 0.00 | 0.00 |

the clustering results are better than those in the rows 7 and 8 and the centroids are close to the ground truth. In the row 10, we ignore the $\overline{w}_1 + \overline{w}_2 + \overline{w}_3 = 1$ constrain and assign to all weighting parameters 1 and the $\delta$ parameter is fixed to 1.2. The obtained centroids are quite similar to those obtained in the first 6 rows. But when we check the typicality of the points, we see that only few points close to the centroids have high typicality values.

We also carried out the experiments on the X12 data set using the different initial values for parameters. The obtained centroids can be seen in Table 13. There is not any change in the prototypes when $\delta$ was increased keeping all other parameters fixed for the X10 data set, and the centroids are also not changed essentially. In the row 7, the results are not good. The centroids are very close to each other which yielded coincident clusters. In the rows 8 and 10, the obtained results are similar to those obtained on X10 database.

From the above experiments, one can infer that there is a balance between the $\delta$ and the weighting parameters. If the weighting parameters adjusted optimally, the influence of the $\delta$ is reduced and if the weighting parameters are not chosen properly, the $\delta$ value can be adapted for obtaining the more proper results. The $\overline{w}_1$ should be selected bigger than $\overline{w}_2$ and $\overline{w}_3$ for giving more weights to the typicality for reducing the effect of outliers.

## 5. Conclusions

In this paper, an efficient clustering algorithm, neutrosophic *c*-means clustering algorithm (NCM), has been presented to partition the data, especially the fuzzy and indistinct data. The traditional methods only describe the degree to every group. For some samples in the border between different groups, it is difficult determine which group it belongs to. Moreover, if a membership is calculated, it makes the centers of groups inaccurate. NCM is designed to handle these disadvantages of the traditional partitioning approaches.

The efficiency of the proposed NCM algorithm is tested on both data clustering and image segmentation applications. We used the popular data sets and images that are synthetically produced and degraded with noise for the experimental works. Experimental results show that the performance of our algorithm is more efficient than performances of FCM, PCM and FPCM. Moreover,

experimental results also show that the performance of our algorithm is more efficient than performance of the FCM–AWA and ASIC algorithms on image segmentation. In addition, we plan to apply the NCM to the more complex data in our future works.

## Conflict of interest

None declared.

## Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable and constructive comments and suggestions which greatly improved the manuscript.

## References

[1] M.R. Andenberg, Cluster Analysis for Applications, Academic Press, New York, 1973.
[2] M. Ménard, C. Demko, P. Loonis, The fuzzy c+2 means: solving the ambiguity rejection in clustering, Pattern Recognit. 33 (2000) 1219–1237.
[3] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1987.
[4] E. Ruspini, A new approach to clustering, Inf. Control 15 (1969) 22–32.
[5] A. Baraldi, P. Blonda, A survey of fuzzy clustering algorithms for pattern recognition – Part I, IEEE Trans. Syst. Man Cybern. B: Cybern. 29 (6) (1999) 778–785.
[6] A. Baraldi, P. Blonda, A survey of fuzzy clustering algorithms for pattern recognition – Part II, IEEE Trans. Syst. Man Cybern. B: Cybern. 29 (6) (1999) 786–801.
[7] Z. Tong, N. Arye, P. Boaz, K-means clustering-based data detection and symbol-timing recovery for burst-mode optical receiver, IEEE Trans. Commun. 54 (8) (2006) 1492–1501.
[8] Z. Wei, et al., Improved K-means clustering algorithm for exploring local protein sequence motifs representing common structural property, IEEE Trans. Nanobiosci. 4 (3) (2005) 255–265.
[9] J.A. Hartigan, M.A. Wong, A K-means clustering algorithm, J. R. Stat. Soc. 28 (1) (1977) 100–108.
[10] D. Arthur, S. Vassilvitskii, k-means++: the advantages of careful seeding, in: Proceedings of the Eighteenth Annual ACM–SIAM Symposium on Discrete Algorithms, 2007, pp. 1027–1035.
[11] J. Kang, L. Min, Q. Luan, X. Li, J. Liu, Novel modified fuzzy c-means algorithm with applications, Digit. Signal Process. 19 (2009) 309–319.
[12] L. Kaufman, P.J. Rousseeuw, Clustering by Means of Medoids, Facul. Mathe. Infor. (1987) 405–416.
[13] M.S. Yang, K.L. Wu, J.N. Hsieh, J. Yu, Alpha-cut implemented fuzzy clustering algorithms and switching regressions, IEEE Trans. Syst. Man Cybern. B: Cybern. 38 (3) (2008) 588–603.
[14] J. Yu, Q. Cheng, H. Huang, Analysis of the weighting exponent in the FCM, IEEE Trans. Syst. Man Cybern. B: Cybern. 34 (1) (2004) 634–639.
[15] R. Krishnapuram, J. Keller, A possibilistic approach to clustering, IEEE Trans. Fuzzy Syst. 1 (2) (1993) 98–110.
[16] N.R. Pal, K. Pal, J.C. Bezdek, A mixed c-means clustering model, in: Proceedings of the Sixth IEEE International Conference on Fuzzy Systems, Barcelona, 1997, pp. 11–21.
[17] D.E. Gustafson, W.C. Kessel, Fuzzy clustering with a fuzzy covariance matrix, in: Proceedings of IEEE CDC, San Diego, CA, vol. 10(12), 1979, pp. 761–766.
[18] R.N. Dave, Clustering relational data containing noise and outliers, Pattern Recognit. Lett. 12 (1991) 657–664.
[19] M. Roubens, Pattern classification problems and fuzzy sets, Fuzzy Sets Syst. 1 (1978) 239–253.
[20] R.J. Hathaway, J.W. Davenport, J.C. Bezdek, Relational duals of the c-means clustering algorithms, Pattern Recognit. 22 (1989) 205–212.
[21] M.P. Windham, Numerical classification of proximity data with assignment measures, J. Classif. 2 (1985) 157–172.
[22] R.J. Hathaway, J.C. Bezdek, Nerf c-means: non-Euclidean relational fuzzy clustering, Pattern Recognit. 27 (1994) 429–437.
[23] R.N. Dave, Clustering of relational data containing noise and outliers, in: Proceedings of the FUZZ'IEEE 98, vol. 2, 1998, pp. 1411–1416.
[24] M.H. Masson, T. Denoeux, ECM: an evidential version of the fuzzy c-means algorithm, Pattern Recognit. 41 (2008) 1384–1397.
[25] M.H. Masson, T. Denoeux, RECM: relational evidential c-means algorithm, Pattern Recognit. Lett. 30 (2009) 1015–1026.
[26] F.C.H., Rhee,C. Hwang, A type-2 fuzzy C-means clustering algorithm, in: Proceedings of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference, vol. 4, 25–28 2001, pp. 1926–1929.
[27] O. Linda, M. Manic, General type-2 fuzzy C-means algorithm for uncertain fuzzy clustering, IEEE Trans. Fuzzy Syst. 20 (5) (2012) 883–897.
[28] F.C.H. Rhee, Uncertain fuzzy clustering: insights and recommendations, IEEE Comput. Intell. Mag. 2 (1) (2007) 44–56.
[29] F. Smarandache, A Unifying Field in Logics Neutrosophic Logic. Neutrosophy, Neutrosophic Set, Neutrosophic Probability, third ed., American Research Press, 2003.
[30] W.B. Kandasamy, F. Smarandache, Neutrosophic Algebraic Structures, Hexis, Phoenix, 2006.
[31] M. Khoshnevisan, S. Bhattacharya, A short note on financial data set detection using neutrosophic probability, in: F. Smarandache (Ed.), Proceedings of the First International Conference on Neutrosophy, Neutrosophic Logic, Neutrosophic Set, Neutrosophic Probability and Statistics, University of New Mexico, 2002, pp. 75–80.
[32] M. Khoshnevisan, S. Singh, Neurofuzzy and neutrosophic approach to compute the rate of change in new economies, in: F. Smarandache (Ed.), Proceedings of the First International Conference on Neutrosophy, Neutrosophic Logic, Neutrosophic Set, Neutrosophic Probability and Statistics, University of New Mexico, 2002, pp. 56–62.
[33] Y. Guo, H.D. Cheng, A new neutrosophic approach to image denoising, New Math. Nat. Comput. 5 (3) (2009) 653–662.
[34] H.D. Cheng, Y. Guo, A new neutrosophic approach to image thresholding, New Math. Nat. Comput. 4 (3) (2009) 291–308.
[35] Y. Guo, H.D. Cheng, A new neutrosophic approach to image segmentation, Pattern Recognit. 42 (2009) 587–595.
[36] A. Sengur, Y. Guo, Color texture image segmentation based on neutrosophic set and wavelet transformation, Comput. Vis. Image Underst. 115 (8) (2011) 1134–1144.
[37] M.S. Yang, Convergence properties of the generalized fuzzy c-means clustering algorithms, Comput. Math. Appl. 25 (12) (1993) 3–11.
[38] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, IEEE Trans. Fuzzy Syst. 13 (4) (2005) 517–530.
[39] R.A. Fisher, The use of multiple measurements in taxonomic problems, Ann. Eugen 7 (1936) 179–188.
[40] S.K. Pal, Fuzzy tools in the management of uncertainty in pattern recognition, image analysis, vision and expert systems, Int. J. Syst. Sci. 22 (1991) 511–549.
[41] Y.M. Zhu, L. Bentabet, O. Dupuis, V. Kaftandjian, D. Babot, M. Rombaut, Automatic determination of mass functions in Dempster–Shafer theory using fuzzy C-means and spatial neighborhood information for image segmentation, Opt. Eng. 41 (4) (2002) 760–770.
[42] Z.M. Wang, Y.C. Soh, Q. Song, S. Kang, Adaptive spatial information-theoretic clustering for image segmentation, Pattern Recognit. 42 (2009) 2029–2044.
[43] E. Karabatak, Y. Guo, A. Sengur, A modified neutrosophic approach to color image segmentation, J. Electron. Imaging 22 (1) (2013) 013005–013015.

**Yanhui Guo** received the B.S. degree in Automatic Control from the Zhengzhou University, PR China, in 1999, the M.S. degree in Pattern Recognition and Intelligence System from the Harbin Institute of Technology, Harbin, Heilongjiang Province, PR China, in 2002, and the Ph.D. degree in the Department of Computer Science, Utah State University, USA, in 2010. He is currently an Assistant Professor in the School of Science at St. Thomas University. His research interests include image processing, pattern recognition, medical image processing, computer-aided detection/diagnosis, fuzzy logic, and neutrosophic theory.

**Abdulkadir Sengur** graduated from the Department of Electronics and Computer Education at Firat University in 1999. He obtained his M.S. degree from the same department and the same university in 2003. His Ph.D. degree was from the Department of Electric and Electronics Engineering at Firat University in 2006. He is an Assoc. Prof. Dr. in the Department of Software Engineering at the Firat University. His research interest areas include pattern recognition, machine learning and image processing.