Egyptian Informatics Journal (2012) **13**, 39–58

Cairo University

**Egyptian Informatics Journal**

www.elsevier.com/locate/eij
www.sciencedirect.com

ORIGINAL ARTICLE

# Determining the number of clusters for kernelized fuzzy C-means algorithms for automatic medical image segmentation

**E.A. Zanaty**

*Computer Science Department, Faculty of Science, Sohag University, Sohag, Egypt*

Received 12 December 2011; revised 18 January 2012; accepted 29 January 2012
Available online 23 February 2012

**KEYWORDS**

Medical image segmentation;
Clustering methods;
FCM;
Kernel function;
Validity indexes

**Abstract** In this paper, we determine the suitable validity criterion of kernelized fuzzy C-means and kernelized fuzzy C-means with spatial constraints for automatic segmentation of magnetic resonance imaging (MRI). For that; the original Euclidean distance in the FCM is replaced by a Gaussian radial basis function classifier (GRBF) and the corresponding algorithms of FCM methods are derived. The derived algorithms are called as the kernelized fuzzy C-means (KFCM) and kernelized fuzzy C-means with spatial constraints (SKFCM). These methods are implemented on eighteen indexes as validation to determine whether indexes are capable to acquire the optimal clusters number. The performance of segmentation is estimated by applying these methods independently on several datasets to prove which method can give good results and with which indexes. Our test spans various indexes covering the classical and the rather more recent indexes that have enjoyed noticeable success in that field. These indexes are evaluated and compared by applying them on various test images, including synthetic images corrupted with noise of varying levels, and simulated volumetric MRI datasets. Comparative analysis is also presented to show whether the validity index indicates the optimal clustering for our datasets.



E-mail address: zanaty22@yahoo.com



Peer review under responsibility of Faculty of Computers and Information, Cairo University.
doi:10.1016/j.eij.2012.01.004

**Production and hosting by Elsevier**

## 1. Introduction

Clustering is one of the most popular classification methods and has found many applications in pattern classification and image segmentation [1–5]. Clustering algorithms attempt to classify a voxel to a tissue class by using the notion of similarity to the class. Unlike the crisp K-means clustering algorithm [4], the FCM algorithm allows partial membership in different tissue classes. Thus, FCM can be used to model the partial volume averaging artifact, where a pixel may contain multiple tissue classes [2,3]. The kernelized fuzzy C-means

(KFCM) [6–9] used a kernel function as a substitute for the inner product in the original space, which is like mapping the space into higher dimensional feature space. There have been a number of other approaches to incorporating kernels into fuzzy clustering algorithms. These include enhancing clustering algorithms designed to handle different shape clusters [8]. More recent results of fuzzy algorithms have been presented in [9] for improving automatic MRI image segmentation. They used the intra-cluster distance measure to give the ideal number of clusters automatically; more discussion can be found in [9]. Also, possibilistic clustering which is pioneered by the possibilistic c-means (PFCM) algorithm was developed in [10–12]. They had been shown that PFCM is more robust to outliers than FCM. However, the robustness of PFCM comes at the expense of the stability of the algorithm [11]. The PCM-based algorithms suffer from the coincident cluster problem, which makes them too sensitive to initialization [12].

Most fuzzy methods have several advantages including yielding regions more homogeneous than other methods; reducing the spurious blobs; removing noisy spots; reduced sensitivity to noise compared to other techniques. However, they require prior knowledge about the number of clusters in the data, which may not be known for new data [13]. In literature, many studies in dealing with this problem are available in [14–18], and, so, there are many cluster validity indexes in this regard. Compactness and separation are two criteria for the clustering evaluation and selection of an optimal clustering scheme [14]. The variation of data within clusters indicates compactness and isolation between clusters indicates separation, respectively.

Though some compatibility or similarity measure can be applied to choose the clusters to be merged, no validity measure is used to guarantee that the clustering result after a merge is better than the one before the merge. Partial results were stated in [19] to answer the questions: "Can the appropriate number of clusters be determined automatically? And if the answer is yes, how?" More existing methods were found in [14–21] to review few validity indexes that can combine with fuzzy c-means algorithms. But, the performance of wide range indexes is not found in any literature before; especially when they applied to kernelized fuzzy c-means (KFCM) or kernelized fuzzy c-means with spatial constraints (SKFCM) methods.

In this paper, we seek the answer to the previous questions for exploring which indexes can achieve high accuracy segmentation whey they performed with KFCM and SKFCM. Our objective is not to improve the segmentation accuracy via enhancing the kernel function, but is to find the indexes with KFCM and SKFCM capable to produce good MRI segmentation. For that; the original Euclidean distance in the FCM algorithm is replaced by the Gaussian radial base function (GRBF)-induced kernel, which is shown to be more robust than FCM (with Euclidean distances). This will make a generalization of the existing FCM methods. The KFCM and SKFCM algorithms based on Gaussian RBF kernel are derived and applied independently on each image. Based on these algorithms, eighteen indexes are implemented to estimate the number of clusters that represents the best structure of a given image. Key existing solutions are evaluated to obtain the cluster validity in the domain of image segmentation. A wide number of various validity indexes from the classical and more recent indexes are examined. As segmentation of medical images is of particular interest in our application, the work here includes the assessment of those indexes on 3D MRI datasets.

The rest of this paper is organized as follows: Section 2 presents the kernel methods. Several criteria to determine the number of clusters are briefly reviewed in Section 3. Experimental comparisons are presented in Section 4. Finally, Section 5 gives our conclusions.

## 2. Kernel methods

The kernel methods [8,13,22–26] are one of the most researched subjects within machine learning community in the recent few years and have widely been applied to pattern recognition and function approximation. A common philosophy behind these algorithms is based on the following kernel (substitution) trick, that is, firstly with a (implicit) nonlinear map, from the data space to the mapped $d$ feature space, $\Psi: X \to F\,(x \to \Psi(x))$, a dataset $\{x, \ldots, x\} \subseteq X$ (an input data space with low dimension) is mapped into a potentially much higher dimensional feature space or inner product $F$, which aims at turning the original nonlinear problem in the input space into potentially a linear one in rather high dimensional feature space so as to facilitate problem solving as proved by Girolami [23]. A kernel $K(x, y)$ in the feature space can be represented as:

$$K(x, y) = (\Psi(x), \Psi(y)) \tag{1}$$

where $(\Psi(x), \Psi(y))$ denotes the inner product operation.

An interesting point about kernel function is that the inner product between $\Psi(x)$ and $\Psi(y)$ can be implicitly computed in $F$, without explicitly using or even knowing the mapping $\Psi$.

So, kernels allow computing inner products in the space, where one could otherwise not practically perform any computations. Three commonly-used kernel functions in literature [25] are:

(1) Gaussian Radial basis function (GRBF) kernel: $K(x, y) = \exp\left(-\|x - y\|^2/\sigma^2\right)$.
(2) Polynomial kernel: $K(x, y) = (\langle x, y\rangle + 1)^d$.
(3) Sigmoid kernel $K(x, y) = \tanh(\alpha\langle x, y\rangle + \beta)$.

where $\sigma$, $d$, $\alpha$ and $\beta$ are the adjustable parameters of the above kernel functions. The main motives of using the kernel methods consist of: (1) inducing a class of robust non-Euclidean distance measures for the original data space to derive new objective functions and thus clustering the non-Euclidean structures in data; (2) enhancing robustness of the original clustering algorithms to noise and outliers, and (3) still retaining computational simplicity.

Sigmoid kernel is a two-layer neural network kernel and is used as a particular kind of two-layer sigmoid neural network. For this, only a set of parameters satisfying the Mercer theorem can be used to define a kernel function [23–26]. The interested reader may refer to [25] for more details. In this section we only stress on GRBF, which is shown to be more robust than FCM (with Euclidean distances) [7].

### 2.1. Fuzzy C-means method (FCM)

Fuzzy C-means clustering (FCM), also known as fuzzy ISODATA, is a data clustering algorithm in which each data point belongs to a cluster to determine a degree specified by its

membership grade [1–3]. Bezdek [1] proposed this algorithm as an alternative to earlier k-means clustering. FCM partitions a collection of $N$ vector $x_i$, $i = 1, \ldots, N$ into $C$ fuzzy groups, and finds a cluster centre in each group such that an objective function of a dissimilarity measure is minimized. The major difference between FCM and k-means is that FCM employs fuzzy partitioning such that a given data point can belong to several groups with the degree of belongingness specified by membership grades between 0 and 1. In FCM, the membership matrix $U = [u_{ij}]$ is allowed to have not only 0 and 1 but also the elements with any values between 0 and 1. This matrix satisfies the constraints:

$$\sum_{i=1}^{C} u_{ij} = 1, \quad \forall j = 1, \ldots, N; \ 0 \leqslant u_{ij} \leqslant 1, \qquad \sum_{j=1}^{N} u_{ij} > 0, \quad \forall i$$

The objective function of FCM can be formulated as follows:

$$J_m = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m \|x_j - c_i\|^2 \tag{2}$$

where $C$ is the number of clusters; $c_i$ is the cluster centre of fuzzy group $i$ and the parameter $m$ is a weighting exponent on each fuzzy membership. Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, updating of membership $u_{ij}$ and the cluster centres $c_i$ by:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left( \frac{\|x_j - c_i\|}{\|x_j - c_k\|} \right)^{2/(m-1)}} \tag{3}$$

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{N} u_{ij}^m}$$

In image clustering, the most commonly used feature is the gray-level value, or intensity of image pixel. Thus the FCM objective function is minimized when high membership values are assigned to pixels whose intensities are close to the centroid of its particular class, and low membership values are assigned when the point is far from the centroid.

### 2.2. Kernelized fuzzy C-means method (KFCM)

The algorithm that uses inner products can implicitly be executed in the feature space $F$. This trick can also be used in clustering, as shown in support vector clustering [22] and kernel (fuzzy) C-means algorithms [23,24]. A common ground of these algorithms is to represent the clustering centre as a linearly-combined sum of all $\Psi(x_k)$, i.e. the clustering centres is located in feature space. A kernelized FCM algorithm is constructed with objective function as following:

$$J_m = \sum_{i=1}^{c} \sum_{j=1}^{N} u_{ij}^m ||\Psi(x_j) - \Psi(c_i))||^2 \tag{4}$$

where $\Psi$ is an implicit nonlinear map as described previously. Unlike Refs. [23,24], $\Psi(c_i)$ here is not expressed as a linearly-combined sum of all $\Psi(x_k)$ anymore, a so-called dual representation, but still reviewed as a mapped point (image) of $c_i$ in the original space, then with the kernel substitution trick, we have:

$$\begin{aligned} \|\Psi(x_j) - \Psi(c_i))\|^2 &= (\Psi(x_j) - \Psi(c_i))^T (\Psi(x_j) - \Psi(c_i)) \\ &= \Psi^T(x_j)\Psi(x_j) - \Psi^T(x_j)\Psi(c_i) \\ &\quad - \Psi(x_j)\Psi^T(c_i) + \Psi^T(c_i)\Psi(c_i) \\ &= K(x_j, x_j) - 2K(x_j, c_i) + K(c_i, c_i) \end{aligned}$$

In GRBF kernel $K(x, c) = \exp(-\|x - c\|^2/\sigma^2)$, $K(x_j, x_j) = 1$, $K(c_i, c_i) = 1$, and $\Psi^T(x_j)\Psi(c_i) = \Psi(x_j)\Psi^T(c_i)$. From Eqs. (2) and (4), we get:

$$J_m = 2 \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m (1 - K(x_j, c_i)) \tag{5}$$

The objective of this paper is to determine the validity criterion of kernelized fuzzy C-means (KFCM) when applied to MRI data sets. It was shown in [22] that the GRBF kernel, has better segmentation results on simulated MR images corrupted by noise and other artifacts than the based polynomials algorithms [21–26]. We confine ourselves to the GRBF kernel to seek the best index that can be used for reliable kernelized fuzzy C-means clustering.

In a similar way to the FCM algorithm, the objective function $J_m$ in Eq. (5) can be minimized under the constraint of $U$. Specifically, taking the first derivatives of $J_m$ with respect to $u_{ij}$ and $c_i$, and zeroing them respectively, two necessary but not sufficient conditions for $J_m$ to be at its local extrema will be obtained. The fuzzy membership matrix $U$ can be obtained from:

$$u_{ij} = \frac{\left(\sum_{k=1}^{C} (1 - K(x_j, c_k))\right)^{1/(m-1)}}{(1 - K(x_j, c_i))^{1/(m-1)}} \tag{6}$$

The cluster center $c_i$ can be obtained from:

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m K(x_j, c_i) x_j}{\sum_{j=1}^{N} u_{ij}^m K(x_j, c_i)} \tag{7}$$

Through the following section, we will only use the GRBF kernel for the simplicity of derivation of Eqs (6) and (7). For other kernel functions, the corresponding equations are a little more complex, because their derivatives are not as simple as the GRBF kernel function. The standard kernelized fuzzy C-means (KFCM) algorithm can be summarized in the following steps:

Step 1: Fix $c, t_{\max}, m > 1$ and $\varepsilon > 0$ for some positive constant.
Step 2: Initialize the memberships $u_{ij}^0, C, m$.
Step 3: For $t = 1, 2, \ldots, t_{\max}$ do
  (a) Update all prototype $c_i^t$ with Eq. (7);
  (b) Update all memberships $u_{ij}^t$ with Eq. (6);
  (c) Compute $E^t = \max_{i,j} |u_{ij}^t - u_{ij}^{t-1}|$, if $E^t \leqslant \varepsilon$, stop;
End;

### 2.3. Kernelized fuzzy C-means with spatial constraints (SKFCM)

In this section, we select three kinds of fuzzy c-means methods which almost cover all objective functions. The objective function consists of two parts: the original objective function and penalty called spatial constraint. All improvements of fuzzy c-means methods lie on modifying spatial constraint formula. Based on this formula, we can divide fuzzy methods into three categories: firstly, the spatial constraint is only based on

Euclidean distance as in Ahmed et al. [2]. In the next category, the spatial constraint is only based on membership values as presented in Zhang et al. [6]. The third one, it uses Euclidean distances based on weighted averaging image window as in Kang et al. [7]. Others recent methods try to enhance one of these objective function such as in [9,12]. Here, we replace the Euclidian distance by GRBF kernel of the KFCM with spatial constraint to induce the generalization of these methods. For example, Ahmed et al. [2] introduced the modified objective function of FCM as:

$$J_m = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \|x_j - c_i\| + \frac{\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}\|x_r - c_i\|^2 \tag{8}$$

The Euclidean distance of objective function in Eq. (8) is replaced by Gaussian RBF kernel as:

$$J_m = 2\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m (1 - K(x_j, c_i)) + \frac{2\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}(1 - K(x_r, c_i)) \tag{9}$$

where $N_j$ stands for the set of neighbors that exist in a window around $x_j$ (not including $x_j$ itself) and $N_R$ is the cardinality of $N_j$. The parameter $\alpha$ controls the effect of the penalty term and lies between zero and one inclusive.

This penalty term $\frac{2\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}(1 - K(x_r, c_i))$ contains spatial neighborhood information, which acts as a regularizer and biases the solution toward piecewise-homogeneous labeling. Such regularization is helpful in segmenting images corrupted by noise.

The objective function $J_m$ under the constraint of $u_{ij}$ and $c_i$ can be solved by using the following theorem [7]:

**Theorem.** *Let* $X = \{x_i, i = 1, 2, \ldots, N | x_i \in R^d\}$ *denotes an image with N pixels to be partitioned into C classes (clusters), where* $x_i$ *represents feature data. The algorithm is an iterative optimization that minimizes the objective function defined by Eq. (9) with the following constraints:*

$$\sum_{i=1}^{C} u_{ij} = 1, \quad \forall j = 1, \ldots, N;\ 0 \leqslant u_{ij} \leqslant 1, \qquad \sum_{j=1}^{N} u_{ij} > 0, \quad \forall i \tag{10}$$

*Then* $u_{ij}$ *and* $c_i$ *must satisfy the following equalities:*

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{(1-K(x_j,c_i))+\frac{\alpha}{N_R}\sum_{r\in N_j}(1-K(x_r,c_i))}{(1-K(x_j,c_k))+\frac{\alpha}{N_R}\sum_{r\in N_j}(1-K(x_r,c_k))}\right)^{1/(m-1)}} \tag{11}$$

$$c_i = \frac{\sum_{j=1} u_{ij}^m \{(1 - K(x_j, c_i)x_j + \frac{\alpha}{N_R}\sum_{r\in N_j}(1 - K(x_r, c_i))x_r\}}{\sum_{j=1}^{N} u_{ij}^m \{(1 - K(x_i, c_i)) + \frac{\alpha}{N_R}\sum_{r\in N_j}(1 - K(x_r, c_i))\}} \tag{12}$$

**Proof.** The minimization of constraint problem $J_m$ in Eq. (9) under constraints can be solved by using the Lagrange multiplier method. Now we define a new objective function with constraint condition (10) as follows:

$$L_m = 2\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m (1 - K(x_j, c_i)) + \frac{2\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}(1 - K(x_r, c_i)) + \sum_{j=1}^{N}\lambda_j\left(1 - \sum_{i=1}^{C} u_{ij}\right)$$

Taking the partial derivative of $L_m$ with respect $u_{ij}$ and $\lambda_i$, and then setting them to zero, we have:

$$\frac{\partial L_m}{\partial u_{ij}} = 0 \Longleftrightarrow 2mu_{ij}^{m-1}(1 - K(x_j, c_i)) + \frac{2\alpha}{N_R} m u_{ij}^{m-1}\sum_{r\in N_j}(1 - K(x_r, c_i)) + \lambda_j(-1) = 0 \tag{13}$$

$$\frac{\partial L_m}{\partial \lambda_j} = 0 \Longleftrightarrow \sum_{i=1}^{C} u_{ij} - 1 = 0 \tag{14}$$

From Eq. (13) we obtain:

$$u_{ij} = \left(\frac{\lambda_j}{2m((1 - K(x_j, c_i)) + \frac{\alpha}{N_R}\left(\sum_{r\in}(1 - K(x_r, c_i))\right)}\right)^{1/(m-1)} \tag{15}$$

Substituting (15) into (14) gives:

$$\left(\frac{\lambda_j}{2m}\right)^{1/(m-1)}\sum_{k=1}^{C}\left(\frac{1}{(1 - K(x_j, c_k)) + \frac{\alpha}{N_R}\sum_{r\in}(1 - K(x_r, c_k))}\right)^{1/(m-1)} = 1 \tag{16}$$

$$\left(\frac{\lambda_j}{2m}\right)^{1/(m-1)} = \frac{1}{\sum_{k=1}^{C}((1 - K(x_j, c_k)) + \frac{\alpha}{N_R}\left(\sum_{r\in}(1 - K(x_r, c_k))\right)^{-1/(m-1)}} \tag{17}$$

Finally, substituting Eq. (17) into Eq. (15), we get:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{(1-K(x_j,c_i))+\frac{\alpha}{N_R}\sum_{r\in N_j}(1-K(x_r,c_i))}{(1-K(x_j,c_k))+\frac{\alpha}{N_R}\sum_{r\in N_j}(1-K(x_r,c_k))}\right)^{1/(m-1)}} \tag{18}$$

In GRBF kernel $K(x, c) = \exp(-\|x - c\|^2/\sigma^2)$, similarly we obtain:

$$\frac{\partial L_m}{\partial c_i} = 0 \Longleftrightarrow 2\sum_{j=1}^{N} u_{ij}^m (1 - K(x_j, c_i))(x_j - c_i)(-1/\sigma^2) + \frac{2\alpha}{N_R} \times \sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}(1 - K(x_r, c_i))(x_r - c_i)(-1/\sigma^2) = 0$$

Then we get:

$$\sum_{j=1}^{N} u_{ij}^m ((1 - K(x_j, c_i))x_j + \frac{\alpha}{N_R}\sum_{r\in N_j}(1 - K(x_r, c_i))x_r) = \sum_{j=1}^{N} u_{ij}^m ((1 - K(x_j, c_i)) + \frac{\alpha}{N_R}\sum_{r\in N_j}(1 - K(x_r, c_i)))c_i$$

$$c_i = \frac{\sum_{j=1} u_{ij}^m ((1 - K(x_j, c_i))x_j + \frac{\alpha}{N_R}\sum_{r\in N_j}(1 - K(x_r, c_i))x_r)}{\sum_{j=1}^{N} u_{ij}^m ((1 - K(x_i, c_i)) + \frac{\alpha}{N_R}\sum_{r\in N_j}(1 - K(x_r, c_i)))} \tag{19}$$

This completes the proof. □

The objective function of Zhang et al. [6] can be defined as:

$$J_m = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \|x_j - c_i\|^2 + \frac{\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}(1-u_{ir})^m$$

Similar to Ahmed et al. [2], the objective function of Zhang et al. [6] can be derived as:

$$J_m = 2\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m (1-K(x_j,c_i)) + \frac{\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \sum_{r\in N_j}(1-u_{ir})^m \tag{20}$$

The objective function $J_m$ is minimized under the constraint of $u_{ij}$ and we get:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{(1-K(x_j,c_i))+\frac{\alpha}{N_R}\sum_{r\in N_j}(1-u_{ir})^m}{(1-K(x_j,c_k))+\frac{\alpha}{N_R}\sum_{r\in N_j}(1-u_{ir})^m}\right)^{1/(m-1)}} \tag{21}$$

Because the penalty function does not depend on $c_i$ the necessary conditions under which Eq. (12) attains its minima is identical to that of standard KFCM (Eq. (7)).

The objective functions of Kang et al. [7] is as follows:

$$J_m = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \|x_j - c_i\|^2 + \alpha\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m \|\bar{x}_j^* - c_i\|^2 \tag{22}$$

Similarly, in Kang et al. [7], the objective function can be derived as:

$$J_m = 2\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m (1-K(x_j,c_i)) + 2\alpha\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^m (1 - K(\bar{x}_j^*, c_i)) \tag{23}$$

where $\bar{x}_j^*$ represents the grey value of pixel in the weighted averaging image window, more discussions can be shown in [7].

Similarly to Eqs. (11) and (12), the membership functions and cluster centers are updated by the following expressions:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{(1-K(x_j,c_i))+\alpha(1-K(\bar{x}_j^*,c_i))}{(1-K(x_j,c_k))+\alpha(1-K(\bar{x}_j^*,c_k))}\right)^{1/(m-1)}} \tag{24}$$

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m \left(K(x_i,c_i)x_i + \alpha(1-K(\bar{x}_j^*,c_i))\bar{x}_j^*\right)}{\sum_{j=1}^{N} u_{ij}^m \{(1-K(x_i,c_i)) + \alpha(1-K(\bar{x}_i^*,c_i))\}} \tag{25}$$

The SKFCM algorithm is almost identical to the KFCM, except in step 3(a) and (b), Eqs. (18) and (19) in Ahmed et al. [2], Eqs. (21) and (7) in Zhang et al. [6], and Eqs. (24) and (25) in Kang et al. [7] are used instead of Eqs. (6) and (7) to update the memberships and centers.

## 3. Indexes for determining number of clusters

One of the most important issues in cluster analysis is the evaluation of clustering results to find the partitioning that best fits the underlying data. Although fuzzy methods [3–7] have several advantages in segmentation accuracy and less sensitive to noise, they have a drawback in requiring prior knowledge about the number of clusters in the data, which may not be known for new data. The final number of clusters is still always sensitive to define the threshold criterion for merging. Though some compatibility or similarity measure can be applied to choose the clusters to be merged, no validity measure is used to guarantee that the clustering result after a merge is better than the one before the merge, i.e. we want to explore which indexes can achieve high accuracy segmentation [19]. In the sequel, we evaluate 18 indexes for determining the number of clusters.

In this section, we will seek the most suitable validity criterion in the following indexes, regrouped into three categories. The first category uses only the membership values. The second one involves both the $U$ matrix and the dataset itself. Statistical indexes are presented in the third one. In our implementation, the under-sampled dataset is obtained by averaging every $2\times2$ pixels ($2\times2\times2$ voxels for 3D data) in the original dataset. One desirable effect here is that the resultant half-sized dataset contains smaller noise, which ought to lead to better cluster estimation. The KFCM and SKFCM are defined by a matrix $U = [u_{ij}]$, where $u_{ij}$ denotes the degree of membership of the vector $x_j$ in the $i$ cluster. Also, a set of cluster representatives have been defined. We denote a criterion by $q$, and we search for the minimum or maximum in the plot of $q$ versus $C$ (number of clusters). Also, in case that $q$ exhibits a trend with respect to the number of clusters, we seek a significant knee of decrease (or increase) in the plot of $q$.

### 3.1. Indexes involving only the membership values

#### 3.1.1. The partition coefficient (PC)

The partition coefficient is proposed by Bezdek et al. [27], and defined as:

$$PC = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^2 \tag{26}$$

The $PC$ index values range in $[1/C, 1]$, where $C$ is the number of clusters. The closer the index to unity the "crisper" the clustering is. In case that all membership values to cluster partition are equal, that is, $u_{ij} = 1/C$, the $PC$ coefficient obtains its lowest value. Thus, a value close to $1/C$ indicates that there is no clustering tendency in the considered dataset or the clustering algorithm failed to reveal it.

#### 3.1.2. The partition entropy coefficient (PE)

The partition entropy coefficient is defined as [27]:

$$PE = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}\cdot\log(u_{ij}) \tag{27}$$

The index is computed for values of $C$ greater than 1 and its values range in $[0, \log C]$. The closer the value of $PE$ to 0, the harder the clustering is. As in the previous case, the values of index close to the upper bound (i.e. $\log C$), indicate absence of any clustering structure in the dataset or inability of the algorithm to extract it.

#### 3.1.3. The modification of the PC index (MPC)

The drawback of $PC$ is its monotonous dependency on the number of clusters. Thus, we seek significant knees of increase (for $PC$) or decrease (for $PE$) in plot of the indexes versus the number of clusters. Modification of the PC index proposed by Dave [28] can reduce the monotonic tendency by using the following formula:

$$MPC(C) = 1 - \frac{C}{C-1}(1 - C^*PC) \tag{28}$$

where $0 \leqslant MPC(C) \leqslant 1$. Note that the $MPC$ index is equivalent to the non-fuzziness index (NFI). In general, an optimal cluster number $C^*$ is found by solving $\max_{2\leqslant C\leqslant N-1} MPC(C)$ to produce a best clustering performance for the data set $X$.

### 3.2. Indexes involving the membership values and the dataset

#### 3.2.1. The Xie–Beni index ($S$)

The Xie–Beni index, also called the compactness and separation validity function, is defined as [29]:

$$S = \left\{ \left(\frac{1}{N}\right) \sum_{i=1}^{C} \sigma_i^2 \right\} \Big/ \{D_{\min}\}^2 \tag{29}$$

where

$$\sigma_i^2 = \sum_{j=1}^{N} u_{ij} \|x_j - c_i\|^2$$

where $x_j$: $j = 1, \ldots, N$ is a set of $N$ feature vectors that is to be partitioned (clustered) into $C$ clusters, for each cluster $c_i$, $i = 1, \ldots, C$, represents its prototype, or center, and $D_{\min}$ is the minimum distance between the prototypes (cluster centers). A large $D_{\min}$ value means a lower reciprocal value. Each $\sigma^2$ is a fuzzy weighted mean-square error for the $i$th cluster, which is smaller for more compact clusters.

#### 3.2.2. The modified Xie–Beni index ($XB$)

It is a modification [30] of the Xie–Beni index by summing only the members of each cluster rather than over all $N$ exemplars for each cluster. Also the reciprocal $XB = 1/S$ is taken so that a larger value of $XB$ indicates a better clustering and the $XB$ is called modified Xie–Beni clustering validity measure.

#### 3.2.3. The I-indexes ($I$)

Consider a data set of $N$ points partitioned into $C$ clusters. The $I$ index [31] is defined as follows:

$$I = \left(\frac{D_{\max}}{C \times E_C}\right)^p \tag{30}$$

where

$$E_C = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ji} \|x_i - c_j\|$$

The value of $C$ for which this index is maximized is considered to be the correct number of clusters. The first factor $D_{\max}$ is the maximum distance between the prototypes (cluster centers). It will increase with the number $C$, hence reducing the index as the number of cluster increases. The second factor $\frac{1}{C}$ will try to reduce the index as the number of cluster increases. The third factor $\frac{1}{E_C}$, which measures the total fuzzy dispersion, will penalize the index as it is increased. The power of $p$ is used to control the contrast between the different cluster configurations. In our implementation, we take $p = 2$.

#### 3.2.4. The Davies–Bouldin index ($DB$)

Assume a similarity measure $R(v_i, v_j) = R_{ij}$ between two clusters $v_i$ and $v_j$ is defined based on a measure of dispersion $s(c_i) = s_i$ of a cluster $v_i$, and a dissimilarity measure $d(v_i, v_j) = d_{ij}$ between two clusters $v_i$ and $v_j$. $R_{ij}$ is defined to be non negative and symmetric. Then the Davies–Bouldin ($DB$) index is defined as [32]:

$$DB = \frac{1}{C} \sum_{j=1}^{C} \max R_{ij}, \quad i = 1, \ldots, C, i \neq j \tag{31}$$

where

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}$$

The similarity between clusters is obtained and the maximum value is denoted as max $R_{ij}$. If $c_i$ denotes the centroid of cluster $v_i$, with:

$$s_i = \sqrt{\frac{1}{card(v_i)} \sum\nolimits_{x \in v_i} \|x - c_i\|^2} \tag{32}$$

$$d_{ij} = \max \|c_i - c_j\| i, \quad j = 1, \ldots, C$$

The number of clusters which minimizes the $DB$ index is the optimal one.

#### 3.2.5. The cluster validity measure ($VM$)

The cluster validity measure (VM) is defined as [33]:

$$VM = (C + (f \times G(2,1) + 1)).\frac{D_a}{D_e} \tag{33}$$

where $D_a$ which measures the compactness of the clusters, is defined as:

$$D_a = \frac{1}{N} \sum_{i=1}^{C} \sum_{x \in v_i} \|x - c_i\|^2 \tag{34}$$

$D_e$ which measures the average separation between two clusters over all possible pairs of clusters, is defined as:

$$D_e = \text{average}(\|c_i - c_j\|^2), \quad i = 1, 2, \ldots, C, \; j = i + 1, \ldots, C \tag{35}$$

$f$ is some natural constant; $G(2, 1)$ is a GRBF with mean value equal to 2 and standard deviation equal to 1, and $c_i$ is the cluster center of the cluster $v_i$. The $VM$ measure should be minimized to get a good segmentation result coming from compact and well separated clusters.

#### 3.2.6. The Fukuyama–Sugeno index ($FS$)

The Fukuyama–Sugeno index is defined as [34]:

$$FS = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m (\|x_i - c_j\|^2 - \|c_j - \bar{c}\|^2) \tag{36}$$

where $\bar{c} = \sum_{i=1}^{c} c_i / C$. It is clear that for compact and well-separated clusters we expect small values for $FS$. The first term in the parenthesis measures the compactness of the clusters and the second one measure the distances between two clusters centers.

#### 3.2.7. The fuzzy hyper volume ($FHV$)

The fuzzy hyper volume is proposed by Gath and Geva [35] based on the concepts of hyper volume and density. The fuzzy hyper volume is given by:

$$FH = \sum_{j=1}^{C} V_j \tag{37}$$

$$V_j = \left| \sum_{j} \right|^{1/2} = \left( \frac{\sum_{i=1}^{N} u_{ij}^m (x_i - c_j)(x_i - c_j)^T}{\sum_{i=1}^{N} u_{ij}^m} \right)^{1/2} \tag{38}$$

Small values of *FH* indicate the existence of compact clusters.

*3.2.8. The average partition density (PA)*

The average partition density is defined as [32]:

$$PA = \frac{1}{C}\sum_{j=1}^{C}\frac{S_j}{V_j} \tag{39}$$

with $S_j = \sum_{x\in X_j} u_{ij}$, where $x_j$ is the set of data points within a window around $_j$ (i.e. the center of $c_j$ cluster), $S_j$ is called the sum of the central members of the $c_j$ cluster.

*3.2.9. The partition density index (PD)*

The partition density index is given by [32]:

$$PD = S/FH$$

where

$$S = \sum_{j=1}^{C} S_j \tag{40}$$

The *PD* and *PA* measures should be at a minimum to get good segmentation results.

*3.2.10. The separation and compactness index (SC)*

A validity function proposed by Zahid et al. [36] is defined by:

$$SC(C) = SC_1(C) + SC_2(C) \tag{41}$$

where

$$SC_1(C) = \frac{\sum_{i=1}^{n_c}\|c_i - \overline{c_i}\|^2/C}{\sum_{i=1}^{C}(\sum_{j=1}^{C}(u_{ij}^m)\|x_j - c_i\|^2/\sum_{j=1}^{C}u_{ij})}$$

and

$$SC_2(C) = \frac{\sum_{i=1}^{C-1}\sum_{l=i+1}^{C}(\sum_{j=1}^{C}(\min(u_{ij},u_{lj}))^2)/\sum_{j=1}^{C}\min(u_{ij},u_{lj})}{\sum_{j=1}^{C}(\max_{1\leqslant i\leqslant C}u_{ij})^2/\sum_{j=1}^{C}\max_{1\leqslant i\leqslant C}u_{ij}}$$

Both $SC_1$ and $SC_2$ measure the ratio of separation and compactness. $SC_1$ considers the geometrical properties of the data structure and membership functions. $SC_2$ considers only the fuzzy memberships. In general, we find an optimal $C^*$ by solving $\min_{2\leqslant C\leqslant N} SC(C)$ to produce the best clustering performance for the data set $X$.

*3.2.11. The partition coefficient and exponential separation (PCAES) index*

The PCAES validity index is defined as [37]:

$$\mathrm{PCAES}_i = \sum_{j=1}^{N} u_{ij}^2/u_M - \exp\left(-\min_{k\neq i}\{\|c_i - c_k\|^2\}/\beta_T\right) \tag{42}$$

where

$$u_M = \min_{1\leqslant i\leqslant C}\left\{\sum_{j=1}^{N} u_{ij}^2\right\}$$

and

$$\beta_T = \frac{\sum_{l=1}^{C}\|c_1 - \overline{c}\|^2}{C}$$

Obviously we have $-C \leqslant \mathrm{PCAES(C)} \leqslant C$.

A large PCAES($C$) value means that each of these $c$ clusters is compact and separated from other clusters. A small PCAES($C$) value means that some of these $c$ clusters are not compact or separated from other clusters. The maximum of PCAES($C$) with respect to $c$, could be used to detect the data structure with a compact partition and well-separated clusters. Thus, an optimal $c^*$ can be found by solving $\min_{2\leqslant C\leqslant N}\mathrm{PCAES}(C)$ to produce a best clustering performance for the data set $X$.

*3.2.12. The PBMF index*

The *PBMF*-index [38] can provide a measure of goodness of clustering on different partitions of a data set and is defined as follows:

$$PBMF = \frac{1}{c}\times\frac{E_1\times\max_{i,j}\|c_i - c_j\|}{\sum_{i=1}^{C}\sum_{j=1}^{N}u_{ij}^m\|x_j - c_i\|} \tag{43}$$

where $C$ is the number of clusters. Here, $E_1 = \sum_{j=1}^{N} u_{ij}\|x_j - c_i\|$.

It is seen that the factor $E_1$ in the expression of the index is a constant term for a particular data set. The maximum value of the index is supposed to give the appropriate number of clusters.

*3.2.13. The compose within and between scattering (CWB) index*

*CWB* index [39] is defined by:

$$CWB = \alpha Scat(C) + Dis(C) \tag{44}$$

where

$$Scat(c) = \frac{\frac{1}{C}\sum_{i=1}^{C}[\sigma(c_i)^T\cdot\sigma(c_i)]^{1/2}}{[\sigma(X)^T\cdot\sigma(X)]^{1/2}} \qquad Dis(C) = \frac{D_{\max}}{D_{\min}}\sum_{i=1}^{C}\left(\sum_{r=1}^{C}\|c_i - c_r\|\right)^{-1}$$

$$\sigma(X) = \frac{1}{N}\sum_{j=1}^{N}(x_j - \overline{x})^2 \qquad \overline{x}:\ \text{center of the whole dataset}$$

$$\sigma(c_i) = \frac{1}{N}\sum_{j=1}^{N}u_{ij}(x_j - c_i)^2 \qquad \text{where} i = 1,\ldots,C$$

$$D_{\max} = \max\{c_i - c_r\},\quad i,r = 1,\ldots,C \text{ and } i\neq r$$
$$D_{\min} = \min\{c_i - c_r\},\quad i,r = 1,\ldots,C \text{ and } i\neq r$$
$$\alpha = Dis(c_{\max})$$

*CWB* tends to find an optimum value of both compactness and separation in fuzzy $c$-partitions. In this index, *Scat(C)* is average scattering for $c$ classes and *Dis(C)* is a distance function associated with distance between class centers. The first term represents the compactness and second term the separation. These two terms usually show opposite trends as $C$ is changed. The minimum value of the index is supposed to give the appropriate number of clusters.

*3.3. Statistical indexes*

Some of the widely adopted criteria for statistical model selection are used for determining the number of clusters. Recently,

El-Melegy et al. [19] presented two indexes for this reason, one is based on Akaike's information criterion (AIC) [40] and the other is based on Cross-Validation [41].

*3.3.1. The index based on Akaike's information criterion (AIC)*

The classical AIC is defined as:

$$AIC = D_a + 2\mu\sigma^2 \tag{45}$$

where $\mu(C) = (C-1)N + C$ in soft case, and $\mu(C) = N + C$ in case of hard, is the number of degree of freedom of the model, $D_a$ can be computed from Eq. (34), and $\sigma$ the noise level, can be estimated from

$$\sigma^2 = \frac{D_a(C^*)}{\rho N - \mu(C^*)} \tag{46}$$

where $C^*$ is the maximum number of clusters, $\rho$ is the co-dimension of the model ($\rho = 1$). The smaller the $AIC$ value is, the better the clustering performance for the data set.

*3.3.2. The index based on cross-validation (V)*

This index is based on cross-validation [19], which is an old, standard tool in statistics [41,42]. The data are divided into two sets, one used for determining the clusters and the other one is used to validate the obtained clusters. The underlying idea here is to validate them on a dataset different from the one used for cluster estimation. For the task of image segmentation, the two subsets of data can be formed in several ways. One way, which we will follow, is to use an under-sampled version of the dataset for cluster estimation and the original dataset for validation.

## 4. Results and discussion

The experiments were performed with two types of data. The first type of data consists of two simple synthetic images (synthetic1 and synthetic2), one corrupted by 9%, 12% salt and pepper noise, and the other corrupted by Gaussian noise of standard deviation (ST) 50% and 60%; and the image size is $64 \times 64$ pixels, as shown in Fig. 1a and b, respectively. The second type of data includes T1-weighted 3D MRI brain data with slice thickness of 1 mm, corrupted by 3% and 6% noise, and no intensity inhomogeneities [43]. The image size is $129 \times 129$ pixels obtained from the classical simulated brain database of McGill University [43]. Two slices drawn from the simulated MRI are shown in Fig. 1d and e. On other hand, we will point to the original data as clean data (0% noise). The quality of the segmentation algorithm is of vital importance to the segmentation process. The comparison score $S$ for each algorithm as proposed in [44] is defined as follows:

$$S = \left| \frac{A \cap A_{ref}}{A \cup A_{ref}} \right| \tag{47}$$

where $A$ represents the set of pixels belonging to a class as found by a particular method and $A_{ref}$ represents the reference cluster pixels.

*4.1. Kernelized fuzzy C-means algorithms*

To assess the capabilities of the validity indexes to accurately identify the number of clusters present in an image, both KFCM and SKFCM methods were implemented. Through our implementation, we set the following parameters: $\sigma = 150$ (GRBF kernel width),$\alpha = 0.5$, $m = 2$, $\varepsilon = 0.00001$, and $N_R = 0.5$ (a $2 \times 2$ window centered around each pixel, except the central pixel itself). Note that the correct number of clusters for synthetic1, and synthetic2 is 2 and 4 clusters respectively. For the 3D simulated data, the correct number of clusters is 10. The standard KFCM algorithms (using iterative process Eqs. (6) and (7)) are applied independently on each image The eighteen indexes are used to estimate the best cluster of each image. In case of SKFCM, the objective functions of Ahmed et al. [2] is used which always gives stable and good results. We use the iterative process in Eqs. (11) and (12), more discussion can be shown in [7]. The outcome of each index on the different test images is shown in Table 1.

*4.1.1. In the case of KFCM algorithm*

***Dataset1 (synthetic1)***: the KFCM clustering algorithm is applied to synthetic1 corrupted by 9%, 12% salt and pepper noise for the cluster number $C = 2$. By optimizing the validity functions, most of the indexes indicate that 2 is an optimal cluster number which actually matches the structure of the image set except *FS*, *PE*, *PC*, *SC PD* indexes in the original image and *PD*, *PBMF*, *CWB*, *FS*, *PC*, *PCAES* and *V* indexes for 9% noise. The *PD*, *PBMF* and *CWB* indexes gave the cluster number 3 for this dataset. However, *FS*, *PC*, *PCAES* and *V* consider that 5 is a good cluster number estimate. Furthermore, for 12% noise, *PA*, *FHV*, *I*, *AIC*, and *V* still achieve optimal clusters. The estimated cluster numbers by the validity indexes are shown in the synthetic1 column of Table 1.

***Dataset2 (synthetic2)***: the KFCM clustering algorithm is applied to synthetic2 image corrupted by 50% and 60%

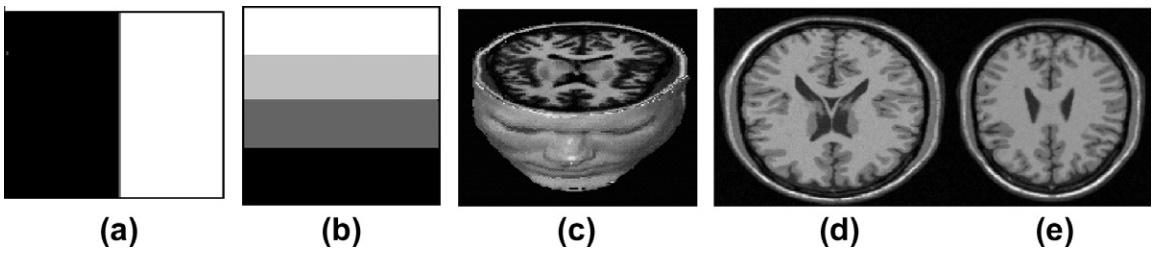


**Figure 1** Test images: (a) synthetic1, (b) synthetic2, (c) 3D simulated data, (d) and (e) two original slices from the 3D simulated data (slice91 and slice100).

**Table 1** Number of clusters obtained by 18 indexes using the KFCM and SKFCM algorithms.

| | KFCM | | | | | | | | | SKFCM | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Synthetic1 | | | Synthetic2 Gaussian noise ST | | | 3D simulated data | | | Synthetic1 | | | Synthetic2 Gaussian noise ST | | | 3D simulated data | | |
| | 0% noise | 9% noise | 12% noise | 0% | 50% | 60% | 0% noise | 3% noise | 6% noise | 0% noise | 9% noise | 12% noise | 0% | 50% | 60% | 0% noise | 3% noise | 6% noise |
| | Obtained number of clusters | | | | | | | | | Obtained number of clusters | | | | | | | | |
| *PD* | 2 | 3 | 4 | 3 | 2 | 2 | 7 | 10 | 9 | 2 | 2 | 2 | 4 | 2 | 2 | 7 | 2 | 9 |
| *PA* | 2 | 2 | 2 | 3 | 2 | 2 | 10 | 2 | 2 | 2 | 2 | 5 | 4 | 2 | 2 | 10 | 2 | 8 |
| *FHV* | 2 | 2 | 2 | 4 | 4 | 4 | 10 | 10 | 9 | 2 | 2 | 2 | 4 | 4 | 4 | 10 | 10 | 10 |
| *FS* | 4 | 5 | 5 | 2 | 2 | 3 | 8 | 6 | 8 | 3 | 4 | 5 | 2 | 7 | 6 | 10 | 10 | 8 |
| *PE* | 3 | 2 | 2 | 2 | 3 | 5 | 5 | 2 | 6 | 2 | 5 | 2 | 2 | 7 | 5 | 10 | 10 | 6 |
| *PC* | 3 | 5 | 4 | 4 | 3 | 3 | 8 | 2 | 7 | 2 | 5 | 4 | 4 | 7 | 5 | 9 | 10 | 8 |
| *S* | 2 | 2 | 3 | 4 | 2 | 2 | 5 | 2 | 2 | 2 | 2 | 3 | 4 | 2 | 2 | 10 | 2 | 2 |
| *XB* | 2 | 2 | 6 | 3 | 2 | 3 | 10 | 2 | 2 | 2 | 2 | 5 | 4 | 2 | 3 | 7 | 2 | 4 |
| *DB* | 9 | 2 | 3 | 4 | 7 | 5 | 8 | 10 | 9 | 2 | 4 | 6 | 4 | 5 | 5 | 10 | 10 | 10 |
| *I* | 9 | 2 | 2 | 4 | 4 | 4 | 10 | 10 | 10 | 2 | 2 | 2 | 4 | 4 | 4 | 10 | 10 | 10 |
| *VM* | 2 | 2 | 5 | 4 | 2 | 2 | 5 | 2 | 6 | 2 | 5 | 5 | 4 | 7 | 2 | 7 | 8 | 6 |
| *AIC* | 2 | 2 | 2 | 4 | 4 | 4 | 10 | 10 | 10 | 2 | 2 | 2 | 4 | 4 | 4 | 10 | 10 | 10 |
| *MPC* | 2 | 4 | 4 | 5 | 7 | 7 | 9 | 6 | 6 | 3 | 5 | 4 | 3 | 7 | 7 | 13 | 10 | 6 |
| *SC* | 3 | 2 | 3 | 4 | 2 | 2 | 10 | 2 | 2 | 2 | 2 | 3 | 4 | 2 | 2 | 5 | 2 | 6 |
| *PCAES* | 2 | 5 | 5 | 4 | 5 | 5 | 10 | 10 | 9 | 2 | 5 | 5 | 4 | 7 | 5 | 10 | 10 | 9 |
| *PBMF* | 2 | 2 | 2 | 4 | 4 | 4 | 9 | 10 | 9 | 2 | 2 | 2 | 4 | 3 | 4 | 10 | 7 | 10 |
| *CWB* | 2 | 3 | 5 | 4 | 3 | 2 | 9 | 3 | 4 | 2 | 4 | 5 | 4 | 6 | 2 | 7 | 4 | 4 |
| *V* | 2 | 5 | 2 | 4 | 6 | 7 | 10 | 9 | 9 | 2 | 5 | 4 | 4 | 7 | 7 | 10 | 10 | 9 |

Gaussian noise of ST. Intuitively, 4 clusters are suitable for the data set. The estimated cluster numbers by the validity indexes are shown on the synthetic2 column of Table 1. By optimizing the validity functions, most of the indexes gave optimal cluster number except *PD*, *PA*, *FS*, *PE*, *XB*, and *MPC* for the original image. For the noisy image of ST 50%, *FHV*, *I*, *AIC*, and *PBMF* indicate that 4 is an optimal cluster number which actually matches the structure of the image set. *PD*, *PA*, *FS*, *S*, *XB*, *VM*, and *SC* indicate that 2 is the best cluster number estimate. *DB* and *MPC* indexes gave the optimal cluster number 7. *PCAES* indicates that 5 is the best cluster number estimate. *V* and (*PE*, *PC*, *CWB*) indexes gave the optimal cluster number 6 and 3, respectively. According to the index values shown in Table 1, only *FHV*, *I*, and *AIC* indexes indicate that 4 is the cluster number estimate for the noise of ST up to 60%.

***Dataset3 (simulated volumetric MR data)***: we tested the efficiency of the validity indexes for a T1-weighted MR data with 3% and 6% noise respectively. As shown in Table 1, *PA*, *FHV*, *XB*, *I*, *AIC*, *SC*, *PCAES*, and *V* indicate that 10 is an optimal cluster number for the original image set. For 3% noise, *PD*, *FHV*, *I*, *DB*, *AIC*, *PCAES*, *PBMF*, and *V* indexes gave the optimal cluster. However only *I*, *AIC*, and *PBMF* gave optimal results for this dataset with 6% noise while *PD*, *DB, PCAES*, and *V* gave 9 clusters for this dataset. Others indexes achieved inconsistent results.

Overall, the *AIC*, *FHV*, and *I* indexes gave the optimal number of clusters 2, 4, and 10 for the three test datasets with low noise level which actually matches the structure of the images.

#### 4.1.2. In the case of SKFCM algorithm

***Dataset1 (synthetic1)***: the SKFCM algorithm applied to synthetic1 image corrupted by 9% and 12% salt and pepper noise. By optimizing the validity functions, most of the indexes indicate that 2 are optimal clusters for this dataset which actually matches the structure of the image except: *FS* and *MPC* for original image and *FS*, *PE*, *PC*, *DB*, *VM*, *MPC*, *PCAES*, *CWB* and *V* for 9% noise. However, *PE*, *PC*, *VM*, *MPC*, *PCAES* and *V* considered that 5 be the optimal cluster number. For 12% noise dataset, only *PD*, *FHV*, *PE*, *I*, *AIC*, and *PBMF* gave the actual clusters as shown in synthetic1 of Table 1.

***Dataset2 (synthetic2)***: the SKFCM algorithm is applied to synthetic2 image corrupted by Gaussian noise of ST 50%, 60% respectively. The estimated cluster numbers by the validity indexes are shown in the synthetic2 of Table 1. Most of indexes indicate that 4 are optimal clusters for original dataset except: *FS*, *PE*, and *MPC* indexes. *FHV*, *DB*, *AIC* and *I* indexes gave the optimal cluster number 4 clusters for this dataset which actually matches the structure of the image. But *PD*, *PA*, *XB*, *SC* gave 2 clusters, *PBMF* gave three clusters, *FS*, *PE*, *PC*, *VM*, *MPC*, *PCEAS*, *CWB*, and *V* indexes indicate that 7 is the optimal cluster number. In the case of dataset with noise 60% of ST, only *FHV*, *I*, *AIC*, and *PBMF* gave the actual cluster number.

***Dataset3 (simulated volumetric MR data)***: we tested the efficiency of the validity indexes for a simulated volumetric MR data (with 3% and 6% noise). As shown in Table 1, most of indexes gave optimal cluster number except *PD*, *PC*, *XB*, *VM*, *SC*, and *CWB* for original image. The *FS*, *PE*, *PC*, *DB*, *AIC*, *I*, *MPC*, *PCAES*, and *V* indexes indicate that 10 is the optimal cluster number for 3% noise dataset which actually matches the structure of the image. The *PD*, *PA*, *S*, *XB*, and *SC* indexes considered that 2 is the optimal cluster number. However, *FHV* and *VM* indexes considered 8 and 9 are the best cluster number. For 6% noise of dataset, only *DB*, *AIC*, *I*, *PBMF*, *FHV* indexes gave the actual cluster number. But *PB*, *PCAES*, and *V* indicate that 9 is best cluster number.

### 4.2. Different noise levels investigation

The performance of each index against noise is evaluated. Figs. 2–8 depict the relationship between the number of clusters and various levels of noise for all indexes when KFCM is applied

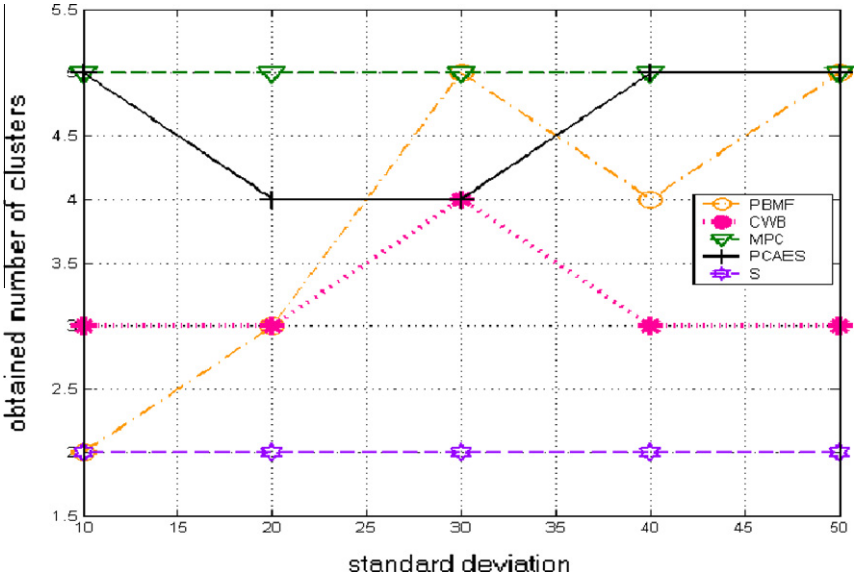


**Figure 2** The relationship between number of clusters and noise level for *PBMF*, *CWB*, *MPC*, *PCAES* and *S* indexes when the KFCM is applied to the synthetic1 image.

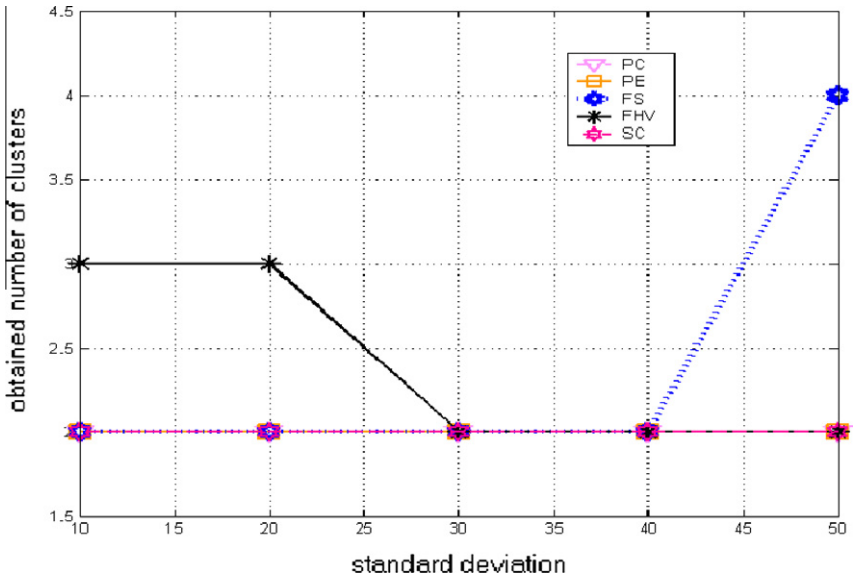


**Figure 3** The relationship between number of clusters and noise level for *PC*, *PE*, *FS*, *FHV* and *SC* indexes when the KFCM is applied to the synthetic1 image.

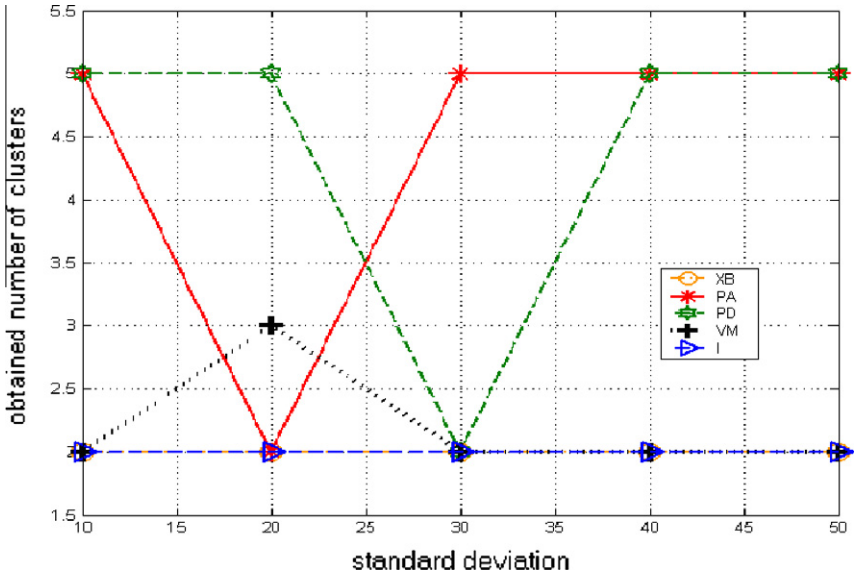


**Figure 4** The relationship between number of clusters and noise level for *XB*, *PA*, *PD*, *VM* and *I* indexes when the KFCM is applied to the synthetic1 image.

to the two synthetic images. It is clear that when *MPC*, *XB*, *AIC* and *SC* indexes are used, the optimal number of cluster is constant for different level of noise less than 12%, while for the *CWB* index this relationship is unstable. For *FS*, as the level noise increases, the obtained number of clusters increases. The *FHV*, *I* and *PBMF* indexes give inconsistent behavior on the synthetic2 and synthetic1, respectively.

Figs. 9–17 show the relationship between the number of clusters and noise level for all indexes when applying the SKFCM algorithm to the synthetic images. When *MPC*, *S*, *PC*, *XB*, *PA*, *SC*, *V* and *AIC* indexes are used, the obtained number of clusters is constant for various levels of noise, whereas this relationship is unstable for *CWB* and *PBMF* indexes. The *FHV*, *I* indexes seem to be working better than the others for the KFCM algorithm. However, it has shown tendency to be affected by noise. On the other hand, for the SKFCM algorithm *PBMF* index seems to be working better than the others. Things look discouraging as no index has shown optimum performance throughout all noise levels (using fuzzy KFCM and SKFCM cases). It is important to note that some indexes, such as the *AIC*, *I* and *FHV* indexes have demonstrated better performance with the fuzzy KFCM and SKFCM algorithms rather than others. Overall, *AIC*, *FHV*, and *I* present good stable cluster number at various

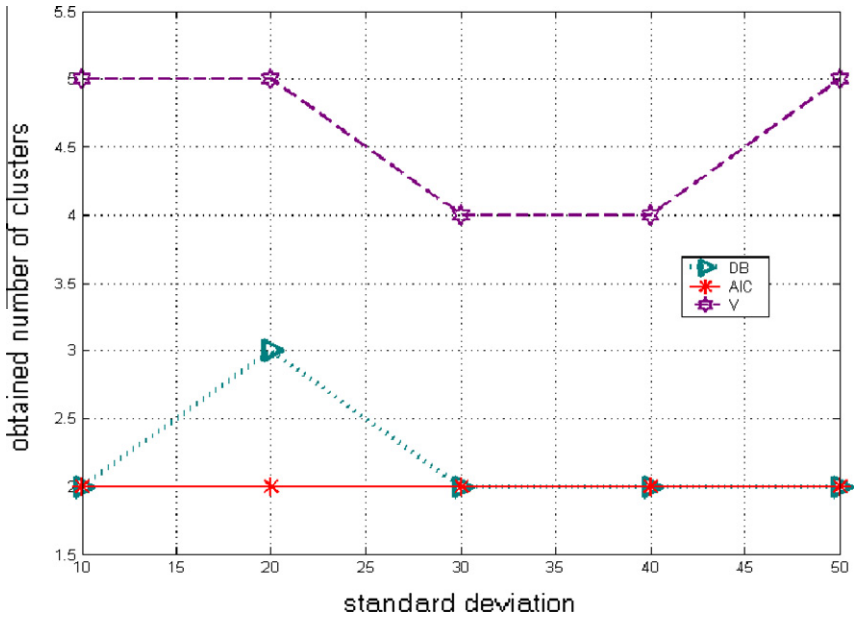


**Figure 5** The relationship between number of clusters and noise level for *AIC*, *DB* and *V* indexes when the KFCM is applied to the synthetic1 image.

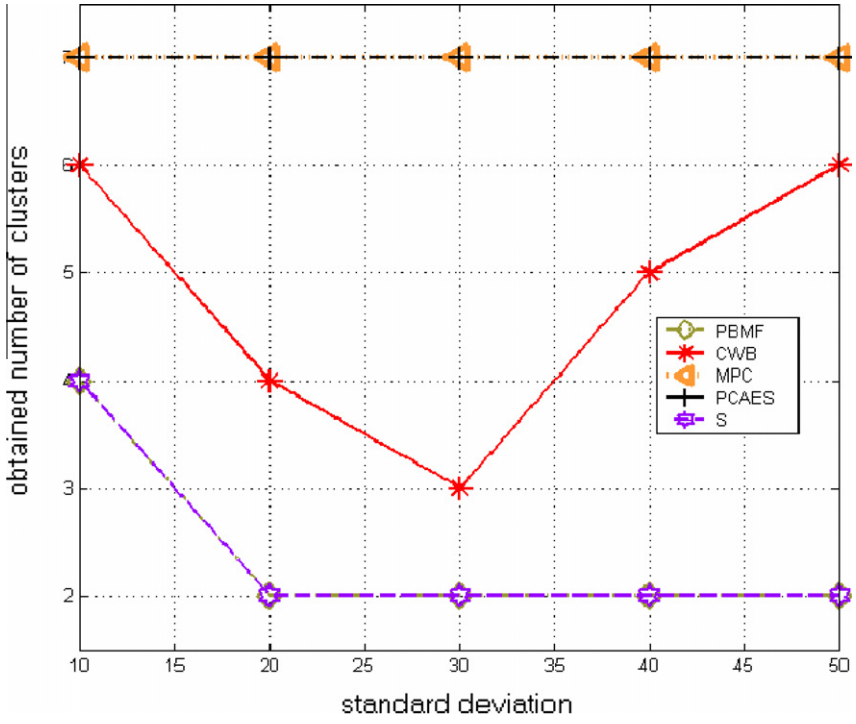


**Figure 6** The relationship between number of clusters and noise level for *PBMF*, *CWB*, *MPC*, *PCAES* and *S* indexes when the KFCM is applied to the synthetic2 image.

levels of noise, especially *AIC* gives the optimal cluster number in all our tests in case of noisy and non noisy data sets.

On the other hand, in case of KFCM, one can read from Table 1 that although many indexes give accurate results on the 3D volume, only the *AIC*, *FHV*, and *I* yield correct or almost correct results on different levels of noises.

Analogously, on applying KFCM algorithm to noisy synthetic images, the corresponding relationships between the found number of clusters and noise standard deviation have revealed that most indexes have constant outcomes for various levels of noise, whereas this relationship is unstable for *CWB*, *PCAES*, *FS*, and *PC* indexes.

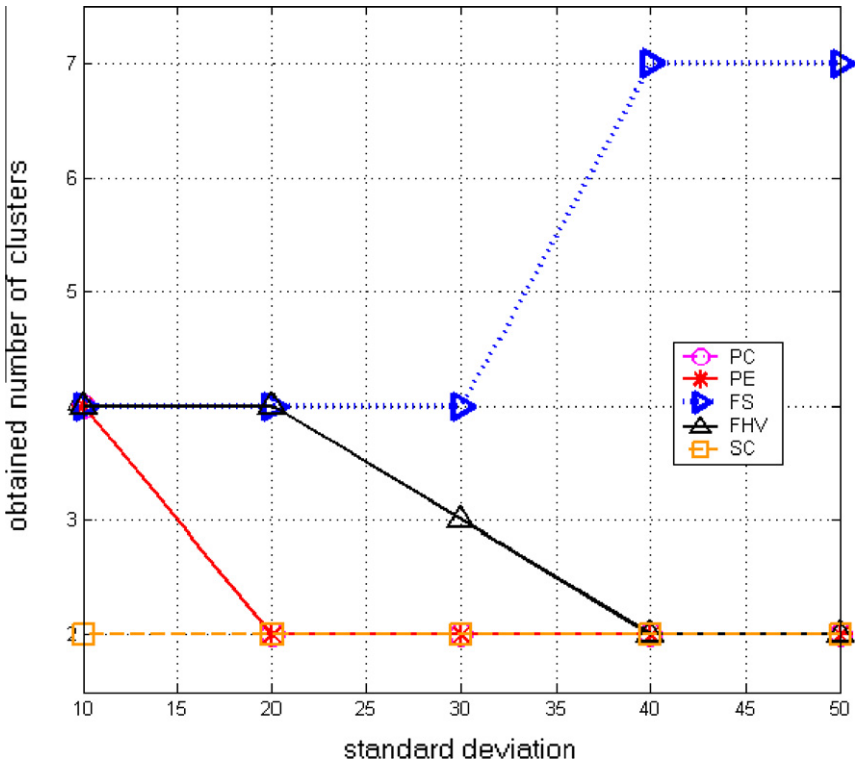


**Figure 7** The relationship between number of clusters and noise level for *PC*, *PE*, *FS*, *FHV* and *SC* indexes when the KFCM is applied to the synthetic2 image.

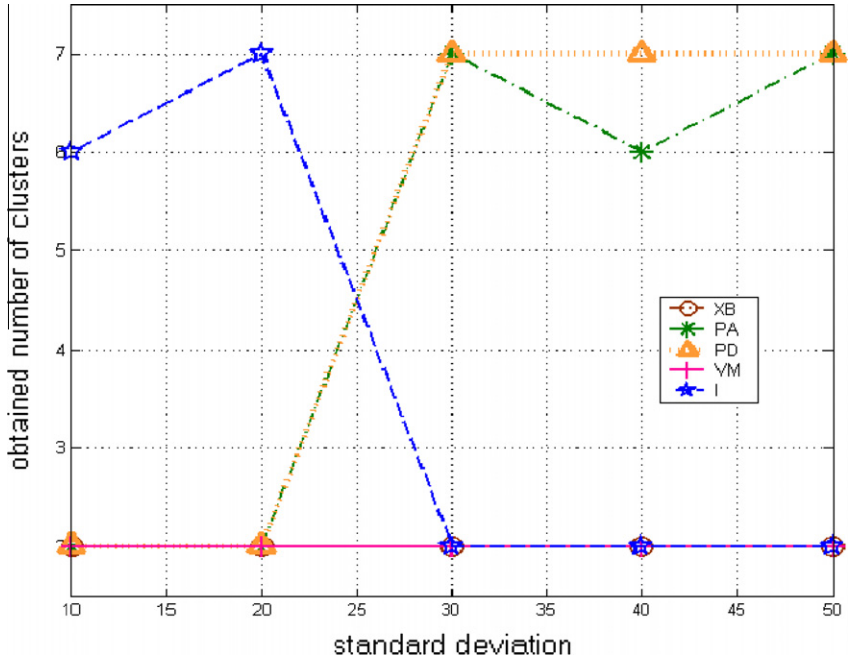


**Figure 8** The relationship between number of clusters and noise level for *XB*, *PA*, *PD*, *VM* and *I* indexes when the KFCM is applied to the synthetic2 image.

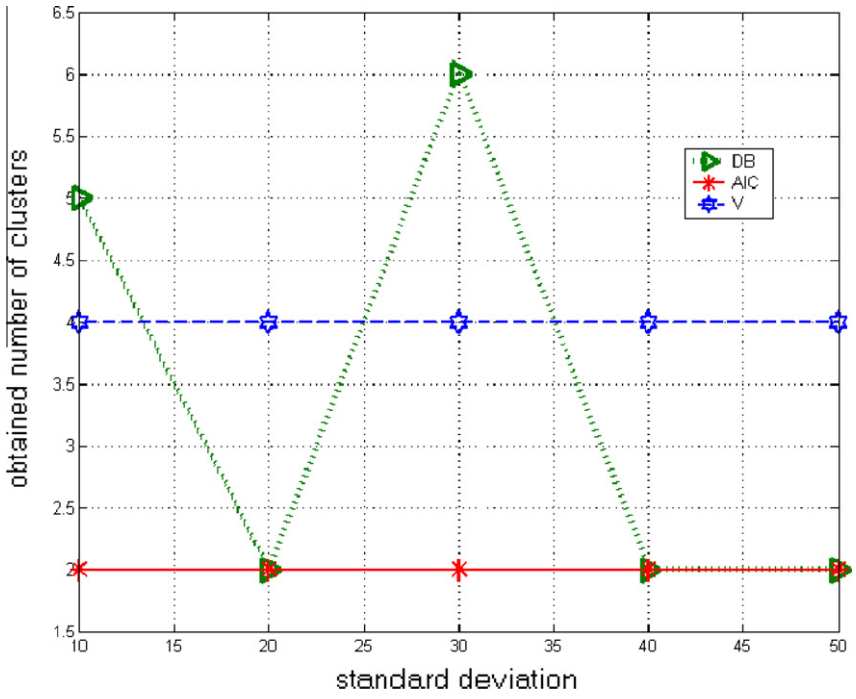


**Figure 9** The relationship between number of clusters and noise level for *AIC*, *DB* and *V* indexes when the KFCM is applied to the synthetic2 image.

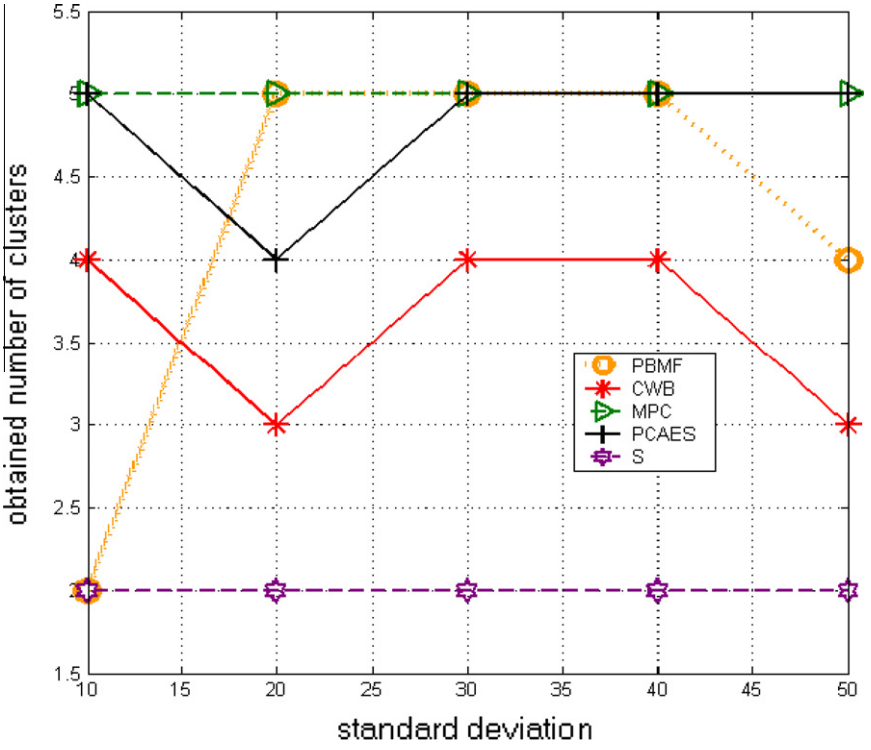


**Figure 10** The relationship between number of clusters and noise level for *PBMF*, *CWB*, *MPC*, *PCAES* and *S* index es when the SKFCM is applied to the synthetic1 image.

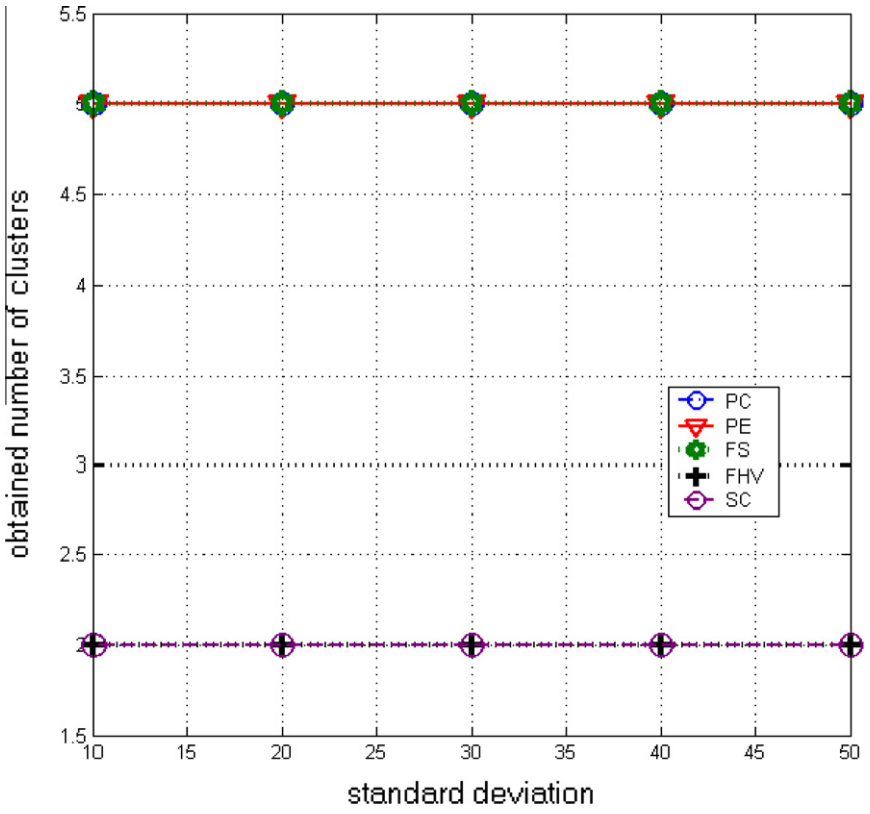


**Figure 11** The relationship between number of clusters and noise level for *PC*, *PE*, *FS*, *FHV* and *SC* indexes when the SKFCM is applied to the synthetic1 image.

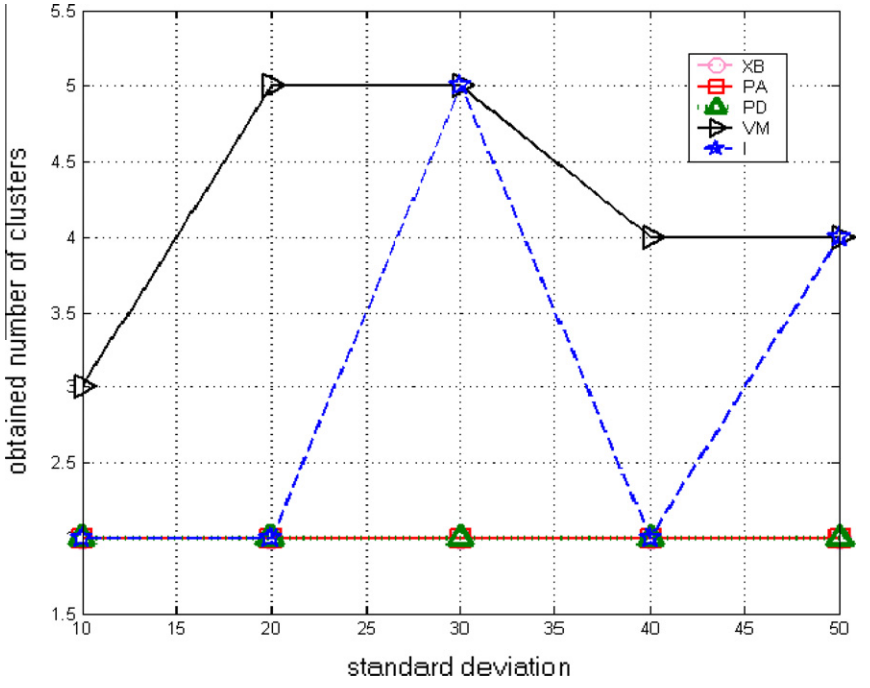


**Figure 12** The relationship between number of clusters and noise level for *XB*, *PA*, *PD*, *VM* and *I* indexes when the SKFCM is applied to the synthetic1 image.

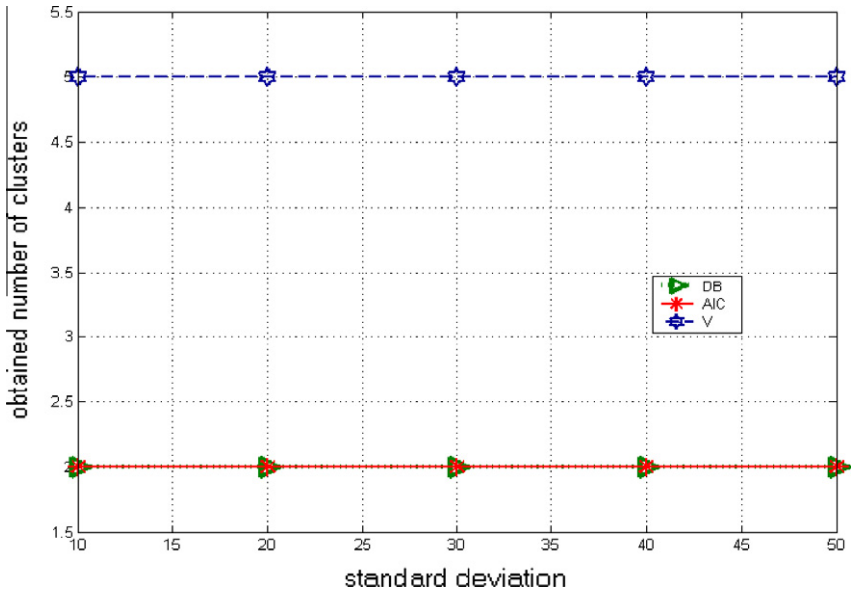


**Figure 13** The relationship between number of clusters and noise level for *AIC*, *DB* and *V* indexes when the SKFCM is applied to the synthetic1 image.

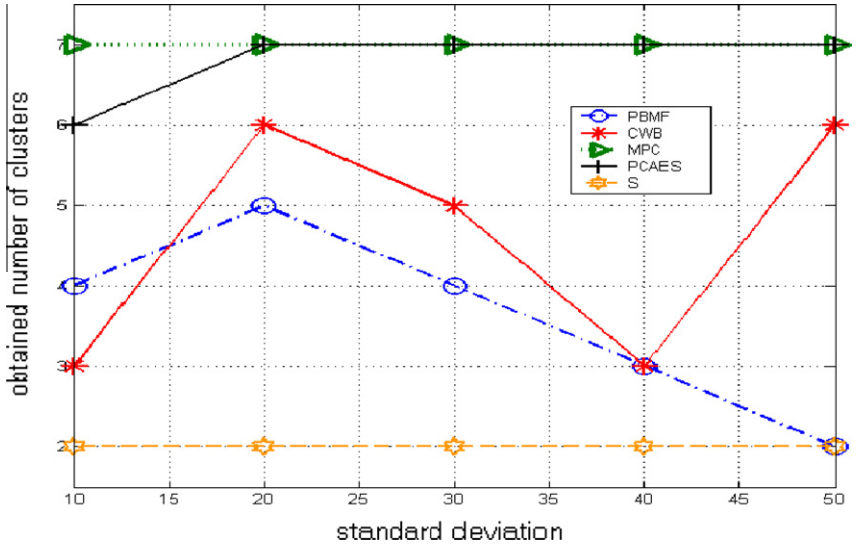


**Figure 14** The relationship between number of clusters and noise level for *PBMF*, *CWB*, *MPC*, *PCAES* and *S* indexes when the SKFCM is applied to the synthetic2 image.

### *4.3. Segmentation accuracy*

In the previous section, we noted that *I*, *FHV*, and *AIC* indexes always give better results than others. Our tests are focused on applying the standard FCM and most popular SKFCM such as: Ahmed et al. [2], Zhang et al. [6], and Kang et al. [7] with these indexes on T1-weighted MR phantom with nine slices thickness of 1 mm, 3% noise. We set the parameter $\sigma = 150$ (GRBF kernel width),$\alpha = 0.5$, $m = 2$, and $N_R = 0.5$. Table 2 shows the corresponding accuracy scores of the four methods: standard FCM and SKFCM of Ahmed et al. [2], Zhang et al. [6], and Kang et al. [7] for the nine classes. Obviously, the standard KFCM gives the worst segmentation accuracy in case *I* and *FHV* indexes, because *I* and *FHV* failed to determine the true clusters number of slice 8, while all methods with *AIC* data give satisfactory results. On the other hand, the SKFCM of Ahmed et al. [2] and Kang et al. [7] acquire the best segmentation performance in case of *I* and

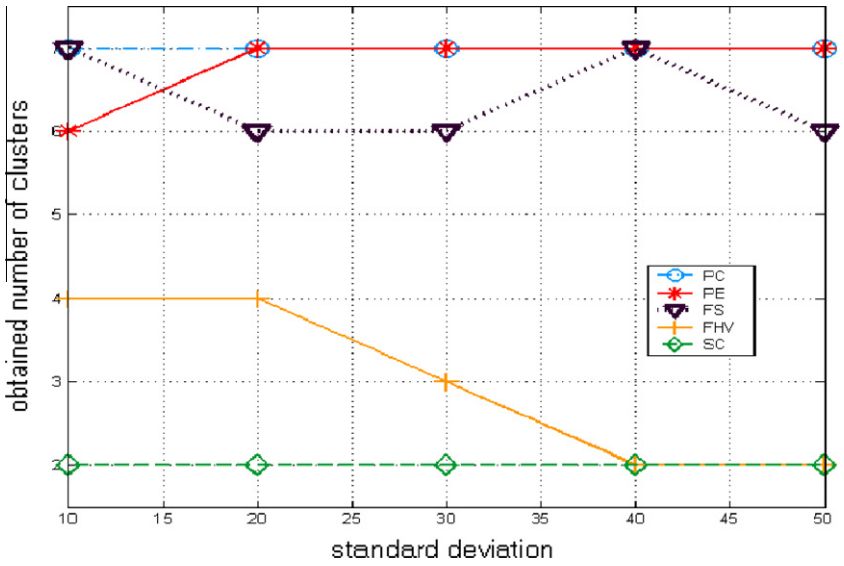


**Figure 15** The relationship between number of clusters and noise level for *PC*, *PE*, *FS*, *FHV* and *SC* indexes when the SKFCM is applied to the synthetic2 image.

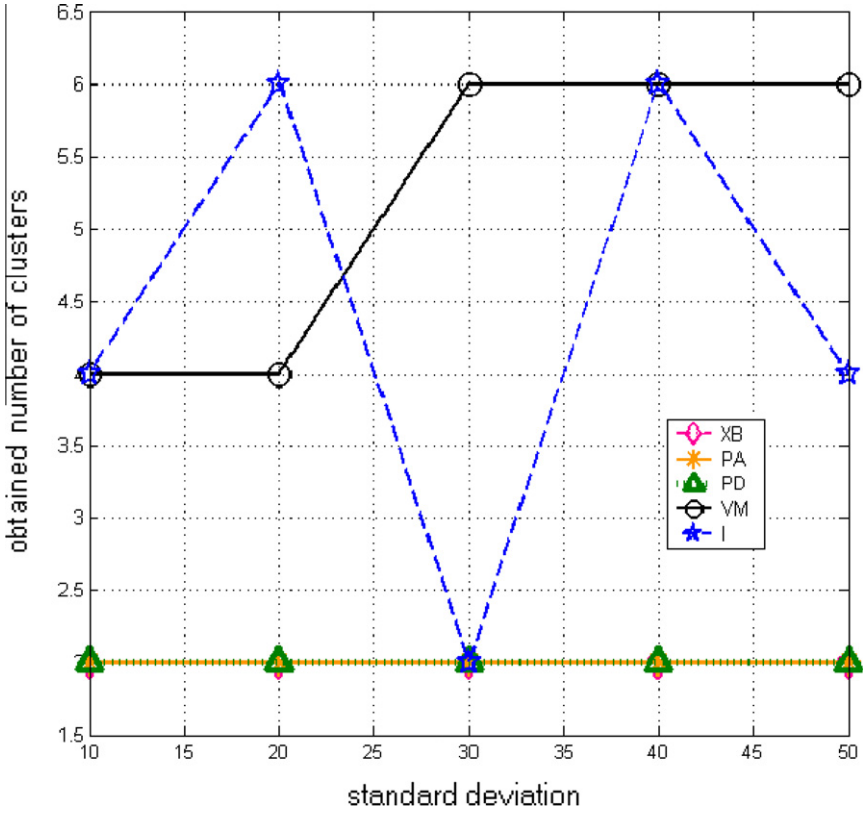


**Figure 16** The relationship between number of clusters and noise level for *XB*, *PA*, *PD*, *VM* and *I* indexes when the SKFCM is applied to the synthetic2 image.

*FHV* respectively. Overall, the SKFCM of Kang [7] with *AIC* index is more stable and achieves much better performance than the others in different classes even with misleading of true tissue of validity indexes.

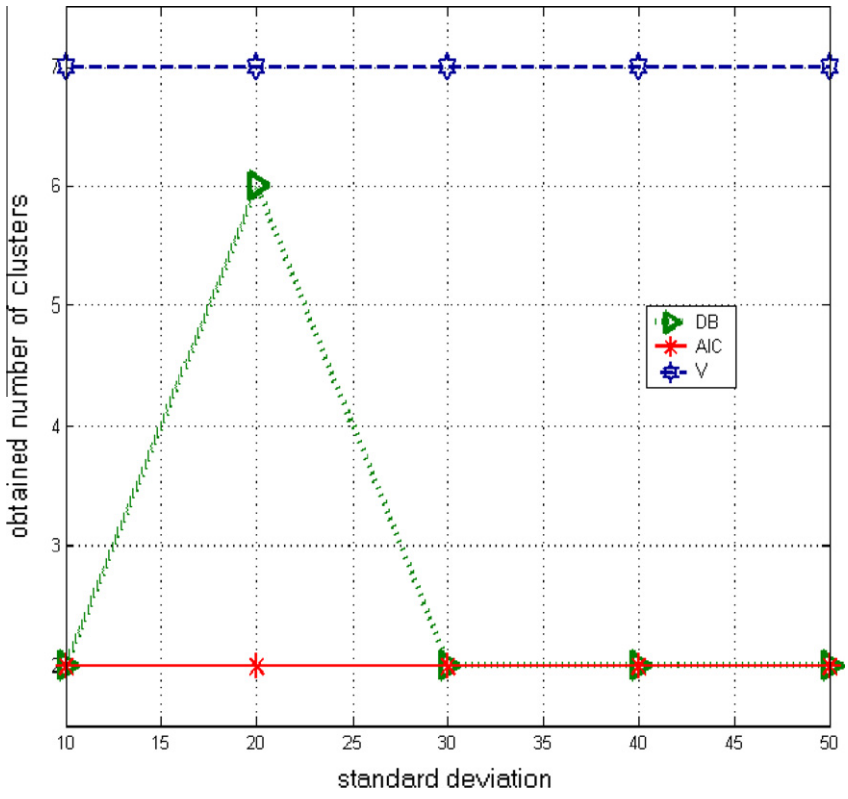


**Figure 17** The relationship between number of clusters and noise level for *AIC*, *DB* and *V* indexes when the SKFCM is applied to the synthetic2 image.

**Table 2** Segmentation accuracy (%) of four methods on brain classes.

| Index | Method | Class 1 | Class 2 | Class 3 | Class 4 | Class 5 | Class 6 | Class 7 | Class 8 | Class 9 | Overall (%) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| I | Standard KFCM | 66.87 | 55.77 | 59.087 | 64.0 | 70.32 | 37.96 | 63.99 | 10.12 | 97.99 | 58.456 |
| | Ahmed SKFCM [3] | 67.55 | 61.14 | 78.83 | 73.88 | 67.96 | 61.87 | 89.21 | 15.27 | 97.27 | 68.108 |
| | Kang SKFCM [8] | 79.54 | 77.55 | 78.34 | 82.01 | 78.65 | 81.98 | 8 | 18.54 | 99.54 | 74.98 |
| | Zhang SKFCM [7] | 56.87 | 60.43 | 66.98 | 70.54 | 76.09 | 45.98 | 66.87 | 16.43 | 96.09 | 61.808 |
| *FHV* | Standard KFCM | 67.55 | 51.14 | 67.83 | 91.03 | 67.96 | 21.87 | 59.21 | 11.27 | 97.26 | 59.457 |
| | Ahmed SKFCM [3] | 75.46 | 71.88 | 99.98 | 82.21 | 96.63 | 82.31 | 55.70 | 1.50 | 96.82 | 73.611 |
| | Kang SKFCM [8] | 78.09 | 74.65 | 73.87 | 83.65 | 98.32 | 90.34 | 65.76 | 20.9 | 98.98 | 76.062 |
| | Zhang SKFCM [7] | 76.98 | 68.32 | 77.65 | 88.23 | 89.43 | 95.21 | 66.98 | 10.43 | 88.54 | 73.532 |
| *AIC* | Standard KFCM | 53.52 | 64.38 | 75.19 | 89.3 | 62.76 | 29.09 | 83.09 | 42.76 | 98.95 | 66.563 |
| | Ahmed SKFCM [3] | 64.92 | 87.64 | 77.84 | 86.18 | 66.17 | 89.18 | 99.95 | 56.3 | 99.03 | 80.801 |
| | Kang SKFCM [8] | 77.98 | 86.72 | 89.54 | 88.34 | 85.12 | 77.02 | 60.0 | 76.89 | 100.0 | 82.401 |
| | Zhang SKFCM [7] | 77.65 | 85.12 | 87.23 | 90.87 | 81.54 | 74.05 | 55.21 | 64.25 | 93.76 | 78.85 |

## 5. Conclusion

In clustering, the role of a validity index is very important. The hope is that the number of clusters within an image can be determined automatically. The aim of this paper was to consider the performance of 18 of the most popular indexes by applying them in turn to a range of simulated and real data sets, including 2D and 3D data sets, corrupted with different types of noise of varying levels. To the best of our knowledge, no such comprehensive survey and comparison has been reported before in literature.

From the results of the experimentation, it is not possible to identify definitively an index which will work well in all cases, and hence be the most suitable as a general index for all cases. However some cluster validity indexes can guide the selection of the appropriate number of clusters existing in a dataset. We have arranged these indexes into three categories. In the first category: *AIC*, *FHV*, and *I* indexes appear to be good general indexes and have exhibited the best overall performance in all the experiments, outperforming all other indexes. The *PBMF*, *MPC*, *XB*, *PD*, and *DB* indexes have shown acceptable results, but have shown unstable performance on all test

images under varying noise levels. The third category: *V*, *S*, *SC*, *PC*, *PCAES*, *CWB*, *VM*, *FS*, *PA*, and *PE* gave incorrect results in all test images.

Overall, on synthetic images, *AIC*, *I* and *FHV* indexes yield the true number of clusters, whereas the *FHV* index shows better performance in KFCM compared SKFCM. Furthermore, the tests strongly suggest that the *AIC* index should be considered as the most robust index for general data and in particular, for determining the correct number of clusters using KFCM and SKFCM for MR medical images. Moreover, our tests prove that one can confide *AIC* method for determining the correct number of clusters using KFCM and SKFCM for MR medical images. The most recent SKFCM with *AIC* has obtained good segmentation performance (i.e. up to 90% in synthetic images and up to 80% for MRI images).

This initial investigation should be expanded to consider further testing on different real data sets from a wide range of applications including medical images and reverse engineered data. It would also be interesting to consider the effect of systematic and random error noise levels within the data sets to further establish the effect of this error on the performance of the indexes. Further tests should also be carried out on clustering improvement of the fuzzy methods using a wide range of validity indexes for automatic clustering algorithms, via the help of specialists within the fields of application.

## References


[1] Bezdek JC. Pattern recognition with fuzzy objective function algorithms. New York: Plenum Press; 1981.

[2] Ahmed MN, Yamany SM, Mohamed N, Farag AA, Moriarty T. A modified fuzzy C-means algorithm for bias field estimation and segmentation of MRI data. IEEE Trans Med Imag 2002;21:193–9.

[3] Maulik U, Bandyopadhyay S. Fuzzy partitioning using a real coded variable length genetic algorithm for pixel classification. IEEE Trans Geosci Remote Sens 2003;41(5):1075–81.

[4] Wu S, Liew AWC, Yan H. Cluster analysis of gene expression data based on self-splitting and merging competitive learning. IEEE Trans Inform Technol Biomed 2004;8:5–15.

[5] Zhu L, Chung FL, Wang S. Generalized fuzzy C-means clustering algorithm with improved fuzzy partitions. IEEE Trans Syst Man Cybernet: Part B 2009;39(3):578–91.

[6] Zhang Dao-Qiang, Chen Song-Can. A novel kernelized fuzzy C-means algorithm with application in medical image segmentation. Artif Intell Med 2004;32:37–50.

[7] Kang Jiayin, Min Lequan, Luan Qingxian, Li Xiao, Liu Jinzhu. Novel modified fuzzy C-means algorithm with applications. Digital Signal Process 2009;19:309–19.

[8] Kim DW, Lee KY, Lee D, Lee KH. A kernel-based subtractive clustering method. Pattern Recog Lett 2005;26(7):879–91.

[9] Zanaty EA, Aljahdali Sultan, Debnath Narayan. A kernelized fuzzy C-means algorithm for automatic magnetic resonance image segmentation. J Comput Methods Sci Eng (JCMSE) 2009: 123–36.

[10] Timm H, Borgelt C, Doring C, Kruse R. An extension to possibilistic fuzzy cluster analysis. Fuzzy Sets Syst 2004;147(1): 3–16.

[11] Zhang JS, Leung YW. Improved possibilistic c-means clustering algorithms. IEEE Trans Fuzzy Syst 2004;12(2):209–17.

[12] Ji Ze-Xuan, Sun Quan-Sen, Xia De-Shen. A modified possibilistic fuzzy c-means clustering algorithm for biasfield estimation and segmentation of brain MR image. Comput Med Imaging Graphics 2011;35(5):383–97.

[13] Yuhua G, Lawrence OH. Kernel based fuzzy ant clustering with partition validity. In: IEEE international conference on fuzzy systems. Vancouver (BC, Canada): Sheraton Vancouver Wall Centre Hotel; 2006. p. 16–21.

[14] Wang W, Zhang Y. On fuzzy cluster validity indices. Fuzzy Sets Syst 2007;158:2095–117.

[15] Zhang Yunjie, Wang Weina, Zhang Xiaona, Lic Yi. A cluster validity index for fuzzy clustering. Inform Sci 2008;178:1205–18.

[16] Zanaty EA. A new modified fuzzy c-means for segmenting magnetic resonance images (MRIS). IJICIS J 2011;11(2):209–22.

[17] Jegatha Deborah L, Baskaran R, Kannan A. Survey on internal validity measure for cluster validation. Int J Comput Sci Eng Survey (IJCSES) 2010;1(2).

[18] Alp Erilli N, Yolcu Ufuk, Eğrioğlu Erol, Hakan Aladag Ç, Öner Yüksel. Determining the most proper number of cluster in fuzzy clustering by using artificial neural networks. Expert Syst Appl 2011;38:2248–52.

[19] El-Melegy MT, Zanaty EA, Abd-Elhafiez Walaa M, Farag Aly. On cluster validity indexes in fuzzy and hard clustering algorithms for image segmentation. In: IEEE international conference on computer vision, vol. 6; 2007. p. VI 5–8.

[20] Xu Y, Richard G, Brereton A. A comparative study of cluster validation indices applied to genotyping data. Chemomet Intell Lab Syst 2005;78:30–40.

[21] Pakhira MK, Bandyopadhyay S, Maulic U. Validity index for crisp and fuzzy clusters. Pattern Recog 2004;37:487–501.

[22] Nasser H, Sweilam AA, Tharwat NK, Moniem Abdel. Support vector machine for diagnosis cancer disease: a comparative study. Egypt Inform J 2011;11(2):81–92.

[23] Girolami M. Mercer kernel-based clustering in feature space. IEEE Trans Neural Networks 2002;13(3):780–4.

[24] Zhang DQ, Chen SC. Fuzzy clustering using kernel methods. In: Proceedings of the international conference on control and automation, Xiamen, China, June 2002.

[25] Zanaty EA, Afifi Ashraf. Support vector machine with universal kernels. Int J Appl Artif Intell 2011;2(5).

[26] Zhang D-Q, Chen S-C. Clustering incomplete data using kernel-based fuzzy C-means algorithm. Neural Process Lett 2003;18(3):155–62.

[27] Bezdeck JC, Ehrlich R, Full W. FCM: fuzzy c-means algorithm. Comput Geosci 1984.

[28] Dave RN. Validating fuzzy partition obtained through c-shells clustering. Pattern Recog Lett 1996;17:613–23.

[29] Xie XL, Beni G. A validity measure for fuzzy clustering. IEEE Trans Pattern Anal Mach Intell 1991;13:841–7.

[30] Carl GL. A fuzzy clustering and fuzzy merging algorithm. Technical report CS-UNR-101; 1999.

[31] Maulik U, Bandyopadhyay S. Performance evaluation of some clustering algorithms and validity indices. IEEE Trans Pattern Anal Mach Intell 2002;24(12).

[32] Davies DL, Bouldin DW. A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 1979;1:224–7.

[33] Liao T, Compsc B. Image segmentation-hybrid method combining clustering and region merging. Thesis. Monash University; 2003.

[34] Fukuyama Y, Sugeno M. A new method of choosing the number of clusters for the fuzzy C-means method. In: Proceeding of fifth fuzzy system symposium; 1989. p. 247–50.

[35] Gath Geva AB. Unsupervised optimal fuzzy clustering. IEEE Trans Pattern Anal Mach Intell 1989;11:773–81.

[36] Zahid N, Limouri M, Essaid A. A new cluster-validity for fuzzy clustering. Pattern Recog 1999;32:1089–97.

[37] Wu KL, Yang MS. A cluster validity index for fuzzy clustering. Pattern Recog Lett 2005;26:1275–91.

[38] Malay KP, Sanghamitra B, Ujjwal M. A study of some fuzzy cluster validity indices, genetic clustering and application to pixel classification. Fuzzy Sets Syst 2005;155:191–214.

[39] Yun X, Brereton GRG. A comparative study of cluster validation indices applied to genotyping data. Chemomet Intell Lab Syst 2005;78:30–40.

[40] Andre T, Antonini M, Barlaud M, Gray RM. Entropy-based distortion measure and bit allocation for wavelet image compression. IEEE Trans Image Process 2007;16(12):3058–64.
[41] Saha S, Bandyopadhyay S. Application of a new symmetry-based cluster validity index for satellite image segmentation. IEEE Trans Pattern Anal Mach Intell 2008;5(2):166–70.
[42] Mezer Aviv, Yovel Yossi, Pasternak Ofer, Gorfine Tali, Assaf Yaniv. Cluster analysis of resting-state fMRI time series. NeuroImage 2009;45:1117–25.
[43] BrainWeb. Simulated brain database. McConnell Brain Imaging Centre, Montreal Neurological Institute, McGill University. <http://www.bic.mni.mcgill.ca/brainweb>.
[44] Masulli F, Schenone A. A fuzzy clustering based segmentation system as support to diagnosis in medical imaging. Artif Intell Med 1999;16(2):29–47.