Multimedia Tools and Applications
https://doi.org/10.1007/s11042-020-09975-3

Automatic fuzzy genetic algorithm in clustering for images based on the extracted intervals

Dinh Phamtoan[1,2,3] · **Tai Vovan**[4]

Received: 14 March 2020 / Revised: 22 August 2020 / Accepted: 24 September 2020 /
Published online: 13 October 2020

Abstract
This research proposes the method to extract the characteristics of images to become the intervals. These intervals are used to build the automatic fuzzy genetic algorithm for images (AFGI). In the proposed model, the overlap measure is the criterion to evaluate the closeness of intervals, and the new Davies and Bouldin index is the objective function. The AFGI can determine the proper number of clusters, the images in each cluster, and the probability to belong to clusters of images at the same time. The experiments with different types of images illustrate the steps of AFGI, and show its significant benefit in comparing to other algorithms.

Keywords Cluster analysis · Fuzzy genetic algorithm · Image processing · Interval data · Pattern recognition · Unsupervised learning

1 Introduction

In the information age, the problem of storage, extraction and recognition data are one of the big challenges to the scientists. For this problem, clustering technique has a basic role. Therefore, it is especially interested in many statisticians [3, 13, 35, 38, 42]. Building cluster is to divide a dataset into groups according to certain characteristics of the elements. Cluster analysis for discrete elements (CDE) was studied in the first time with many great contributions both theory and application [3, 4, 28, 35, 36, 39]. With the big and complex data such as images, each object needs to be considered as a distribution, clustering for the

✉ Tai Vovan
vvtai@ctu.edu.vn

Dinh Phamtoan
phamtoandinh@vanlanguni.edu.vn

1 University of Science, Ho Chi Minh City, Vietnam

2 Vietnam National University, Ho Chi Minh City, Vietnam

3 Faculty of Engineering, Van lang University, Ho Chi Minh City, Vietnam

4 College of Natural Science, Can Tho University, Can Tho, Vietnam

probability density functions (CDF) is proposed. In image recognition, CDF has given more benefit than CDE. The important results in the recent years for CDF were studied in [4, 35, 37, 38]. In the both CDE and CDF, statisticians have been used a lot of different measures as the criteria for clustering. Regarding CDF, the problem for finding the proper number of groups has been settled.

There are two kinds of cluster analysis: crisp and fuzzy clustering. In the crisp clustering, each element belongs to a cluster with probability as 1. Therefore, some boundary elements maybe not evaluated precisely. In the fuzzy clustering, each element can belong to one or more clusters with different levels of membership. The higher level of membership in one cluster is, the greater probability of element belonging to that cluster is. This shows the flexibility and advantages of the fuzzy clustering in comparison with the crisp cluster. In the both CDE and CDF, the fuzzy cluster analysis algorithms have been proposed [2, 22, 24, 41, 43].

In some areas, interval data is often recorded nowadays. Therefore, clustering model for interval data (CID) should be considered. This is also a research direction that is highly applicable in practice [19]. Although CID has only been studied in recent years, it has also received many interesting results. Some momentous researches for CDI are analyzed as follows. De Souza et al. [8] used the dynamic algorithm with the adaptive squared Euclidean distance to build clusters. This adaptive distance is parameterized according to the intra-class structure of the partition, and it is able to recognize clusters with different shapes and sizes. Peng and Li [27] studied the clustering algorithm using the modified dissimilarity measures for intervals. Additionally, they also argued several approaches to build groups, and gave the relations among them. DeCarvalho et al. [7] used the Euclidean distance between each element and the center of groups to build CDI. Sato-Ilic [32] developed the algorithm of DeCarvalho et al. [7] in building fuzzy cluster for intervals. With the Hausdorff distance, Hung et al. [16] proposed the method to find the relevant number of groups. For the new measure, overlap ratio, Kabir et al. [19] gave a method to build CID. Based on the fuzzy c-means algorithm for discrete elements, Jeng et al. [18] and Sara et al. [31] developed the fuzzy clustering algorithm for the intervals with city-block distance, Euclidean distance, and Hausdorff distance. However, these clustering algorithms required the prior knowledge of data. Currently, based on traditional genetic algorithm and the overlap distance, Tai et al. [40] has given the significant distributions in building CID. However, this is non-fuzzy algorithm. It means that the algorithm could not to determine the probability to belong to clusters of each element. Moreover, with the complex data such as images, these methods often obtained the bad results. To decrease error rate in cluster analysis, many statisticians have used the genetic algorithm (GA) because of its outstanding advantages. GA was firstly developed by Holland [17] to find the optimal solution for the algorithms. Some results of the GA for CDE and CDF have also introduced by [12, 20, 21, 25, 33, 37, 38]. Tai et al. [40] was firstly proposed GA for CID. However, it had disadvantage for the complex intervals.

In data processing, image is specially considered because it is visible, and applicable to numerous fields. For example, in agriculture, it is used to automatically classify potatoes, fruit [1] or detect fruit on tree [26]. Besides, in the environmental problem, it is used for detecting oil spills. In medicine, it is used for detecting breast cancer in woman [5]. In internet security, it is used to filter sexy images consisted of virus. There are a lot of methods to classify an image set into different groups. Almost these methods based on the extracted features from image such as colour, texture, co-occurrence matrix, etc. Extracting feature of images into the probability density functions, based on the colour to clustering studied by Tai and Thao [35, 37]. In addition, image is usually used in numerical examples, not

a major object. Moreover, in state-of-the-art studies, estimation pdfs are just performed in one-dimensional space. In particular for image object, this is popular and until now, estimation of pdfs from the image's colour in three-dimensional spaces has been still restricted. For example, although the work of Tai et al. [35] has discussed in clustering for pdfs, estimating pdfs from image is only executed in one-dimensional space, and it also an object in numerical example section. The studies related to texture or co-occurrence matrix are preferred. Setia et al. [34] used cluster co-occurrence matrices of local relational features to classify images. Eleyan et al. [9] introduced a new approach to recognize face based on the Gray level co-occurrence matrix. Almost of these researches have been based on discrete data [10, 11]. Tai et al. [40] currently has clustered for the images by the genetic algorithm based on the extracted intervals but it still has limitations in many cases.

In this article, we use the overlap measure (d_{OLID}) of two elements in one-dimension, and improve it in multi-dimension to evaluate the difference of the elements. Based on this measure, the DB index [6] of discrete elements, and the IDB index in [40], the New DB index (NDB) is given. This index is the objective function in the proposed algorithm. The AFGI also solves the problem to find the pertinent number of groups. In addition, the important contribution of this study is the clustering analysis for images from the AFGI. We use 40 features of Gray level co-occurrence matrix for each image, and illustrate it into typical intervals. After that, these intervals are used as input data to cluster for the images by the proposed algorithm. We can find the appropriate number of clusters, the elements in each cluster, and the probability to belong to clusters of each element at the same time.

The next section of the article is structured as follows. The defines the overlap measure in the one-dimensional and multi-dimensional cases are given in Section 2. This section also gives the indexes to evaluate the clustering result. This section also illustrates the method to extract the data of image to become the featured intervals. Section 3 proposes the automatic fuzzy genetic algorithm in clustering for images from the extracted data. Section 4 applies the AFGI for four image sets. The final section is the conclusion.

2 The overlap measure and the problem of extracting the characteristics for images

2.1 The overlap measure

Let $a = \left[\underline{a}, \overline{a}\right]$ and $b = \left[\underline{b}, \overline{b}\right]$ be two one-dimensional intervals. Set $m_a = \frac{\underline{a}+\overline{a}}{2}$, $v_a = \frac{\underline{a}-\overline{a}}{2}$, $m_b = \frac{\underline{b}+\overline{b}}{2}$, and $v_b = \frac{\underline{b}-\overline{b}}{2}$. The overlap measure between a and b is given by (1):

$$d_{OLID}(a,b) = m(a,b)\left(1 - \frac{R(a,b)}{2v_a + 1}\right), \tag{1}$$

where $R(a;b)$ is the overlap region between a and b, and $m(a,b)$ is determined by

$$m(a,b) = \max\{\min_{a' \in [\underline{a},\overline{a}]}\{\min_{b' \in [\underline{b},\overline{b}]} d_E(a',b')\}\},$$

with $d_E(a',b')$ is the Euclidean distance between a' and b'.

Considering two one - dimensional intervals a and b, we have 5 cases: b contains a, a contains b, b overlaps with a on the left side of a, b overlaps with a on the right side of a, b and a does not overlap area together: From these cases, Tai et al. [40] has detailed (1) to apply. For multi dimensions case, the overlap measure is defined as follows:

Definition 1 Let α and β be two p-dimensional intervals:

$$\alpha = \{\alpha_1, \alpha_2, ..., \alpha_p\},\ \ \beta = \{\beta_1, \beta_2, ..., \beta_p\}$$

Then, the overlap measure of α and β is defined by (2).

$$d_{OLID}(\alpha, \beta) = \sum_{i=1}^{p} \max\{d_{OLID}(\alpha_i, \beta_i), d_{OLID}(\beta_i, \alpha_i)\}. \tag{2}$$

Considering the common distance such as Hausdorff (d_H), City-block (d_C), Euclide (d_E), and Minkowski (d_M), we see that they only based on the lower bound and upper bound of intervals. The overlap area of intervals was not considered in these distances.

For two intervals, there are two factors which are to relate to their closeness or differences. They are centers and overlap area. d_{OLID} defined by (1) and (2) has considered these two factors, so it overcomes d_E, d_C, d_H in measuring the relationship of intervals. For example, given 8 intervals $a_1 = [5, 7]$, $a_2 = [5, 8]$, $a_3 = [7, 8]$, $a_4 = [4, 9]$, $a_5 = [5, 9]$, $a_6 = [4, 8]$, $a_7 = [7, 9]$, and $b = [2, 6]$, the similarities between $a_1, a_2, a_3, a_4, a_5, a_6, a_7$ and b are respectively measured by d_C, d_H, d_E, and d_{OLID} (see Table 1).

Table 1 shows that d_C and d_E does not give the difference between (a_2, b) and (a_4, b). d_H can not determine the difference (a_1, b), (a_2, b), (a_4, b) and (a_5, b), while d_{OLID} has overcome the limitations of the above distances.

2.2 The indexes for measuring and building clusters

Definition 2 Let N p-dimensional intervals dataset $\{x_1, x_2, \ldots, x_N\}$ grouped into k clusters C_i, $i = 1, 2, \ldots, k$, the New Davies and Bouldin (NDB) index is defined as follows.

$$NDB = \frac{1}{N} \sum_{i=1}^{N} \max_{i \neq j} \left\{ \frac{\frac{1}{|C_i|} \sum_{x_i \in C_i} d_{OLID}(x_i, \hat{x}_i) + \frac{1}{|C_j|} \sum_{x_j \in C_j} d_{OLID}(x_j, \hat{x}_j)}{d_E(\hat{x_i}, \hat{x_j})} \right\}, \tag{3}$$

where

x_i is the interval of groups C_i,
$|C_i|$ is the number of intervals in group C_i,
$\hat{x}_i$ is the center of group C_i.

The NDB index is actuated from the DB index in [6], and it is the objective function in this study. It has difference with the IDB index in [40] by $d_E(\hat{x_i}, \hat{x_j})$. NDB considers both the closeness and the segregation of clusters. The more detached the groups are, the significant the NDB index is.

Table 1 The similarity between a_i and b

Distance	a_1	a_2	a_3	a_4	a_5	a_6	a_7
d_C	10	13	29	13	18	8	34
d_H	3	3	5	3	3	2	5
d_E	3.16	3.61	5.39	3.61	4.24	2.83	5.83
d_{OLID}	0.67	1.50	3	2	2.40	1.20	4

Definition 3 Let $U = \{u_1, u_2, ..., u_R\}$, $V = \{v_1, v_2, ..., v_C\}$ be two partitions of the same data set having R and C clusters, respectively. These are the used parameters to evaluate the quality of the established algorithms:

– CR index [15]:

$$CR = \frac{\sum_{i=1}^{R}\sum_{j=1}^{C}\binom{n_{ij}}{2} - \binom{n}{2}^{-1}\sum_{i=1}^{R}\binom{n_i}{2}\sum_{j=1}^{C}\binom{n_j}{2}}{\frac{1}{2}\left[\sum_{i=1}^{R}\binom{n_i}{2} + \sum_{j=1}^{C}\binom{n_j}{2}\right] - \binom{n}{2}^{-1}\sum_{i=1}^{R}\binom{n_i}{2}\sum_{j=1}^{C}\binom{n_j}{2}},$$

– HI index [14]:

$$HI = 1 + \left[2\sum_{i=1}^{R}\sum_{j=1}^{C} n_{ij}^2 - \left(\sum_{i=1}^{R} n_i^2 + \sum_{j=1}^{C} n_j^2\right)\right] / \binom{n}{2},$$

– MI index [23]

$$MI = \left[-1/2\left\{\sum_{i=1}^{R}(\sum_{j=1}^{C} n_{\mathrm{ij}})^2 + \sum_{j=1}^{C}(\sum_{i=1}^{R} n_{\mathrm{ij}})^2\right\} + \sum_{j=1}^{C}\sum_{i=1}^{R} n_{\mathrm{ij}}^2\right] / \binom{n}{2},$$

– RI index [29]:

$$RI = \left[\binom{n}{2} - \left[1/2\left\{\sum_{i=1}^{R}(\sum_{j=1}^{C} n_{\mathrm{ij}})^2 + \sum_{j=1}^{C}(\sum_{i=1}^{R} n_{\mathrm{ij}})^2\right\} - \sum_{j=1}^{C}\sum_{i=1}^{R} n_{\mathrm{ij}}^2\right]\right] / \binom{n}{2},$$

where

n_{ij} is the number of elements to belong to cluster u_i and v_j, $1 \le i \le R$, $1 \le j \le C$,
n_i, n_j are the number of elements in cluster u_i and u_j, respectively,
n is the total number of elements.

For the built algorithms, the smaller of MI is, the better of algorithm is. For the parameters (CR, HI, RI), their values have the opposite meaning.

2.3 Extracting the featured intervals for images

The gray level co-occurrence matrix (GLCM) for a image with size $M \times N$ is the P matrix to have the size $g \times g$, where g is the number of gray-level used to construct the matrix. Each element $p_{d\theta}(i, j)$ of P shows the probability for occurrence the intensity i and j with distance d and orientation angle θ. It is given by (4).

$$\begin{aligned} p_{d\theta}(i, j) &= \{((r, c), (r', c')) \in M \times N | d = ||(r, c), (r', c')||, \\ \theta &= \Theta((r, c), (r', c')), I(r, c) = i, I(r', c') = j\}. \end{aligned} \tag{4}$$

After calculating GLCM for each image, we continue to extract these characteristics to become the intervals by (5):

$$[\mu_x - r_1/2, \mu_x + r_1/2], [\mu_y - r_2/2, \mu_y + r_2/2], \tag{5}$$

where r_1 and r_2 are the value of uniform distribution in [1;4], and

$$\mu_x = \frac{1}{N_y}\sum_{j}^{N_y}\left(\frac{1}{N_x}\sum_{i}^{N_x}(i)\,p_{d\theta}(i,j)\right); \mu_y = \frac{1}{N_y}\sum_{i}^{N_x}\left(\frac{1}{N_x}\sum_{j}^{N_y}(j)\,p_{d\theta}(i,j)\right), \quad (6)$$

with N_x and N_y are first and second size of image, and $p_{d\theta}(i, j)$ is calculated by (4).

Formulate (6) is considered to be the average of the structural features of image extracted by (4), according to the first and second dimensions of the GLCM. From the value of the average, we create a two-dimensional interval by (5). It means that each image will be characterized by a two-dimensional interval that presents for two sizes of GLCM. This can represent well the texture of the image, and reduce the calculation cost in clustering.

3 The proposed algorithm

3.1 The algorithm

Let N images $X = \{I_1, I_2, ..., I_N\}$. The automatic fuzzy genetic algorithm in clustering for images has the following steps:

Step 1 Extract the characteristics of images by (4), and (5) to have the interval data: $V = \{v_1, v_2, ..., v_N\}$, where $v_i, i = 1, 2, ..., N$ are the two p-dimensional intervals.

Step 2 When $t = 0$, initialize the interval data

$$\mathbf{V^{(0)}} = \left\{v_1^{(0)}, v_2^{(0)}, \ldots, v_N^{(0)}\right\} = V.$$

Step 3 Update the prototype intervals by (7):

$$v_i^{(t+1)} = \frac{\sum_{j=1}^{N} f\left(v_i^{(t)}, v_j^{(t)}\right).v_j^{(t)}}{\sum_{j=1}^{N} f\left(v_i^{(t)}, v_j^{(t)}\right)}, i, j = 1, ..., N \quad (7)$$

where

$$f\left(v_i^{(t)}, v_j^{(t)}\right) = \begin{cases} \exp\left(-\frac{d_{OLID}\left(v_i^{(t)}, v_j^{(t)}\right)}{\lambda}\right) & \text{if } d_{OLID}\left(v_i^{(t)}, v_j^{(t)}\right) \le \mu\alpha_{ij}(t), \\ 0 & \text{if } d_{OLID}\left(v_i^{(t)}, v_j^{(t)}\right) > \mu\alpha_{ij}(t), \end{cases}$$

with

- $\alpha_{ij}(t) = \frac{\alpha_{ij}(t-1)}{1+\alpha_{ij}(t-1).f\left(v_i^{(t-1)}, v_j^{(t-1)}\right)}$ is the balance factor ($\alpha_{ij}(0) = 1$).
- $\mu = \frac{1}{\binom{2}{N}}\sum_{i<j} d_{OLID}\left(v_i^{(t)}, v_j^{(t)}\right)$.
- $\lambda = \frac{\sigma}{r}$ with $\sigma = \sqrt{\frac{1}{\binom{2}{N}}\sum_{i<j}\left[d\left(v_i^{(t)}, v_j^{(t)}\right) - \mu\right]^2}$, and r is a constant.

The value of λ affects to the result of building clusters. When $\lambda \to 0$, each cluster contains only one element, and when $\lambda \to \infty$, we only have one cluster. Test on many several experiments, we take $\lambda = \sigma/16$ for all examples in this article.

Step 4 Repeat Step 3 until $\max_i\{d_{OLID}(v_i^{(t+1)}, v_i^{(t)}\} < \varepsilon$.

Through each updating for $v^{(t+1)}$ by (7), the objects in the same group will gradually converge to its representative interval. The algorithm will stop when the maximum distance between intervals representing of two consecutive steps is smaller than ε. In this paper, $\varepsilon = 10^{-4}$ is chosen for all numerical examples.

Step 5 For data, the chromosomes is coded into the non-integer in $[\min(V); \max(V)]$ with the length cp.

Step 6 Calculate the NDB index for the first chromosome by (3). In this process, the formula (8) is used as the provisional step:

$$U = \arg\max\{d_{OLID}(x, \hat{x}_i)\}, i = 1, ..., c. \tag{8}$$

Step 7 Perform the selection, crossover, and mutation operators.

- Crossover: The children chromosome is born from the parents P_1 and P_2 by (9)

$$Child = P_1 + v[0, 1].Ra.(P_2 - P_1), \tag{9}$$

  where ,

  - Ra is the probability of crossover process chosen as 0.85 for the numerical examples.
  - $v[0, 1]$ is non-integers in $[0, 1]$, which has the length equal P_1 and P_2.

- Mutation: Let x be the previous result of the mutated gene, the new value x' of x is calculated as follows.

$$x' = \begin{cases} (1 \pm 2\gamma)\, x & \text{if } x \neq 0, \\ \pm 2\gamma & \text{if } x = 0, \end{cases}$$

  where γ is a random value in $[0, 1]$. In this study, we take '+' and '-' with the same probability.
- Selection: the roulette wheel method of Lai [20] is used for the next generation.

Step 8 Update the NDB index for the new chromosome created form Step 7.

Step 9 Repeat Step 6, Step 7, and Step 8 until $iter < miter$, where $iter$ and $miter$ are the current and maximum iterations of the genetic algorithm, respectively. This study chooses $miter = 1000$ to perform.

Step 10 From c clusters and the optimal chromosome m_i, the probability belonging to cluster of images is determined by (10).

$$\mu_{ij} = \frac{d_{OLID}(m_i, a_h)^2}{\sum_{j=1}^{c} d_{OLID}(m_j, a_h)^2}, 1 \leq i, j \leq c, 1 \leq h \leq N. \tag{10}$$

3.2 The convergence of algorithm

The AFGI has the two main phases. Phase 1 (Step 1 to Step 4) finds the appropriate number of clusters c. After that, c is used for input of Phase 2 (the remaining steps of the algorithm). Phase 2 determines the intervals for clusters, and the probability of elements belonging to

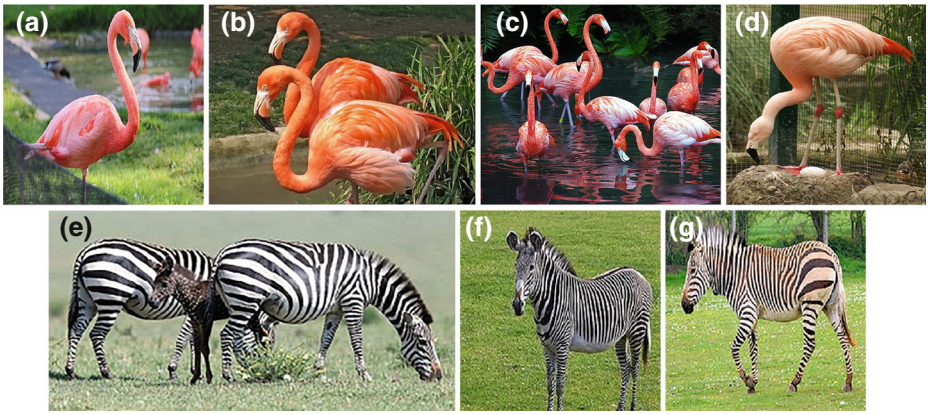

Fig. 1 The images about flamingo and horse of 2 groups

each group. The convergence of Phase 1 is presented in Tai et al. [40]. The converges of Phase 2 appears when the number of iterations reach the maximum value ($miter$).

4 Numerical examples

This section considers four image sets. With the small number, Example 1 illustrates the step by step of the AFGI. It also shows suitability of the AFGI. For complex of the images, the remaining examples focus the comparison of AFGI with the existing ones. The compared algorithms are AFGI using the Hausdorff distance (AFGI-H), Euclidean distance (AFGI-E), and City-block distance (AFGI-C). In addition, we also compare AFCI with other algorithms such as k-mean with Hausdorff distance (k-mean-H), k-mean with City-block distance (k-mean-C), k-mean with Euclidean distance (k-mean-E), k-mean with overlap measure (k-mean-O), the algorithms of De Carvalho et al. [7], De Souza et al. [8], Hung et al. [16], Jeng et al. [18], Sara et al. [30], and Tai et al. [40].

4.1 Example 1

In this example, we use two groups of images (flamingo and horse) with the numbers of each group as 4 and 3 images, respectively. The images are presented in Fig. 1.

Table 2 The featured intervals for 7 images

Image	Dimension 1	Dimension 2
1	[2.663, 5.663]	[2.167,6.167]
2	[1.604, 5.604]	[1.606,5.606]
3	[1.824, 3.824]	[1.826,3.826]
4	[1.705,5.705]	[1.708,5.708]
5	[4.071, 6.071]	[4.072,6.072]
6	[3.584, 5.584]	[3.592,5.592]
7	[4.573,5.573]	[3.579,6.579]

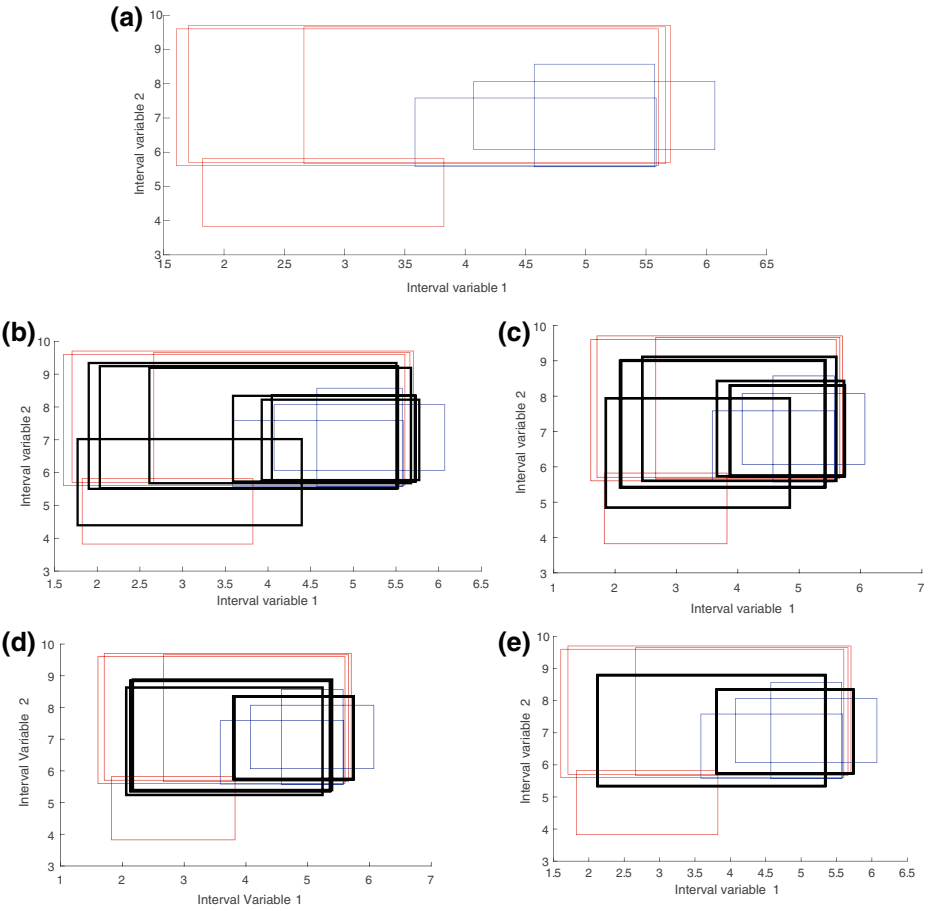

Fig. 2 The convergence of the AFGI in Phase 1 with 7 intervals

Step 1 The extracted intervals for 7 images is given in Table 2.

Step 2 to Step 4 After 5 iterations of Phase 1, the algorithm will stop. These iterations are shown by Fig. 2 and present in Table 3.

Table 3 The convergence of 7 intervals into two protoype ones

Image	Interval 1	Interval 2
1	[2.123, 5.341]	[2.006, 5.464]
2	[2.123, 5.341]	[2.006, 5.464]
3	[2.123, 5.341]	[2.006, 5.464]
4	[2.123, 5.341]	[2.006, 5.464]
5	[3.804, 5.739]	[3.475, 6.078]
6	[3.804, 5.739]	[3.475, 6.078]
7	[3.804, 5.739]	[3.475, 6.078]

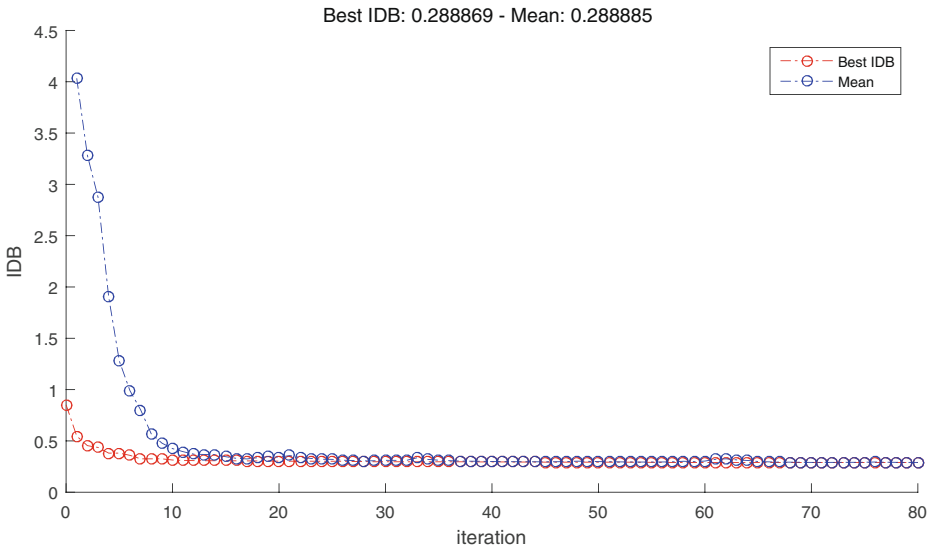

Fig. 3 The convergence of the algorithm in the Phase 2 with 7 intervals

From Fig. 2 and Table 3, we obtain 2 clusters. Continue to perform Phase 2, the steps is detailed as follows:

Step 5 Code the chromosomes, we obtain:

- $Varmin = [1.604, 3.824, 1.606, 3.826, 1.604, 3.824, 1.606, 3.826]$.
- $Varmax = [4.573, 6.071, 4.072, 6.580, 4.573, 6.071, 4.072, 6.580]$.
- First chromosome: 3.690 , 5.129 , 2.690 , 6.414 , 1.623 , 3.829 , 2.384 , 3.962.
- $NDB = 0.478$, $U = [1\ 2\ 1\ 1\ 1\ 1\ 1]$.

Step 6 Run operators:

- Crossover operator: 85 out of 100 chromosomes are randomly selected to crossover with together by formula (9).
- Mutation operator: 15 remain chromosomes will be running the point mutation processes. (see Appendix A).
- Selection operator: The Roulette wheel method [20] is used in this operator.

Table 4 The probability to belong to clusters of 7 images

Image	$\mu_{i,1}$	$\mu_{i,2}$
1	0.905	0.095
2	0.998	0.002
3	0.892	0.109
4	0.995	0.005
5	0.001	0.999
6	0.044	0.957
7	0.019	0.981

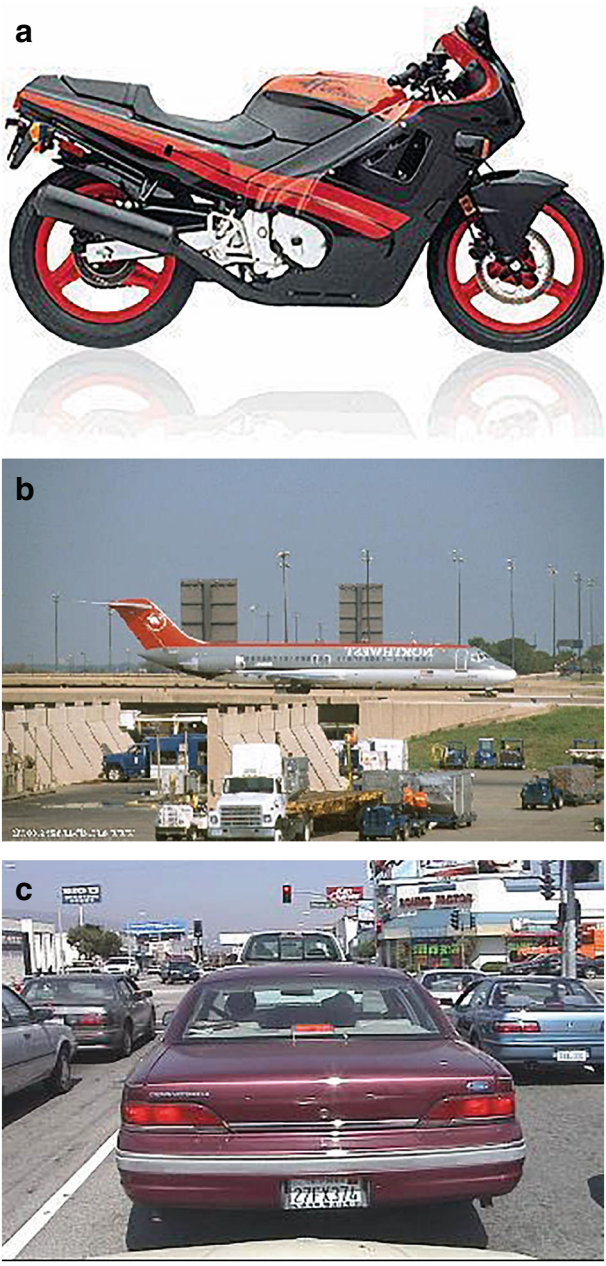

Fig. 4 Three samples for images

Step 7 Compute the NDB index for 100 new chromosomes, we have the $NDB = 0.410$, and the best chromosome: $U = [1\ 2\ 2\ 1\ 2\ 1\ 1]$.

Step 8 Return Step 6 and Step 7 with 65 iterations, the algorithm converges.

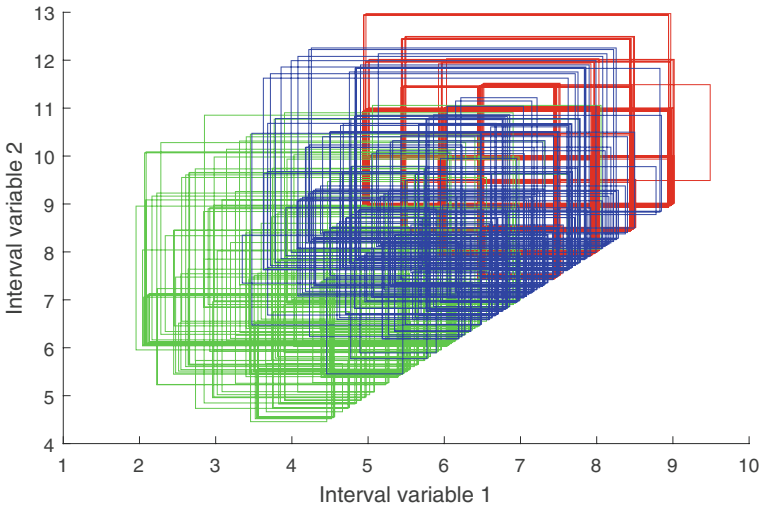

Fig. 5 The extracted Intervals for 300 images

Step 9 After Step 8, the algorithm will be stopped (see Fig. 3) and published the following results:

- The best chromosome: $m = [4.145, 6.001, 4.072, 6.072, 1.604, 5.083, 1.606, 5.582]$.
- The optimal objective function: $NDB = 0.289$.
- The result of cluster analysis: $U = [1\ 1\ 1\ 1\ 2\ 2\ 2]$.
  It means that we have two clusters:

$$C_1 = \{I_1, I_2, I_3, I_4\}; C_2 = \{I_5, I_6, I_7\}.$$

Step 10 Calculate the probability to belong to clusters of each image, we have Table 4:

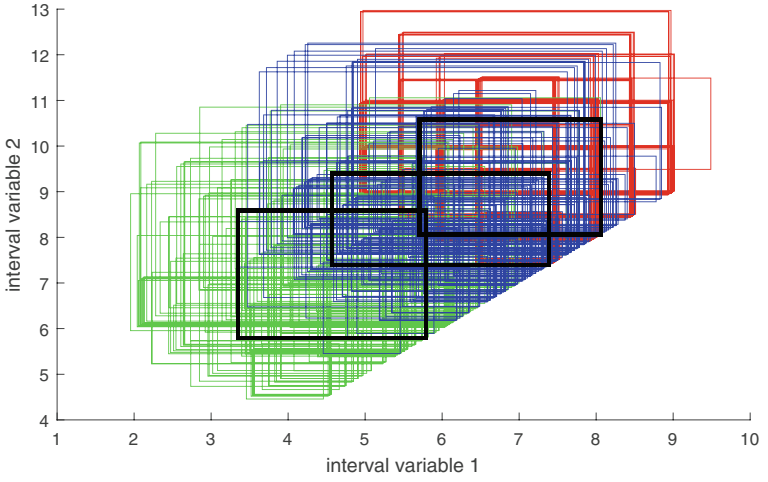

Fig. 6 The convergence of 300 intervals for Phase 1

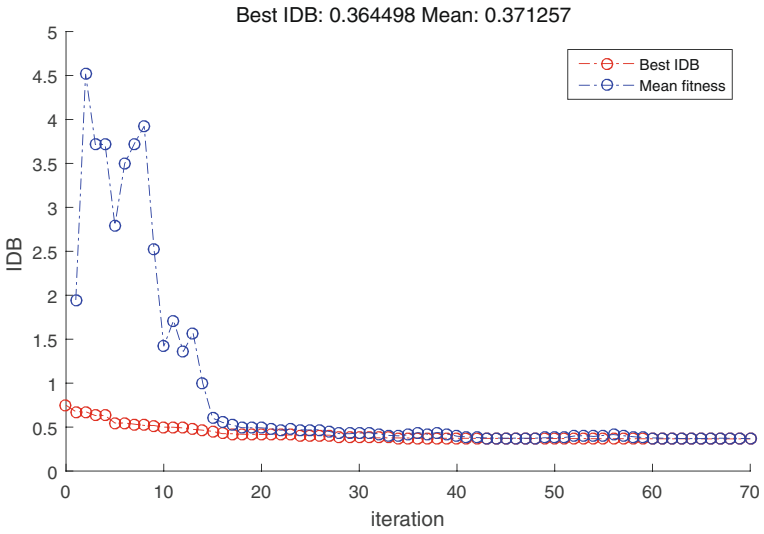

Fig. 7 The convergence of the AFGI for Phase 2 with 300 intervals

4.2 Example 2

This section considers 300 images divided to 3 groups with 100 images for each group. This data is taken from http://www.vision.caltech.edu/html-files/archive.html. Some samples for images are shown in Fig. 4.

Extracting 300 images, we have the intervals shown by Fig. 5.

Performing Phase 1, we obtain Fig. 6.

Figure 6 shows that 300 intervals converge into the 3 black rectangles. Therefore, the appropriate number of groups is 3.

Run Phase 2, the convergence is shown by Fig. 7, and the results obtain as follows:

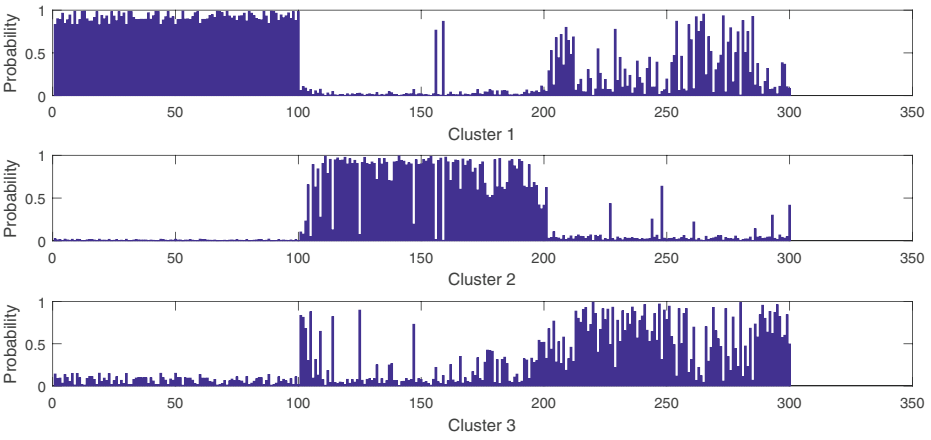

Fig. 8 The probability belongs to three clusters of 300 images

Table 5 Comparing the proposed algorithm and the existing ones for 300 images

Method	CR	RI	MI	HI
Proposed	0.848	0.933	0.067	0.865
AFGI-E	0.672	0.853	0.147	0.707
AFGI-C	0.665	0.850	0.150	0.700
AFGI-H	0.630	0.834	0.166	0.668
Tai et al. (2020) [40]	0.749	0.863	0.137	0.786
Jeng et al. (2019) [18]	0.663	0.849	0.151	0.699
Sara et al. (2019) [30]	0.656	0.846	0.154	0.692
De Carvalho et al. (2007) [7]	0.521	0.743	0.198	0.612
De Souza et al. (2004) [8]	0.654	0.781	0.176	0.721
k-means-O	0.456	0.723	0.277	0.446
k-means-C	0.459	0.724	0.276	0.449
k-means-E	0.488	0.724	0.276	0.449
k-means-H	0.462	0.726	0.274	0.452

- The optimal objective function: $NDB = 0.365$.
- The result of cluster analysis:

$$U = [1, \underbrace{...}_{98}, 1, 3, \underbrace{...}_{6}, 3, 2, \underbrace{...}_{80}, 2, 1, 1, 1, 1, 1, 1, 3, \underbrace{...}_{92}, 3]$$

It means that we have 3 clusters:

$$C_1 = \{I_1, ..., I_{100}, I_{201}, I_{202}, I_{203}, I_{204}, I_{205}, I_{206}\};$$
$$C_2 = \{I_{109}, ..., I_{200}\};$$
$$C_3 = \{I_{101}, I_{102}, I_{103}, I_{104}, I_{105}, I_{106}, I_{107}, I_{108}, I_{207}, ..., I_{300}\}.$$

The probability of images to belong to clusters is shown by Fig. 8.

Comparing AFGI and other algorithms, we obtain Table 5.

Table 5 shows that the AFGI has good result in implementing. It is also sees AFGI has the best result in comparison to others with all parameters.

In short, the AFGI has given the right result about the number of groups with this image data. The probability to belong to its right cluster is also suitable. In addition, Table 5 indicates that the AFGI has the best result for all parameters in comparison to the existing ones.

Fig. 9 The image samples of 519 flowers

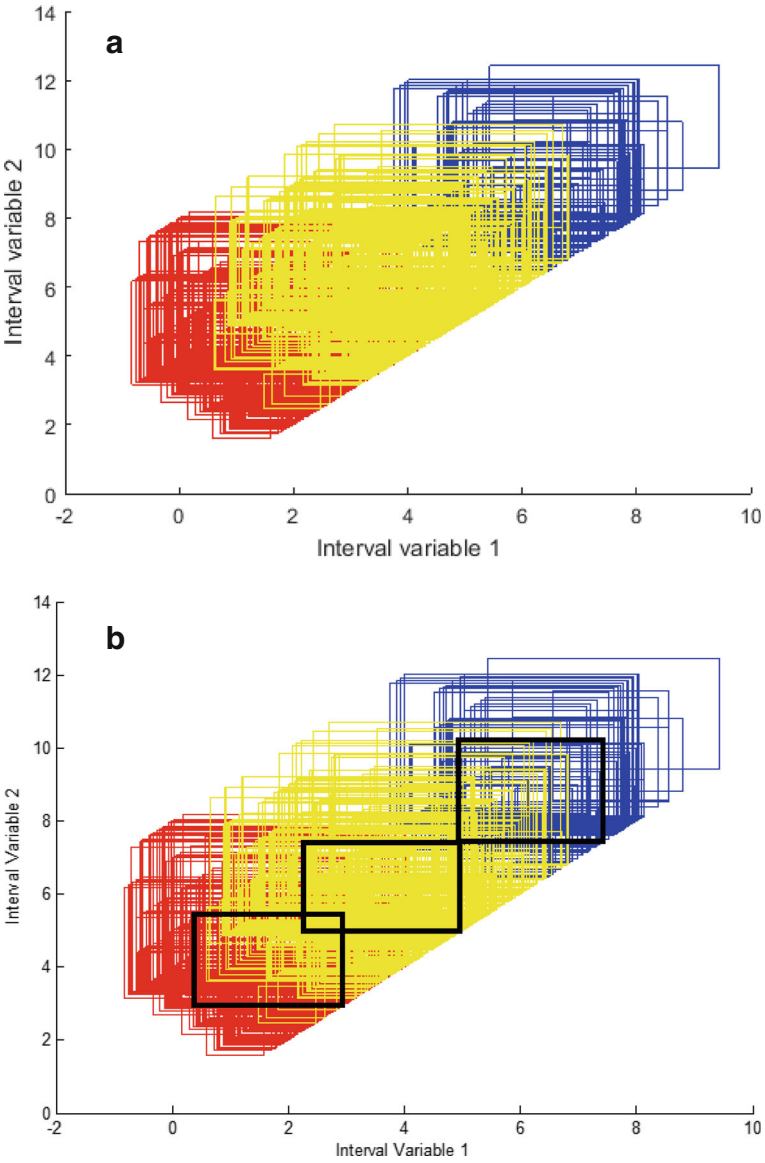

Fig. 10 The extracted intervals (**a**) and the convergence of 519 intervals (**b**) for Phase 1

4.3 Example 3

This example considers 519 images with 192 Sen flowers, 76 Gazania flowers, and 251 Passion flowers. This image set is taken from http://www.robots.ox.ac.uk/~vgg/data/flowers/102/categories.html. Some sample images of three groups are given by Fig. 9.

Extracting the characteristics of 519 flowers by the intervals, and implementing Phase 1 with 18 iterations, we have Fig. 10.

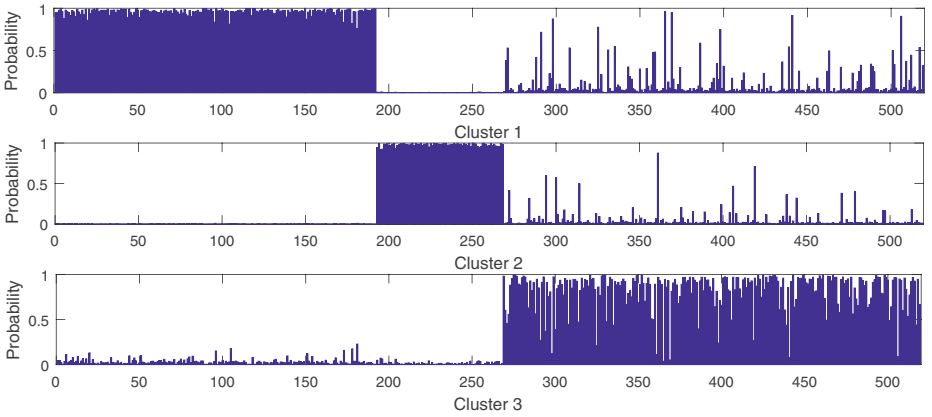

Fig. 11 The optimal result of Phase 2 for 519 images

From Fig. 10b, we obtain 3 clusters. Performing Phase 2 with 3 clusters, after 45 iterations, the algorithm will stop (see Fig. 11).

At that time, we have the following results:

- $NDB = 0.268$.
- Three clusters:

$$C_1 = \{I_1, I_2, ..., I_{191}\}; C_2 = \{I_{193}, I_{194}, ..., I_{268}\}; C_3 = \{I_{192}, I_{269}, I_{194}, ..., I_{519}\}.$$

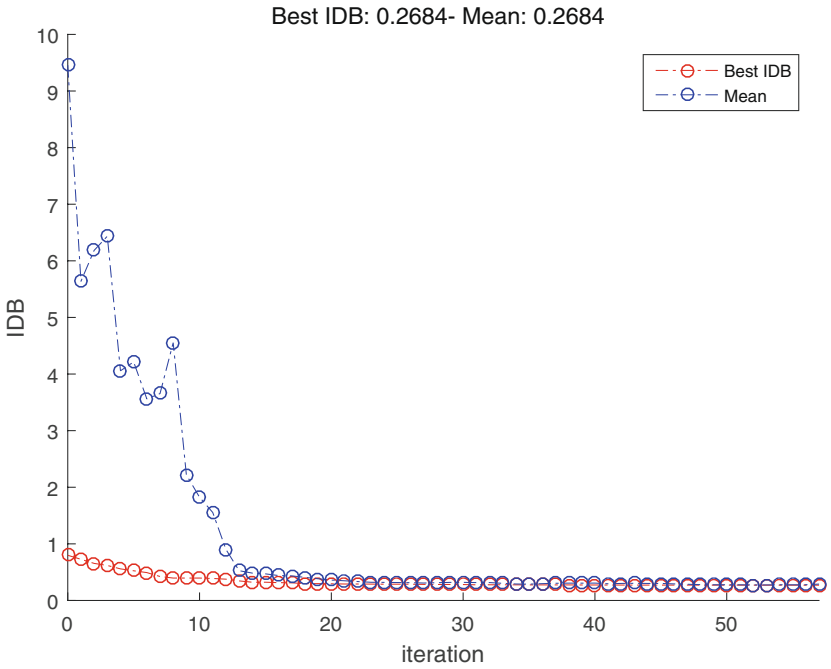

Fig. 12 The probability belongs three clusters of 519 images

Table 6 Comparing the AFGI and the existing ones for 519 images

Method	CR	RI	MI	HI
Proposed	0.995	0.998	0.002	0.995
AFGI-E	0.976	0.988	0.012	0.977
AFGI-C	0.976	0.988	0.012	0.977
AFGI-H	0.934	0.969	0.031	0.938
Tai et al. (2020) [40]	0.993	0.990	0.003	0.984
Sara et al. (2019) [30]	0.864	0.936	0.064	0.871
Jeng et al. (2019) [18]	0.853	0.930	0.070	0.758
Hung et al. (2017) [16]	0.969	0.985	0.015	0.971
De Carvalho et al. (2007) [7]	0.933	0.968	0.032	0.936
De Souza et al. (2004) [8]	0.933	0.968	0.032	0.936
k-means-E	0.861	0.933	0.067	0.861
k-means-H	0.861	0.933	0.067	0.861
k-means-C	0.858	0.933	0.067	0.865

The probability to belong to 3 clusters of the images is given by Fig. 12:

Comparing to the existing ones, we obtain Table 6.

One time, from Table 6, we see that the proposed model has the most optimization with all considered parameters.

4.4 Example 4

This example considers 1400 images with 70 groups. This image set is provided from http://www.dabi.temple.edu/~shape/MPEG7/dataset.html. Some sample images are presented in Fig. 13.

Perform the Phase 1 for 1400 images, we have 70 clusters shown by Fig. 14.

For Phase 2, the convergence of the AFGI is present by Fig. 15, and probability to belong to some clusters is given by Fig. 16.

Comparing result of the proposed and other algorithms, we have Table 7.

Table 7 also shows that the AFGI has the best result when it is compared to the existing ones Fig. 16.

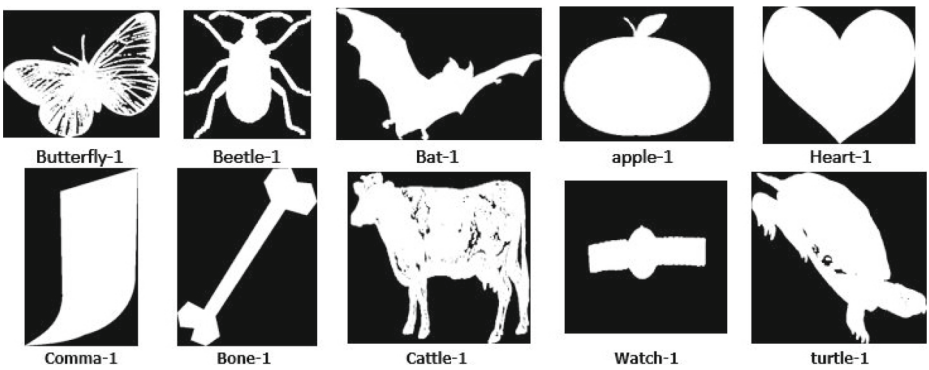

Fig. 13 Some samples for MPEG-7 dataset

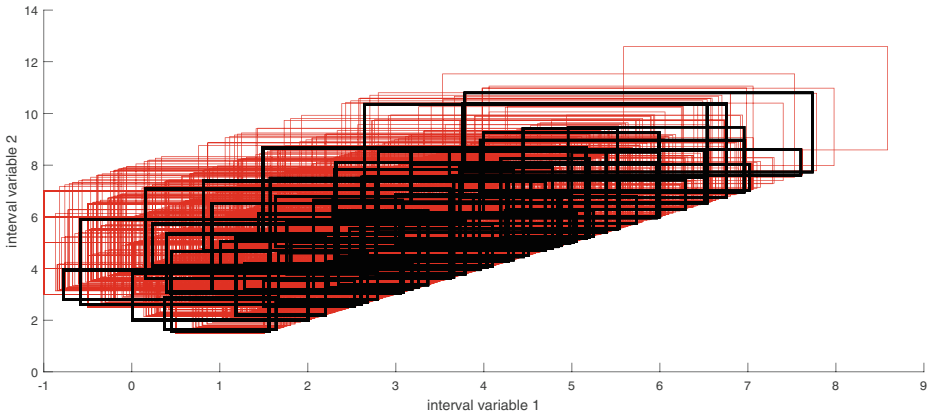

Fig. 14 The convergence of 1400 intervals into 70 clusters

5 Conclusion

This research has the important contributions for unsupervised learning technique. First, we has been proposed a method to extract the characteristics from the texture of images into intervals. These intervals are considered as the good input data for recognizing images. Second, the study gives a criterion to measure the differences of intervals. This measure has more advantages than the popular ones in building CID. Third, it establishes the automatic fuzzy genetic algorithm for images with outstanding advantages. The AFGI not only finds the apposite number of groups, the images in each group but also determines the probability

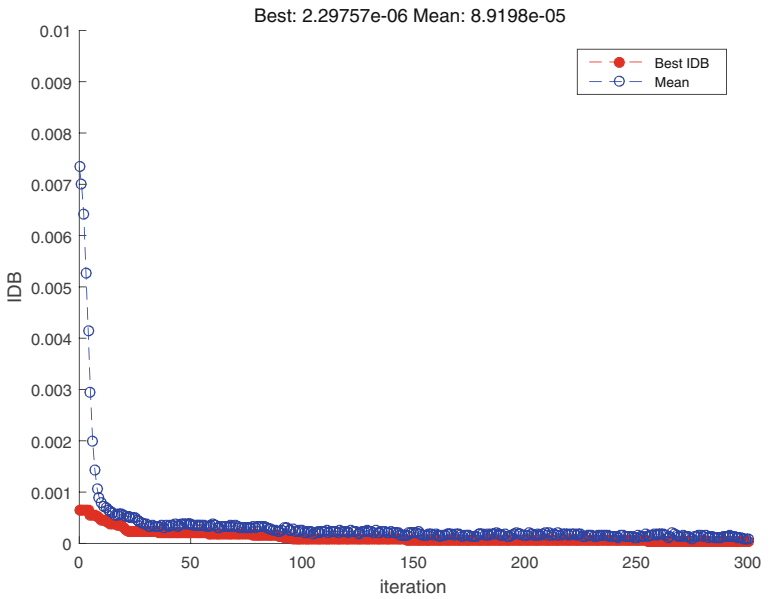

Fig. 15 The convergence of the AFGI for Phase 2 with 1400 images

Table 7 Comparing the proposed algorithm and the existing ones for 1400 images

Method	CR	RI	MI	HI
Proposed	0.693	0.969	0.031	0.941
AFGI-E	0.564	0.878	0.122	0.877
AFGI-C	0.526	0.748	0.252	0.827
AFGI-H	0.533	0.729	0.271	0.839
Tai et al. (2020) [40]	0.673	0.947	0.048	0.938
Sara et al. (2019) [30]	0.337	0.948	0.052	0.897
Jeng et al. (2019)[18]	0.331	0.969	0.031	0.938
Hung et al. (2016) [16]	0.569	0.937	0.063	0.931
De Carvalho et al. (2007) [7]	0.333	0.938	0.062	0.936
De Souza et al. (2004) [8]	0.343	0.935	0.065	0.933
k-means-E	0.328	0.968	0.032	0.937
k-means-H	0.339	0.937	0.063	0.874
k-means-C	0.331	0.969	0.031	0.938

to belong to groups of each image. The proposed method can execute effectively by the established Matlab procedure. It is illustrated step by step by the numerical examples. They also show the outstanding advantage of the proposed algorithm in comparison to the existing ones.

With the compared data sets to have the difference about the number and character of images, the proposed algorithm has obtained the suitable result, and given the most optimal in comparison with the existing ones. Furthermore, the AFGI has very potential in applying to the intelligent systems that relate to recognize the images. It can become the first step for these systems applied in medicine, security, environment. In the proposed algorithm, we can expand the extraction the two-dimensional intervals by the p-dimensional intervals ($p > 2$) to increase the effectiveness. However, we might have to deal with a computational problem in this case. The cloud computing and the parallel algorithm can be the good methods to solve this problem. All of the above issues will be our further research in the future.

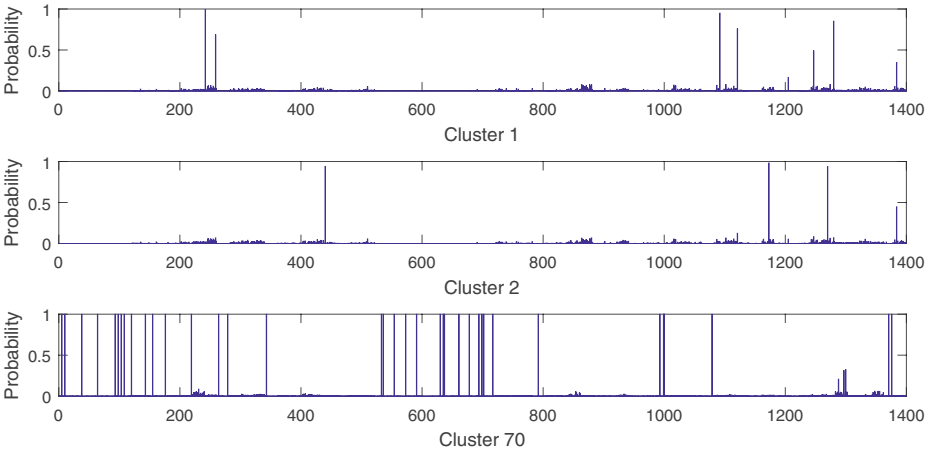

Fig. 16 The probability to belong to 1, 2 and 70 clusters of 1400 images

Appendix A

Table 8 The created chromosomes by operators in first iteration

No.	Value							
1	1.728	3.918	2.077	4.321	2.313	5.475	3.345	6.346
2	2.763	4.039	1.928	5.291	3.485	5.767	3.416	6.372
3	2.177	4.950	2.257	4.685	4.433	5.255	3.825	4.773
4	3.902	5.982	2.552	6.349	1.892	4.193	3.280	5.601
5	3.087	5.231	3.754	6.451	2.177	5.722	2.821	4.464
6	3.882	3.827	3.080	4.723	2.299	5.152	2.577	5.821
7	3.486	4.039	3.903	5.291	3.485	4.747	1.854	4.219
8	2.325	4.668	2.920	4.584	2.881	5.255	3.499	4.465
9	3.320	4.830	2.691	4.465	1.888	5.334	2.171	6.209
10	3.134	4.792	2.601	6.076	3.558	5.248	2.698	6.091
11	1.604	3.824	3.596	4.864	1.604	4.553	3.405	3.826
12	2.315	5.982	1.970	5.410	3.885	4.889	3.375	4.079
13	2.469	5.772	2.118	6.349	2.638	4.966	2.399	5.601
14	1.604	5.473	2.310	4.327	2.824	5.062	3.419	6.468
15	2.287	5.258	3.603	6.327	2.313	5.475	2.041	4.703
16	3.752	3.998	3.790	4.360	4.533	5.997	2.688	6.494
17	2.177	4.950	2.257	3.925	4.433	5.447	2.864	4.773
18	2.576	4.675	3.015	5.040	2.079	5.428	2.992	4.313
19	3.579	5.982	2.273	6.055	2.774	4.377	2.636	5.601
20	2.157	4.410	3.146	6.383	2.015	5.939	1.864	5.027
21	4.528	4.437	1.773	4.399	1.912	4.080	2.678	4.971
22	3.274	4.624	3.432	5.337	3.392	5.087	3.146	5.717
23	1.917	4.865	2.168	4.987	4.486	5.019	1.987	5.080
24	2.763	4.039	1.928	3.897	3.485	5.767	3.416	6.563
25	3.791	4.387	3.117	5.043	2.212	4.835	2.407	4.655
26	1.905	5.473	2.507	4.327	2.591	4.207	3.419	6.468
27	1.728	5.306	2.077	3.934	2.313	6.063	3.298	5.387
28	2.610	5.228	3.512	3.925	3.094	5.930	2.396	5.161
29	2.763	4.039	2.507	5.291	2.591	5.767	1.692	4.664
30	3.545	5.825	4.049	6.460	2.749	3.943	2.790	5.824
31	2.074	3.973	3.589	4.132	4.199	5.192	1.964	5.080
32	1.728	3.918	3.416	5.627	2.920	4.150	2.554	6.346
33	2.177	4.950	2.257	6.451	4.433	5.722	3.825	4.773
34	3.015	5.830	3.146	5.790	2.015	4.978	2.346	4.971
35	3.630	5.046	3.010	4.053	2.053	5.324	3.180	4.702
36	2.543	4.960	2.552	6.019	3.200	3.908	3.241	5.601
37	2.920	5.047	3.219	4.911	4.262	5.054	3.095	6.152
38	3.087	5.231	2.077	4.321	2.177	5.722	3.345	4.464
39	3.902	5.007	2.346	6.349	1.892	4.193	2.367	5.601

Table 8 (continued)

No.	Value							
40	1.728	4.039	2.077	5.291	3.485	5.767	3.416	6.346
41	2.262	5.359	3.080	5.339	2.299	3.874	1.674	5.299
42	2.177	4.207	3.589	4.685	4.433	5.255	1.964	6.452
43	4.552	4.121	3.596	5.337	3.104	5.087	3.405	5.504
44	2.442	4.940	4.049	5.248	2.749	5.876	2.790	5.157
45	2.753	5.871	3.246	5.954	3.531	4.203	1.991	6.256
46	1.728	5.871	2.077	4.321	2.313	5.475	2.145	6.346
47	2.275	3.958	2.662	3.851	2.528	4.395	1.705	6.013
48	4.118	5.828	1.776	5.536	1.669	5.381	3.551	4.115
49	4.213	5.048	2.468	6.035	3.070	5.275	3.775	4.016
50	4.076	4.940	2.105	5.248	2.594	5.243	2.585	4.818
51	2.478	5.727	3.696	5.776	2.950	5.912	2.241	4.439
52	2.763	4.865	1.928	5.291	4.486	5.019	3.890	6.372
53	1.902	4.050	2.077	4.321	1.723	3.860	3.345	6.346
54	1.691	4.950	4.023	4.498	4.279	4.212	2.388	4.357
55	1.659	4.251	2.014	6.035	3.599	4.433	3.775	4.016
56	1.639	5.017	2.341	5.396	4.250	5.024	2.271	6.211
57	2.920	5.258	2.137	6.101	1.957	4.562	2.238	4.703
58	3.869	5.383	2.913	5.618	1.733	4.835	2.407	5.750
59	4.225	4.843	3.426	6.064	3.485	5.697	3.416	4.731
60	2.262	4.624	3.432	4.877	4.313	4.552	2.149	5.882
61	3.063	5.982	3.808	5.410	3.885	4.403	1.834	5.523
62	3.150	5.409	3.113	4.724	1.737	4.262	2.353	6.397
63	4.213	5.048	2.077	6.376	2.313	5.475	3.345	4.000
64	3.147	3.852	2.282	4.997	3.775	4.351	1.992	4.825
65	2.298	4.830	1.833	4.647	2.191	5.334	2.171	6.077
66	3.430	4.634	2.785	6.232	3.968	4.162	3.748	4.472
67	2.887	5.768	2.515	5.238	4.337	4.066	2.703	3.869
68	3.087	5.937	3.754	5.296	3.043	5.722	2.821	4.464
69	4.245	4.675	3.931	4.295	4.120	5.428	4.071	5.087
70	2.177	4.302	2.257	4.685	4.433	5.255	3.825	3.955
71	2.612	4.353	3.195	3.933	4.440	4.446	3.691	5.456
72	2.177	3.967	2.257	4.685	1.957	5.255	2.238	4.773
73	2.790	5.836	3.754	6.451	2.673	5.376	2.555	4.464
74	2.545	5.160	1.643	4.354	3.531	5.928	1.991	4.830
75	2.612	4.837	3.788	4.591	1.762	5.428	2.080	5.456
76	4.118	5.539	3.808	5.536	1.669	4.873	1.834	5.595
77	2.447	4.110	2.123	3.925	2.436	4.194	1.646	5.161
78	4.225	5.245	2.793	6.064	2.899	4.552	2.682	4.731
79	2.585	5.212	2.077	5.470	3.197	4.592	3.917	4.657
80	2.287	5.258	3.603	6.327	2.601	4.103	2.041	4.703
81	1.728	3.918	2.077	5.419	4.444	3.909	3.345	6.346

Table 8 (continued)

No.	Value							
82	4.084	4.554	3.931	4.295	4.120	3.861	3.122	5.505
83	3.709	4.830	1.643	4.647	3.948	5.334	2.171	6.209
84	3.878	6.019	3.772	4.988	4.250	4.574	2.601	6.152
85	1.905	5.745	3.294	4.498	3.781	4.150	2.388	3.882
86	3.545	5.825	3.081	6.460	3.273	5.823	2.790	5.157
87	3.902	5.982	2.552	5.349	1.892	5.193	3.280	5.601
88	1.905	5.473	2.310	4.327	2.280	5.778	2.419	6.468
89	2.074	4.207	2.589	4.419	4.199	5.412	1.964	6.452
90	2.516	4.409	2.446	4.724	3.152	4.334	2.353	6.545
91	2.878	5.716	2.552	3.954	4.194	5.209	3.468	6.051
92	1.742	4.833	3.392	5.263	3.019	4.997	2.071	5.810
93	3.882	3.827	1.860	4.723	2.772	6.027	2.577	4.821
94	4.291	5.204	2.449	3.998	3.413	4.489	2.141	4.682
95	2.262	5.933	3.417	5.721	4.313	3.874	3.602	5.615
96	3.763	5.826	2.507	5.335	2.591	4.207	1.692	4.664
97	1.708	5.712	2.661	5.195	3.341	5.970	3.163	5.519
98	3.763	5.039	1.928	5.291	3.485	5.767	3.416	6.372
99	1.878	5.716	3.552	3.954	4.194	5.209	3.468	6.051
100	3.868	3.937	3.671	5.092	3.514	4.791	2.176	4.019

References

1. Arivazhagan S, Shebiah RN, Nidhyanandhan SS, Ganesan L (2010) Fruit recognition using color and texture features. Journal of Emerging Trends in Computing and Information Sciences 1(2):90–94
2. Bora DJ, Gupta AK (2014) Impact of exponent parameter value for the partition matrix on the performance of fuzzy c means algorithm. arXiv:1406.4007
3. Cabanes G, Bennani Y, Destenay R, Hardy A (2013) A new topological clustering algorithm for interval data. Pattern Recogn 46(11):3030–3039
4. Chen JH, Hung WL (2015) An automatic clustering algorithm for probability density functions. J Stat Comput Simul 85(15):3047–3063
5. Cheng HD, Shan J, Ju W, Guo Y, Zhang L (2010) Automated breast cancer detection and classification using ultrasound images: a survey. Pattern Recogn 43(1):299–317
6. Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell (2):224–227
7. De Carvalho FDA, Pimentel JT, Bezerra LX (2007) Clustering of symbolic interval data based on a single adaptive l^1 distance. In: Neural networks 2007, international joint conference, pp 224–229
8. De Souza RM, de Carvalho FDA, Silva FC (2004) Clustering of interval-valued data using adaptive squared euclidean distances. In: International conference on neural, pp 775–780
9. Eleyan A, Demirel H (2011) Co-occurrence matrix and its statistical features as a new approach for face recognition. Turk J Electr Eng Comput Sci 19(1):97–107
10. Engin MA, Cavusoglu B (2019) Rotation invariant curvelet based image retrieval & classification via Gaussian mixture model and co-occurrence features. Multimedia Tools and Applications 78(6):6581–6605
11. Fadl S, Megahed A, Han Q, Qiong L (2020) Frame duplication and shuffling forgery detection technique in surveillance videos based on temporal average and gray level co-occurrence matrix. Multimedia Tools and Applications 1–25
12. Ge Y, Yin BC, Sun YF, Jing GD (2014) Expansion of 3d face sample set based on genetic algorithm. Multimedia Tools and Applications 70(2):781–797
13. He Z, Ho C-H (2019) An improved clustering algorithm based on finite Gaussian mixture model. Multimedia Tools and Applications 78(17):24285–24299

14. Hubert L (1977) Nominal scale response agreement as a generalized correlation. Br J Math Stat Psychol 30(1):98–103
15. Hubert L, Arabie P (1985) Comparing clusterings. J Classif 2:193–218
16. Hung WL, Yang JH, Shen KF (2016) Self-updating clustering algorithm for interval-valued data. Fuzzy Systems 1494–1500
17. Holland JH (1973) Genetic algorithms and the optimal allocation of trials. SIAM J Comput 2(2):88–105
18. Jeng JT, Chen CM, Chang SC, Chuang CC (2019) IPFCM Clustering algorithm under Euclidean and Hausdorff distance measure for symbolic interval data. Int J Fuzzy Syst 21:2102–2119
19. Kabir S, Wagner C, Havens TC, Anderson DT, Aickelin U (2017) Novel similarity measure for interval-valued data based on overlapping ratio. Fuzzy Systems IEEE International Conference 1–6
20. Lai CC (2005) A novel clustering approach using hierarchical genetic algorithms. Intelligent Automation & Soft Computing 11(3):143–153
21. Liu Y, Wu X, Shen Y (2011) Automatic clustering using genetic algorithms. Appl Math Comput 218(4):1267–1279
22. Malarvizhi N, Selvarani P, Raj P (2019) Adaptive fuzzy genetic algorithm for multi biometric authentication. Multimedia Tools and Applications 1–14
23. Mirkin BG, Chernyi LB (1970) Measurement of the distance between distinct partitions of a finite set of objects. Autom Tel 5:120–127
24. Nair LR, Subramaniam K, Venkatesan GP (2019) An effective image retrieval system using machine learning and fuzzy c-means clustering approach. Multimedia Tools and Applications 1–18
25. Nguyen-Trang T, Tai VV (2017) A new approach for determining the prior probabilities in the classification problem by Bayesian method. ADAC 11(3):629–643
26. Patel HN, Jain R, Joshi MV (2011) Fruit detection using improved multiple features based algorithm. Int J Comput Appl 13(2):1–5
27. Peng W, Li T (2006) Interval data clustering with applications. In: Tools with artificial intelligence. 2006, 18th IEEE international conference, pp 355–362
28. Pham-Gia T, Turkkan N, Tai VV (2008) Statistical discrimination analysis using the maximum function. Communications in Statistics—Simulation and Computation® 37(2):320–336
29. Rand WM (1971) Objective criteria for the evaluation of clustering methods. J Am Stat Assoc 66(336):846–850
30. Rodríguez SI, de Carvalho FD (2019) A new fuzzy clustering algorithm for interval-valued data based on city-block distance. In: 2019 IEEE International Conference on Fuzzy Systems, pp 1–6
31. Sara IRR, Francisco ATC (2019) Francisco. a new fuzzy clustering algorithm for interval-valued data based on city-block distance. In: 2019 IEEE International Conference on Fuzzy Systems, pp 1–9
32. Sato-Ilic M (2011) Symbolic clustering with interval-valued data. Procedia Computer Science 6:358–363
33. Selvi AS, Kumar KPM, Dhanasekeran S, Maheswari PU, Ramesh S, Pandi SS (2020) De-noising of images from salt and pepper noise using hybrid filter, fuzzy logic noise detector and genetic optimization algorithm (HFGOA). Multimedia Tools and Applications 79(5):4115–4131
34. Setia L, Teynor A, Halawani A, Burkhardt H (2006) Image classification using cluster cooccurrence matrices of local relational features. In: Proceedings of the 8th ACM international workshop on multimedia information retrieval, pp 173–182
35. Tai VV, NguyenTrang T (2018) Similar coefficient for cluster of probability density functions. Communications in Statistics-Theory and Methods 47(8):1792–1811
36. Tai VV, Trang TN (2018) Similar coefficient of cluster for discrete elements. Sankhya B 80(1):19–36
37. Tai VV, Trung NT, Vo-Duy T, Ho-Huu V, Nguyen-Trang T (2017) Modified genetic algorithm-based clustering for probability density functions. J Stat Comput Simul 87(10):1964–1979
38. Tai VV (2017) L1-distance and classification problem by bayesian method. J Appl Stat 44(3):385–401
39. Tai VV, Phamtoan D, Tranthituy D (2019) Automatic genetic algorithm in clustering for discrete elements. Communications in Statistics-Simulation and Computation 1–16
40. Tai V, Phamtoan D, Lehoang T, Nguyentrang T (2020) An automatic clustering for interval data using the genetic algorithm. Ann Oper Res. https://doi.org/10.1007/s10479-020-03606-8
41. Zhang X, Jian M, Sun Y, Wang H, Zhang C (2020) Improving image segmentation based on patch-weighted distance and fuzzy clustering. Multimedia Tools and Applications 79(1-2):633–657
42. Zhao Y, Guo Y, Sun R, Liu Z, Guo D (2019) Unsupervised video summarization via clustering validity index. Multimedia Tools and Applications 1–14
43. Zhou XG, Lu M, Huang XX (2018) C-means clustering algorithm based on intuitionistic fuzzy sets and its application in satisfaction evaluation. Journal of Information Hiding and Multimedia Signal Processing 9(2):484–495