I.J. Intelligent Systems and Applications, 2019, 6, 49-68
Published Online June 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2019.06.05

A Comparative Analysis of Firefly and Fuzzy-Firefly based Kernelized Hybrid C-Means Algorithms

[1]B.K. Tripathy, [2]Anmol Agrawal, [3]A. Jayaram Reddy
[1,2]School of Computer Science and Engineering, VIT, Vellore, 632014, India
E-mail: [1]tripathybk@vit.ac.in, [2]anmol98agr@gmail.com, [3]ajayaramreddy@vit.ac.in

Received: 28 July 2018; Revised: 15 September 2018; Accepted: 31 October 2018; Published: 08 June 2019

Abstract**—**In most of the clustering algorithms, the assignment of initial centroids is performed randomly, which affects both the final outcome and the number of iterations required. Another aspect of the approaches in clustering algorithms is the use of Euclidean distance as the measure of similarity between data points, which is handicapped by linear separability of input data. The purpose of this paper is to combine suitable techniques so that both the above problems can be handled suitably leading to efficient algorithms. For the initial assignment of centroids we use Firefly and Fuzzy Firefly algorithms. We replace the Euclidean distance by Kernels (Gaussian and Hyper-tangent) leading to hybridized versions. For experimental analysis we use five different images from different domains as input. Two efficiency measures; Davis Bouldin index (DB) and Dunn index (D) are used for comparison. The tabular values, their graphical representations and output images are generated to support the claims. The analysis proves the superiority of the optimized algorithms over their existing counterparts. We also find that Hyper-tangent kernel with Rough Intuitionistic Fuzzy C-Means algorithm using Fuzzy Firefly algorithm produces the best results and has a much faster convergence rate. The analysis of medical, satellite or geographical images can be done more efficiently using the proposed optimized algorithms. It is supposed to play an important role in image segmentation and analysis.

Index Terms**—**Data Clustering, Image segmentation, Kernel function, Firefly, Fuzzy Firefly, DB Index, Dunn Index.

I. Abbreviations Used in the Paper

FCM: Fuzzy C-Means.
IFCM: Intuitionistic Fuzzy C-Means.
RFCM: Rough Fuzzy C-Means.
RIFCM: Rough Intuitionistic Fuzzy C-Means.
FCMFA: Fuzzy C-Means with Firefly algorithm.
FCMFFA: Fuzzy C-Means with Fuzzy Firefly algorithm.
GKFCM: Gaussian Kernelized Fuzzy C-Means.
HKFCM: Hyper-tangent Fuzzy C-Means.
IFCMFA: Intuitionistic Fuzzy C-Means with Firefly algorithm.
IFCMFFA: Intuitionistic Fuzzy C-Means with Fuzzy Firefly algorithm.
GKIFCM: Gaussian Kernelized Intuitionistic Fuzzy C-Means.
HKIFCM: Hyper-tangent Kernelized Intuitionistic Fuzzy C-Means.
RFCMFA: Rough Fuzzy C-Means with Firefly algorithm
RFCMFFA: Rough Fuzzy C-Means with Fuzzy Firefly algorithm.
GKRFCM: Gaussian Kernelized Rough Fuzzy C-Means
HKRFCM: Hyper-tangent Rough Fuzzy C-Means
RIFCMFA: Rough Intuitionistic Fuzzy C-Means with Firefly Algorithm.
RIFCMFFA: Rough Intuitionistic Fuzzy C-Means with Fuzzy Firefly Algorithm.
GKRIFCM: Gaussian Kernelized Rough Intuitionistic Fuzzy C-Means.
HKRIFCM: Hyper-tangent Kernelized Rough Intuitionistic Fuzzy C-Means.

II. Introduction

Data clustering techniques are widely used in image segmentation over the past decade. Image segmentation involves the splitting and grouping of similar pixels of an image. With respect to position of elements in various clusters, clustering techniques can be categorized as: (a) Hard clustering and (b) Soft clustering. The data points in the case of hard clustering, can belong to at most one cluster i.e. they either belong to the cluster or not. In the case of soft or fuzzy clustering, the data points can belong to more than one clusters based on certain membership values. They use the Fuzzy Set concept [1]. Fuzzy C-Means (FCM) [2] is one of the simplest and most popular fuzzy clustering algorithm that uses the fuzzy set concept. Later, intuitionistic fuzzy sets [3] and rough sets were introduced [4]. Applying these models, several new clustering algorithms were developed such as Intuitionistic Fuzzy C-Means (IFCM) [5] that used the concept of intuitionistic fuzzy sets, Rough Fuzzy C-Means (RFCM) [6,7] that used the concept of rough fuzzy sets and, Rough Intuitionistic Fuzzy C-Means (RIFCM) [8], that used the concept of both, intuitionistic fuzzy sets and rough fuzzy sets.

In all these mentioned clustering algorithms, the Euclidean metric was used as a similarity measure. The

Euclidean distance-based clustering algorithms have the problem of linearly separable datasets. However, this issue was corrected by using the kernel function. The kernel function projects the feature space into a higher dimension by applying an appropriate non-linear mapping function which ensures that the complex clusters are linearly separable which are otherwise not linearly separable in its original feature space. Thus, in an attempt to avail this generality several kernel function based algorithms have been developed [9]. Some of these algorithms are the Kernel based K-means clustering using rough sets [10], Kernel based rough fuzzy c-means algorithm [11], Kernel based Rough Intuitionistic Fuzzy C-Means algorithm [12]. A comparative analysis of uncertainty-based kernelized c-means algorithms has been provided in [13]. Fuzzy clustering algorithm for multidimensional data on ordinary scale [14] was proposed in 2017. All these algorithms involved random initialization of cluster centroids. This resulted in slow convergence and hence led to more computational cost.

III. Related Work

In 2009, a metaheuristic inspired by the flashing behaviour of fireflies was proposed [15]. An improved version of this algorithm, namely, the Fuzzy Firefly algorithm was proposed by T. Hassanzadeh [16]. Stabilization of Rough Sets Based clustering algorithm using Firefly algorithm was proposed by Jain [17] where the firefly algorithm was used to assign the initial cluster centroids. Image Segmentation using Hybridized Firefly Algorithm and Intuitionistic Fuzzy C-Means was proposed by Chinta [18], where IFCM was combined with Firefly algorithm. In this paper, we make a comparative analysis of relative efficiencies of the hybrid algorithms obtained from basic FCM, IFCM, RFCM and RIFCM algorithm as well as their kernelized versions combined with two optimisation algorithms; firefly and fuzzy firefly for selection of initial centroids of clusters and make a comparative analysis of their performances. Two performance indices, Davis Bouldin index (DB) [19] and Dunn index (D) [20] has been used as efficiency measures of these algorithms. We have also established the relative relations between these algorithms. In section 2, we describe the various algorithms used in this analysis. In section 3 and 4, we discuss the methodology and analyze the results. The summary of our analysis is discussed in section 5 and section 6 contains the conclusion.

IV. Definition and Notation

A. Clustering Algorithms

Clustering can be considered as the most important unsupervised learning problems. It is defined as the unsupervised classification of observations, data items, or feature vectors into groups (clusters) [21]. The following clustering algorithms have been used in this paper.

a. Fuzzy C-Means:

In Fuzzy C-Means, a data point may belong to multiple clusters. Each element has a membership value associated with it. This membership value is used to assign the data element to the clusters. The performance of clusters is measured by the objective functions

$$J = \sum_{i=1}^{c} \sum_{k=1}^{n} (\mu_{ik})^m d^2(x_k, v_i) \tag{1}$$

The clusters are expected to be compact and thus should have minimum J value.

ALGORITHM:

1. *Assign initial 'c' cluster centroids where 'c' is the number of clusters.*
2. *Calculate the distance d_{ik} between the data points x_k and centroids v_i using* Euclidean function or some other appropriate distance measure.
3. *Compute μ (membership matrix) as:*

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}}} \tag{2}$$

Here, value of m=2 ('m' is called the fuzzifier).

4. *Calculate the cluster centroid as follows*

$$v_i = \frac{\sum_{j=1}^{N} (\mu_{ij})^m x_j}{\sum_{j=1}^{N} (\mu_{ij})^m} \tag{3}$$

5. *Repeat the above steps until* $\left\| U^{(k+1)} - U^{(k)} \right\| < \varepsilon$.

b. Intuitionistic Fuzzy C-Means:

The Intuitionistic Fuzzy C- Means algorithm uses a new parameter known as 'hesitation value' which improves the accuracy of the clustering.

ALGORITHM

1. *Assign initial centers for 'c' clusters.*
2. *Calculate the distance d_{ik} between the data points x_k and centroids v_i using Euclidean function or some other appropriate distance measure.*
3. *Compute U (membership matrix) using (2)*
4. *Compute the hesitation matrix π as:*

$$\pi_A(x) = 1 - \mu_A(x) - \frac{1-\mu_A(x)}{1+\lambda\mu_A(x)}, \forall x \tag{4}$$

5. *Compute the modified membership matrix using*

$$\mu_{ik}(x) = \mu_{ik}(x) + \pi_{ik}(x), \forall i, k, x \tag{5}$$

6. *Calculate the new centroids of the cluster using:*

$$v_i = \frac{\sum_{j=1}^{N} (\mu'_{ij})^m x_j}{\sum_{j=1}^{N} (\mu'_{ij})^m} \tag{6}$$

7. *Calculate the new partition matrix by following the steps ii to v.*
8. *If* $\left\| U^{(k+1)} - U^{(k)} \right\| < \varepsilon$ *, then stop, else repeat from step iv.*

c. Rough Fuzzy C-Means:

Rough Fuzzy C-Means clustering algorithm combines the concepts of rough set and fuzzy set theory. In rough sets, the concepts of lower and upper approximations deal with uncertainty, vagueness and incompleteness. The concept of membership function in fuzzy set helps to enhance and evaluate overlapping clusters.

ALGORITHM:

1. *Assign initial means* v_i *for c clusters.*
2. *Compute* μ_{ik} *(membership matrix) using (2)*
3. *Let* μ_{ik} *be the maximum and* μ_{jk} *be the next to maximum membership values of data points* x_k *to cluster centroids* v_i *and* v_j *.*
4. *If* $\mu_{ik} - \mu_{jk} < \varepsilon$ *then*
5. $x_k \in \overline{B}U_i$ *and* $x_k \in \overline{B}U_j$ *and* x_k *cannot be a member of any lower approximation.*
6. *Else* $x_k \in \underline{B}U_i$
7. *Calculate the new cluster means by using (7) where* $0 \le w_{low}, w_{up} \le 1$ *such that* $w_{low} + w_{up} = 1$

$$v_i = \begin{cases} w_{low} \cdot \dfrac{\sum_{x_k \in \underline{B}U_i} x_k}{|\underline{B}U_i|} + w_{up} \dfrac{\sum_{x_k \in \overline{B}U_i - \underline{B}U_i} \mu_{ik}^m x_k}{\sum_{x_k \in \overline{B}U_i - \underline{B}U_i} \mu_{ik}^m}, & if\ \underline{B}U_i \ne \phi \wedge \overline{B}U_i - \underline{B}U_i \ne \phi; \\ \dfrac{\sum_{x_k \in \overline{B}U_i - \underline{B}U_i} \mu_{ik}^m x_k}{\sum_{x_k \in \overline{B}U_i - \underline{B}U_i} \mu_{ik}^m}, & if\ \underline{B}U_i = \phi \wedge \overline{B}U_i - \underline{B}U_i \ne \phi; \\ \dfrac{\sum_{x_k \in \underline{B}U_i} x_k}{|\underline{B}U_i|}, & ELSE. \end{cases} \tag{7}$$

8. *Repeat from step ii until the terminating condition is satisfied or until there are no more assignment of objects*

d. Rough Intuitionistic Fuzzy C-Means:

Rough Intuitionistic Fuzzy C-Means was developed in 2013. In RIFCM, each cluster can be defined by three properties (i) a centroid, (ii) a crisp lower approximation and (iii) an intuitionistic fuzzy boundary.

ALGORITHM:

1. *Select c objects from the data set and assign one each to the c clusters as initial centroids*
2. *Compute* d_{ik} *the distance between the data points* x_k *and the centroid* v_k *by using some appropriate distance measure.*
3. *Compute the initial matrix U*
4. *If* $d_{ik} = 0$ *or* $x_k \in \underline{B}U_i$ *then* $\mu_{ik} = 1$*. Else* μ_{ik} *is computed by using the formula (2).*
5. *Compute* π_{ik} *by using equation (4).*
6. *Compute* μ'_{ik} *by the formula (8) and normalize*

$$\mu'_{ik}(x) = \mu_{ik}(x) + \pi_{ik}(x), \forall i, k, x \tag{8}$$

7. *Let* μ'_{ik} *be the maximum and* μ'_{jk} *be the next to maximum values of the object* x_k *to the clusters with centroids* v_i *and* v_j *respectively among all the clusters*
8. *If* $\mu'_{ik} - \mu'_{jk} < \varepsilon$ *(for some preassigned value* ε *) then* $x_k \in \overline{B}U_i$ *and* $x_k \in \overline{B}U_j$ *and* x_k *cannot be a member of any lower approximation*
9. *Else* $x_k \in \underline{B}U_i$
10. *Calculate the new cluster centres by using the following formula (9), where* $0 \le w_{low}, w_{up} \le 1$ *such that* $w_{low} + w_{up} = 1$

$$v_i = \begin{cases} w_{low} \dfrac{\sum_{x_k \in \underline{B}U_i} x_k}{|\underline{B}U_i|} + w_{up} \dfrac{\sum_{x_k \in \overline{B}U_i \setminus \underline{B}U_i} (\mu'_{ik})^m x_k}{\sum_{x_k \in \overline{B}U_i \setminus \underline{B}U_i} (\mu'_{ik})^m}, & if\ \underline{B}U \ne \phi \wedge \overline{B}U_i - \underline{B}U_i \ne \phi; \\ \dfrac{\sum_{x_k \in \overline{B}U_i \setminus \underline{B}U_i} (\mu'_{ik})^m x_k}{\sum_{x_k \in \overline{B}U_i \setminus \underline{B}U_i} (\mu'_{ik})^m}, & if\ \underline{B}U_i = \phi \wedge \overline{B}U_i - \underline{B}U_i \ne \phi; \\ \dfrac{\sum_{x_k \in \underline{B}U_i} x_k}{|\underline{B}U_i|}, & ELSE. \end{cases} \tag{9}$$

11. *Repeat steps 2 to 9 until the difference between two consecutive values of U is less than a preassigned value.*

B. Optimization Algorithms:

a. Firefly Algorithm:

Firefly Algorithm was proposed by Yang (2009) [15]. It is a bio-inspired meta-heuristic which mimics the behaviour of fireflies. Biologically, fireflies are attracted to luminous objects. In this algorithm, each firefly has its own brightness value and hence attracts all the other fireflies having lower brightness. The movement of the brightest firefly is random. The degree of attraction between two fireflies varies inversely to the distance between them. The brightness of a firefly is computed using an objective function which is problem-specific. The attractiveness (*β)* between two fireflies is determined by the formula:

$$\beta(r_{i,j}) = \beta_0 e^{-\gamma r^2_{i,j}} \tag{10}$$

Here, β_0 is the initial attractiveness value, γ is the co-efficient of light absorption and $\gamma_{i,j}$ is the Euclidean distance between the two fireflies *i* and *j*. In the implementation of this algorithm, we take β_0 =1 and γ is in the range 0.01 to 100.

$$x_i = x_i + \beta_0 e^{-\gamma r^2_{i,j}} (x_i - x_j) + \alpha(rand - 1/2) \tag{11}$$

The above equation is for the movement of firefly *i* to the brighter firefly *j*. α ϵ [0,1] denotes the randomization parameter. rand is a random number generator function uniformly distributed in the range [0,1]. This ensures that the fireflies are not stuck at a local optimum.

ALGORITHM:

1. *Define the initial parameters.*
2. *Generate initial population of fireflies* x_i , *i = 1, 2…n*
3. *Compute the light intensities* I_i *at* x_i *, i = 1, 2…n*
4. *Repeat:*

For i = 1to n (n being the number of fireflies)
For j = 1 to n
If $(I_j > I_i)$
Move firefly i towards firefly j in d dimensions
Attractiveness varies with distance r via $[-\gamma r]$
Compute new solutions and update light intensities
If there is no firefly brighter than I_i *, move* I_i *randomly*
Assign ranks to the fireflies and find the current best until either maximum iteration limit is reached or minimum change of the objective function occurs.

The most important characteristic of Firefly algorithm is its ability to avoid the local optima as fireflies covering the whole solution space are initialized randomly. This ensures that at-least one firefly has a high intensity. All the other fireflies start moving towards this brightest firefly. Since every firefly is associated with a degree of randomness, the whole solution space is thoroughly covered by the population of fireflies.

b. Fuzzy-Firefly Algorithm:

The fuzzy firefly algorithm, proposed by T. Hassanzade in 2014 [16], increased the area of exploration by each firefly and improves the convergence rate. To do this, in each iteration k-brighter fireflies are selected which attract the other less brighter fireflies. Here, k is a user-defined parameter that depends upon the complexity of the problem and the swarm population. Taking h as a brighter firefly with fitness value f(p_h) and the local optimum firefly has its fitness value f(p_g), the degree of attractiveness of the firefly h is defined as:

$$\psi(h) = \frac{\beta}{f(p_h) - f(p_g)} \tag{12}$$

Here, β is defined as

$$\beta = \frac{f(p_g)}{\ell} \tag{13}$$

Here, *l* is a user-set parameter. The movement of a less brighter firefly i towards one of the k-brighter fireflies h is formulated as:

$$X_i = x_i + \left(\beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + \sum_{h=1}^{k} \psi(h)\beta_0 e^{-\gamma r_{ih}^2}(x_h - x_i)\right).\alpha(rand - 1/2) \tag{14}$$

In the implementation of this algorithm, the value of β_0 is taken as 1, γ is taken in the range [0.01,100], k is taken as 15 and α ϵ [0,1].

C. Similarity Measures:

Out of the several measures used to find similarity between two data points, the most popular one is the Euclidean distance, has the limitation of being sensitive to initial assignment of centroids and being stable for only linearly separable data points. The second limitation can be solved by Kernel based clustering approach wherein non-linear boundaries are created to segregate the data points efficiently. This is possible by transforming the data points present in the ordinary plane to a higher dimensional feature plane known as the kernel space. Some non-linear mapping function is used to ensure this kind of transformation. This subsection describes some of the similarity measures.

a. Euclidean Distance:

The Euclidean distance d_{ij} between any two data points *i* and *j* is described as:

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2} \tag{15}$$

This formula holds true across any n-dimensional space. Here $x_1, x_2, ...x_n$ and $y_1, y_2, ...y_n$ are attributes of x and y respectively.

b. Kernel Distance:

Let 'a' denote a data point. $\phi(a)$ denotes the transformation of 'a' from ordinary plane to a higher dimension kernel space. Inner product space is computed as. $K(a,b) = \phi(a).\phi(b)$.

Let $a = (a_1, a_2, ...a_n)$ and $b = (b_1, b_2, ...b_n)$ be two points in the n-dimensional space. There are several kernel functions available in the literature. The Kernel functions to be used in this paper are as follows:

1. Gaussian Kernel:

$$G(a,b) = \exp\left(-\frac{\sum_{i=1}^{n}(a_i - b_i)^2}{2\sigma^2}\right) \tag{16}$$

2. Hyper-tangent Kernel:

$$H(a,b) = 1 - \tan h\left(-\frac{\sum_{i=1}^{n}(a_i - b_i)^2}{2\sigma^2}\right) \tag{17}$$

Where,

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}\|a_i - a'\|^2 \text{ and } a' = \frac{1}{N}\sum_{i=1}^{N} a_i \tag{18}$$

The general form of kernel distance formula is denoted by $D(x,y) = K(x,x) + K(y,y) - 2K(x,y)$. However, we know that $K(x,x) = 1$(Property of Similarity). Thus, the kernel distance becomes $D(x,y) = 2(1 - K(x,y))$

D. Performance Indices:

Performances indices are used for measuring the efficiency of clustering algorithms. There are several performance indices available in the literature. The Davis-Bouldin (DB) and Dunn (D) indexes are some of the most commonly used performance indices. Their results depends on the number of clusters required.

a. Davis-Bouldin (DB) index:

The DB index is the ratio of sum of distance within the cluster to between the clusters. It is given by the formula:

$$DB = \frac{1}{c}\sum_{i=1}^{c}\max_{k \neq i}\left\{\frac{S(v_i) + S(v_k)}{d(v_i, v_k)}\right\}, \; for \; 1 < k, i < c. \tag{19}$$

It aims to minimize the separation within the cluster and maximize the between cluster separation. Hence a low value of DB index indicates good clustering.

b. Dunn (D) index:

D index is used to identify the compact and separated clusters. It is calculated as:

$$Dunn = \min_{i}\left[\min_{k \neq i}\left\{\frac{d(v_i, v_k)}{S(v_l)}\right\}\right], \; for 1 < k, i, l < c \tag{20}$$

It objective is to maximize the between-cluster distance and minimize the within-cluster distance. Therefore, a greater D index value indicates higher efficiency.

V. Methodology

The swarm of fireflies is initialized to random values and the metaheuristic is allowed to calculate the intensity of each firefly. These fireflies are allowed to move around following equation (10) and their intensities are recalculated. At the end of this cycle, the best firefly (centroid) values are passed as the initial values of clustering algorithms. We have used this technique for the Kernelized (Gaussian and Hyper-tangent) versions of the algorithms FCM, IFCM, RFCM, RIFCM and made a comparative analysis among themselves and the existing clustering algorithms in this direction. It is observed that the algorithms obtained through our approach not only show significant improvement (verified through the computation of the measuring indices DUNN and DB and results obtained) but also their rates of convergences are high. In this paper we have used five different type of images for our experimental purpose.

VI. Results and Analysis

Implementation of algorithms have been carried out in Python 3.6 with Spyder 3.1.4 IDE. NumPy library has been used in the implementation of algorithms and matplotlib library has been used to plot the output figures. In the experimental analysis, we have used five different kinds of images in order to make the study extensive.

ORIGINAL DATASET:

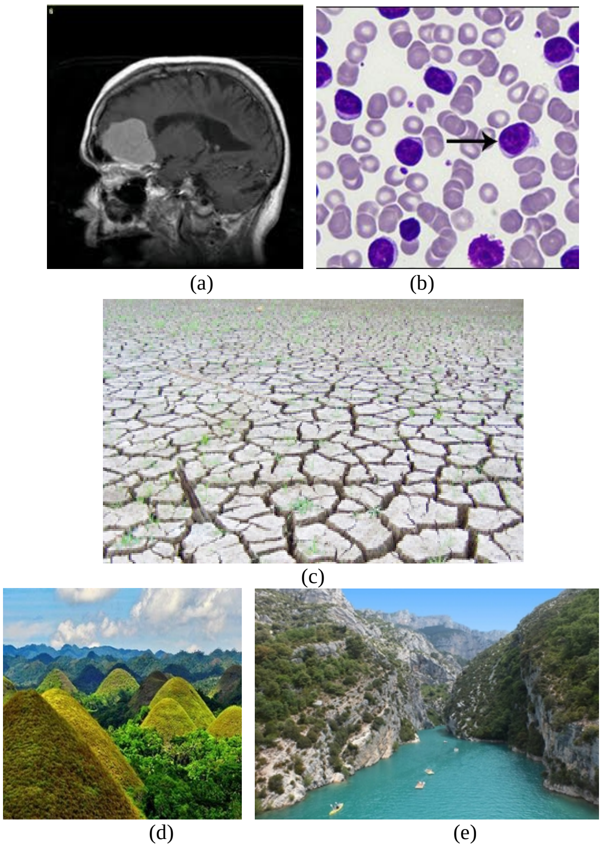


Fig.1. Original Dataset.

Figure (a) (221 x 228) represents MRI scan of a section of a human brain. The lighter region on the forehead region indicates presence of a tumour. Figure (b) (250 x 250) represents the proliferation of abnormal WBCs among the RBCs and normal WBCs. Figure (c) (307 x 154) is a picture of a draught area. The cracks on the land helps in the comparison of various clustering algorithms. Figure (d) (250 x 250) represents a geographical image of hills. Figure (e) (309 x 212) represents a geographical image of river-valley. The image can be segmented into two major segments: (i) The river and the sky (blue) and (ii) The vegetation (green).

A. Segmentation of Tumour in Brain Mri Scan:

a. Using Euclidean distance:

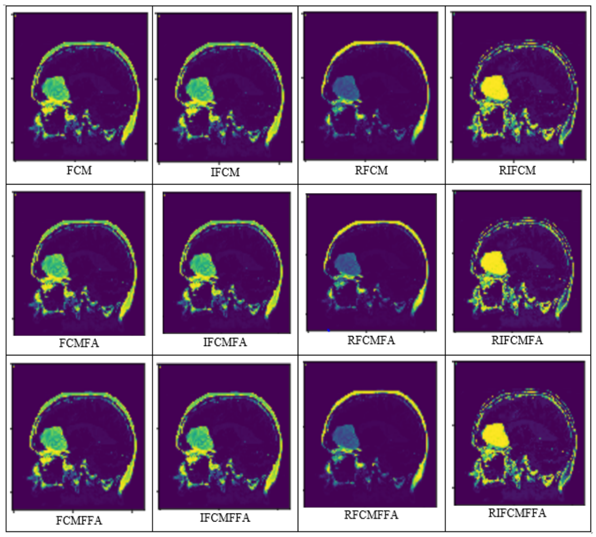


Fig.2. Segmentation outputs of tumour in brain MRI scan using Euclidean distance

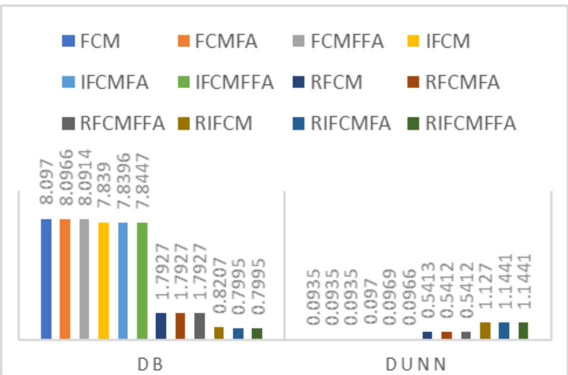


Fig.3. Comparison of performance of various algorithms suing the Euclidean metric

It can be inferred from the above output (Fig. 2) that FCM and IFCM produces roughly similar result while RFCM and RIFCM produces much better results. The output produced by FCM and IFCM are slightly blurred and noisy specially at the edges. Output produced by FCMFA, FCMFFA, IFCMFA, IFCMFFA, RFCMFA, RFCMFFA RIFCMFA, RIFCMFFA are marginally better than their original counterparts as is clear from the values of Dunn and DB indices and have a better convergence rate as shown in Fig 4. It can be observed that IFCM and IFCMFA works slightly better than IFCMFFA but have much higher convergence rate. In case of RIFCM, the convergence improves more rapidly than any other algorithm when combined with Firefly and Fuzzy-Firefly algorithm. The convergence rate is fastest in the case of IFCMFFA.

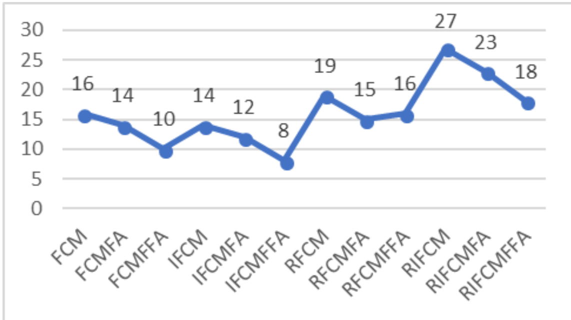


Fig.4. Comparison of number of iterations required for segmentation with respect to Euclidean distance

Overall, by looking at the performance indices, the following relation can be established:

FCM<FCMFA<FCMFFA,
IFCMFFA<IFCM<IFCMFA,
RFCM≈RFCMFA≈RFCMFFA,
RIFCM<RIFCMFA≈RIFCMFFA,
FCM<IFCM<RFCM<RIFCM.

b. Using Gaussian Kernel:

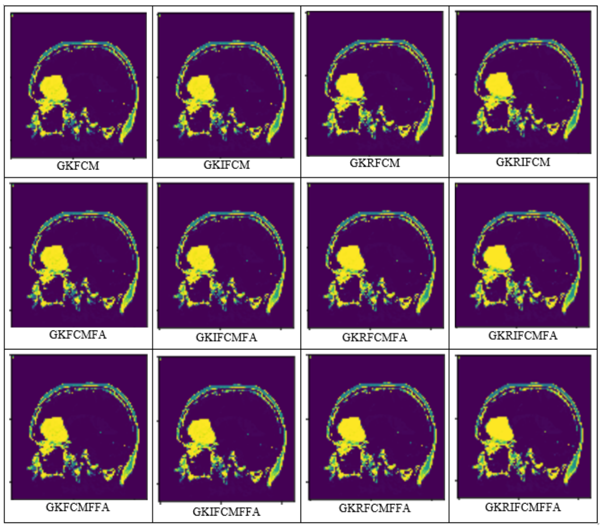


Fig.5. Segmentation output of tumour in brain MRI scan using Gaussian Kernel

Table 1. Performance analysis indices for brain tumour segmentation.

Dist. Function	Algorithm	Number of Cluster=3			Number of Cluster=4		
		#iteratio	DB	Dunn	#iteratio	DB	Dunn
Euclidean Distance	FCM	28	17.0474	0.0412	16	8.0970	0.0935
	IFCM	28	16.5964	0.0420	14	7.8390	0.0970
	RFCM	27	3.2600	0.2190	19	1.7927	0.5413
	RIFCM	19	2.0683	0.8514	27	0.8207	1.1270
	FCMFA	23	17.0471	0.0412	14	8.0966	0.0935
	IFCMFA	29	16.5955	0.0419	12	7.8396	0.0969
	RFCMFA	13	3.2600	0.2190	15	1.7927	0.5412
	RIFCMFA	11	1.4394	0.5200	23	0.7995	1.1441
	FCMFFA	19	17.0463	0.0412	10	8.0914	0.0935
	IFCMFFA	15	16.6106	0.0420	8	7.8447	0.0966
	RFCMFFA	21	3.2599	0.2190	16	1.7927	0.5412
	RIFCMFFA	11	1.4394	0.5200	18	0.7995	1.1441
Gaussian Kernel	GKFCM	20	0.0021	6.1558	23	0.0006	15.4141
	GKIFCM	23	0.1354	6.2586	18	0.0441	15.4779
	GKRFCM	48	0.0530	18.1913	21	0.0096	118.2917
	GKRIFCM	15	0.0117	82.1403	29	0.0035	297.9356
	GKFCMFA	14	0.0021	6.1560	14	0.0006	15.4147
	GKIFCMFA	11	0.1354	6.2586	13	0.0440	15.4762
	GKRFCMFA	14	0.0530	18.1914	10	0.0096	118.2918
	GKRIFCMFA	14	0.0117	82.1407	13	0.0035	314.9880
	GKFCMFFA	14	0.0021	6.1561	14	0.0006	15.4480
	GKIFCMFFA	12	0.1354	6.2586	11	0.0438	15.5861
	GKRFCMFFA	13	0.0530	18.1915	9	0.0096	118.3178
	GKRIFCMFFA	14	0.0112	86.3369	8	0.0034	315.0408
Hyper-tangent Kernel	HKFCM	18	0.0025	6.9640	31	0.0020	6.9641
	HKIFCM	18	0.1325	7.0495	13	0.0473	15.7339
	HKRFCM	33	0.0546	18.7862	15	0.0100	112.5565
	HKRIFCM	33	0.0103	98.3862	43	0.0036	315.0355
	HKFCMFA	13	0.0024	6.9648	15	0.0007	15.4989
	HKIFCMFA	16	0.1324	7.0495	11	0.0439	15.5437
	HKRFCMFA	19	0.0546	18.7862	12	0.0099	115.6368
	HKRIFCMFA	21	0.0103	98.4875	18	0.0034	315.0355
	HKFCMFFA	7	0.0024	6.9652	11	0.0007	15.5262
	HKIFCMFFA	11	3.4744	7.0494	9	3.4652	15.7382
	HKRFCMFFA	12	0.0546	18.7862	8	0.0099	115.6067
	HKRIFCMFFA	11	0.0103	98.4898	15	0.0034	315.0355

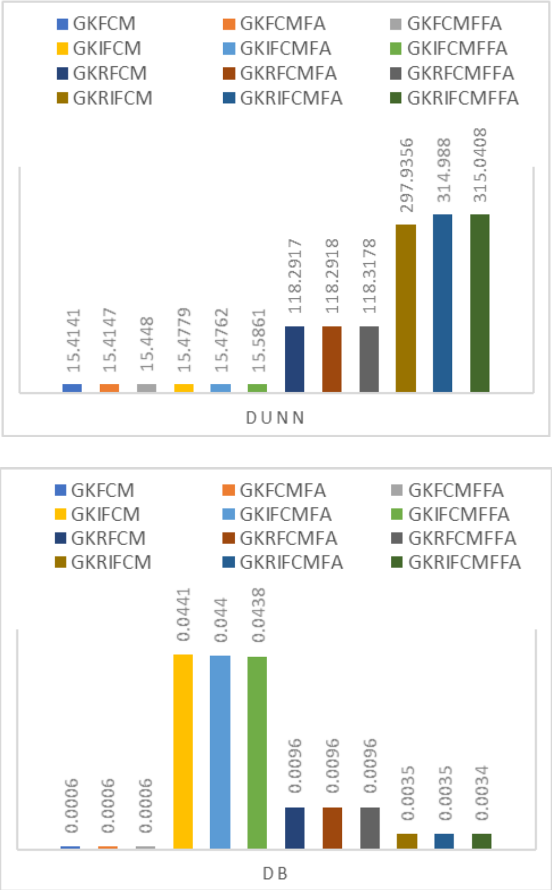


Fig.6. Comparison of performance of various algorithms using the Gaussian Kernel

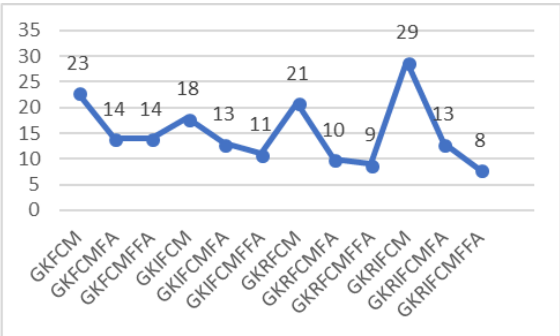


Fig.7. Comparison of number of iterations required for segmentation with respect to Gaussian Kernel

It can be easily inferred from table 1 that the results produced using Gaussian Kernel (Fig. 5) is much better than that produced using Euclidean distance (Fig. 2). The indices show that Firefly and Fuzzy Firefly versions of the algorithm works better than their conventional counterpart in all the three cases and show much better convergence rate. Figure 7 shows a significant improvement of convergence rate of GKRIFCM when combined with Firefly and Fuzzy Firefly algorithms. Thus, GKRIFCMFFA shows the best results as well as the best convergence rate. Finally, the following relations can be established based on the performance indices:

GKFCM<GKFCMFA<GKFCMFFA,
GKIFCM≈GKIFCMFA<GKIFCMFFA,
GKRFCM<GKRFCMFA<GKRFCMFFA,
GKRIFCM<GKRIFCMFA<GKRIFCMFFA
GKIFCM<GKFCM<GKRFCM<GKRIFCM

3. Using Hyper-tangent Kernel:

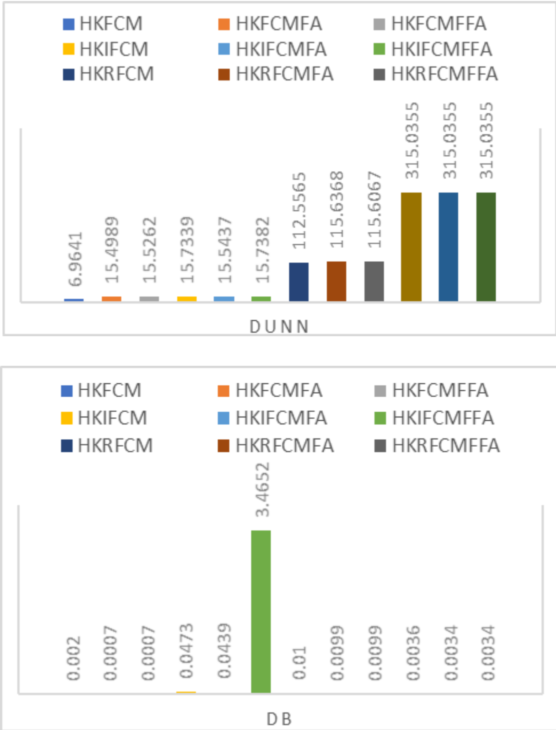


Fig.8. Comparison of performance analysis of various algorithms using the Hyper-tangent Kernel

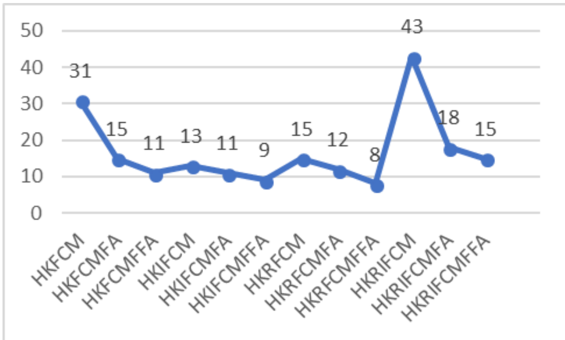


Fig.9. Comparison of number of iterations required for segmentation with respect to Hyper-tangent Kernel

It can be inferred from Fig. 8 that the results produced by Hyper-tangent kernel is almost similar to the results produced by the Gaussian Kernel. Even here, the differences in the output are not noticeable and we have to rely on the performance indices to compare them. Data shows that output produced by HKFCMFA, HKFCMFAA, HKRFCMFA, HKRFCMFAA are better than the unoptimized versions. However, it can be observed that HKIFCMFAA does not perform well when compared to HKIFCM and HKIFCMFA. Convergence rate is better in optimized versions of the algorithms with HKRIFCM showing substantial improvements. It can also be observed that HKRIFCMFA and HKRIFCMFFA shows the best results. The convergence rate is best for HKRFCMFFA, while slowest for HKRIFCM. The performance of the algorithms can be related as follows:

HKFCM<KFCMFA<HKFCMFFA,
HKIFCMFFA<HKIFCM<HKIFCMFA,
HKRFCM<HKRFCMFA<HKRFCMFFA,
HKRIFCM<HKRIFCMFA≈HKRIFCMFFA
HKIFCM<HKFCM<HKRFCM<HKRIFCM

B. *Segmentation Of Blood Cancer Cells:*

a. *Using Euclidean distance:*

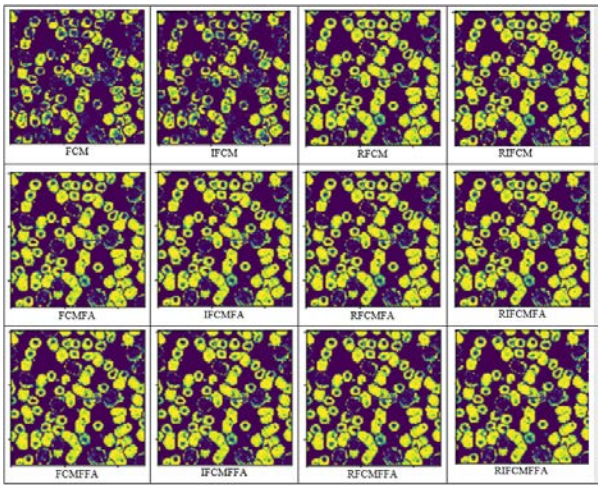


Fig.10. Output of segmentation of blood cancer cells using Euclidean distance

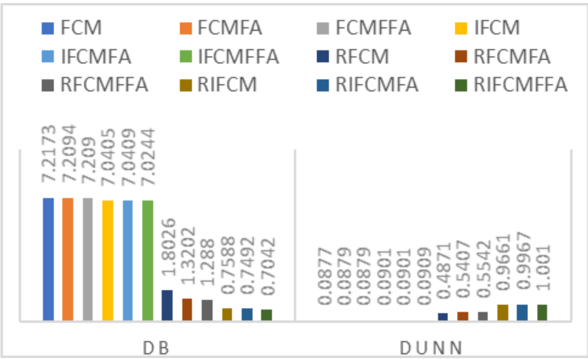


Fig.11. Comparison of performance of various algorithms with respect to Euclidean Distance

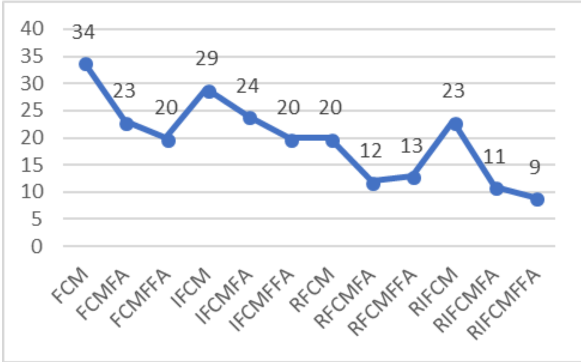


Fig.12. Comparison of number of iterations required for segmentation using to Euclidean distance

It can be easily observed from the above output that performance of FCM and IFCM are quite similar. RFCM and RIFCM produces much better result than FCM and IFCM. The firefly and fuzzy firefly versions of all the three algorithms outperform their unoptimized versions both in quality of the output and the convergence rate as is clear from Fig. 11 and 12. RIFCMFFA produces the best result while FCM produces the worst results both in terms of cluster quality and convergence rate amongst all the twelve cases. The following relations can thus be established:

FCM<FCMFA<FCMFFA,
IFCM<IFCMFA<IFCMFFA,
RFCM<RFCMFA<RFCMFFA,
RIFCM<RIFCMFA<RIFCMFFA,
FCM<IFCM<RFCM<RIFCM

b. *Using Gaussian Kernel:*

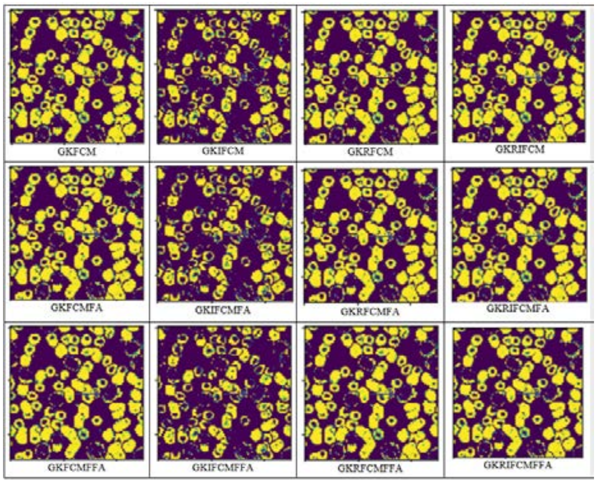


Fig.13. Output of segmentation of blood cancer cells using Gaussian Kernel

Table 2. Performance indices for blood cancer cells segmentation.

Dist. Function	Algorithm	Cluster=3			Cluster=4		
		#i	DB	Dunn	#i	DB	Dunn
Euclidean Distance	FCM	11	7.8496	0.1446	34	7.2173	0.0877
	IFCM	11	7.7675	0.1480	29	7.0405	0.0901
	RFCM	23	2.8246	0.2305	20	1.8026	0.4871
	RIFCM	21	0.8436	1.9683	23	0.7588	0.9661
	FCMFA	8	7.8494	0.1446	23	7.2094	0.0879
	IFCMFA	6	7.7673	0.1481	24	7.0409	0.0901
	RFCMFA	10	1.7249	0.9082	12	1.3202	0.5407
	RIFCMFA	10	0.8435	1.9648	11	0.7492	0.9967
	FCMFFA	8	7.8486	0.1446	20	7.2090	0.0879
	IFCMFFA	5	7.7666	0.1481	20	7.0244	0.0909
	RFCMFFA	5	1.7248	0.9083	13	1.2880	0.5542
	RIFCMFFA	5	0.8436	1.9647	9	0.7042	1.001
Gaussian Kernel	GKFCM	16	0.0010	19.2985	23	0.0006	8.8262
	GKIFCM	10	0.0549	19.3398	19	0.0421	8.9770
	GKRFCM	22	0.0090	167.3301	13	0.0106	125.5488
	GKRIFCM	16	0.0032	545.0577	44	0.0017	269.7745
	GKFCMFA	6	0.0008	19.2991	21	0.0006	8.8265
	GKIFCMFA	8	0.0549	19.3398	15	0.0420	8.8361
	GKRFCMFA	5	0.0089	167.3261	12	0.0043	84.9657
	GKRIFCMFA	8	0.0032	545.0582	39	0.0017	269.7745
	GKFCMFFA	6	0.0008	19.2985	20	0.0006	8.8241
	GKIFCMFFA	4	0.0549	19.3408	6	0.0225	68.3863
	GKRFCMFFA	6	0.0089	167.3011	7	0.0106	84.5302
	GKRIFCMFFA	7	0.0032	545.0714	13	0.0017	269.7762
Hyper-tangent Kernel	HKFCM	9	0.0010	21.1514	20	0.0007	9.2740
	HKIFCM	9	0.0541	21.2074	18	0.0462	9.2792
	HKRFCM	20	0.0072	145.1430	12	0.0045	76.9258
	HKRIFCM	7	0.0031	551.4250	10	0.0016	272.4641
	HKFCMFA	6	0.0010	21.1520	20	0.0006	9.2752
	HKIFCMFA	5	0.0541	21.2133	16	0.0462	9.2797
	HKRFCMFA	8	0.0092	166.5903	11	0.0045	76.9258
	HKRIFCMFA	7	0.0031	551.5176	10	0.0016	278.2701
	HKFCMFFA	5	0.0009	21.1529	6	0.0004	65.2348
	HKIFCMFFA	9	2.8761	21.2130	11	6.9347	9.2775
	HKRFCMFFA	5	0.0089	168.5143	6	0.0102	137.5797
	HKRIFCMFFA	7	0.0031	551.5298	7	0.0026	278.7326

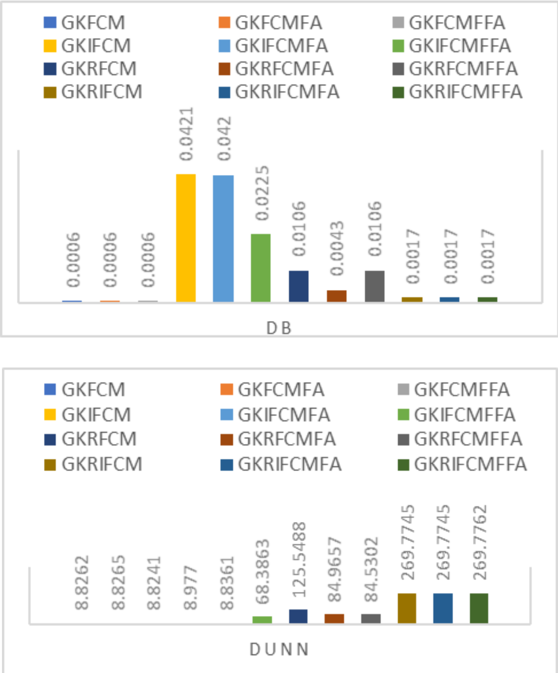


Fig.14. Comparison of performance of algorithms with respect to Gaussian Kernel

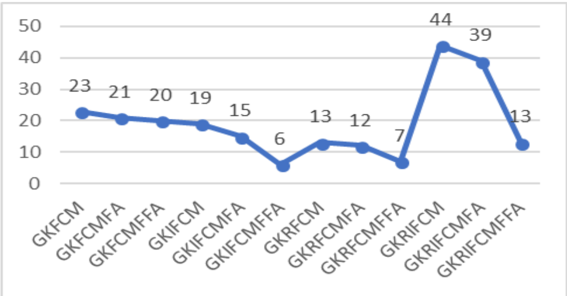


Fig.15. Comparison of number of iterations required for segmentation with respect to Gaussian Kernel

It can be inferred from Fig. 13 and 14 that the outputs are significantly better than those rendered using Euclidean measures. It is also evident that GKFCM, GKRFCM and GKRIFCM produces better results than GKIFCM. It can be established by referring to the performance indices that in GKFCM, GKRFCM and GKRIFCM both firefly and fuzzy firefly gives better results. In case of GKIFCM, it is difficult to establish any concrete relation. GKIFCMFFA shows the best convergence rate closely followed by GKRFCMFFA.

The optimized versions show better convergence rate in all the cases. We may conclude the following relations:

GKFCM<GKFCMFA<GKFCMFFA,
GKIFCM≈GKIFCMFA<GKIFCMFFA,
GKRFCM<GKRFCMFFA<GKRFCMFA,
GKRIFCM<GKRIFCMFA<GKRIFCMFFA,
GKIFCM<GKFCM<GKRFCM<GKRIFCM

c. Using Hyper-tangent Kernel

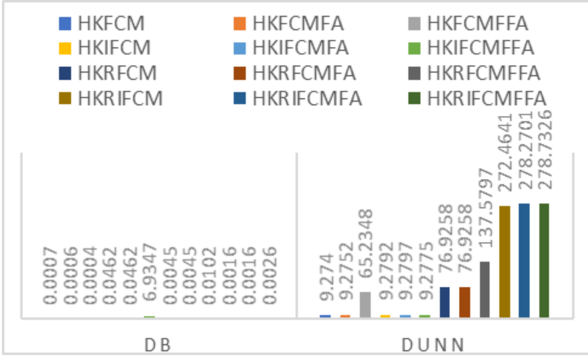


Fig.16. Comparison of performance of algorithms with respect to Hyper-tangent Kernel

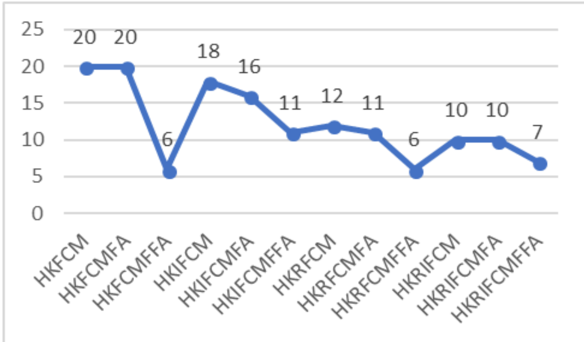


Fig.17. Comparison of number of iterations required for segmentation with respect to Hyper-tangent Kernel

These results are quite similar to those obtained using Gaussian Kernel. Results obtained by HKFCM, HKRFCM and HKRIFCM are evidently better than those obtained from HKIFCM. HKFCMFA, HKFCMFFA, HKRFCMFA, HKRFCMFFA, HKRIFCMFA, HKRIFCMFFA gives better results than the original algorithms. But in the case of HKIFCM, performance increases when combined with firefly algorithm but decreases marginally when combined with fuzzy firefly algorithm. HKFCMFFA and HKRFCMFFA show the best convergence rate closely followed by HKRIFCMFFA. Thus, the following relations can be established based on the performance indices.

HKFCM<HKFCMFA<HKFCMFFA,
HKIFCMFFA<HKIFCM<HKIFCMFA,
HKRFCM<HKRFCMFA<HKRFCMFFA,
HKRIFCM<HKRIFCMFA, HKRIFCMFFA,
HKIFCM<HKFCM<HKRFCM<HKRIFCM

C. Segmentation of Draught Image:

a. Using Euclidean Distance:

Table 3. Performance indices for draught image segmentation

Dist. Function	Algorithm	Cluster=3			Cluster=4		
		#i	DB	Dunn	#i	DB	Dunn
Euclidean Distance	FCM	25	9.9014	0.0828	56	8.3780	0.0893
	IFCM	21	9.6706	0.0851	38	8.1521	0.0914
	RFCM	24	2.6322	0.3126	49	2.2558	0.3630
	RIFCM	42	1.6525	0.5417	48	1.4608	0.5617
	FCMFA	16	9.8992	0.0829	53	8.3780	0.0893
	IFCMFA	21	9.6706	0.0852	26	8.1537	0.0929
	RFCMFA	13	2.6322	0.3126	13	2.1934	0.3631
	RIFCMFA	16	1.6456	0.5010	46	1.4413	0.5686
	FCMFFA	14	9.9013	0.0828	50	8.3779	0.0893
	IFCMFFA	11	9.6683	0.0854	24	8.1521	0.0914
	RFCMFFA	6	2.6196	0.3365	9	2.2018	0.3830
	RIFCMFFA	18	1.6451	0.5011	45	1.4413	0.5686
Gaussian Kernel	GKFCM	19	0.0016	8.3143	67	0.0010	9.3227
	GKIFCM	23	0.0804	8.3780	53	0.0523	9.7718
	GKRFCM	26	0.0139	49.8469	46	0.0092	55.6372
	GKRIFCM	31	0.0065	120.3396	57	0.0048	201.5259
	GKFCMFA	21	0.0015	8.3316	65	0.0009	9.4690
	GKIFCMFA	18	0.0804	8.3785	46	0.0523	9.7719
	GKRFCMFA	16	0.0137	51.6018	15	0.0090	55.9217
	GKRIFCMFA	16	0.0064	120.6207	24	0.0048	201.5264
	GKFCMFFA	15	0.0015	8.3310	52	0.0009	9.4669
	GKIFCMFFA	7	0.0803	8.3785	41	0.0528	9.2363
	GKRFCMFFA	14	0.0137	51.6008	11	0.0090	56.2344
	GKRIFCMFFA	6	0.0065	120.3574	18	0.0045	202.1196
Hyper-tangent Kernel	HKFCM	21	0.0019	8.4174	62	0.0012	9.4129
	HKIFCM	22	0.0839	8.4584	39	0.0532	9.1642
	HKRFCM	17	0.0137	52.3852	34	0.0087	94.2714
	HKRIFCM	30	0.0063	130.6972	67	0.0043	233.3251
	HKFCMFA	18	0.0019	8.4348	57	0.0011	9.4134
	HKIFCMFA	19	0.0839	8.4585	53	0.0532	9.1646
	HKRFCMFA	14	0.0137	54.6485	27	0.0087	94.2714
	HKRIFCMFA	25	0.0062	130.6972	47	0.0043	233.3251
	HKFCMFFA	16	0.0019	8.4350	57	0.0011	9.4176
	HKIFCMFFA	14	3.8229	8.5288	40	5.5002	9.8156
	HKRFCMFFA	11	0.0137	55.8882	19	0.0087	94.2717
	HKRIFCMFFA	23	0.0062	130.6972	13	0.0043	238.8526

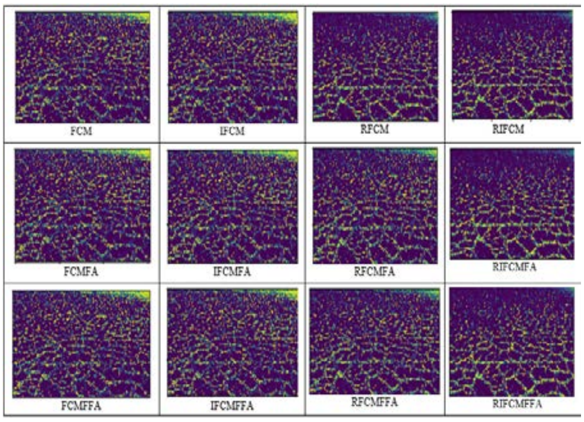


Fig.18. Output of segmentation of draught image using Euclidean distance.

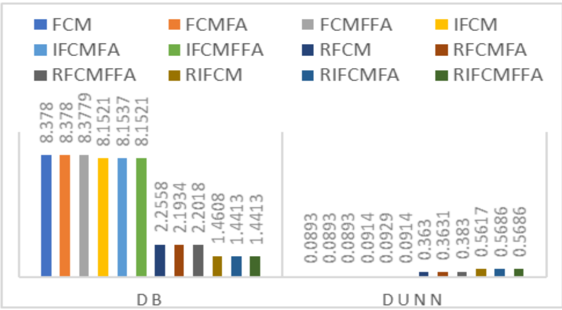


Fig.19. Comparison of performance of algorithms using Euclidean distance

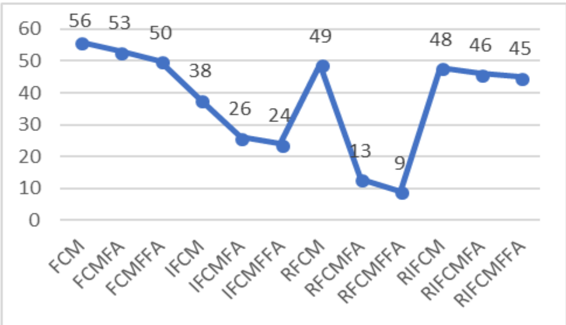


Fig.20. Comparison of number of iterations required for segmentation with respect to Euclidean distance

It can be observed from Fig. 18 that RIFCM performs much better than FCM, IFCM and RFCM. The output produced by FCM and IFCM are quite blurred and unrecognizable. The cracks in the ground are much sharper and distinguishable in case of RFCM and even better in case of RIFCM. The performance values obtained supports our observation. The firefly and fuzzy firefly versions show marginal improvements in the cluster quality and significant improvement in the convergence rate. RFCMFFA gives the best convergence rate, which is significantly better than its original version (RFCM). Thus, the following relation holds good:

FCM<FCMFA<FCMFFA,
IFCM<IFCMFA<IFCMFFA,
RFCM<RFCMFA<RFCMFFA,
RIFCM<RIFCMFA<RIFCMFFA,
FCM<IFCM<RFCM<RIFCM

b. Using Gaussian Kernel:

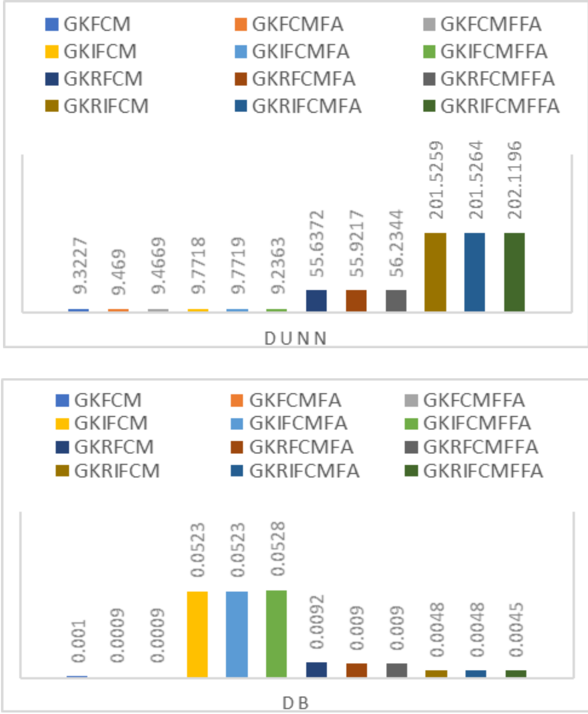


Fig.21. Comparison of performance of algorithms with respect to Gaussian Kernel

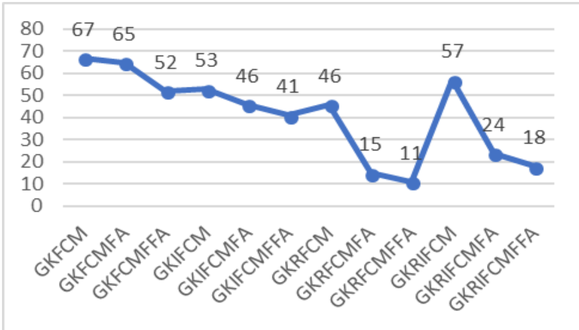


Fig.22. Comparison of number of iterations required for segmentation with respect to Gaussian Kernel

It can be inferred from the values of these indices that both FCM, RFCM and RIFCM perform better than IFCM. Firefly and fuzzy firefly algorithms improve the result in all the three cases. The convergence rate also shows significant improvements in the optimized versions particularly in GKRFCM and GKRIFCM, with GKRFCMFFA showing the best result among the twelve cases. The performance of algorithms can be related as follows:

GKFCM<GKFCMFA<GKFCMFFA,
GKIFCM<GKIFCMFA<GKIFCMFFA,
GKRFCM<GKRFCMFA<GKRFCMFFA,
GKRIFCMFFA<GKRIFCM< GKRIFCMFA,
GKIFCM<GKFCM<GKRFCM<GKRIFCM

c. Using Hyper-tangent Kernel:

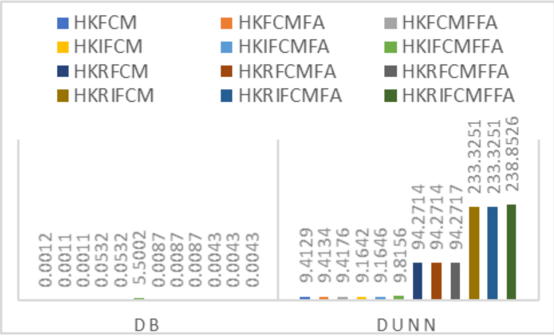


Fig.23. Comparison of performance of algorithms with respect to Hyper-tangent Kernel

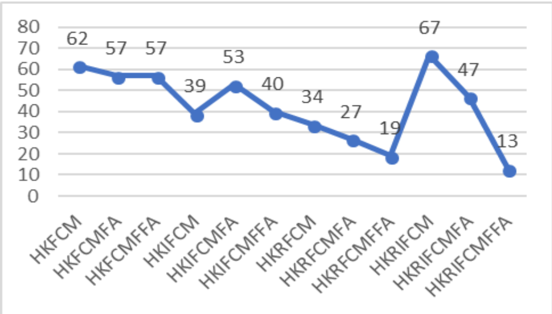


Fig.24. Comparison of number of iterations required for segmentation with respect to Hyper-tangent Kernel

The output produced by Hyper-tangent kernel is similar to that produced by Gaussian Kernel. Convergence rate also shows good improvement. HKRIFCM does not show any stable relation both in the cluster quality and the convergence rate. The improvement in convergence rate from HKRIFCM to HKRIFCMFFA is quite remarkable. Finally, the following relation can be established:

HKFCM<HKFCMFA<HKFCMFFA,
HKIFCMFFA<HKIFCM<HKIFCMFA,
HKRFCM<HKRFCMFA<HKRFCMFFA,
HKRIFCM<HKRIFCMFA<HKRIFCMFFA,
HKIFCM<HKFCM<HKRFCM<HKRIFCM

D. Segmentation of Geographical Image- Hills

a. Using Euclidean Distance:

It can be observed that the results produced by FCM and IFCM are quite similar. Also results produced by RFCM and RIFCM are very similar and it is quite difficult to differentiate between them. So, we rely on the performance indices for our analysis. The results produced by Firefly and Fuzzy Firefly are better than their existing counterparts both in terms of quality of cluster and convergence rate. The results produced by RIFCMFFA is better than all other algorithms. IFCMFFA shows the best convergence rate followed by RFCMFFA. The following relation can be established:

FCM<FCMFA< FCMFFA,
IFCM<IFCMFA≈IFCMFFA,
RFCM<RFCMFA<RFCMFFA,
RIFCM<RIFCMFA<RIFCMFFA,
FCM<IFCM<RFCM<RIFCM

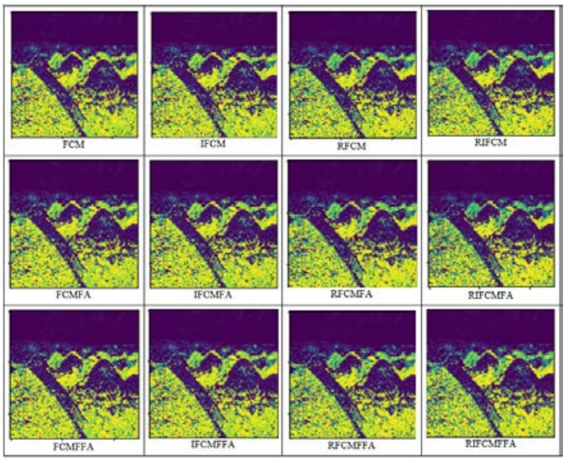


Fig.25. Segmentation outputs of geographical image (hills) using Euclidean distance

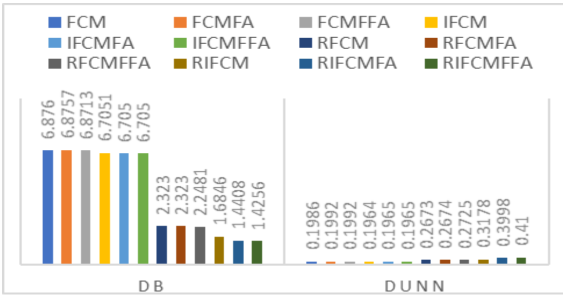


Fig.26. Comparison of performance of various algorithms using Euclidean metric

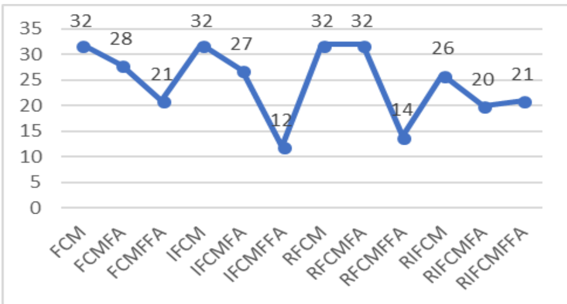


Fig.27. Comparison of number of iterations required for segmentation using to Euclidean distance

Table 4. Performance analysis indices for geographical image segmentation

Dist. Function	Algorithm	Cluster=3			Cluster=4		
		#i	DB	Dunn	#i	DB	Dunn
Euclidean Distance	FCM	14	8.8505	0.1846	32	6.8760	0.1986
	IFCM	17	8.7029	0.1898	32	6.7051	0.1964
	RFCM	15	2.7168	0.3639	32	2.3230	0.2673
	RIFCM	26	1.8060	0.5045	26	1.6846	0.3178
	FCMFA	12	8.8495	0.1847	28	6.8757	0.1992
	IFCMFA	15	8.7029	0.1898	27	6.7050	0.1965
	RFCMFA	13	2.7168	0.3640	32	2.3230	0.2674
	RIFCMFA	19	1.8068	0.5303	20	1.4408	0.3998
	FCMFFA	12	8.8495	0.1847	21	6.8713	0.1992
	IFCMFFA	9	8.7002	0.1901	12	6.7050	0.1965
	RFCMFFA	7	2.7168	0.3640	14	2.2481	0.2725
	RIFCMFFA	14	1.8098	0.5421	21	1.4256	0.4100
Gaussian Kernel	GKFCM	14	0.0004	53.9552	33	0.0002	42.5575
	GKIFCM	16	0.0305	53.5421	30	0.0172	40.6055
	GKRFCM	20	0.0112	63.2520	32	0.0077	52.2640
	GKRIFCM	36	0.0074	112.1361	44	0.0046	69.9995
	GKFCMFA	9	0.0004	53.9550	31	0.0001	42.7377
	GKIFCMFA	14	0.0305	53.5018	28	0.0172	40.6050
	GKRFCMFA	13	0.0111	63.2520	17	0.0043	53.2340
	GKRIFCMFA	17	0.0059	112.3657	32	0.0046	69.9996
	GKFCMFFA	7	0.0003	54.0436	23	0.0001	42.5637
	GKIFCMFFA	9	0.0304	53.5430	28	0.0172	40.6052
	GKRFCMFFA	5	0.0109	63.5655	18	0.0043	53.2341
	GKRIFCMFFA	12	0.0059	112.3656	10	0.0046	70.0430
Hyper-tangent Kernel	HKFCM	15	0.0004	55.2892	35	0.0001	41.2516
	HKIFCM	19	0.0297	54.6804	31	0.0168	39.4011
	HKRFCM	25	0.0145	52.4792	42	0.0095	55.5478
	HKRIFCM	17	0.0057	120.3086	39	0.0046	70.9375
	HKFCMFA	13	0.0004	55.2857	30	0.0001	41.2521
	HKIFCMFA	12	0.0298	54.1337	28	0.0168	39.3993
	HKRFCMFA	22	0.0114	53.1856	25	0.0096	55.9206
	HKRIFCMFA	15	0.0056	120.5731	34	0.0046	70.9376
	HKFCMFFA	10	0.0004	55.2960	27	0.0001	41.2446
	HKIFCMFFA	6	4.6483	54.6679	21	8.7054	39.0522
	HKRFCMFFA	20	0.0109	63.6731	21	0.0077	56.2906
	HKRIFCMFFA	12	0.0055	122.9860	32	0.0045	70.9376

b. Using Gaussian Kernel:

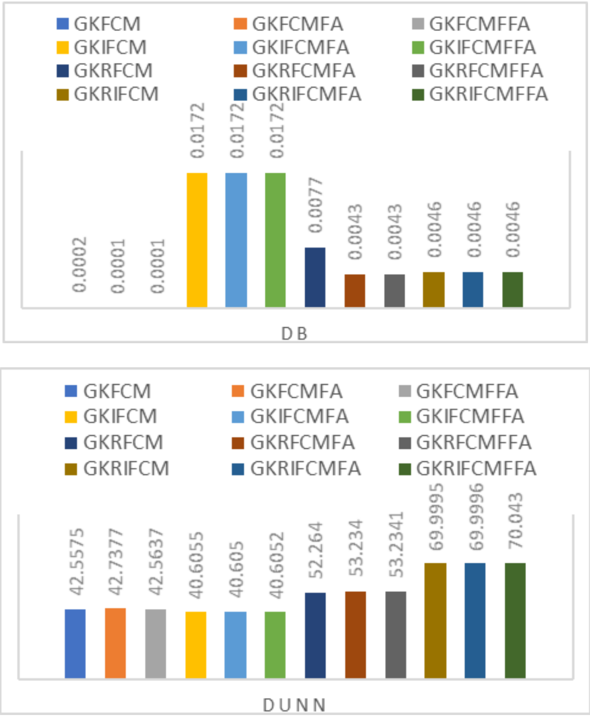


Fig.28. Comparison of performance of algorithms using Gaussian Kernel

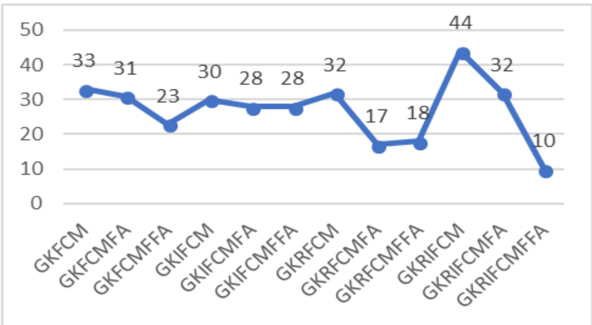


Fig.29. Comparison of number of iterations required for segmentation with respect to Gaussian kernel

The performance indices show that algorithms combined with Gaussian kernel performs significantly better than their Euclidean counterparts. The firefly and Fuzzy firefly versions of all the algorithms show better results than their naïve counterparts. While the firefly and fuzzy firefly algorithms have better convergence rate, there is a drastic improvement in the convergence rate of GKRIFCM. GKIFCM does not show any stable relation. GKRIFCMFFA shows the best convergence rate among the twelve cases. The following relation can thus be established:

GKFCM<GKFCMFA<GKFCMFFA,
GKIFCMFA<GKIFCMFFA<GKIFCM,
GKRFCM<GKRFCMFA< GKRFCMFFA,
GKRIFCM<GKRIFCMFA< GKRIFCMFFA,
GKIFCM<GKFCM<GKRFCM<GKRIFCM.

c. Using Hyper-tangent kernel:

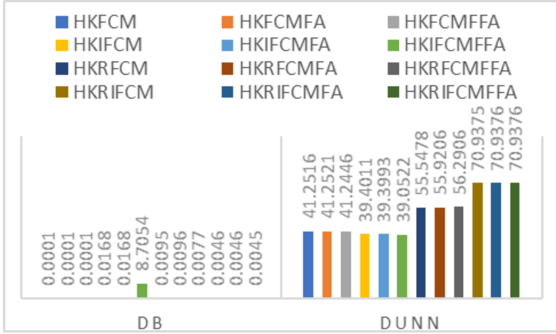


Fig.30. Comparison of performance of various algorithms using Hyper-tangent Kernel

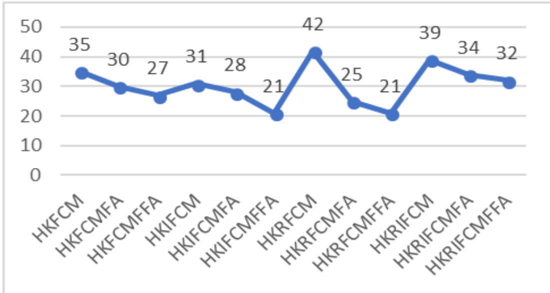


Fig.31. Comparison of number of iterations required for segmentation with respect to Hyper-tangent kernel

It can be inferred from the performance indices that HKRIFCMFA and HKRIFCMFFA shows the best results. The convergence rates are much lower in firefly and fuzzy firefly versions of all the algorithms except in case of HKIFCM which shows unstable relation. HKIFCMFFA and HKRFCMFFA shows the best convergence rates. The following relation thus holds true:

HKFCM<HKFCMFA<HKFCMFFA,
HKIFCMFFA<HKIFCM≈HKIFCMFA,
HKRFCM<HKRFCMFA<HKRFCMFFA,
HKRIFCM<HKRIFCMFA<HKRIFCMFFA

E. Segmentation Geographical Image- River-Valley

a. Using Euclidean Distance:

Fig. 30 shows the segmentation of vegetation from water body. The performance indices show that Firefly and Fuzzy Firefly versions of the algorithm shows better result than their naïve counterpart. There is remarkable improvement in the convergence rate of all the algorithms when combined with firefly and fuzzy firefly algorithms. RIFCMFAA shows the best result than all other algorithms, while RFCMFFA shows the best convergence rate. The following relation holds good:

FCM<FCMFA<FCMFFA,
IFCM<IFCMFA<IFCMFFA,
RFCM<RFCMFA<RFCMFFA,
RIFCM, RIFCMFA, RIFCMFFA,
FCM<IFCM<RFCM<RIFCM.

Table 5. Performance analysis indices for River-valley image segmentation

Dist. Function	Algorithm	Cluster=3			Cluster=4		
		#i	DB	Dunn	#i	DB	Dunn
Euclidean Distance	FCM	22	8.7063	0.1973	47	6.9550	0.2093
	IFCM	19	8.5614	0.2008	42	6.7756	0.2129
	RFCM	17	2.6516	0.6034	35	2.2247	0.4635
	RIFCM	17	1.8047	0.8730	21	1.5839	0.5154
	FCMFA	15	8.7040	0.1977	45	6.9562	0.2107
	IFCMFA	12	8.5613	0.2008	39	6.7755	0.2129
	RFCMFA	17	2.6616	0.6363	23	2.2247	0.4635
	RIFCMFA	15	1.8038	0.8387	20	1.5819	0.5693
	FCMFFA	13	8.7038	0.1977	21	6.9562	0.2108
	IFCMFFA	9	8.5601	0.2010	26	6.7750	0.2124
	RFCMFFA	11	2.6616	0.6364	6	2.2402	0.4812
	RIFCMFFA	9	1.8025	0.8760	11	1.5813	0.6481
Gaussian Kernel	GKFCM	18	0.0008	33.7921	48	0.0004	32.1088
	GKIFCM	16	0.0461	33.2485	42	0.0272	32.1366
	GKRFCM	10	0.0148	95.3710	51	0.0106	60.7647
	GKRIFCM	17	0.0078	181.4355	39	0.0058	106.6725
	GKFCMFA	12	0.0008	33.7906	37	0.0004	31.7838
	GKIFCMFA	12	0.0461	33.7248	14	0.0272	32.1262
	GKRFCMFA	9	0.0148	95.3804	33	0.0106	60.7647
	GKRIFCMFA	9	0.0076	181.5749	33	0.0056	106.7067
	GKFCMFFA	14	0.0008	33.7929	30	0.0004	31.7895
	GKIFCMFFA	12	0.0461	33.7248	10	0.0272	32.1383
	GKRFCMFFA	8	0.0147	97.7294	27	0.0106	60.7648
	GKRIFCMFFA	8	0.0074	181.5954	12	0.0054	106.7528
Hyper-tangent Kernel	HKFCM	19	0.0010	34.7748	33	0.0005	31.9072
	HKIFCM	15	0.0444	34.4471	34	0.0264	31.9268
	HKRFCM	18	0.0143	100.6311	41	0.0110	55.1118
	HKRIFCM	20	0.0072	193.8864	42	0.0054	110.6459
	HKFCMFA	14	0.0009	34.7759	25	0.0004	32.2439
	HKIFCMFA	12	0.0444	34.3826	34	0.0263	31.9264
	HKRFCMFA	10	0.0143	100.6312	33	0.0110	55.1118
	HKRIFCMFA	7	0.0070	194.1187	37	0.0054	110.9897
	HKFCMFFA	11	0.0009	34.7759	20	0.0004	32.2460
	HKIFCMFFA	5	3.3350	34.4256	31	5.5649	31.0475
	HKRFCMFFA	7	0.0143	100.6271	10	0.0108	55.1119
	HKRIFCMFFA	5	0.0070	194.1440	11	0.0054	111.6459

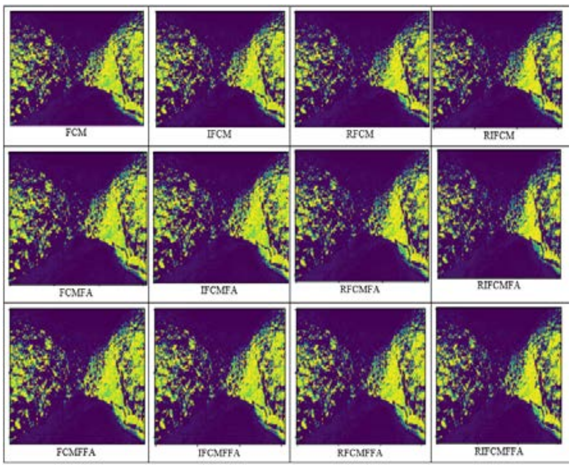


Fig.32. Output of segmentation of geographical image (river-valley) using Euclidean distance

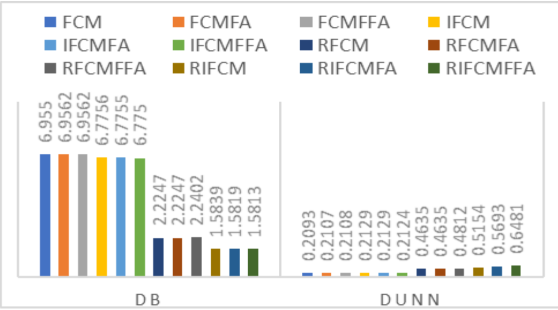


Fig.33. Comparison of performance of various algorithms using the Euclidean metric

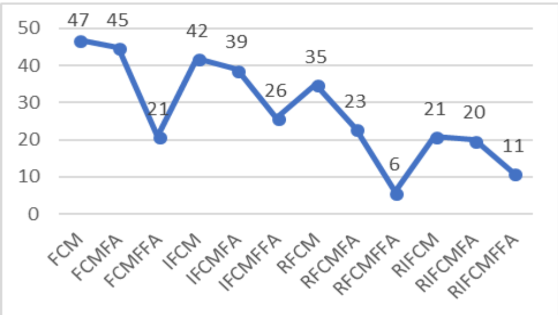
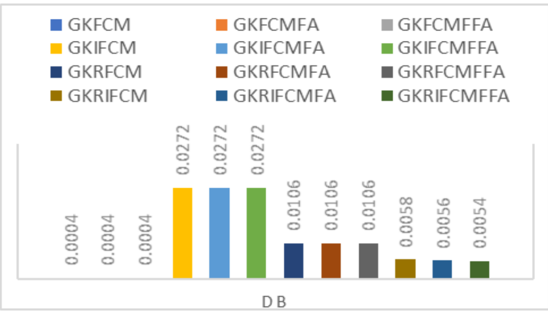


Fig.34. Comparison of number of iterations required for segmentation with respect to Euclidean distance

b. Using Gaussian Kernel:



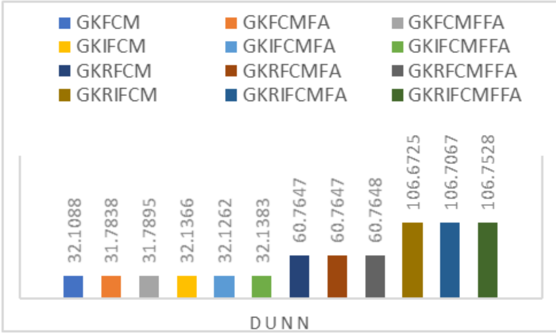


Fig.35. Comparison of performance of algorithms with respect to Gaussian Kernel

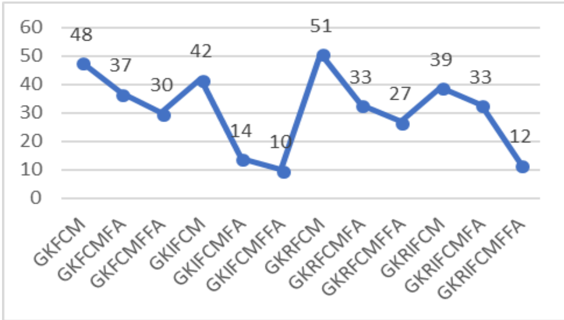


Fig.36. Comparison of number of iterations required for segmentation with respect to Gaussian Kernel

It is evident from the values of performance indices that Gaussian Kernelized versions produce much better results. The indices show that GKFCM, GKRFCM, GKRIFCM perform better than GKIFCM. The graph in fig. 36 shows the improvement of algorithms as they transition from their naïve form to firefly and then to fuzzy firefly. The indices prove that GKRIFCMFFA gives the best result in very few iterations. GKIFCMFFA shows the best convergence rate. Referring to the bar graph, the following relation between efficiency of algorithms can be established:

GKFCM<GKFCMFA<GKFCMFFA,
GKIFCMFA<GKIFCM<GKIFCMFFA,
GKRFCM<GKRFCMFA<GKRFCMFFA,
GKRIFCM<GKRIFCMFA<GKRIFCMFFA,
GKIFCM<GKFCM<GKRFCM<GKRIFCM.

c. Using Hyper-tangent Kernel:

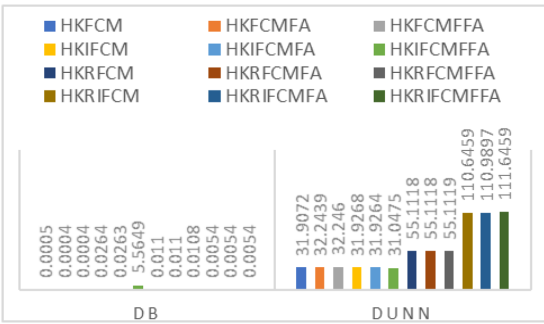


Fig.37. Comparison of performance of various algorithms using Hyper-tangent Kernel

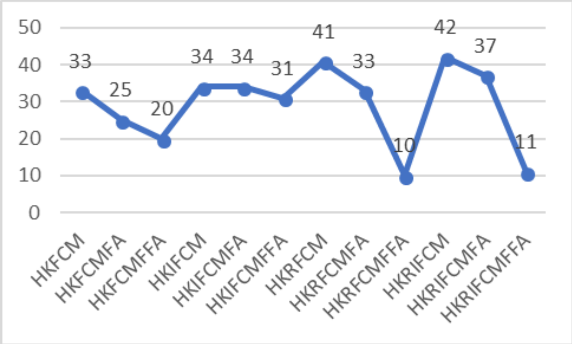


Fig.38. Comparison of number of iterations required for segmentation using the Hyper-tangent Kernel

Hyper-tangent kernelized versions produces better results than Gaussian versions. The graph in fig. 36 shows remarkable improvements specially in case of HKRFCM and HKRIFCM during their transition from original version to firefly and fuzzy firefly versions. The performance indices show that HKRIFCMFFA forms the best clusters. From fig 38, it can be observed that HKRFCMFFA shows the best convergence rate closely followed by HKRIFCMFFA. Referring the bar graphs, the following order of efficiency of algorithms can be established:

HKFCM<HKFCMFA<HKFCMFFA,
HKIFCMFFA<HKIFCM<HKIFCMFA,
HKRFCM<HKRFCMFA<HKRFCMFFA,
HKRFCM<HKRFCMFA<HKRFCMFFA,
HKIFCM<HKFCM<HKRFCM<HKRIFCM

VII. Summary

Through the above observations, it is quite evident that Firefly and Fuzzy Firefly show improvements both in performance indices as well as the convergence rate, which has been verified through DB and Dunn indices. It can be said that RIFCM gives the best results while IFCM and FCM shows comparable results with IFCM giving slightly better results than FCM and RFCM gives better results than FCM and IFCM. However, in kernelized versions, FCM, RFCM and RIFCM outperform IFCM. Moreover, it is established that Hyper-tangent Kernel gives slightly better result than Gaussian Kernel. The kernelized versions gave significantly better results than their Euclidean counterpart. So, RIFCM combined with Hyper-tangent kernel and fuzzy firefly algorithm gives the best result amongst all other cases. Also, the performance of kernelized IFCM combined with firefly and fuzzy firefly does not show any stable relation. Thus, we could establish that firefly and fuzzy firefly prove to be efficient optimization algorithms and improve the performances of clustering algorithms on combination.

VIII. Conclusion

In this paper, we have successfully fused firefly and fuzzy firefly algorithms with the existing clustering algorithms and analyzed the performance of these algorithms with the existing algorithms. Fusing clustering algorithms with firefly and fuzzy firefly algorithm renders stability to the clustering output and improves the convergence rate of the algorithms. Fuzzy firefly manages to outperform the firefly algorithm by producing better outputs in fewer iterations. These meta heuristic algorithms find the cluster centers that are closer to the actual cluster centers, thus giving better results than those produced by assigning random centers. Replacing the Euclidean distance formula with the Gaussian and Hyper-tangent kernels enables the algorithms to cluster non-linearly separable data. We have successfully established relations among these algorithms.

References

[1] Zadeh, L.A., “Fuzzy Sets”, *Information and Control*, Vol. 8 No.3, 1965. pp.338 – 353.

[2] Bezdek, J. C., Ehrlich, R., and Full, W., “FCM: The fuzzy c-means clustering algorithm”, *Computers & Geosciences*, Vol. 10 No. 2-3, 1984, pp.191-203.

[3] Atanassov, K.T., “Intuitionistic Fuzzy Sets”, *Fuzzy sets and Systems*, Vol. 20 No.1, 1986, pp.87-96.

[4] Pawlak, Z., “Rough sets”, *International Journal of Parallel Programming*, Vol.11 No.5, 1982, pp.341-356

[5] Chaira, T., “A novel intuitionistic fuzzy C means clustering algorithm and its application to medical images”, *Applied Soft Computing*, Vol 11 No.2, 2011, pp.1711-1717.

[6] Maji, P. and Pal, S.K, “RFCM: A Hybrid Clustering Algorithm using rough and fuzzy set”, *Fundamenta Informaticae*, Vol. 8 No. 4, 2007, pp.475-496.

[7] Mitra, S., Banka, H. and Pedrycz, W., “Rough-Fuzzy Collaborative Clustering”, *IEEE Transactions on System, Man, and Cybernetics, Part B: Cybernetics*, Vol. 36 No. 4, 2006, pp.795-805.

[8] Bhargava, R. Tripathy, B. K., Tripathy, A., Dhull, R., Verma, E. and Swarnalatha, P., “Rough intuitionistic fuzzy C-means algorithm and a comparative analysis”, *Proceedings of ACM Compute-2013, International Conference, SITE, VIT University, 21-22 August.*

[9] Zhang, D. and Chen, S., “Fuzzy clustering using kernel method”, in *International conference on Control and Automation, Xiamen, China*, 2002, pp.123-127.

[10] Tripathy, B. K., Ghosh, A., and Panda, G. K., “Kernel Based K-Means Clustering Using Rough Set”, in Proceedings of 2012 International Conference on Computer Communication and Informatics (ICCCI -2012), Jan. 10 – 12, Coimbatore, INDIA, pp.1 -5.

[11] Tripathy, B.K., and Bhargav, R., “Kernel Based Rough-Fuzzy C-Means”, in International Conference on Pattern Recognition and Machine Intelligence (PReMI), ISI Calcutta, December 2013, LNCS 8251, pp.148-157.

[12] Tripathy, B.K., Tripathy, A., Govindarajulu, K. and Bhargav, R., “On Kernel Based Rough Intuitionistic Fuzzy C-means Algorithm and a Comparative Analysis” *Smart Innovation, Systems and Technologies, 2014.*

[13] Tripathy, B.K., and Mittal, D., "Efficiency Analysis of Kernel Functions in Uncertainty Based C-Means Algorithms", *International Conference on Advances in Computing, Communications and Informatics, ICACCI 2015, Article number 7275709*, pp. 807-813.
[14] Tyshchenko, Oleksii, Bodyanskiy, Yevgeniy, Hu, Zhengbing and Samitova, Viktoriia. "Fuzzy Clustering Data Given in the Ordinal Scale", in *International Journal of Intelligent Systems and Applications*, 2017, Vol 9. No. 1, pp. 67-74.
[15] Yang, Xin-She (2009), "Firefly algorithms for multimodal optimization", *O. Watanabe and T. Zeugmann (Eds.): SAGA 2009*, Vol. 5792, pp. 169–178.
[16] Hassanzadeh, T. and Kanan, H. R., "Fuzzy FA: A modified firefly algorithm", *Applied Artificial Intelligence.* Vol 28 No. 1, 2014, pp. 47-65.
[17] Jain, A., Chinta, S. and Tripathy, B.K., "Stabilizing Rough Sets Based Clustering Algorithms Using Firefly Algorithm over Image Datasets", in *2nd International Conference on Information and Communication Technology for Intelligent Systems (ICTIS 2017)*, 2017, pp.325-332.
[18] Chinta, S., Jain, A. and Tripathy, B. K., "Image Segmentation Using Hybridized Firefly Algorithm and Intuitionistic Fuzzy C-Means", in *1st International Conference On Smart Systems, Innovations and Computing, Manipal University, Jaipur*, 2018.
[19] Davis, D.L. and Bouldin, D.W., "A cluster separation measure", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI Vol 1 No.2, 1979, pp.224 – 227.
[20] Dunn, J. C., "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters", *Journal of Cybernetics*, Vol 3 No. 3, 1974, pp.32-57.
[21] Jain, A. K., Murty, M. N. and Flynn, P. J., "Data clustering: a review", *ACM Computing Surveys*, Vol. 31 No. 3, 1999, pp.264-323.

Authors' Profiles

B.K. Tripathy is working in School of Computer Science and Engineering, VIT, Vellore, India. He has received several fellowships, funded projects, published over 500 technical papers and has supervised around 50 research degrees. He has published two text books and has edited five research volumes for IGI publications. He is a member of IEEE, ACM, IRSS, CSI and IMS and is associated with more than 75 journals. His research interest includes fuzzy sets and systems, rough sets and knowledge engineering, data clustering, social network analysis, soft computing, granular computing, neighborhood systems, soft set theory and applications, multiset theory, list theory and multi-criteria decision making.

Anmol Agrawal was born on July 1, 1998. He is currently a third year Computer Science and Engineering student at Vellore Institute of Technology, Vellore, India. He will receive his B.Tech degree in 2020. He is a part of a research group under the supervision of Dr. B. K. Tripathy. His current research interests include data mining, machine learning, artificial intelligence and soft computing.

A. Jayaram Reddy is working in the SITE school of VIT, Vellore. He is pursuing his Ph. D in computer science and engineering. His research focus is on soft computing, rough sets and clustering, data mining and machine learning.

How to cite this paper: B.K. Tripathy, Anmol Agrawal, A. Jayaram Reddy, "A Comparative Analysis of Firefly and Fuzzy-Firefly based Kernelized Hybrid C-Means Algorithms", International Journal of Intelligent Systems and Applications(IJISA), Vol.11, No.6, pp.49-68, 2019. DOI: 10.5815/ijisa.2019.06.05