


# A Novel Fuzzy Logic-Based Improved Cuckoo Search Algorithm

Krishna Gopal Dhal, Midnapore College (Autonomous), India\*

 <https://orcid.org/0000-0002-6748-0569>

Arunita Das, Midnapore College (Autonomous), India

Jorge Gálvez, Universidad de Guadalajara, Mexico

## ABSTRACT

Cuckoo search (CS) algorithm is a nature-inspired optimization algorithm (NIOA) with less control parameters that is stable, versatile, and easy to implement. CS has good global search capabilities, but it is prone to local optima problems. As a result, it may be possible to improve the classic CS algorithm's optimization capability. Centered on fuzzy set theory, this paper introduces an improved CS version. The population of solutions has been divided into two fuzzy sets, and each solution is assigned to one of the sets based on its fitness. The fuzzy collection centroids, global best solution advice, and Lévy distribution dependent mutation are all used to boost the population's solutions. With well-accepted objective functions such as Otsu inter class variance and Kapur's entropy, the experimental analysis has been conducted on the CEC-2014 test suite and image multi-level thresholding domain. The proposed fuzzy cuckoo search (FCS) algorithm is compared to the classical CS, PSO, FA, SMA, and BA algorithm and provides satisfactory results.

## KEYWORDS

Cuckoo Search, Fuzzy Set, Image Segmentation, Multi-Level Thresholding, Optimization, Swarm Intelligence

## 1. INTRODUCTION

Natural-Inspired Optimization Algorithms (NIOAs) (Dhal, Das, Ray et al, 2020; Dhal, Das, Ray et al, 2021; Dhal, Ray, Das, & Das, 2019) have been proposed as an alternative to conventional mathematical approaches for solving complex optimization problems. These techniques are used in Artificial Intelligence (AI) mechanisms, in which a population works together to solve a problem. This inspiring nature has piqued the interest of many researchers who abstract natural phenomena in computational terms to solve complex engineering problems by adapting human collaboration to create NIOA operators and alternative mechanisms for solving them. A plethora of NIOAs have recently been created, based not only on natural laws but also on physical, social, and biological principles (Dhal, Das, Ray et al, 2020; Dhal, Ray, Das, & Das, 2019). The No Free Lunch Theorem (NFL) states that no single optimization algorithm can solve any optimization task (Dhal, Das, Ray et al, 2020; Dhal, Ray, Das, & Das, 2019). This justifies the creation of a large number of NIOAs. As a result, the output of NIOAs is highly dependent on the problem to be solved as well as the structure of the algorithm in question. In this context, the development of novel NIOAs is an accessible and exciting research area, as many issues such as the balance between exploration and exploitation

DOI: 10.4018/IJAMC.292516

\*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

stages, self-adaptivity, parameter-less results, and convergence continue to perplex the NIOAs community. Cuckoo Search Algorithm (CSA) (Yang & Deb, 2009) is a simple and efficient NIOA that has shown to be successful on a variety of optimization tasks. However, the efficacy of CSA is strongly influenced by the capacity for discovery and exploitation, and it may be possible to improve its performance by solving complex optimization problems. There is no interaction or knowledge sharing between solutions in traditional CSA, and only Lévy flight with a fixed step size has been used for exploration and exploitation (Dhal, Das, Ray et al, 2019; Dhal & Das, 2017; Dhal et al., 2017). As a result, researchers are able to effectively develop the CSA by using various methods to address the aforementioned issues. Parameter adaptation (Mareli & Twala, 2018), integration of different random number generators (Yang & Deb, 2009), communication and sharing of information (Dhal, Das, Ray et al, 2019; Dhal et al., 2017; Yang & Deb, 2009), global and local search strategies (Dhal, Das, Ray et al, 2019; Dhal & Das, 2017), hybridization with other NIOA (Chi et al., 2019; Zhang et al., 2019), and initial population development, adaptive population size (Dhal, Das, Sahoo et al, 2021; Dhal et al., 2017; Mlakar et al., 2016), etc are some important strategies to boost the CSA. Parameter adaptation methods, including population size, have a significant impact on the efficiency of the CS algorithm (Mlakar et al., 2016). Particle Swarm Optimizer (PSO) (Abdelbar et al., 2005; Gaxiola et al., 2019; Nobile et al., 2018), Bat Algorithm (BA) (Pérez et al., 2015a; Pérez et al., 2015b), Firefly Algorithm (FA) (Hassanzadeh & Kanan, 2014), and CS (Guerrero et al., 2015; Yang & Deb, 2009) also have used fuzzy logic to derive the rules for parameter adaptation. In this research, fuzzy logic has been applied in a unique way. The population has been divided into two fuzzy sets, each of which has some belongingness, which is determined by the solution's fitness. The solutions are improved by using fuzzy set centroids, global best solution guidance, and Lévy distribution dependent mutation. The validation of the proposed fuzzy CS has been performed over CEC-2014 test suite (Liang et al., 2014) and multi-level thresholding (Dhal, Das, Ray et al, 2020) based image segmentation field with the assistance of well-known objective functions.

According to the literature, CS has been commonly used in the thresholding-based image segmentation domain, but it also has some shortcomings. For example, Brajevic and Tuba (2014) used CS and Firefly Algorithm (FA) in multi-thresholding image segmentation with the Kapur's entropy and the Otsu criterion as objective function. By considering all objective functions, CS and FA outperformed PSO and DE. CS and FA, on the other hand, produced competitive results. Considering Kapur's entropy and the Otsu criterion in the multi-thresholding region, Alihodzic and Tuba (2014) improved the efficiency of the Bat Algorithm (BA) by integrating elements of the DE and ABC algorithms into it. Experiments showed that the improved BA outperformed many existing NIOAs, including classical BA, CS, PSO, DE, FA, etc. Tuba et. al. (2017) used the Elephant Herding Optimization (EHO) algorithm in conjunction with Kapur's and Otsu to threshold multi-level images. In terms of mean and standard deviation of objective function values, the EHO algorithm outperformed four other swarm intelligence algorithms namely PSO, DE, CS, and FA. Suresh and Lal (2017) designed a Chaotic DPSO (CDPSO) for multi-thresholding that was based on a chaotic sequence and used Cross and Tsallis entropies as objective functions. In terms of robustness, computational time, stability, and fitness value, the results of the comparative study clearly demonstrated that CDPSO was very successful in this satellite image segmentation domain, providing better results than CS, HS, DE, and classical PSO. Experiments, on the other hand, showed that Cross entropy is more powerful than Tsallis' entropy. According to (2015), CS had a problem with longer computation times. As a result, the researchers improved CS in the multi-thresholding domain. Suresh and Lal (2016), for example, developed a CS-based model for satellite image multi-thresholding with the help of three useful objective functions: Otsu, Kapur, and Tsallis. In this analysis, Mantegna and McCulloch algorithm based Lévy flight generation strategies were used to implement two variants of the CS algorithm. McCulloch was seen to generate a Lévy distribution that is more stable and computationally efficient. As a consequence, the CS algorithm using the McCulloch strategy outperforms the CS algorithm using the Mantegna method. Classical CS, ABC, and Darwinian PSO (DPSO) algorithms were all

used to optimize the three aforementioned objective functions. Experimental results showed that CS with McCulloch provided better results than its competitors in terms of computational time, values of objective functions and convergence rate. To segment histopathological images, Dhal et al. (2018) developed a Parameterless Cuckoo Search (PLCS) algorithm with a Fuzzy Entropy Model. By achieving better threshold values with less computational time, the PLCS significantly outperformed the traditional CS and PSO algorithms. Manic et. al. (2021) employed chaotic Cuckoo Search (CCS) with Shannon, Kapur, and Otsu criteria for the segmentation of the brain MRI images and produced satisfactory results. Jiao et. al. (2021) proposed an improved CS based entropic thresholding approach for image segmentation under different conditions. The proposed technique gave superior results to PSO and ant colony optimizer (ACO). Duan et. al. (2021) also proposed an improved CS (ICS) by incorporation parameter adaptation strategy and dynamic weighted random-walk method for better exploration and exploitation respectively. Numerical results clearly demonstrated that proposed ICS with Otsu objective function provided promising outcomes. Kalyani et. al. (2021) employed CS with the assistance of Kapur, Otsu and minimum cross entropies for the multi-level image segmentation. Experiments showed that CS with Otsu achieved the best results. Minjares et. al. (2021) developed an approach based on the Cuckoo Search Algorithm (CSA) and the Generalized Gaussian (GG) distribution to find the optimal threshold. Experiments had been performed to segments Skin lesions and results were satisfactory. Rahaman and Singh (2021) devised an adaptive CS (ACS) algorithm and applied for multi-level image segmentation with the assistance of Otsu and Tsallis entropy. Experimental results showed that ACS with Otsu outperformed the other tested segmentation models. Tan et. al. (2021) developed and improved CS (ICS) by incorporating adaptive control parameter mechanism and hybrid search strategy for the enhancement of global and local search ability respectively. The authors also developed modified fuzzy entropy for the proper image segmentation. Experimental results showed that the proposed ICS with modified fuzzy entropy gave competitive results compare to other tested segmentation strategies. As a result, this paper has used proposed fuzzy CS (FCS) with two common objective functions, Otsu and Kapur's entropy, over an image multi-level thresholding domain. In terms of optimization ability, accuracy, computational time, and image segmentation quality parameters, experimental results show that the FCS outperforms the classical CS, FA, BA, Slime mould algorithm (SMA) Li et al., (2020), and PSO. The main contribution of the paper is the construction of a new improved CS algorithm with the assistance of fuzzy set based population and centroid based searching strategies.

The remainder of the paper is structured as follows: section 2 discusses fuzzy CS, section 3 performs a brief mathematical implementation of objective functions, section 4 discusses performance assessment and experimental findings, and section 5 concludes the paper.

## 2. PROPOSED METHODOLOGY

In this section, the methodology of the proposed fuzzy cuckoo search (FCS) is discussed.

### 2.1. Classical Cuckoo Search Algorithm

Cuckoo search (CS) is a powerful optimization algorithm proposed by X. S Yang and Suash Deb in 2009, inspired by some cuckoo species' obligate brood parasitism by laying their eggs in other host birds' nests (Yang & Deb, 2009). The aforementioned parasitic behaviour in cuckoo birds refers to some species of cuckoo birds' aggressive and highly active reproduction strategy, which is based on an evolutionary predisposition to lay eggs in the nests of the host birds. This action helped to ensure the survival of their species. This natural observation is implemented in the computational field by treating the host bird eggs as initial solutions and the cuckoo eggs as alternative solutions, with the aim of arriving at a near-optimal solution via efficient iterations that include the substitution of weaker solutions with better ones.

As a result, there are three perfect guidelines or principles for the cuckoo search algorithm's glowing action (Dhal, Das, Ray et al, 2019; Dhal et al., 2017; Yang & Deb, 2009):

- a) Each cuckoo lays one egg at a time, then selects a random nest to lay its egg in.
- b) The best nests with high-class eggs would be passed on to the next generation.
- c) The number of accessible host nests is limited. The host bird has a chance  $P_a \in [0,1]$  of discovering the eggs.

The following is an explanation of how the cuckoo search works. The location of the cuckoo egg within the search space is represented by a solution in the original cuckoo search algorithm corresponding to cuckoo nests. The initial population of solutions of size  $n$  is generated mathematically as follows:

$$X_i = low + (up - low) \times \partial \quad (1)$$

$X_i$  is the  $i^{th}$  individual.  $up$  and  $low$  are the upper and lower bound of the search space of the objective function.  $\partial$  is the random variable belonging to  $[0,1]$  and usually generated from uniform distribution. Generation of new solution signifies the exploitation of the current solutions is carried out by using the Lévy flight distribution expressed as:

$$X_i^{t+1} = X_i^t + \alpha \cdot Lévy() \quad (2)$$

$\alpha > 0$  represents a scaling factor of the step size drawn from Lévy distribution.

$Lévy()$  i.e. Lévy Flight is a non-Gaussian random process in which the stationary increments follow a Levy stable distribution. The features of this distribution are as follows (Dhal et al., 2017; Yang & Deb, 2009):

- a) Lévy Flight's mean and variance are both infinite.
- b) Since it is a heavy-tailed random walk, the tail of the Lévy distribution falls off much more gently than a Gaussian distribution. Furthermore, the distribution of heavy tailed data is not exponentially bounded.
- c) The Lévy distribution's variation has divergence characteristics. As a consequence, extremely long jumps could be possible. Therefore, it is capable of exploring a large amount of search space.
- d) Since the probability of repeating the same area is lower in the Lévy distribution than in the Gaussian distribution, it allows for faster algorithm convergence. By integrating the ability to escape from local minima, the algorithm's efficiency is improved.

Different algorithms, such as the Rejection algorithm, McCulloch's algorithm, Mantegna's algorithm, and others (Yang & Deb, 2009), can be used to produce Lévy Flight. Mantegna's algorithm was used in this research. As mentioned below, it generates random numbers using a symmetric Lévy stable distribution.

$$\sigma = \left[ \frac{\Gamma(1 + \beta) \sin \frac{\pi\beta}{2}}{\Gamma\left(\frac{1 + \beta}{2}\right) \beta \cdot 2^{\frac{(\beta-1)}{2}}} \right]^{\frac{1}{\beta}} \quad (3)$$

Where, “ $\Gamma$ ” is the gamma function,  $0 < \beta \leq 2$ .  $A$  is the standard deviation. As per Mantegna’s algorithm, the step length  $v$  can be calculated as,

$$v = \frac{x}{|y|^{1/\beta}} \quad (4)$$

Here,  $\hat{x}$  and  $\hat{y}$  are taken from normal distribution and  $\sigma_{\hat{x}} = \sigma, \sigma_{\hat{y}} = 1$ . Where  $\sigma$  is the standard deviation. The resulting distribution has the same behavior of Lévy distribution for large values of the random variables. Mantegna’s algorithm is preferred in this study for generating levy distribution because of its simple calculation steps than McCulloch’s algorithm and its faster computational speed in the range  $0.75 \leq \beta \leq 1.95$ .

Depending on the discussion the algorithm of the traditional CS is as follows:

**Algorithm 1: Classical Cuckoo Search Algorithm**

<b>1)</b>	<b>Objective function has been taken</b>
<b>2)</b>	Generate initial population of size $n$ as per Eq.(1)
<b>3)</b>	<b>While</b> (Stop Criterion) <b>do</b>
<b>4)</b>	Get a Cuckoo randomly using Levy Flight as per Eq.(2)
<b>5)</b>	Evaluate the fitness $(F_i)$
<b>6)</b>	Choose another solution randomly (say $j^{\text{th}}$ solution)
<b>7)</b>	<b>if</b> $(F_i > F_j)$
<b>8)</b>	Replace $j^{\text{th}}$ solution by $i^{\text{th}}$ solution
<b>9)</b>	<b>end if</b>
<b>10)</b>	A fraction $(P_a)$ of worst nest are abandoned and built using Eq.(1)
<b>11)</b>	<b>end While</b>
<b>12)</b>	<b>Output:</b> the most fit or global best solution $G_{\text{best}}$ .

In terms of the number of parameters, the classic CS algorithm is a very simple algorithm. Traditional CS algorithm, however, has several flaws. They are listed below. (Dhal, Das, Ray et al, 2019; Dhal et al., 2017; Yang & Deb, 2009):

- a) There is no communication or knowledge exchange between the solutions.
- b) In conventional CS, there is no guidance from the global best (*Gbest*).
- c) For both global and local quest, only Lévy flight with a fixed phase size was used. It doesn't behave in an ergodic way. In any metaheuristic algorithm, the appropriate step size for diversification and intensification steps is critical.

To address the CS algorithm's shortcomings, this paper proposes a fuzzy CS (FCS) variant based on fuzzy set based population, levy distribution based mutation, and guidance of global best and centroid of the fuzzy population set. In the following section, the proposed methodologies are discussed.

## 2.2. Fuzzy Cuckoo Search Algorithm

The step by step implementation of the proposed FCS is given as follows:

### 2.2.1. Fuzzy Population

In the proposed FCS, population is considered as fuzzy set (Zimmermann, 2011). Suppose, the initial population of size  $n$  i.e.  $\{X_i \mid i = 1, 2, 3, \dots, n\}$  has been generated randomly using Eq.(1). We can easily divide the population into the best set ( $B$ ) and worst set ( $W$ ) depending on the fitness values. For a maximization problem, solutions with higher fitness values belong to the best set ( $B$ ) and the rest others belongs to worst set ( $W$ ). In each set of  $B$  or  $W$ , an solution either belongs or does not belong if we take  $B$  and  $W$  as crisp sets. It means the membership or belongingness of a solution can be full i.e. 1 or null i.e. 0. However, this study considers the  $B$  and  $W$  as fuzzy sets where the memberships of the solutions are within  $[0, 1]$ . The fuzzy set  $B$  is represented as follows:

$$B = \left\{ \frac{\mu_1^B}{X_1}, \frac{\mu_2^B}{X_2}, \dots, \frac{\mu_i^B}{X_i}, \dots, \frac{\mu_n^B}{X_n} \right\} \quad (5)$$

Where,  $\mu_i^B$  indicates the membership of the solution  $X_i$  into the best ( $B$ ) fuzzy set. The following mathematical rules have been used to find  $\mu_i^B$  depending on the fitness  $F_i$  of the solution  $X_i$ .

- 1) If it is a maximization problem and for all  $X_i$  within its range fitness of  $X_i$  i.e.  $F(X_i) = +ve$

$$\text{then } \mu_i^B = \left| \frac{F_i}{\max_{\forall i} F_i} \right|.$$

- 2) If it is a maximization problem and for all  $X_i$  within its range fitness of  $X_i$  i.e.  $F(X_i) = -ve$

$$\text{then } \mu_i^B = \frac{1}{\left| \frac{F_i}{\max_{\forall i} F_i} \right|}.$$

3) If it is a minimization problem and for all  $X_i$  within its range fitness of  $X_i$  i.e.  $F(X_i) = +ve$

$$\text{then } \mu_i^B = \frac{1}{\left| \frac{F_i}{\min_{\forall i} F_i} \right|}.$$

4) If it is a minimization problem and for all  $X_i$  within its range fitness of  $X_i$  i.e.  $F(X_i) = -ve$

$$\text{then } \mu_i^B = \left| \frac{F_i}{\min_{\forall i} F_i} \right|.$$

The most fit solution or global best (Gbest) solution has the membership of 1 in the set B i.e.  $\mu^B(\text{Gbest}) = 1$  and for other solutions the membership values are greater than zero i.e.  $0 \leq \mu^B(X_i) < 1$  for  $X_i \neq \text{Gbest}$ .

In the same way, the fuzzy set W for worst solutions is represented by the following manner:

$$W = \left\{ \frac{\mu_1^W}{X_1}, \frac{\mu_2^W}{X_2}, \dots, \frac{\mu_i^W}{X_i}, \dots, \frac{\mu_n^W}{X_n} \right\} \quad (6)$$

where,  $\mu_i^W = 1 - \mu_i^B$  indicates the membership of the solution  $X_i$  into the worst (W) fuzzy set and hence,  $\mu_i^B + \mu_i^W = 1$ . For all solution, the membership values are greater than zero i.e.  $0 \leq \mu^W(X_i) \leq 1$ .

### 2.2.2. Centroids Calculation and Solution Modification

Next, find the centroids of the fuzzy sets B i.e.  $(C_B)$  and W i.e.  $(C_W)$  which will guide the solutions to improve themselves. According to the literature, the centroid-based solution generation technique is superior to selecting two members and determining their centre of gravity because it contains all the information of the population set's solutions (Erol & Eksin, 2006). As a result, the improvement focused on the centroid functions as a contraction operator. Differential Evolution (DE) is one of many efficient algorithms that select two random solutions to change other solutions. However, the centroid of the solutions considers this form of random selection, which is crucial to performance. As a consequence, the centroid increases the algorithm's convergence and accuracy. The fuzzy set's centroids are determined using the following formula (Lei et al., 2018).

$$C_B = \frac{\sum_{X_i} \mu^B(X_i) \cdot X_i}{\sum_{X_i} \mu^B(X_i)} \quad (7)$$

and

$$C_W = \frac{\sum_{X_i} \mu^W(X_i) \cdot X_i}{\sum_{X_i} \mu^W(X_i)} \quad (8)$$

The solutions are modified by the following strategies as discussed below:

### 2.2.3. Solution Modification

If the membership of the corresponding solution is greater in the set B then the solution is modified depending on centroid  $C_B$ , global best solution  $Gbest$ . Otherwise if the belongingness of the solution is greater in W set then solution is modified depending on centroid  $C_W$ , global best solution  $Gbest$ . Mathematically it is expressed as:

$$\text{if} \left( \mu^B(X_i) > \mu^W(X_i) \right)$$

$$X_i = C_B + (C_B - X_i) \cdot \delta 1 + (Gbest - X_i) \cdot \varepsilon 1 \quad (9)$$

else

$$X_i = C_W + (C_W - X_i) \cdot \delta 2 + (Gbest - X_i) \cdot \varepsilon 2 \quad (10)$$

where,  $\delta 1, \delta 2 \in [1, -1]$  and  $\varepsilon 1, \varepsilon 2 \in [0, 1]$  are random numbers generated from uniform distributions. The fitness of the newly modified solution has been measured and random replacement policy is utilized to upgrade the population by following the same procedure of classical cuckoo search.

Here, the exploitation or local search has been done using the Eq.(9) where centroid of the best set has been modified using the factors  $(C_B - X_i)$  and  $(Gbest - X_i)$ . These factors help to move the concerned solution  $X_i$  towards the centroid of the best set  $C_B$  and the global best solution  $Gbest$ . Guidance of the  $Gbest$  always play an important role in the solution improvement. The fractions of these two factors added to the centroid  $C_B$  which carries the information of all the solutions of the set. It is reported that solutions generated around the centroid of the population set greatly emphasizes the local search or exploitation ability of the algorithm (Erol & Eksin, 2006).

The exploration has been performed over worst solutions or the solutions with higher membership in the worst set through the Eq.(10). The centroid of the worst set  $C_W$  is modified using the factors  $(C_W - X_i)$  and  $(Gbest - X_i)$ . The difference between the worst solution  $X_i$  and  $Gbest$  mainly helps to performs the exploration. It is reported that centroid based guidance helps to move the worst solutions in a population towards the better regions of the search space. Hence, this study utilizes  $(C_W - X_i)$  factor to modify the worst solutions (Mejía-de-Dios & Mezura-Montes, 2019).

When the algorithm going to converge, the solutions of the population becomes nearly same or near to the  $Gbest$  solution. Therefore their membership in the best set is higher and according to the formed rule local search or exploitation will be done over the solutions using Eq.(9). So, FCS follows the important property of the NIOA i.e. performs local search during the convergence time.

### 2.2.4. Worst Solutions Replacement

In the proposed FCS, a fraction of worst solutions  $(P_a)$  are abandoned and regenerated using Lévy flight based mutation as per Eq.(2). This step helps to maintain the balance between exploration and exploitation. It also enhances the diversity of the population which significantly resists the premature convergence.

Depending on the above discussion the algorithm of the proposed FCS is as Algorithm 2.



Algorithm 2

1)	<b>Objective function has been taken.</b>		
2)	Initialize the population of cuckoos, $X = \{X_i \mid i = 1, 2, 3, \dots, n\}$ using Eq.(1) where, $n$ is the number of cuckoos and $X_i$ is the $i^{\text{th}}$ cuckoo.		
3)	Consider the population set ( $P$ ) as two fuzzy sets best ( $B$ ) and worst ( $W$ ) which are represented as:		
	$B = \left\{ \frac{\mu_1^B}{X_1}, \frac{\mu_2^B}{X_2}, \dots, \frac{\mu_i^B}{X_i}, \dots, \frac{\mu_n^B}{X_n} \right\}$ <p>Where, <math>\mu_i^B</math> indicates the membership of the solution <math>X_i</math> into the best (<math>B</math>) fuzzy set. <math>\mu_i^B</math> has been computed depending on the rules mentioned in the previous section.</p>		
	$W = \left\{ \frac{\mu_1^W}{X_1}, \frac{\mu_2^W}{X_2}, \dots, \frac{\mu_i^W}{X_i}, \dots, \frac{\mu_n^W}{X_n} \right\}$ <p>Where, <math>\mu_i^W = 1 - \mu_i^B</math> indicates the membership of the solution <math>X_i</math> into the worst (<math>W</math>) fuzzy set and hence, <math>\mu_i^B + \mu_i^W = 1</math>.</p>		
4)	Now find the centroids of the fuzzy sets $B(C_B)$ and $W(C_W)$ by the following expressions.		
	$C_B = \frac{\sum_{X_i} \mu^B(X_i) \cdot X_i}{\sum_{X_i} \mu^B(X_i)}$	and	$C_W = \frac{\sum_{X_i} \mu^W(X_i) \cdot X_i}{\sum_{X_i} \mu^W(X_i)}$
5)	<p>Take a random cuckoo or solution <math>X_i</math> from the population</p> <p style="text-align: center;">if (<math>\mu^B(X_i) &gt; \mu^W(X_i)</math>)</p> <p style="text-align: center;"><math>X_i = C_B + (C_B - X_i) \cdot \delta 1 + (Gbest - X_i) \cdot \varepsilon 1</math></p> <p style="text-align: center;">else</p> <p style="text-align: center;"><math>X_i = C_W + (C_W - X_i) \cdot \delta 2 + (Gbest - X_i) \cdot \varepsilon 2</math></p> <p>Where, <math>\delta 1, \delta 2 \in [1, -1]</math> and <math>\varepsilon 1, \varepsilon 2 \in [0, 1]</math> are random numbers generated from uniform distributions.</p>		
6)	Evaluate its quality or fitness value ( $F_i$ ) of $X_i$		
7)	Choose a nest with another solution among $n$ randomly and say this solution is $j$ .		
8)	if ( $F_i > F_j$ ) then replace $X_j$ by $X_i$ else do nothing.		

Algorithm 2 continued on next page

Algorithm 2 continued

1)	Objective function has been taken.
9)	A fraction ( $P_a$ ) of worst solutions are abandoned and new solutions ( $X_N$ ) are generated using Lévy flight around the abandoned solutions ( $X_K$ ) by using the Eq. (2).
10)	Rank the solutions and find <i>Gbest</i> solution.
11)	Steps 3-11 are repeated until the termination condition.

### 2.3. Parameter Settings

Parameter setting is critical for any Nature-Inspired Optimization Algorithm (NIOA), and it is usually done by trial and error. The NIOAs' parameter settings in this analysis are as follows:

Fuzzy CS (FCS): In FCS, Lévy flight parameters are set as follows:  $P_a = 0.25$ ,  $\beta = 1.5$ ,  $\alpha = 0.1$ . Population size ( $n$ ) = 30 has been taken during experiment.

Bat Algorithm (BA) (Pérez et al., 2015a): The values of the parameters of BA are as follows: loudness ( $A$ ) = 0.9, pulse rate ( $r$ ) = 0.1, minimum frequency ( $f_{min}$ ) = 0, maximum frequency ( $f_{max} = 2$ ), population size ( $n$ ) = 40.

Particle Swarm Optimizer (PSO) (Nobile et al., 2018): For PSO,  $c_1$  and  $c_2$  stand for the acceleration coefficients which control the effect of local and global best solution over the current solution. Both  $c_1$  and  $c_2$  are set to 2 during the experiment. Population size ( $n$ ) = 40 has been considered for the experiment.

Cuckoo Search (CS) (Yang & Deb, 2009): CS has three main parameters, namely abandoned nest probability ( $P_a$ ), Lévy flight parameters  $\beta$  and  $\alpha$ . In traditional CS,  $0.75 \leq \beta \leq 1.95$ ,  $0 \leq P_a \leq 1$ , and  $\alpha > 0$ . Here the setting of these parameters has been performed is as follows:  $P_a = 0.25$ ,  $\beta = 1.5$ ,  $\alpha = 0.1$ , population size ( $n$ ) = 40.

Firefly Algorithm (FA) (Hassanzadeh & Kanan, 2014): FA has four main parameters, namely Attractiveness  $\beta_0 = 1$ , Light absorption coefficient  $s = 1$ , Lévy flight parameters  $\alpha = 0.1$ ,  $\beta = 1.5$ , population size ( $n$ ) = 80.

Slime mould algorithm (SMA) (Li et al., 2020): For SMA,  $z = 0.03$  and population size ( $n$ ) = 40.

The termination condition for all NIOA is the number of Fitness Evaluations (FEs). The maximum number of FEs (i.e. MAX\_FE) has been taken as  $10000 \times D$ . Where,  $D$  is the dimension of the search space for the Congress on Evolutionary Computation-2014 (CEC'14) Test Functions and  $1000 \times D$  for the multi-level thresholding based domain where  $D$  is the number of thresholds. Present studies show that current researchers prefer FEs over number of iterations (NIs) because function evaluations are also a crucial performance index used to measure the efficiency of evolutionary algorithms. Compared to computational complexity, function evaluation allows considering some technical aspects such as the computer system where the experiments run and implementation details, which directly influence the running CPU time focusing only on the ability of the algorithm to search within the solution space (Dhal, Das, Gálvez et al, 2020).

### 3. TEST FUNCTIONS SUITE AND MULTI-LEVEL THRESHOLDING PROBLEM

The experiment has been performed over Congress on Evolutionary Computation-2014 (CEC'14) benchmark suite (Liang et al., 2014) (Table 1) which consists of 30 benchmark functions that are divided into four classes:

- 1) Unimodal functions (1–3),
- 2) Simple multi-modal functions (4–16),
- 3) Hybrid functions (17–22), and
- 4) Composition functions (23–30).

There is only one global optimum and no local optimum in unimodal functions. This suite's unimodal functions are non-separable and rotated. There are two types of multi-modal functions: separable and non-separable. Additionally, they are rotated and/or shifted. To create hybrid functions, the variables are divided into subcomponents at random, and then different basic functions are used for each subcomponent. Composition functions are made up of two or more simple functions added together. Hybrid functions are used as the building blocks for composition functions in this suite. The characteristics of these hybrid and composition functions are determined by the basic functions' characteristics. The following methods were used to assess the efficacy of the proposed algorithm:

- a) Best solution (best) found over 30 runs,
- b) Worst solution (worst) found over 30 runs,
- c) Mean objective function (mean) value over 30 runs,
- d) Standard deviation (std. dev.) over 30 runs, and
- e) Median (Med.) of 30 runs.

Table 1. Summary of the Congress on Evolutionary Computation-2014 (CEC'14) Test Functions

Func. Type	Class	No.	Functions	$F_i^* = F_i(x^*)$ (Global Best)
Unimodal Functions	1	F1	Rotated High Conditioned Elliptic Function	100
		F2	Rotated Bent Cigar Function	200
		F3	Rotated Discus Function	300

Table 1 continued on next page

Table 1 continued

Func. Type	Class	No.	Functions	$F_i^* = F_i(x^*)$ (Global Best)
Simple Multimodal Functions	2	F4	Shifted and Rotated Rosenbrock's Function	400
		F5	Shifted and Rotated Ackley's Function	500
		F6	Shifted and Rotated Weierstrass Function	600
		F7	Shifted and Rotated Griewank's Function	700
		F8	Shifted Rastrigin's Function	800
		F9	Shifted and Rotated Rastrigin's Function	900
		F10	Shifted Schwefel's Function	1000
		F11	Shifted and Rotated Schwefel's Function	1100
		F12	Shifted and Rotated Katsuura Function	1200
		F13	Shifted and Rotated Happy Cat Function	1300
		F14	Shifted and Rotated HGBat Function	1400
		F15	Shifted and Rotated Expanded Griewank's plus Rosenbrock's Function	1500
F16	Shifted and Rotated Expanded Scaffer's F6 Function	1600		
Hybrid Functions	3	F17	Hybrid Function1 (N=3)	1700
		F18	Hybrid Function 2 (N=3)	1800
		F19	Hybrid Function 3 (N=4)	1900
		F20	Hybrid Function 4 (N=4)	2000
		F21	Hybrid Function 5 (N=5)	2100
		F22	Hybrid Function 6 (N=5)	2200
Composition Functions	4	F23	Composition Function 1 (N=5)	2300
		F24	Composition Function 2 (N=3)	2400
		F25	Composition Function 3 (N=3)	2500
		F26	Composition Function 4 (N=5)	2600
		F27	Composition Function 5 (N=5)	2700
		F28	Composition Function 6 (N=5)	2800
		F29	Composition Function 7 (N=3)	2900
		F30	Composition Function 8 (N=3)	3000

Search Range: [-100, 100]<sup>D</sup> D is the dimension of the search space , N is the number of basic functions

### 3.1. Mutli-level Thresholding

There are many problems in the field of image processing where an efficient search of solutions within a complex search domain is needed to find an optimal solution. One of them is multi-thresholding, which is an effective image segmentation technique. The multi-thresholding problem is an exponential combinatorial optimization problem that is typically formulated using a complex objective function

criterion that can only be solved using nondeterministic methods. As a result, researchers are addressing these issues by employing Nature-Inspired Optimization Algorithms (NIOAs) as a multi-thresholding problem alternative methodology (Dhal, Das, Ray et al, 2020).

We find the threshold points in multi-level thresholding based image segmentation in such a way that the segmented classes on the histogram satisfy the desired property. This is accomplished by maximizing or minimizing an objective function that includes thresholds as parameters.

Suppose, an image  $I$  with  $L$  gray levels are classified into  $K$  classes  $(C_1, C_2, \dots, C_i, \dots, C_K)$  using a set of  $(K - 1)$  threshold points  $T = (t_1, t_2, \dots, t_i, \dots, t_{K-1})$ , where  $t_1 < t_2 < \dots < t_{K-1}$ . Here for 8 bit image  $L = 256$  and gray level lie within the range  $[0, 255]$ . Therefore, a pixel with gray level  $g$  is belongs to class  $C_i$  if  $t_{i-1} < g < t_i$  for  $i = 1, 2, \dots, K$ . Thus single objective thresholding problem is the process of selecting the set of thresholds  $T'$  which optimizes the objective function  $F(T)$  such that

$$T' = \arg \max / \min_{0 \leq T \leq L-1} \{F(T)\} \quad (11)$$

In this study, two well-known objective functions are used which are discussed as follows.

### 3.1.1. Otsu's Method for Multi-level Thresholding

The Otsu methods for multi-level thresholding is based solely on maximization of inter class variances or minimization of intra class variances of segmented regions or classes (Alihodzic & Tuba, 2014; Brajevic & Tuba, 2014; Tuba et al., 2017). Suppose,  $\mu_1, \mu_2, \dots, \mu_m$  are the mean intensity of the classes 1, 2, ...,  $m$  respectively for multi-level thresholding problem.  $\mu$  is the global mean of the image.

$$\mu_0 = \sum_{i=0}^{t_1-1} \frac{ip_i}{w_0}, \mu_1 = \sum_{i=t_1}^{t_2-1} \frac{ip_i}{w_1}, \mu_j = \sum_{i=t_j}^{t_{j+1}-1} \frac{ip_i}{w_j}, \mu_m = \sum_{i=t_m}^{L-1} \frac{ip_i}{w_m}, \text{ and } \mu = \sum_{i=0}^{L-1} ip_i \quad (12)$$

where,

$$w_0 = \sum_{i=0}^{t_1-1} p_i, w_1 = \sum_{i=t_1}^{t_2-1} p_i, w_j = \sum_{i=t_j}^{t_{j+1}-1} p_i, w_m = \sum_{i=t_m}^{L-1} p_i \quad (13)$$

where,  $0 \leq p_i \leq 1$  denotes the probability of the state  $i$ . In the case of gray level image, it represents the occurrence of the  $i^{th}$  gray level in the image and  $\sum_{i=1}^{L-1} p_i = 1$ . Then the inter class variance of each class has been given as:

Therefore, Otsu based multi-level thresholding process can be formulized as follows:

$$(t_1, t_2, \dots, t_m) = \text{Argmax} \left( \sum_{i=0}^m \sigma_i \right) \quad (14)$$

where,

$$\sigma_0 = w_0 (\mu_0 - \mu)^2, \sigma_1 = w_1 (\mu_1 - \mu)^2, \sigma_j = w_j (\mu_j - \mu)^2, \text{ and } \sigma_m = w_m (\mu_m - \mu)^2 \quad (15)$$

### 3.1.2. Kapur's Entropy for Multi-level Thresholding

Let  $P = (p_1, p_2, p_3, \dots, p_n) \in \Delta_n$  where

$$\Delta_n = \left\{ (p_1, p_2, \dots, p_n) \mid p_i \geq 0, i = 1, 2, \dots, n; n \geq 2; \sum_{i=1}^n p_i = 1 \right\}$$

is a set of discrete finite  $n$ -ary probability distributions. Then entropy of the total image can be defined as (Alihodzic & Tuba, 2014; Brajevic & Tuba, 2014; Tuba et al., 2017):

$$H(P) = -\sum_{i=1}^n p_i \log_2 p_i \quad (16)$$

$I$  denote a 8 bit gray level digital image of dimension  $M \times N$ .  $P$  is the normalized histogram for image with  $L = 256$  gray levels. Now, if there are  $n - 1$  thresholds  $(t)$ , partitioning the normalized histogram into  $n$  classes, then the entropy for each class may be computed as,

$$H_2(t) = -\sum_{i=t_1+1}^{t_2} \frac{p_i}{P_2} \ln \frac{p_i}{P_2},$$

$$H_n(t) = -\sum_{i=t_{n-1}+1}^{L-1} \frac{p_i}{P_n} \ln \frac{p_i}{P_n}. \quad (17)$$

where,

$$P_1(t) = \sum_{i=0}^{t_1} p_i, P_2(t) = \sum_{i=t_1+1}^{t_2} p_i, \dots, P_n(t) = \sum_{i=t_{n-1}+1}^{L-1} p_i \quad (18)$$

where, for ease of computation, two dummy thresholds  $t_0 = 0, t_n = L - 1$  are introduced with  $t_0 < t_1 < \dots < t_{n-1} < t_n$ . Then the optimum threshold value can be found by

$$\varphi(t_1, t_2, \dots, t_n) = \text{Arg max} \left( \left[ H_1(t) + H_2(t) + \dots + H_n(t) \right] \right) \quad (19)$$

## 4. EXPERIMENTAL RESULTS

All the experiments have been done using MatlabR2018 with Windows-8 OS, x64-based PC, Intel Core-i5-CPU, 4 GB RAM. The optimization ability of the proposed FCS has been tested over CEC'14 benchmark suite where different types of mathematical functions have been considered. In addition to that, the performance of the FCS is also evaluated by employing in pathology image segmentation domain. FCS has been employed for finding the optimal threshold points for the proper segmentation of White Blood Cells (WBCs) from the blood pathology images. Here, 100 color pathology images have been taken for experiment which will prove the real life applicability of the proposed FCS algorithm. To validate the performance of the proposed FCS, it is compared with six state-of-the-arts NIOA which are classical CS, FA, BA, SMA, and PSO. The details of the results over CEC'14 and pathology images are presented in the following sections.

### 4.1. Results over CEC'14

Tables 2, 3, and 4 show the results of the CEC'14 test feature suite. The experimental findings are presented for seven of the CEC'14 benchmark suite's total of thirty features. This study considers one unimodal function (F1), two multimodal functions (F6 and F8), four hybrid functions (F17 and F20), and four composition functions (F24 and F20). Tables 2, 3, and 4 show the best, worst, mean, standard deviation, and median of NIOAs across different functions for dimensions 10, 30, and 50, respectively. In the tables, the best results are highlighted in bold. Tables clearly show that FCS delivers the best results in any dimension and for any form of function. The results of traditional CS, SMA, and FA are second best.

Table 2. Results over test functions with Dimension D = 10

Function	Algorithm	Best	Worst	Mean	Median	Std. dev.
F1 (Unimodal)	FCS	<b>1.00E+02</b>	<b>1.00e+02</b>	<b>1.00e+02</b>	<b>1.00e+02</b>	<b>1.29e-03</b>
	CS	1.09E+02	2.31e+02	1.49e+02	1.39e+02	2.99e+01
	FA	7.87E+04	4.84E+05	2.24E+05	2.24E+05	1.01E+05
	PSO	1.55E+05	3.04E+06	1.48E+06	1.32E+06	7.51E+05
	BA	4.50E+03	3.65E+08	8.63E+07	6.05E+07	7.83E+07
	SMA	6.54E+04	3.98E+05	2.68E+05	2.68E+05	2.02E+05
F6 (Multimodal)	FCS	<b>6.00e+02</b>	<b>6.02e+02</b>	<b>6.01e+02</b>	<b>6.01E+02</b>	<b>3.41e-01</b>
	CS	6.02E+02	6.05E+02	6.04E+02	6.04E+02	6.13E-01
	FA	6.01E+02	<b>6.02E+02</b>	<b>6.01E+02</b>	<b>6.01E+02</b>	3.89E-01
	PSO	6.04E+02	6.07E+02	6.05E+02	6.05E+02	9.01E-01
	BA	6.07E+02	6.13E+02	6.11E+02	6.11E+02	0.15E+01
	SMA	6.01E+02	<b>6.02E+02</b>	<b>6.01E+02</b>	<b>6.01E+02</b>	6.09E-01
F8 (Multimodal)	FCS	<b>8.00e+02</b>	<b>8.02e+02</b>	<b>8.00e+02</b>	<b>8.00e+02</b>	<b>5.73e-01</b>
	CS	8.02E+02	8.07E+02	8.05E+02	8.05E+02	1.34E+00
	FA	8.04E+02	8.15E+02	8.08E+02	8.08E+02	2.89E+00
	PSO	8.19E+02	8.39E+02	8.30E+02	8.32E+02	4.88E+00
	BA	8.29E+02	9.09E+02	8.54E+02	8.48E+02	1.98E+01
	SMA	8.11E+02	8.14E+02	8.06E+02	8.06E+02	1.09E+00

Table 2 continued on next page

Table 2 continued

Function	Algorithm	Best	Worst	Mean	Median	Std. dev.
F17 (Hybrid)	FCS	<b>1.70 e + 03</b>	<b>1.75 e + 03</b>	<b>1.72 e + 03</b>	<b>1.72 e + 03</b>	<b>1.31 e + 01</b>
	CS	1.72E+03	1.78E+03	1.75E+03	1.74E+03	1.78E+01
	FA	2.65E+03	3.03E+04	4.93E+03	4.38E+03	2.14E+03
	PSO	6.65E+03	2.98E+04	1.50E+04	1.44E+04	5.92E+03
	BA	3.71E+03	4.57E+06	8.68E+05	4.08E+05	1.08E+06
	SMA	2.68E+03	2.88E+04	4.68E+03	4.12E+03	3.32E+03
F20 (Hybrid)	FCS	<b>2.00 e + 03</b>	<b>2.00 e + 03</b>	<b>2.00 e + 03</b>	<b>2.00 e + 03</b>	<b>2.43 e - 01</b>
	CS	<b>2.00E+03</b>	<b>2.00E+03</b>	<b>2.00E+03</b>	<b>2.00E+03</b>	5.19E-01
	FA	2.07E+03	1.14E+04	2.78E+03	2.37E+03	6.70E+02
	PSO	2.09E+03	3.96E+03	2.45E+03	2.32E+03	3.87E+03
	BA	2.59E+03	4.14E+06	4.14E+05	1.00E+06	7.58E+04
	SMA	2.08E+03	2.04E+04	2.99E+03	2.41E+03	1.66E+03
F24 (Composite)	FCS	<b>2.51 e + 03</b>	<b>2.51 e + 03</b>	<b>2.51 e + 03</b>	2.51 e + 03	3.01 e + 00
	CS	<b>2.51E+03</b>	2.52E+03	2.52E+03	2.52E+03	4.12E+00
	FA	<b>2.51E+03</b>	2.62E+03	<b>2.51E+03</b>	2.51E+03	0.37E+01
	PSO	2.53E+03	2.55E+03	2.54E+03	2.54E+03	4.57E+00
	BA	2.55E+03	2.64E+03	2.60E+03	2.60E+03	2.10E+01
	SMA	<b>2.51E+03</b>	2.65E+03	<b>2.51E+03</b>	2.51E+03	1.03E+01
F25 (Composite)	FCS	<b>2.61E+03</b>	<b>2.67E+03</b>	<b>2.61E+03</b>	<b>2.64E+03</b>	2.46E+01
	CS	2.62E+03	2.69E+03	2.65E+03	2.65E+03	1.66E+01
	FA	<b>2.61E+03</b>	2.70E+03	<b>2.61E+03</b>	<b>2.64E+03</b>	3.06E+01
	PSO	2.65E+03	2.70E+03	2.69E+03	2.70E+03	1.71E+01
	BA	2.72E+03	2.78E+03	2.74E+03	2.74E+03	<b>1.30E+01</b>
	SMA	<b>2.64E+03</b>	2.68E+03	2.62E+03	<b>2.64E+03</b>	2.96E+01



Table 3. Results over test functions with Dimension D = 30

Function	Algorithm	Best	Worst	Mean	Median	Std. dev.
F1 (Unimodal)	FCS	<b>1.92E+05</b>	<b>1.94E+06</b>	<b>6.32E+05</b>	<b>5.52E+05</b>	<b>2.45E+05</b>
	CS	5.13E+05	2.27E+06	1.38E+06	1.41E+06	4.45E+05
	FA	2.72E+06	2.05E+07	8.28E+06	6.89E+06	4.56E+06
	PSO	4.95E+07	1.62E+08	1.10E+08	1.03E+08	2.85E+07
	BA	1.93E+08	1.63E+09	7.95E+08	7.77E+08	4.42E+08
	SMA	2.78E+06	2.14E+07	8.55E+06	7.12E+06	7.56E+06
F6 (Multimodal)	FCS	<b>6.04E+02</b>	<b>6.12E+02</b>	<b>6.08E+02</b>	<b>6.08E+02</b>	<b>1.19E+00</b>
	CS	6.23E+02	6.27E+02	6.25E+02	6.25E+02	1.30E+00
	FA	6.05E+02	<b>6.12E+02</b>	<b>6.08E+02</b>	<b>6.08E+02</b>	1.93E+00
	PSO	6.25E+02	6.34E+02	6.30E+02	6.30E+02	1.71E+00
	BA	6.37E+02	6.48E+02	6.42E+02	6.41E+02	3.02E+00
	SMA	6.11E+02	6.42E+02	6.28E+02	6.30E+02	2.06E+00
F8 (Multimodal)	FCS	<b>8.20E+02</b>	<b>8.41E+02</b>	<b>8.31E+02</b>	<b>8.31E+02</b>	<b>3.45E-02</b>
	CS	8.31E+02	8.64E+02	8.44E+02	8.44E+02	8.05E+00
	FA	8.24E+02	8.80E+02	8.47E+02	8.46E+02	1.26E+01
	PSO	9.95E+02	1.03E+03	1.01E+03	1.01E+03	1.18E+01
	BA	9.17E+02	1.08E+03	1.01E+03	1.01E+03	4.34E+01
	SMA	8.36E+02	8.81E+02	8.49E+02	8.47E+02	0.26E+01
F17 (Hybrid)	FCS	<b>3.00E+03</b>	<b>4.54E+03</b>	<b>3.61E+03</b>	<b>3.59E+03</b>	<b>4.03E+02</b>
	CS	3.45E+03	5.31E+03	4.56E+03	4.65E+03	4.58E+02
	FA	4.91E+04	1.54E+06	5.98E+05	5.23E+05	4.27E+05
	PSO	1.78E+06	5.62E+06	3.11E+06	3.08E+06	8.22E+05
	BA	8.96E+05	1.55E+08	4.44E+07	2.51E+07	4.39E+07
	SMA	2.77E+05	2.58E+06	6.66E+05	5.95E+05	3.33E+06
F20 (Hybrid)	FCS	2.04E+03	<b>2.08E+03</b>	<b>2.05E+03</b>	<b>2.06E+03</b>	2.01E+01
	CS	<b>2.03E+03</b>	2.09E+03	2.06E+03	<b>2.06E+03</b>	<b>1.41E+01</b>
	FA	2.93E+03	1.46E+04	5.70E+03	4.81E+03	2.69E+03
	PSO	3.55E+03	9.18E+03	5.46E+03	4.96E+03	1.49E+03
	BA	2.89E+04	4.27E+06	3.82E+05	7.76E+05	1.67E+05
	SMA	2.91E+03	1.48E+04	5.72E+03	4.85E+03	3.39E+03
F24 (Composite)	FCS	<b>2.61E+03</b>	<b>2.61E+03</b>	<b>2.61E+03</b>	<b>2.61E+03</b>	<b>0.92E+00</b>
	CS	2.62E+03	2.62E+03	2.62E+03	2.62E+03	1.06E+00
	FA	2.62E+03	2.62E+03	2.63E+03	2.63E+03	1.29E+01
	PSO	2.69E+03	2.71E+03	2.70E+03	2.70E+03	4.85E+00
	BA	2.66E+03	2.75E+03	2.71E+03	2.40E+01	2.71E+03
	SMA	2.62E+03	2.62E+03	2.63E+03	2.63E+03	1.09E+01

Table 3 continued on next page

Table 3 continued

Function	Algorithm	Best	Worst	Mean	Median	Std. dev.
F25 (Composite)	FCS	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>8.02E-01</b>
	CS	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>8.38E-01</b>
	FA	<b>2.70E+03</b>	2.71E+03	<b>2.70E+03</b>	2.71E+03	0.15E+00
	PSO	2.71E+03	2.72E+03	2.72E+03	2.71E+03	3.00E+00
	BA	2.72E+03	2.78E+03	2.74E+03	1.30E+01	2.74E+03
	SMA	<b>2.70E+03</b>	2.71E+03	<b>2.70E+03</b>	2.71E+03	0.98E+00

Table 4. Results over test functions with Dimension D = 50

Function	Algorithm	Best	Worst	Mean	Median	Std. dev.
F1 (Unimodal)	FCS	<b>8.88E+04</b>	<b>1.95E+06</b>	<b>8.59E+05</b>	<b>8.01E+05</b>	<b>3.65E+05</b>
	CS	4.06E+06	1.09E+07	7.22E+06	7.33E+06	1.63E+06
	FA	6.98E+06	1.98E+07	1.16E+07	1.06E+07	3.30E+06
	PSO	2.37E+08	5.54E+08	3.97E+08	3.98E+08	7.73E+07
	BA	8.23E+07	3.34E+09	1.56E+09	1.42E+09	8.83E+08
	SMA	6.98E+06	2.08E+07	1.32E+07	1.28E+07	4.35E+06
F6 (Multimodal)	FCS	<b>6.12E+02</b>	<b>6.25E+02</b>	<b>6.18E+02</b>	<b>6.18E+02</b>	<b>1.17E+00</b>
	CS	6.43E+02	6.53E+02	6.49E+02	6.49E+02	2.40E+00
	FA	<b>6.12E+02</b>	6.28E+02	6.19E+02	<b>6.18E+02</b>	3.96E+00
	PSO	6.25E+02	6.34E+02	6.30E+02	6.30E+02	1.71E+00
	BA	6.70E+02	6.85E+02	6.77E+02	6.76E+02	4.07E+00
	SMA	6.20E+02	6.30E+02	6.25E+02	6.25E+02	2.01E+00
F8 (Multimodal)	FCS	<b>8.20E+02</b>	<b>8.63E+02</b>	<b>8.35E+02</b>	<b>8.37E+02</b>	<b>4.63E+00</b>
	CS	8.83E+02	9.55E+02	9.23E+02	9.26E+02	1.67E+01
	FA	8.59E+02	9.46E+02	9.07E+02	9.05E+02	2.38E+01
	PSO	9.95E+02	1.03E+03	1.01E+03	1.01E+03	1.18E+01
	BA	1.00E+03	1.35E+03	1.16E+03	1.17E+03	6.01E+01
	SMA	8.68E+02	9.50E+02	9.27E+02	9.16E+02	3.07E+01
F17 (Hybrid)	FCS	<b>2.26E+04</b>	<b>1.49E+05</b>	<b>6.00E+04</b>	<b>5.33E+04</b>	<b>4.55E+04</b>
	CS	6.89E+04	3.94E+05	2.04E+05	1.83E+05	8.37E+04
	FA	1.47E+05	4.39E+06	1.33E+06	9.06E+05	1.10E+06
	PSO	7.38E+06	2.83E+07	1.81E+07	1.76E+07	5.41E+06
	BA	1.05E+07	2.60E+08	1.30E+08	1.33E+08	6.35E+07
	SMA	1.50E+05	4.35E+06	1.46E+06	8.86E+05	2.17E+06

Table 4 continued on next page

Table 4 continued

Function	Algorithm	Best	Worst	Mean	Median	Std. dev.
F20 (Hybrid)	FCS	<b>2.14E+03</b>	<b>2.46E+03</b>	<b>2.27E+03</b>	<b>2.28E+03</b>	<b>5.83E+01</b>
	CS	2.25E+03	2.51E+03	2.35E+03	2.34E+03	6.76E+01
	FA	3.22E+03	1.51E+04	7.53E+03	6.84E+03	3.00E+03
	PSO	1.30E+04	4.53E+04	2.66E+04	2.49E+04	7.93E+03
	BA	3.50E+04	1.55E+06	9.00E+00	3.01E+05	1.46E+05
	SMA	3.22E+03	1.59E+04	6.97E+03	6.54E+03	2.98E+03
F24 (Composite)	FCS	<b>2.65E+03</b>	<b>2.66E+03</b>	<b>2.65E+03</b>	<b>2.65E+03</b>	<b>1.27E+00</b>
	CS	2.66E+03	<b>2.66E+03</b>	2.66E+03	2.66E+03	1.43E+00
	FA	2.67E+03	2.68E+03	2.67E+03	2.67E+03	0.27E+01
	PSO	2.83E+03	2.87E+03	2.84E+03	2.84E+03	9.10E+00
	BA	2.75E+03	3.03E+03	2.87E+03	6.30E+01	2.86E+03
	SMA	2.69E+03	2.88E+03	2.69E+03	2.70E+03	0.65E+01
F25 (Composite)	FCS	<b>2.70E+03</b>	<b>2.71E+03</b>	<b>2.70E+03</b>	<b>2.70E+03</b>	<b>1.88E+00</b>
	CS	2.71E+03	2.73E+03	2.72E+03	2.72E+03	3.13E+00
	FA	2.71E+03	2.72E+03	2.71E+03	2.71E+03	0.29E+01
	PSO	2.75E+03	2.79E+03	2.77E+03	2.77E+03	1.04E+00
	BA	2.74E+03	2.83E+03	2.77E+03	2.00E+00	2.77E+03
	SMA	2.71E+03	2.72E+03	2.71E+03	2.71E+03	0.91E+01

#### 4.2. Results over Multi-level Thresholding Domain

NIOA were proposed as stochastic search mechanisms for solving complex engineering problems involving multiple global and local minima, where conventional mathematical methods are ineffective. Since multi-level color image segmentation leads to a multimodal optimization challenge, NIOA is viewed as an alternative strategy for determining the best set of threshold values. Their performance should be evaluated using a multi-level color pathology image segmentation scheme. In this part, the performance of the proposed FCS has been tested using Otsu and Kapur's entropy as objective functions on well-known benchmark images of Acute Lymphoblastic Leukemia (ALL) patients, namely ALL-IDB (Labati et al., 2011). All the images in the datasets were taken with a PowerShot G5 camera and stored in JPG format with a 24 bit colour depth and a native resolution of 2592 x 1944 pixels. The images correspond to different microscope magnifications (ranging from 300 to 500). The ALL-IDB database is divided into two versions (ALL-IDB1 and ALL-IDB2), both of which are focused on segmentation and classification. The ALL-IDB1 can also be used to test algorithms' segmentation capabilities as well as classification systems' accuracy. There are 108 images in this dataset. ALL-IDB2 is belongs to the ALL-IDB1 dataset that includes cropped areas of interest of normal and blast cells. There are 260 pictures in all, with lymphoblasts representing for 50% of them. The grey level characteristics of ALL-IDB2 images are identical to those of ALL-IDB1. Proper Enhancement (Dhal, Ray, Das, Biswas et al, 2019) and segmentation (Dhal, Gálvez, & Das, 2020) of digital pathology images are critical and play a key role in a computer-aided diagnosis (CAD) system.

For segmentation and image classification, the ALL-IDB dataset is used. It focuses on Acute Lymphoblastic Leukemia (ALL), a severe blood disease that can be lethal in as little as a few weeks if

left untreated. ALL is more prevalent in children, with a peak occurrence between the ages of two and five years. The ALL disease is related to the lymphocytes in the bone marrow and into the peripheral blood. Lymphocytes of ALL patients are called lymphoblasts. Segmentation of White Blood Cell (WBC), especially lymphocytes/ lymphoblasts is an essential step in an automatic method for the detection of Acute Lymphoblastic Leukemia.

This study used 100 images from the aforementioned datasets to demonstrate the robustness of the FCS. To make a fair comparison among NIOA, each execution of the tested objective functions considers the number of function evaluations  $NFE = 1,000 * d$ , as stop criterion of the optimization process. This criterion has been selected to promote compatibility, with previously published works in the literature. The experiments are evaluated considering the number of threshold values ( $nt$ ) set to 3, 4, and 5 which correspond to the  $d$ -dimensional search space in an optimization problem formulation. For each image, an optimization process is achieved by the evaluation of each objective function. Standard deviations, average fitness values, computational time, and common quality image metrics such as Peak Signal to Noise Ratio (PSNR), Quality Index based on Local Variance (QILV), and Feature Similarity Index (FSIM) are used to compare the numerical results obtained by the tested segmentation methods. To remove the random effect of numerical results among independent runs, a non-parametric framework based on Wilcoxon rank sum test (García et al., 2009) is used to statistically validate and corroborate the numerical results. Three segmentation quality parameters are considered to judge the segmentation ability of the utilized algorithms. The brief description of the quality parameters are given in Table 5.

Table 5. Quality parameters to evaluate the performance of the proposed segmentation methods.

Sl.	Parameters	Formulation	Remarks
1.	Feature Similarity Index (FSIM) (Dhal, Gálvez, Ray et al., 2020; Zhang et al., 2011)	$FSIM = \frac{\sum_{x \in \mathbb{C}} S_L(x) \cdot PC_m(x)}{\sum_{x \in \mathbb{C}} PC_m(x)}$	Defines the quality score which reflects the significance of a local structure. High Value reflects better results.
3.	Peak Signal to Noise Ratio (PSNR) (Aja-Fernández et al., 2006)	$PSNR = 10 \log_{10} \frac{(2^b - 1)^2}{\sqrt{MSE}}$	Represents the ratio between the maximum possible power of a signal and the power of corrupting noise. High PSNR value indicates better result.
4.	Quality Index based on Local Variance (QILV) (Aja-Fernández et al., 2006; Dhal, Gálvez, Ray et al., 2020)	$QILV(I, J) = \frac{2\mu_{V_i} \mu_{V_j}}{\mu_{V_i}^2 + \mu_{V_j}^2} \Delta \frac{2\sigma_{V_i} \sigma_{V_j}}{\sigma_{V_i}^2 + \sigma_{V_j}^2} \Delta \frac{\sigma_{V_i V_j}}{\sigma_{V_i} \sigma_{V_j}}$	Finds the similarity between segmented image and uncompressed or distortion-free image. High QILV value indicates better result.

For each image, each NIOA-based thresholding model has been run 40 times and the best run was registered. The images of various blast cells from Acute Lymphoblastic Leukemia are shown in Fig. 1.

Figure 1. Original Acute Lymphoblastic Leukemia images

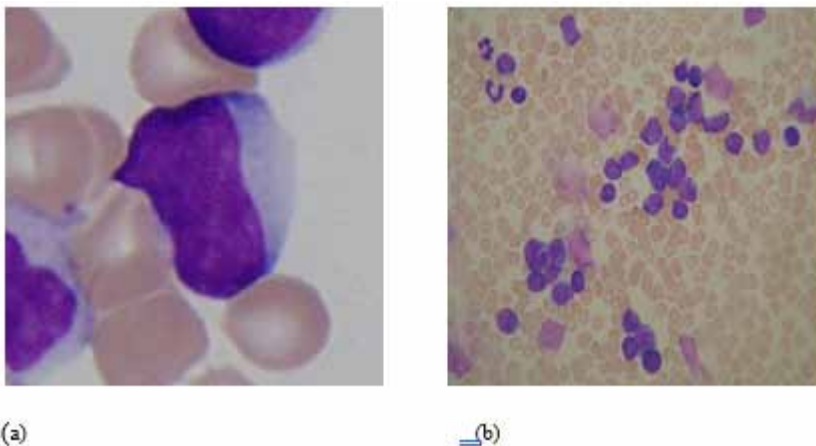


Figure 2. FCS based segmented outputs for Fig.1(a) using the Otsu and Kapur's entropy objective functions

$f$	$\alpha t = 3$	$\alpha t = 4$	$\alpha t = 5$
Kapur			
Otsu			

Figure 3. FCS based segmented outputs for Fig.1(b) using the Otsu and Kapur's entropy objective functions

$f$	$\alpha t = 3$	$\alpha t = 4$	$\alpha t = 5$
Kapur			
Otsu			

Table 6. Numerical comparison for Kapur's entropy objective function. (Best results given in bold)

Number of thresholds ( <i>ni</i> )	NIOA	$\bar{f}$	$\sigma_f$	Time (src.)	PSNR	QILV	FSIM
3	FCS	<b>2.27E+01</b>	<b>2.21E-03</b>	2.426	<b>16.08</b>	<b>0.4799</b>	<b>0.7108</b>
	CS	2.21E+01	7.23E-01	<b>2.216</b>	15.45	0.4588	0.6906
	FA	2.21E+01	5.34E-01	3.187	15.38	0.4491	0.6883
	PSO	2.20E+01	7.02E-02	3.881	14.91	0.4486	0.6767
	BA	2.20E+01	1.12E-01	3.892	14.92	0.4486	0.6765
	SMA	2.20E+01	5.58E-02	3.174	14.61	0.4486	0.6767
4	FCS	<b>2.76E+01</b>	<b>2.21E-02</b>	2.588	<b>16.76</b>	<b>0.5318</b>	<b>0.7308</b>
	CS	2.56E+01	1.45E-01	<b>2.501</b>	15.69	0.4902	0.7044
	FA	2.59E+01	3.29E-01	3.784	15.88	0.5129	0.7170
	PSO	2.54E+01	8.09E-02	3.989	15.33	0.4876	0.6989
	BA	2.56E+01	7.45E-01	4.001	15.70	0.5118	0.7026
	SMA	2.58E+01	2.25E-01	3.698	15.88	0.5128	0.7158
5	FCS	<b>3.15E+01</b>	<b>8.24E-05</b>	3.001	<b>18.51</b>	<b>0.7144</b>	<b>0.7699</b>
	CS	2.87E+01	3.76E-04	<b>2.997</b>	17.52	0.6789	0.7123
	FA	2.89E+01	4.09E-04	4.106	17.68	0.6887	0.7211
	PSO	2.82E+01	4.21E-04	4.602	17.17	0.6568	0.7064
	BA	2.87E+01	4.21E-03	4.765	17.67	0.6786	0.7185
	SMA	2.87E+01	5.33E-04	4.004	17.61	0.6821	0.7113

The segmented images by FCS with Otsu and Kapur's entropy corresponding to Figs.1 are given as Figs.2 and 3. Table 6 represents average numerical values of PSNR, QILV, FSIM, standard deviation ( $\sigma_f$ ), Computational time, and fitness function ( $\bar{f}$ ) values over 100 images of all the tested NIOAs with Kapur's entropy as objective function. The average is calculated by sum of best run value of each 100 image divided by number of images. Average fitness and standard deviation clearly show that FCS provides the best results over thresholds 3, 4, and 5. FA provides the second best results by considering the same numerical analysis. Performance of CS deteriorates when number of thresholds is 5. Therefore it can be said that CS suffering when dimension increases. PSO gives worst results among all the tested NIOAs. Computational effort of classical CS algorithm is least. Whereas, FCS is the second best according to the computational time. Values of PSNR, FSIM, and QILV demonstrates that FCS with Kapur's entropy is the best choice for the image WBC segmentation of ALL hematology images. Comparison among the NIOAs of Kapur's entropy objective function maximization ability has been given as Fig. 4.

Figure 4. Average Fitness Comparison of NIOAs for Kapur's Entropy

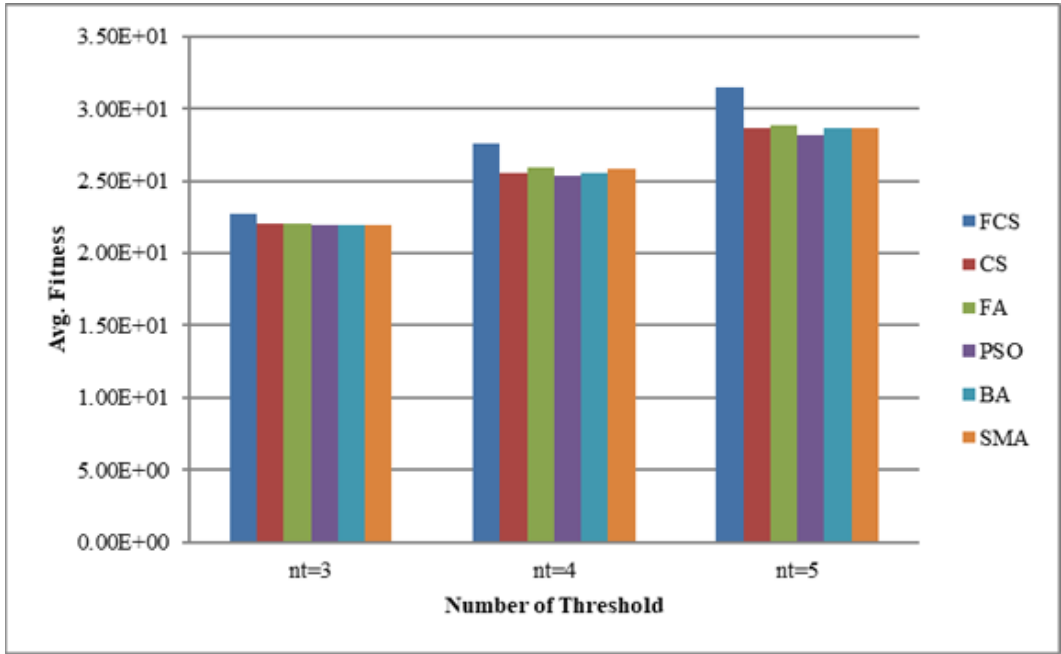


Table 7. Numerical comparison for Otsu's criterion objective function. (Best results given in bold)

Number of thresholds (nt)	NIOA	$\bar{f}$	$\sigma_f$	Time (sec.)	PSNR	QILV	FSIM
3	FCS	<b>8.10E+02</b>	<b>5.49E-02</b>	2.822	<b>16.35</b>	<b>0.8240</b>	<b>0.7792</b>
	CS	7.87E+02	7.21E-02	<b>2.206</b>	15.91	0.7899	0.7514
	FA	7.89E+02	9.35E-01	3.128	15.95	0.7910	0.7529
	PSO	7.85E+02	7.22E-02	3.902	15.90	0.7869	0.7517
	BA	7.86E+02	3.32E-01	3.911	15.90	0.7877	0.7518
	SMA	7.87E+02	8.04E-02	3.107	15.93	0.7895	0.7515
4	FCS	<b>8.16E+02</b>	<b>6.22E-03</b>	3.191	<b>17.92</b>	<b>0.8619</b>	<b>0.7926</b>
	CS	7.99E+02	5.76E-01	<b>3.068</b>	17.74	0.8548	0.7643
	FA	8.01E+02	6.01E-02	4.283	17.85	0.8598	0.7697
	PSO	7.91E+02	3.27E-02	4.511	16.98	0.8513	0.7636
	BA	7.98E+02	5.54E-02	4.544	17.80	0.8547	0.7642
	SMA	8.01E+02	7.55E-02	4.107	17.88	0.8561	0.7638
5	FCS	<b>8.22E+02</b>	<b>1.62E-02</b>	3.381	<b>20.91</b>	<b>0.8975</b>	<b>0.8257</b>
	CS	8.04E+02	1.82E-02	<b>3.291</b>	19.22	0.8662	0.7853
	FA	8.10E+02	8.89E-03	4.651	20.73	0.8797	0.8201
	PSO	8.02E+02	8.40E-02	4.871	19.06	0.8663	0.7809
	BA	8.04E+02	4.44E-01	4.859	19.43	0.8661	0.7856
	SMA	8.08E+02	2.88E-02	4.523	19.85	0.8742	0.8144

Table 7 represents average numerical values of PSNR, QILV, FSIM, standard deviation ( $\sigma_f$ ), Computational time, and fitness function ( $\bar{f}$ ) values over 100 images of all the tested NIOAs with Otsu interclass variance as objective function. Average fitness and standard deviation clearly show that FCS provides the best results over thresholds 3, 4, and 5. Here, again FA provides the second best results. Optimization ability of the CS deteriorates when number of thresholds is 5. PSO gives worst results among all the tested NIOAs. Classical CS algorithm takes least time to find the threshold values. Whereas, FCS is the second best according to the computational time. Values of PSNR, FSIM, and QILV demonstrates that FCS with Otsu is the best choice for the image WBC segmentation of ALL hematology images. Comparison among the NIOAs of Otsu objective function maximization ability has been given as Fig. 5.

From Tables 6 and 7, we can also conclude that if numbers of thresholds increase, value of PSNR, QILV, and FSIM also increase for the both objective functions.

Figure 5. Average Fitness Comparison of NIOAs for Otsu Criterion

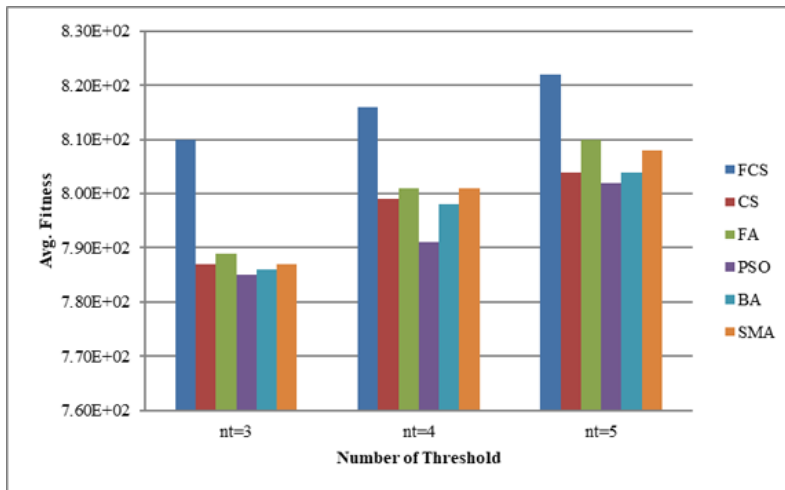


Table 8. Comparison among optimization algorithms depending on Wilcoxon p-values

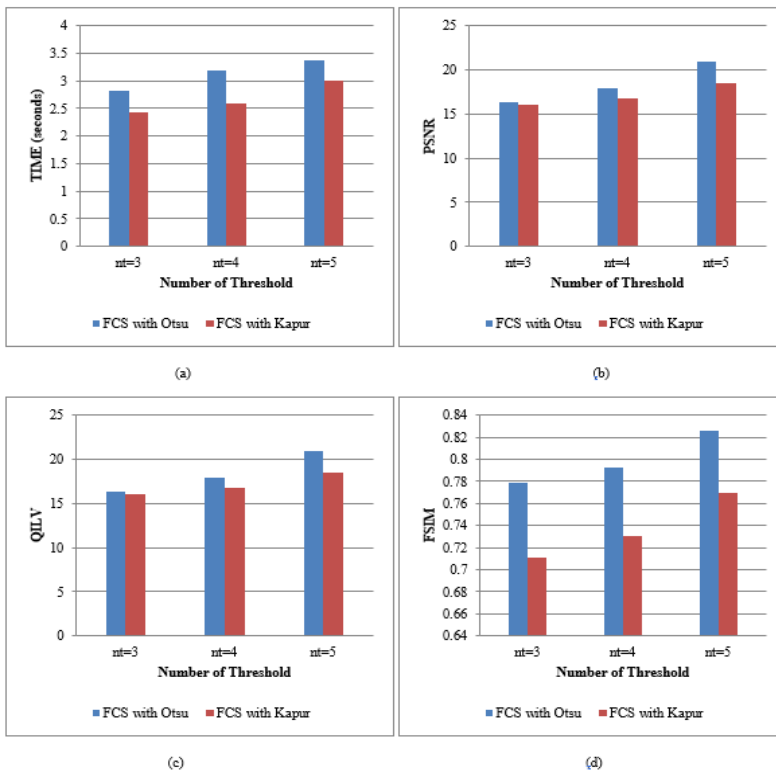
Thresholds → Algorithms ↓	Kapur's Entropy						Otsu Criterion					
	nt = 3		nt = 4		nt = 5		nt = 3		nt = 4		nt = 5	
	p	h	p	h	p	h	p	h	p	h	P	h
FCS vs. CS	<0.05	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
FCS vs. FA	<0.05	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
FCS vs. PSO	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
FCS vs. BA	<0.05	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1
FCS vs. SMA	<0.05	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1	<0.05+	1

"+" indicates significant difference



The fitness values of the FCS have been compared with other NIOAs using a nonparametric significance proof known as the Wilcoxon’s rank test (García et al., 2009) that is conducted with 100 independent samples (40 runs). Such proof allows assessing result differences among two related methods. Wilcoxon’s rank test at the 5% significance level is used to determine if the results obtained with the best performing algorithm vary statistically significantly from the final results of the other competitors. A  $p$ -value of less than 0.05 (5% significance level) strongly supports the rejection of the null hypothesis, suggesting that the best algorithm’s results vary statistically significantly from those of the other peer algorithms and that the discrepancy is not due to chance. Table 8 reports the  $p$ -values produced by Wilcoxon’s test for a pairwise comparison of the fitness function between two groups formed as FCS vs. CS, FCS vs. FA, FCS vs. PSO, FCS vs. BA, and FCS vs. SMA for 3, 4, and 5 number of thresholds. All of the  $p$  values in Table 8 are less than 0.05 (5% significance level), and  $h = 1$  is clear proof against the null hypothesis, showing that the FCS fitness values for the performance are statistically higher, and this is not a fluke. From Tables 6 and 7 it is clearly noticed that FCS with Otsu is better than FCS with Kapur’s entropy for WBC segmentation of ALL hematology images. The values of computational time, PSNR, QILV, and FSIM corresponding to FCS with Otsu and FCS with Kapur’s entropy have been compared and represented as Fig. 6. It can be seen from the figures that FCS with Otsu provides better values of quality parameters compare to FCS with Kapur’s entropy. However, the computational effort of FCS with Otsu is slightly higher than FCS with Kapur’s entropy. Not only FCS, all the tested NIOA with Otsu gave superior outcomes in terms of segmentation quality parameters compare to Kapur’s entropy with the concerned NIOA. Therefore, it can be said that Otsu is superior to Kapur’s entropy to segment the WBC of the ALL hematology images.

Figure 6. Comparison of FCS with Otsu and FCS with Kapur’s Entropy based on: (a) Computational time; (b) PSNR; (c) QILV; (d) FSIM.



## 5. CONCLUSION

The Fuzzy Cuckoo Search (FCS) algorithm is a fuzzy population-based improved Cuckoo Search algorithm presented in this study. In the FCS algorithm, the population is divided into two groups: best and worst, with each solution belonging to one of the set based on fitness. The solutions are enhanced with the aid of fuzzy set centroids, global best solution guidance, and levy flight dependent mutation. With Otsu and Kapur's entropy as objective functions, the proposed FCS algorithm is employed to the CEC'14 function suites and multi-level thresholding based image segmentation domain. In comparison to other studied NIOA, such as CS, BA, FA, SMA, and PSO, the experimental study shows that FCS performs exceptionally well in minimizing the CEC'14 test functions. FCS also outperforms the other algorithms in Otsu and Kapur's entropy-based multi-level color hematology image segmentation domain. In comparison to FCS with Kapur's entropy, the numerical results show that FCS with Otsu as the objective function produces the best segmented outcomes of color hematology images in terms of segmentation consistency parameters. The computational effort of the Otsu with FCS, on the other hand, is higher than that of FCS with Kapur's entropy. With the tested NIAOs, Otsu is superior to Kapur's entropy as an objective function for segmentation of the concerned color hematology images, according to the study. The numerical data is supported by a statistical framework that employs a non-parametric approach to identify substantial differences between FCS and the rest of NIOA. The FCS could be used in other engineering and real-time optimization fields in the future. To increase the effectiveness of other common NIOA, the proposed fuzzy population-based technique can be implemented.

## FUNDING

There is no funding associated with this research.

## CONFLICT OF INTEREST

On behalf of all authors, the corresponding author states that there is no conflict of interest. The authors declare that they have no conflict of interest.

## ETHICAL APPROVAL

This article does not contain any studies with human participants or animals performed by any of the authors.

## REFERENCES

- Abdelbar, A. M., Abdelshahid, S., & Wunsch, D. C. (2005, July). Fuzzy PSO: a generalization of particle swarm optimization. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks* (Vol. 2, pp. 1086-1091). IEEE. doi:10.1109/IJCNN.2005.1556004
- Aja-Fernández, S., San José Estépar, R., Alberola-López, C., & Westin, C. F. (2006). Image quality assessment based on local variance. *EMBC 2006*. doi:10.1109/IEMBS.2006.259516
- Alihodzic, A., & Tuba, M. (2014). Improved bat algorithm applied to multilevel image thresholding. *The Scientific World Journal*, 2014, 2014. doi:10.1155/2014/176718 PMID:25165733
- Bhandari, A. K., Kumar, A., & Singh, G. K. (2015). Tsallis entropy based multilevel thresholding for colored satellite image segmentation using evolutionary algorithms. *Expert Systems with Applications*, 42(22), 8707–8730. doi:10.1016/j.eswa.2015.07.025
- Brajevic, I., & Tuba, M. (2014). Cuckoo search and firefly algorithm applied to multilevel image thresholding. In *Cuckoo search and firefly algorithm* (pp. 115–139). Springer. doi:10.1007/978-3-319-02141-6\_6
- Chi, R., Su, Y. X., Zhang, D. H., Chi, X. X., & Zhang, H. J. (2019). A hybridization of cuckoo search and particle swarm optimization for solving optimization problems. *Neural Computing & Applications*, 31(1), 653–670. doi:10.1007/s00521-017-3012-x
- Dhal, K. G., Das, A., Gálvez, J., Ray, S., & Das, S. (2020). An Overview on Nature-Inspired Optimization Algorithms and Their Possible Application in Image Processing Domain. *Pattern Recognition and Image Analysis*, 30(4), 614–631. doi:10.1134/S1054661820040100
- Dhal, K. G., Das, A., Ray, S., & Das, S. (2019). A Clustering Based Classification Approach Based on Modified Cuckoo Search Algorithm. *Pattern Recognition and Image Analysis*, 29(3), 344–359. doi:10.1134/S1054661819030052
- Dhal, K. G., Das, A., Ray, S., Gálvez, J., & Das, S. (2020). Nature-inspired optimization algorithms and their application in multi-thresholding image segmentation. *Archives of Computational Methods in Engineering*, 27(3), 855–888. doi:10.1007/s11831-019-09334-y
- Dhal, K. G., Das, A., Ray, S., Gálvez, J., & Das, S. (2021). Histogram equalization variants as optimization problems: A review. *Archives of Computational Methods in Engineering*, 28(3), 1471–1496. doi:10.1007/s11831-020-09425-1
- Dhal, K. G., Das, A., Sahoo, S., Das, R., & Das, S. (2021). Measuring the curse of population size over swarm intelligence based algorithms. *Evolving Systems*, 12(3), 779–826. doi:10.1007/s12530-019-09318-0
- Dhal, K. G., & Das, S. (2017). Cuckoo search with search strategies and proper objective function for brightness preserving image enhancement. *Pattern Recognition and Image Analysis*, 27(4), 695–712. doi:10.1134/S1054661817040046
- Dhal, K. G., Gálvez, J., & Das, S. (2020). Toward the modification of flower pollination algorithm in clustering-based image segmentation. *Neural Computing & Applications*, 32(8), 3059–3077. doi:10.1007/s00521-019-04585-z
- Dhal, K. G., Gálvez, J., Ray, S., Das, A., & Das, S. (2020). Acute lymphoblastic leukemia image segmentation driven by stochastic fractal search. *Multimedia Tools and Applications*, 79(17-18), 12227–12255. doi:10.1007/s11042-019-08417-z
- Dhal, K. G., Quraishi, M. I., & Das, S. (2017). An improved cuckoo search based optimal ranged brightness preserved histogram equalization and contrast stretching method. *International Journal of Swarm Intelligence Research*, 8(1), 1–29. doi:10.4018/IJSIR.2017010101
- Dhal, K. G., Ray, S., Das, A., & Das, S. (2019). A survey on nature-inspired optimization algorithms and their application in image enhancement domain. *Archives of Computational Methods in Engineering*, 26(5), 1607–1638. doi:10.1007/s11831-018-9289-9

- Dhal, K. G., Ray, S., Das, S., Biswas, A., & Ghosh, S. (2019). Hue-Preserving and Gamut Problem-Free Histopathology Image Enhancement. *Iranian Journal of Science and Technology. Transaction of Electrical Engineering*, 43(3), 645–672. doi:10.1007/s40998-019-00175-w
- Dhal, K. G., Sen, M., & Das, S. (2018). Multi-Thresholding of Histopathological Images Using Fuzzy Entropy and Parameterless Cuckoo Search. In *Critical Developments and Applications of Swarm Intelligence* (pp. 339–356). IGI Global. doi:10.4018/978-1-5225-5134-8.ch013
- Duan, L., Yang, S., & Zhang, D. (2021). Multilevel thresholding using an improved cuckoo search algorithm for image segmentation. *The Journal of Supercomputing*, 77(7), 1–20. doi:10.1007/s11227-020-03566-7
- Erol, O. K., & Eksin, I. (2006). A new optimization method: Big bang–big crunch. *Advances in Engineering Software*, 37(2), 106–111. doi:10.1016/j.advengsoft.2005.04.005
- García, S., Molina, D., Lozano, M., & Herrera, F. (2009). A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization. *Journal of Heuristics*, 15(6), 617–644. doi:10.1007/s10732-008-9080-4
- Gaxiola, F., Melin, P., Valdez, F., Castro, J. R., & Manzo-Martínez, A. (2019). PSO with dynamic adaptation of parameters for optimization in neural networks with interval type-2 fuzzy numbers weights. *Axioms*, 8(1), 14. doi:10.3390/axioms8010014
- Guerrero, M., Castillo, O., & García, M. (2015, May). Fuzzy dynamic parameters adaptation in the Cuckoo Search Algorithm using fuzzy logic. In *2015 IEEE Congress on Evolutionary Computation (CEC)* (pp. 441-448). IEEE. doi:10.1109/CEC.2015.7256923
- Hassanzadeh, T., & Kanan, H. R. (2014). Fuzzy FA: A modified firefly algorithm. *Applied Artificial Intelligence*, 28(1), 47–65. doi:10.1080/08839514.2014.862773
- Jiao, W., Chen, W., & Zhang, J. (2021). An Improved Cuckoo Search Algorithm for Multithreshold Image Segmentation. *Security and Communication Networks*, 2021, 1–10. Advance online publication. doi:10.1155/2021/6036410
- Kalyani, R., Sathya, P. D., & Sakthivel, V. P. (2021, March). Image segmentation with Kapur, Otsu and minimum cross entropy based multilevel thresholding aided with cuckoo search algorithm. *IOP Conference Series. Materials Science and Engineering*, 1119(1), 012019. doi:10.1088/1757-899X/1119/1/012019
- Labati, R. D., Piuri, V., & Scotti, F. (2011). All-IDB: The acute lymphoblastic leukemia image database for image processing. *2011 18th IEEE International Conference on Image Processing*, 2045–2048. doi:10.1109/ICIP.2011.6115881
- Lei, T., Jia, X., Zhang, Y., He, L., Meng, H., & Nandi, A. K. (2018). Significantly fast and robust fuzzy c-means clustering algorithm based on morphological reconstruction and membership filtering. *IEEE Transactions on Fuzzy Systems*, 26(5), 3027–3041. doi:10.1109/TFUZZ.2018.2796074
- Li, S., Chen, H., Wang, M., Heidari, A. A., & Mirjalili, S. (2020). Slime mould algorithm: A new method for stochastic optimization. *Future Generation Computer Systems*, 111, 300–323. doi:10.1016/j.future.2020.03.055
- Liang, J., Qu, B., & Suganthan, P. (2014). Problem Definitions and Evaluation Criteria for the CEC. Special Session and Competition on Single Objective Real Parameter Numerical Optimization, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore.
- Manic, K. S., Biju, R., Patel, W., Khan, M. A., Raja, N., & Uma, S. (2021). Extraction and Evaluation of Corpus Callosum from 2D Brain MRI Slice: A Study with Cuckoo Search Algorithm. *Computational and Mathematical Methods in Medicine*, 2021, 1–15. Advance online publication. doi:10.1155/2021/5524637 PMID:34381523
- Mareli, M., & Twala, B. (2018). An adaptive Cuckoo search algorithm for optimisation. *Applied Computing and Informatics*, 14(2), 107-115.
- Mejía-de-Dios, J. A., & Mezura-Montes, E. (2019). A new evolutionary optimization method based on center of mass. In *Decision Science in Action* (pp. 65–74). Springer. doi:10.1007/978-981-13-0860-4\_6
- Mlakar, U., Fister, I. Jr, & Fister, I. (2016). Hybrid self-adaptive cuckoo search for global optimization. *Swarm and Evolutionary Computation*, 29, 47–72. doi:10.1016/j.swevo.2016.03.001

Munoz-Minjares, J., Vite-Chavez, O., Flores-Troncoso, J., & Cruz-Duarte, J. M. (2021). Alternative Thresholding Technique for Image Segmentation Based on Cuckoo Search and Generalized Gaussians. *Mathematics*, 9(18), 2287. doi:10.3390/math9182287

Nobile, M. S., Cazzaniga, P., Besozzi, D., Colombo, R., Mauri, G., & Pasi, G. (2018). Fuzzy Self-Tuning PSO: A settings-free algorithm for global optimization. *Swarm and Evolutionary Computation*, 39, 70–85. doi:10.1016/j.swevo.2017.09.001

Pérez, J., Valdez, F., & Castillo, O. (2015a, May). Modification of the bat algorithm using fuzzy logic for dynamical parameter adaptation. In *2015 IEEE Congress on Evolutionary Computation (CEC)* (pp. 464–471). IEEE. doi:10.1109/CEC.2015.7256926

Pérez, J., Valdez, F., & Castillo, O. (2015b). A new bat algorithm with fuzzy logic for dynamical parameter adaptation and its applicability to fuzzy control design. In *Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics* (pp. 65–79). Springer. doi:10.1007/978-3-319-10960-2\_4

Rahaman, J., & Sing, M. (2021). An efficient multilevel thresholding based satellite image segmentation approach using a new adaptive cuckoo search algorithm. *Expert Systems with Applications*, 174, 114633. doi:10.1016/j.eswa.2021.114633

Suresh, S., & Lal, S. (2016). An efficient cuckoo search algorithm based multilevel thresholding for segmentation of satellite images using different objective functions. *Expert Systems with Applications*, 58, 184–209. doi:10.1016/j.eswa.2016.03.032

Suresh, S., & Lal, S. (2017). Multilevel thresholding based on Chaotic Darwinian Particle Swarm Optimization for segmentation of satellite images. *Applied Soft Computing*, 55, 503–522. doi:10.1016/j.asoc.2017.02.005

Tan, Z., Li, K., & Wang, Y. (2021). An improved cuckoo search algorithm for multilevel color image thresholding based on modified fuzzy entropy. *Journal of Ambient Intelligence and Humanized Computing*, 1–14. doi:10.1007/s12652-021-03001-6

Tuba, E., Alihodzic, A., & Tuba, M. (2017, June). Multilevel image thresholding using elephant herding optimization algorithm. In *Engineering of Modern Electric Systems (EMES), 2017 14th International Conference on* (pp. 240–243). IEEE. doi:10.1109/EMES.2017.7980424

Yang, X. S., & Deb, S. (2009, December). Cuckoo search via Lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* (pp. 210–214). IEEE.

Zhang, L., Zhang, L., Mou, X., & Zhang, D. (2011). FSIM: A feature similarity index for image quality assessment. *IEEE Transactions on Image Processing*, 20(8), 2378–2386. doi:10.1109/TIP.2011.2109730 PMID:21292594

Zhang, Z., Ding, S., & Jia, W. (2019). A hybrid optimization algorithm based on cuckoo search and differential evolution for solving constrained engineering problems. *Engineering Applications of Artificial Intelligence*, 85, 254–268. doi:10.1016/j.engappai.2019.06.017

Zimmermann, H. J. (2011). *Fuzzy set theory—and its applications*. Springer Science & Business Media.

*Krishna Gopal Dhal completed his B.Tech and M. Tech from Kalyani Government Engineering College, India. He received his Ph.D. in Engineering from University of Kalyani, West Bengal, India. Currently, he is working as Assistant Professor in the Department of Computer Science and Application, Midnapore College (Autonomous). His research interests are digital image Processing, Medical Imaging, and Nature inspired Optimisation Algorithms.*

*Arunita Das received her M.Tech in Information Technology from Kalyani Government Engineering College (under M.A.K.A.U.T), West Bengal, India. She also completed her B.Sc. and M.Sc. in Computer Science from Vidyasagar University, Paschim Medinipur, West Bengal, India. She is the recipient of the University Gold Medal and Silver Medals (two times) for achieving first position in M.Tech and second position in B.Sc. and M.Sc courses respectively. Currently, she is working as faculty member in the department of Computer Sc. & Application in Midnapore College (Autonomous), Paschim Medinipur, West Bengal, India. Her research interests are Medical Image processing and Nature-Inspired Optimization Algorithms.*

*Jorge Jesus Galvez Rodriguez obtained his Ph.D. degree from the University of Guadalajara, Mexico. He currently is an associate professor at the University of Guadalajara. His research interests include evolutionary computation, computer vision, computer graphic, virtual reality, and robotics.*