ROA-CS: A Hybrid Algorithm for Image Brightness Enhancement

Jeng-Shyang Pan[1,2], Yingying Wang[1], Shu-Chuan Chu[1], Ling Li[1], Ning Zhong[3,4]*

[1] *College of Computer Science and Engineering, Shandong University of Science and Technology, China*
[2] *Department of Information Management, Chaoyang University of Technology, Taiwan*
[3] *Department of Life Science and Informatics, Maebashi Institute of Technology, Japan*
[4] *Faculty of Information Technology, Beijing University of Technology, China*
jengshyangpan@gmail.com, 202282060031@sdust.edu.cn, scchu0803@sdust.edu.cn, moureling@foxmail.com, zhong@maebashi-it.ac.jp

Abstract

Meta-heuristic optimization algorithms have been commonly applied to solve image-related problems for decades. In this paper, a hybrid algorithm of Rafflesia optimization algorithm and cuckoo optimization algorithm (ROA-CS) is proposed. Since ROA has a powerful exploration capability and CS has a powerful exploitation capability. With the advantages of two algorithms, the ROA-CS algorithm is proposed. The two algorithms use Taguchi theory as the communication strategy, and the performance of the algorithms is further improved. The proposed ROA-CS algorithm is tested on 30 benchmark functions of CEC2017, and the improved algorithm outperforms the current state-of-the-art 9 algorithms. Finally, the ROA-CS algorithm is applied to image brightness enhancement to maximize the fitness function by outputting a set of optimal parameters to improve the image's entropy, standard deviation, and edge details. The experimental outcomes show that the algorithm is effective for enhancing image brightness. The proposed ROA-CS algorithm outperforms other image enhancement methods to a large extent in terms of visual effectiveness and quantitative image quality assessment.

Keywords: Rafflesia optimization algorithm, Intelligent evolutionary algorithm, Image enhancement, CEC2017, Cuckoo search algorithm

1 Introduction

As science and technology have rapidly advanced, human society has consequently been propelled forward and made significant progress. At the same time, the problems people face are becoming more and more complex. Examples include path planning in large cities and high-precision instruments in the medical field. The increasing scale of the problems and the demand for efficiency have driven researchers to search for more efficient and optimal techniques [1]. One of the methods that researchers have found is the meta-heuristic algorithm. In the last few years, meta-heuristics have developed rapidly and many efficient meta-heuristics have been proposed, and many meta-heuristics have been applied within the domain of image processing. The core idea of meta-heuristic algorithms is to explore and then optimize them inspired by certain phenomena and laws of living things in nature. Meta-heuristic algorithms show improved self-adaptation and learning capabilities along with increased robustness when compared to regular optimization techniques [2]. The traditional meta-heuristic algorithms include genetic algorithm (GA) [3], particle swarm optimization (PSO) [4-5], differential evolution algorithm (DE) [6], cuckoo search algorithm (CS), and artificial bee colony algorithm (ABC) [7]. Among them, Eberhart and Kennedy co-proposed the PSO algorithm in 1995. It is an evolutionary system that mimics the feeding habits of a congregation of birds. The early explorations conducted by researchers established a strong theoretical basis for the development of later research and provided directions for improvement [8-10].

Many novel optimization algorithms have been proposed by academics in response to the complicated optimization challenges that have been appearing across a range of businesses in recent years. With the goal of enhancing problem solving accuracy and efficiency and resolving challenging problems in real-world applications, these algorithms will help the optimization field grow quickly [11-13]. The cat swarm algorithm (CSO), a global optimization method inspired by cat behavior, was first proposed by Chu et al. [14] in 2006. After this, Tsai et al. [15] used parallel strategies to optimize the cat swarm algorithm and applied it to the aircraft planning flight problem. Tsai et al. [16] also mixed the artificial bee swarm algorithm with the cat swarm algorithm. The QUATRE algorithm was utilized by Du et al. [17-18] to address the communication issue in wireless sensor networks. In 2021, The Phasmatodea population evolution algorithm was first presented by Song et al. [19], which emulates path dependency, convergent evolution, and the traits of population expansion, competition, and convergent evolution found in nature. Pan at al. [20] proposed gannet optimization algorithm (GA) based on gannet habits. In 2022, Fu et al. [21] proposed the Rafflesia optimization algorithm (ROA). ROA is an innovative optimization algorithm that emulates the natural growth traits of plants. At the same time, ROA algorithm was applied to

*Corresponding Author: Shu-Chuan Chu; Email: scchu0803@sdust.edu.cn
DOI: https://doi.org/10.70003/160792642026092705001

the logistics distribution center site selection problem to prove the proposed new algorithm. The effectiveness of the proposed new algorithm is demonstrated. The CS [22] algorithm is a representative meta-heuristic algorithm proposed by Yang et al. in 2009. CS algorithm introduces a levy flight search mechanism by mimicking the parasitic brood rearing behavior of cuckoos in nature, and imitates the flight behavior of birds in nature to perform the merit search operation. The original ROA performs well in terms of exploitation, but falls short in terms of exploration capabilities. To improve the performance of the original ROA, its overall performance can be enhanced by fusing it with other algorithms. Since the traditional CS algorithm has a strong global exploration capability, a hybrid of this algorithm and the ROA algorithm is used. The hybrid strategy uses Taguchi theory [23]. A significant area of image processing is image enhancement [24]. Image enhancement is the processing of the parts of interest in an image according to certain requirements or highlighting the useful features in an image [25]. The uninteresting parts are suppressed to improve the subjective visual effect and image quality, thus increasing the interpretability or perceptibility of the information present in the image [26]. The primary focus of image enhancement is on applications where the improved image will produce a more striking visual impression than the original [27]. Depending on the space where the enhancement process is located, the two primary types of image improvement techniques are air-domain enhancement techniques and frequency domain enhancement techniques [28-29]. In recent years, meta-heuristic algorithms have drawn increasing attention from researchers [30-31].

Histogram equalization (HE) and linear contrast stretching (LCS) are two popular picture enhancing techniques, but images obtained by HE and LCS usually have excessive contrast enhancement. GA is one of the first meta-heuristics applied in the field of image enhancement. In order to assess the fitness of potential solutions, Saitoh [32] suggested using GA to measure the strength of spatial edges seen in an image. Compared to traditional methods, he increased the processing time but was able to obtain better results. In 2007, Braik et al. [33] used PSO to adjust a set of parameters by maximizing the objective function, making the details in the image enhanced. Compared to the GA-based method, it preserves the edge details while simplifying the computation. Each algorithm has its characteristics and mixing two or three algorithms through some strategy can combine the advantages of the algorithms. In 2015, Mahapatra et al. [34] proposed a hybrid image enhancement algorithm based on particle swarm optimization and artificial immune system algorithm which is achieved by enhancing the gray level of the image. In 2022, Khan et al. [27] combined the β-hill climbing technique and political optimizer together and hybridized these two algorithms to find the optimal pixel values to optimize the image.

Although many meta-heuristic algorithms have been applied in the field of image enhancement, there are problems such as poor visual effect, complicated calculation, and poor image evaluation index. The new image brightness enhancement algorithm for ROA-CS presented in this paper can achieve better results in terms of visual effects and image assessment indices when compared to prior approaches. It enriches the development of meta-heuristic algorithms in the field of image enhancement. The ROA algorithm and CS algorithm are hybridized by Taguchi theory and applied to image enhancement to verify the performance of the algorithm. In the application, the hybrid algorithm enhances the information entropy, edge information and standard deviation of the image by optimizing the four optimal parameters in the solution, which in turn maximizes the fitness function. The following is a summary of this paper's main contributions:

1. A Hybrid Rafflesia optimization algorithm is proposed.
2. Based on the Taguchi theory, a novel method combining the Cuckoo search algorithm (CS) with the Rafflesia optimization algorithm was proposed in this.
3. ROA-CS is applied to low brightness image enhancement. ROA-CS algorithm outperforms other image enhancement methods to a large extent in terms of visual effectiveness and quantitative image quality assessment.

The rest of the paper is organized as follows: Section 2 presents the principles of the original ROA and CS algorithms. Section 3 introduces the orthogonal strategy and the proposed ROA-CS algorithm. Section 4 compares the proposed ROA-CS algorithm with other optimization algorithms on CEC2013 and CEC2017 and shows the performance of these algorithms on test functions. Section 5 gives the results of applying the proposed ROA-CS algorithm to image enhancement and compares it with other algorithms. At last, a summary is provided in Section 6.

2 Related Works

2.1 The Original Rafflesia Optimization Algorithm

There are three stages to the Rafflesia optimization algorithm: the insect attraction phase, the insect swallowing phase, and the seed sowing phase.

In the insect attraction phase, Rafflesia relies on its emission of a putrid odor to attract odor-seeking insects such as flies. The ROA algorithm finds the optimal solution by local search, using a combination of autonomous flight and approximation of the optimal individual to update the position of the individual. In this phase, ROA algorithm is divided into two strategies. Strategy I is used to deal with poorly adapted insect individuals, which are replaced by additional individuals. The number of individuals is 1/3 of the overall population size. The positions of the newly added individuals are updated using Equation (1).

$$X_{\{i_k\}} = X_{\{best_k\}} + d \cdot \sin B_k \cos R_k \quad (1)$$

where X_{best} represents the optimal individual in the population. X_i denotes the individual newly added to the population. k denotes the dimension in which it is located (k = 1, 2, ..., D).

The angle between $\overrightarrow{X_{best}X_i}$ and the z-axis is denoted as B_k, and its values range from 0 to π / 2. R_k represents the angle between the projection of $\overrightarrow{X_{best}X_i}$ on the plane made up of the dimensions of the x, y axis, and range of values: 0 to π. d represents the distance between X_i and X_{best} and can be calculated by Equation (2).

$$d = \sqrt{\sum_{k=1}^{D}\left(X_{r_k} - X_{bestk}\right)^2} \tag{2}$$

A random member of the population is denoted by X_r. Individuals with poor fitness values were replaced by newly added individuals using Equation (3).

$$X_{worst_i} = X_i \tag{3}$$

Strategy II is used to update the remaining 2/3 of better adapted individuals that represent insects that continue to fly towards the Rafflesia. Their positions were updated according to a formula related to the model of a natural world insect's wing movements. The motion of an insect when flapping its wings can be decomposed into translational and rotational, and the speed of individual insect motion is equal to the superposition of translational and rotational speeds, using $\overrightarrow{v_1}$ and $\overrightarrow{v_2}$ to represent translational and rotational speeds. The translational and rotational velocities are obtained from Equation (4) and Equation (5).

$$\underset{v_1}{\rightarrow} = \frac{\omega_0}{2}\sqrt{A^2 sin^2\left(\omega_0 t + \theta\right) + B^2 cos^2\left(\omega_1 t + \theta\right)\}} \tag{4}$$

where, the wing's motion amplitude is denoted by A, and the lateral offset is represented by B. Periodic flutter frequency is indicated by ω_0. For the side flutter wing, its frequency period is ω_1. Phase is indicated by θ. t is time.

$$\underset{v_2}{\rightarrow} = \underset{v_2}{\rightarrow}\omega_0 \cos\left(\omega_0 t + \theta + \varphi\right) \tag{5}$$

The translation and rotation phase difference is represented by φ. To sum up, the individual's velocity equation is:

$$\underset{v}{\rightarrow} = \underset{v_1}{\rightarrow} + \underset{v_2}{\rightarrow} \tag{6}$$

The movement of individuals in the population is not only related to their own movement speed, but also influenced by the best individual. For every individual, the distance update equation is:

$$\underset{le}{\rightarrow} = C \cdot \underset{v}{\rightarrow} \cdot t + \left(X_{best} - X\left(t\right)\right) \cdot \left(1 - C\right) \cdot rand \tag{7}$$

C is a random number, taking values in the range (-1, 1). In summary, the equation for individual renewal is:

$$X\left(t\right) = X\left(t\right) + \underset{le}{\rightarrow} \tag{8}$$

The second stage is swallowing insect. After Rafflesia attracts small insects, 'swallowing' of insects by Rafflesia occurs throughout the growth phase due to its unique and special floral chamber structure. The original ROA algorithm is set to reduce the number of individuals with the worst fitness value every certain number of iterations to reduce the impact of unsatisfactory individuals in the population. This is used to simulate the phenomenon of Rafflesia swallowing insects in nature.

Seed propagation stage. Avoid getting stuck in the local optimum to perform a global search to leap from the local optimum. In this stage, the update equation of the individual is represented using Equation (9):

$$X\left(t\right)_k = X_{best_k} + rd \cdot \exp\left(\frac{iter}{Max\_iter} - 1\right) \cdot sign \\ \left(rand - 0.5\right) \tag{9}$$

where k(k = 1, 2, …, D) denotes dimension. The current iteration count is denoted by $iter$, the maximum number of iterations is Max_iter, the distribution range of individuals, denoted as rd, is acquired through Equation (10).

$$rd = rand \cdot \left(ub - lb\right) + lb \tag{10}$$

The random number $rand$ ranges from 0 to 1. ub denotes the upper boundary, while lb represents the lower boundary.

2.2 The Original Cuckoo Search Algorithm

The Cuckoo search algorithm is durable, quick to implement, and has few parameters, making it a powerful development tool. And its convergence speed is slow.

Cuckoos are a special kind of birds in nature that deposit their offspring within other birds' nests. The CS algorithm mainly simulates the incubating egg parasitism of cuckoos while using levy flight to improve the ability of random wandering. CS algorithm requires the assumption of three ideal conditions: ss (1) a cuckoo lays one egg at a time and selects a nest at random to incubate it; (2) the best nest from the collection of nests selected at random is kept for the next generation; (3) the number of parasitic nests that can be selected is fixed and the likelihood that a cuckoo egg will be discovered by its host is p [35], with p taking values in the range (0, 1). If based on the above three principles in the ideal, the formula for updating the location and path of cuckoo nest finding is expressed using Equation (11).

$$X_i^{iter+1} = X_i^{iter} + a \cdot L(s, \lambda) \quad \textbf{(11)}$$

where X_i^{iter} denotes the nest position of the ith nest in the iter generation. *iter* denotes the number of current iterations, taking values from 1 to the maximum number of iterations. a denotes the step scale factor, $a > 0$. $L(s,\lambda)$ denotes the Levy flight, where λ is a random vector conforming to the standard normal distribution.

The next-generation nest location is updated and the fitness value of the objective function is calculated. If the fitness value becomes better after update, the original position is replaced, otherwise the original nest position is retained [36]. After the location update, the value *ra* between 0 and 1 is compared with the probability p of being discovered by randomly generated values. If *ra* is greater than p, it is discovered and a random update of the bird nest's position is required. Finally, the set of bird nest locations with the best fitness value is retained.

3 The Hybrid Algorithm

This section introduces the improved rafflesia optimization algorithm, and describes the Taguchi theory. Taguchi theory combines ROA algorithm with CS algorithm.

3.1 Communication Strategy: Taguchi Method

Taguchi method is a low cost, high yield quality engineering method proposed by Dr. Genichi Taguchi [37]. This method is commonly employed in industry. The features of Taguchi method are: it can make the number of experiments reduced, it is sufficient to use a little quantity of information for analysis in a large experimental data; it saves experimental cost; and it can make the experimental efficiency increased [38].

Taguchi method can find useful information using fewer combinations of experiments. Compared with the full factorial method, it is not able to find a very accurate optimal position, but it can use a small number of experiments to find the optimal trend and has better feasibility. Orthogonal array is one of the crucial ways in Taguchi method. The orthogonal matrix is represented using $L_M(Q^N)$. Where Q denotes the number of levels; M denotes the number of experimental combinations; N represents the number of factors [39], and M and N are positive integers. In this paper, a two-level orthogonal array is used, so the value of Q is 2, there are only two possible values of levels. When the value of Q is 2, the value of M minus the value of N equals 1. At this point, the total number of experiments using the full factorial design method will be 2^N, but the number of experiments using the orthogonal matrix will only be M times.

The factor levels are given equal weights throughout the design. Each row denotes a distinct set of level combinations, while each column represents the values of the factors taken. An example of an orthogonal array $L_8(2^7)$ is shown below:

$$L_8\left(2^7\right) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 & 2 & 2 & 2 \\ 1 & 2 & 2 & 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 2 & 1 & 2 & 2 & 1 & 2 & 1 \\ 2 & 2 & 1 & 1 & 2 & 2 & 1 \\ 2 & 2 & 1 & 2 & 1 & 1 & 2 \end{bmatrix}$$

There are levels 1 and 2. When the value is 1 and 2, choose different algorithms respectively.

3.2 The Proposed Algorithm (ROA-CS)

This section mixes the ROA algorithm with the CS algorithm using the orthogonal strategy from Taguchi theory. The following are the steps of the improved algorithm.

Step 1: Define and set the parameters of the ROA algorithm and CS algorithm.

Step 2: Initialize the population.

Step 3: Divide the population equally into two sub-populations. One population is updated with individuals using the ROA algorithm and the other population is updated with individuals using the CS algorithm. The two algorithms are executed simultaneously and the best adapted individuals are updated separately.

Step 4: Set up the communication between the two algorithms using the orthogonal strategy in Taguchi theory after every 10 iterations to further update the best individual. If updated individual fitness is greater than original individual, the original individual is replaced.

Step 5: The maximum number of iterations is achieved, and the best individual is found.

Some meta-heuristic algorithm experts have recently conducted extensive study on the guiding and update mechanisms, leading them to propose a number of novel approaches with considerable development potential: including fitness-distance balancing (FDB) [40], fitness-distance-constraint (FDC) [41], and Natural Survivor Method (NSM) [42]. The creation of meta-heuristic algorithms now has additional options thanks to these creative methods. Through enhancing the search performance and adaptability of the algorithms, they hope to further advance the area. Taguchi method is used to make ROA algorithm and CS algorithm run in parallel. In the update mechanism, the corresponding values in the orthogonal arrays are made to choose the greatest individual in ROA algorithm and CS algorithm to update the individuals, respectively.

In the orthogonal arrays, orthogonal array value of 1 means that the best position found by the ROA algorithm is selected to update its position, and an orthogonal array value of 2 means that the best position found by the CS algorithm is selected to update its position. In

ROA-CS, the orthogonal strategy provides a method of communication between the two parallel populations, allowing different combinations of the best individuals from each subpopulation in each dimension. The flowchart of ROA-CS is shown in Figure 1.

Algorithm 1 is the proposed algorithm's pseudo code. In the pseudo code, t is the number of the current iteration and *or* denotes the orthogonal array. The first 3 lines indicate the initialization phase of the algorithm; lines 5-9 indicate that population pop1 uses the ROA algorithm to update and obtain the optimal fitness value f1 and the optimal position pos1, and population pop2 uses the CS algorithm to update and obtain the optimal fitness value f2 and the optimal position pos2; the next 10 iterations are a cycle; lines 11-15 indicate that when the value of orthogonal array is 1, the optimal position is updated using the When the orthogonal array value is 1, the ROA algorithm is used to update the optimal position bestpos; when the orthogonal array value is 2, the CS algorithm is used to update the optimal position bestpos; and then Calculate the fitness value for the individual and update if the fitness value becomes better. Lines 17-25 indicate that the optimal fitness value fbest is the minimum of f1 and f2, and when fbest is less than f1, pos1 and f1 are equal to the values of bestpos and fbest, and the same is true when fbest is less than f2. The algorithm iterates in this way to obtain the optimal value.

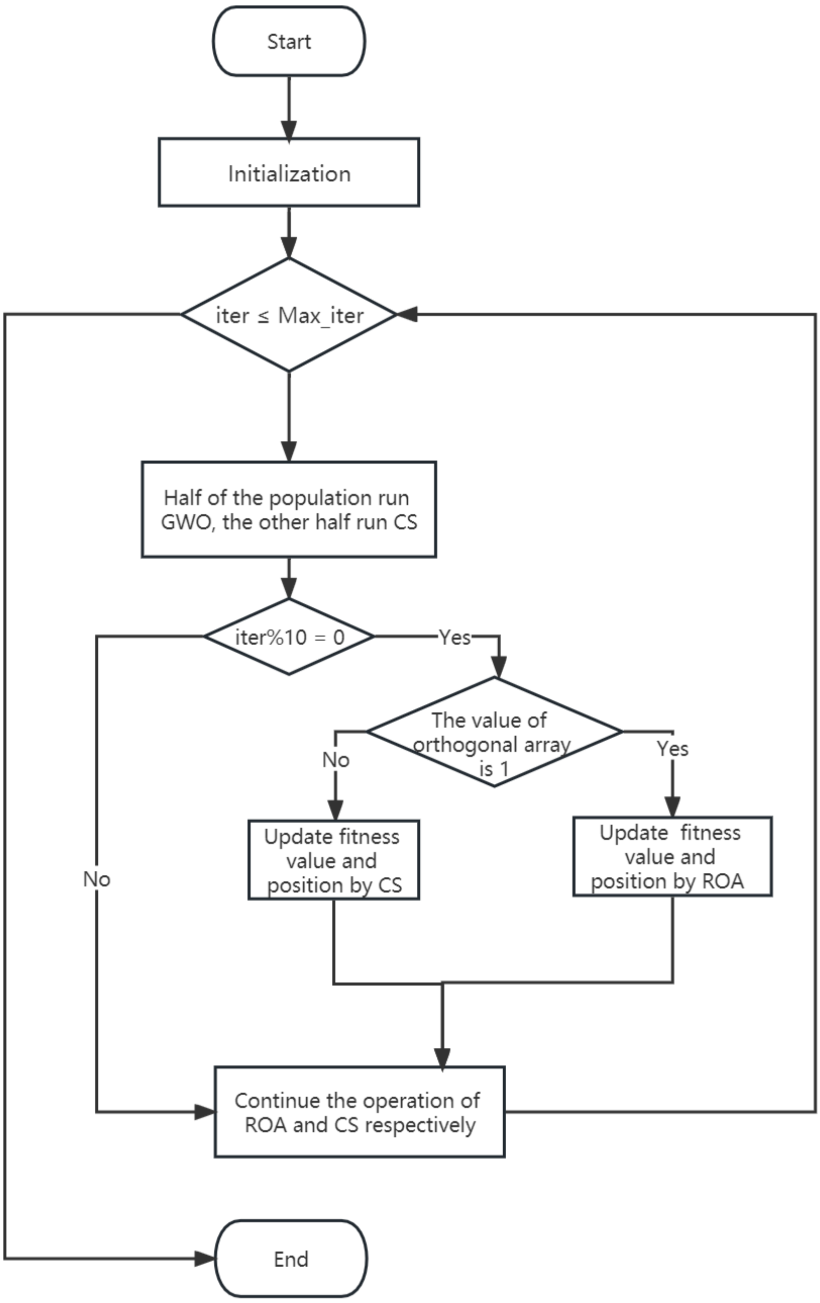


Figure 1. The flowchart of ROA-CS

4 Experimental Results and Analysis

In this section, the improved algorithm is tested on benchmark functions in CEC2017. And the results of all algorithms were compared. All experiments were conducted on the same experimental platform. Experiments were performed using a computer installed with Windows 10 (64-bit), Intel (R) Core (TM) i5-8265U and MATLAB R2019b with 16.0 GB RAM.

4.1 Statistical Results

To guarantee the experiment's fairness, the initial population size of both the improved algorithm and the compared algorithm were set to 30. Each algorithm was run 30 times independently. The average and standard deviation of statistics are employed as two metrics to assess the algorithms' results. The maximum number of fitness function evaluations was set to 10000*Dimension and was used as a criterion for the termination of the search-process life cycle of the algorithms. This allowed for a fair comparison between the algorithms. Friedman tests were used for statistical analysis.

The algorithms compared with ROA-CS in the experiments are described as follows. Adaptive Fitness-Distance Balance based Artificial Rabbits Optimization (AFDB-ARO) Algorithm [43], introduced by Ozkaya at al. in 2024, represents an efficient optimization approach that significantly bolsters the exploratory capacity of the ARO algorithm during the search process through the employment of an AFDB guidance mechanism. Fitness-Distance Balance Artificial Ecosystem Optimization (FDB-AEO) [44] is a new optimisation method proposed by Sonmez et al. in 2022, it combines the Fitness-Distance Balance strategy with the Artificial Ecosystem Optimization (AEO) algorithm. ROA is the original Rafflesia optimization algorithm. Whale optimization algorithm (WOA) [45], an algorithm proposed by Seyedali Mirjalili in 2016 based on whale roundup prey, which simulates the social behavior of whales and introduces a bubble net hunting strategy. PSO, an algorithm proposed in 1995 by Eberhart et al. that simulates the predatory behavior of bird flocks. Moth-Flame optimization (MFO) [46], a meta-heuristic algorithm proposed by Seyedali Mirjalili inspired by the navigation methods of moths in nature; and Bat Algorithm (BA) [47], proposed by Yang et al. in 2010, simulates the behavior of bats in nature using ultrasonic waves to locate obstacles and find prey; Firefly algorithm (FA) [48], a meta-heuristic algorithm proposed by Yang et al., inspired by the flickering behavior of fireflies. Butterfly optimization algorithm (BOA) [49], proposed by Arora et al. and inspired by the mating and foraging behavior of butterflies. The parameter settings for these algorithms used for comparison are shown in the Table 1.

Algorithm 1. The pseudo code of ROA-CS

Input: N: population size; $iter_{\max}$: number of iterations; Dim: population dimension; or: Orthogonal array;
Output: The global optimal value.
Initialize parameters: A, B, ω_0, ω_1, θ, φ, C, $time$;
Populations are divided into two subpopulations, pop1 and pop2.
Initialize the position of each individual.
for $t = 1: iter_{\max}$ **do**
    Calculate and sort the fitness value of each individual.
    Use ROA Algorithm to update pop1.
        Get the best fitness value f1, the best position pos1.
    Use CS Algorithm to update pop2.
        Get the best fitness value f2, the best position pos2.
    if $rem(t, 10) == 0$ **then**
        if $or == 1$ **then**
            Update bestpos by ROA algorithm.
        else
            Update bestpos by CS algorithm.
        end if
        Calculate the fitness value for the individual and update if the fitness value becomes better.
        if $fbest < f1$ **then**
            $pos1 = bestpos$
            $f1 = fbest$
        end if
        if $fbest < f2$ **then**
            $pos2 = bestpos$
            $f2 = fbest$
        end if
        $fbest = \min(f1, f2)$;
    end if
end for

Table 2 shows the Friedman scores of the all compared algorithms. Friedman test serves as an effective comparison tool to reveal significant differences between algorithms. The ROA-CS algorithm exhibits an excellent third place in the overall effectiveness ranking under the consideration of dimensions 30 and 50, as well as the CEC2013 and CEC2017 criteria. The overall ranking of the 9 algorithms is shown at the end of Table 2. FDB-AEO algorithm is ranked first and AFDB-ARO is ranked second, both adopted the Fitness-Distance Balance (FDB) strategy. FDB takes into account both the fitness of a solution and the relative distance of that solution to other solutions in the search space to improve the algorithm's search performance and avoid premature convergence problems. In contrast, the original ROA, WOA, and PSO algorithms were evaluated in the mid-range. The MFO, BA, BOA and FA algorithms, on the other hand, perform poorly in this evaluation and rank low. Based on the Friedman score, we selected the top five ranked algorithms as the subsequent in-depth study for further validation and optimization.

In Table 2, the ROA-CS algorithm significantly outperforms the comparative state-of-the-art algorithms in dimension 30, CEC2017 benchmarking, and ranks first in Friedman scores. Therefore ROA-CS algorithm is considered as an efficient optimization algorithm and has much room for improvement.

Tables 3 present the experimental results of the algorithm in 30-dimensional spaces. The optimal values for each function are indicated in bold numbers. Mean represents average value of the optimal fitness scores attained. Std is the standard deviation associated with these values. The last row in the table records the number of times ROA-CS outperforms the other algorithms. Based on the information presented in the table, ROA-CS completely wins over WOA and PSO on 30 benchmark functions. In the single-peak test functions, the ROA-CS algorithm outperforms all the other algorithms except the AFDB-AEO and FDB-AEO algorithm. In the multi-peak and combined test functions, the FDB-AEO algorithm performs better. However, the performance is much poorer compared to the ROA-CS algorithm. The experimental results demonstrate that the ROA-CS algorithm exhibits the highest level of stability among the tested methods. Among the compared algorithms, WOA and PSO achieve poorer stability. Compared with FDB-AEO, although ROA-CS wins, the advantage is not obvious. Compared with the rest of the algorithms, ROA-CS shows a greater advantage in stability. From the data in Tables 3, it is evident that the ROA-CS algorithm has exhibited outstanding performance on problems with 30 dimensions.

Table 1. Parameter settings for related algorithms

Algorithms	Parameter settings
AFDB-ARO	Population size =30
FDB-AEO	Population size =30
PSO	c1=2, c2=2, Vmax=100, Vmin=-100, w=0.9
WOA	a: linearly decreases from 2 to 0, a2: linearly dicreases from -1 to -2, b=1
CSO	mr =0.4, Seek.srd=0.9, Trace.c=1.05
MFO	b=1
BA	r0 = 0.7, Af = 0.9, Rf = 0.9, Qmin = 0, Qmax = 1
FA	alpha=1.0, beta0=1.0, gamma=0.01, theta=0.97
BOA	p=0.8, power_exponent=0.1, sensory_modality=0.01

Table 2. Friedman scores of the all compared algorithms

Algorithms	Dimension=30		Dimension=50		Mean
	CECC2013	CEC2017	CEC2013	CEC2017	Rank
FDB-AEO	2.0071	2.1333	2.1429	2.0067	2.0725
AFDB-ARO	2.0357	2.1333	2.6143	2.5000	2.3208
ROA_CS	2.4214	2.0667	2.5357	2.3600	2.3460
ROA	5.7500	5.9667	5.9286	5.2667	5.7280
PSO	5.6429	5.8667	5.9643	6.0667	5.8851
WOA	5.7500	6.2000	6.2857	5.8667	6.0256
MFO	6.3214	5.9333	6.6214	6.1667	6.2607
BA	6.8214	5.9667	6.8714	5.8667	6.3816
BOA	8.3571	8.9000	6.0357	9.0000	8.0732
FA	9.8929	9.8333	10.0000	9.9000	9.9066

Table 3. Experimental results of ROA-CS and 16 comparative algorithms on CEC2017 (Dim=30)

Fun.		AFDB_ARO	FDB-AEO	ROA-CS	ROA	WOA	PSO
f1	mean	**1.00E+02**	4.69E+03	3.54E+03	2.07E+03	2.75E+06	5.93E+09
	std	**4.52E-13**	2.45E+03	4.58E+02	2.69E+03	3.57E+06	6.01E+09
f2	mean	1.26E+03	**2.27E+02**	1.00E+10	4.82E+05	8.41E+21	9.45E+37
	std	1.81E+03	3.06E+01	**0.00E+00**	5.66E+05	1.24E+22	1.54E+38
f3	mean	3.00E+02	**3.00E+02**	3.00E+02	3.00E+02	2.27E+05	3.15E+05
	std	2.91E-05	**9.60E-07**	8.98E-04	8.74E-04	9.71E+04	1.49E+05
f4	mean	4.31E+02	**4.18E+02**	4.70E+02	4.73E+02	5.37E+02	1.38E+03
	std	4.83E+01	2.77E+01	**1.60E+01**	2.57E+01	1.79E+01	1.11E+03
f5	mean	6.25E+02	6.06E+02	**5.83E+02**	8.47E+02	7.94E+02	6.42E+02
	std	2.14E+01	2.59E+01	**1.35E+01**	1.54E+01	6.25E+01	2.76E+01
f6	mean	**6.01E+02**	6.04E+02	6.01E+02	6.69E+02	6.73E+02	6.18E+02
	std	**5.14E-02**	1.66E+00	5.29E-01	5.24E+00	9.09E+00	3.24E+00
f7	mean	9.00E+02	8.71E+02	**8.55E+02**	9.94E+02	1.25E+03	1.07E+03
	std	3.95E+01	1.38E+01	**5.46E+00**	9.64E+01	1.04E+02	2.91E+02
f8	mean	9.32E+02	**8.65E+02**	8.88E+02	1.10E+03	9.80E+02	9.28E+02
	std	4.29E+01	**1.92E+01**	2.64E+01	3.86E+01	3.00E+01	3.05E+01
f9	mean	3.53E+03	**1.14E+03**	1.72E+03	1.06E+04	7.96E+03	3.31E+03
	std	6.48E+02	1.57E+02	**5.19E+01**	1.41E+03	1.87E+03	1.76E+02
f10	mean	4.02E+03	**3.48E+03**	4.94E+03	6.29E+03	6.00E+03	6.29E+03
	std	2.01E+02	**1.73E+02**	2.23E+02	8.23E+02	1.51E+03	5.53E+02
f11	mean	1.18E+03	**1.15E+03**	1.17E+03	1.31E+03	1.60E+03	1.88E+04
	std	**1.55E+01**	2.56E+01	3.53E+01	4.59E+01	2.75E+02	2.89E+04
f12	mean	4.78E+04	**1.36E+04**	2.92E+06	6.02E+05	4.28E+07	9.34E+08
	std	1.50E+04	**3.03E+03**	1.37E+06	1.55E+05	3.34E+07	1.60E+09
f13	mean	1.79E+04	**5.15E+03**	4.20E+04	7.81E+04	1.20E+05	6.96E+08
	std	1.76E+04	**3.28E+03**	3.51E+04	5.62E+03	9.89E+04	1.20E+09
f14	mean	4.58E+03	1.53E+03	**1.47E+03**	8.57E+03	3.07E+05	5.77E+05
	std	2.76E+03	**1.55E+01**	1.95E+01	4.48E+03	1.39E+05	9.49E+05
f15	mean	3.36E+03	9.69E+03	**1.59E+03**	7.02E+04	6.97E+04	3.82E+04
	std	2.76E+03	**1.55E+01**	1.95E+01	4.48E+03	1.39E+05	9.49E+05
f16	mean	2.62E+03	2.45E+03	**2.09E+03**	3.54E+03	3.57E+03	2.69E+03
	std	1.94E+02	7.73E+01	**7.16E+01**	2.98E+02	4.17E+02	1.69E+02
f17	mean	2.26E+03	1.97E+03	**1.92E+03**	2.69E+03	2.52E+03	2.23E+03
	std	1.17E+02	**7.90E+01**	9.29E+01	2.66E+02	1.25E+02	1.04E+02
f18	mean	5.06E+04	7.78E+03	**2.85E+03**	1.22E+05	1.49E+06	2.59E+06
	std	3.22E+04	5.07E+03	**6.83E+02**	2.33E+04	1.50E+06	2.75E+06
f19	mean	3.42E+03	5.94E+03	**1.93E+03**	1.37E+05	2.89E+06	4.74E+06
	std	2.26E+03	5.37E+03	**9.61E+00**	7.60E+03	3.61E+06	5.29E+06
f20	mean	2.36E+03	**2.20E+03**	2.34E+03	2.74E+03	2.90E+03	2.47E+03
	std	1.48E+02	**1.16E+02**	1.31E+02	2.87E+02	2.36E+02	2.81E+02
f21	mean	2.43E+03	**2.36E+03**	2.38E+03	2.62E+03	2.59E+03	2.52E+03
	std	4.82E+01	2.34E+01	**2.07E+01**	5.42E+01	2.25E+01	5.92E+01
f22	mean	4.40E+03	**2.30E+03**	**2.30E+03**	7.04E+03	5.57E+03	7.27E+03
	std	1.83E+03	1.47E-12	**0.00E+00**	6.13E+02	2.84E+03	9.26E+02
f23	mean	2.80E+03	2.76E+03	**2.73E+03**	2.93E+03	3.01E+03	2.98E+03
	std	2.38E+01	1.86E+01	**1.74E+01**	3.08E+01	1.44E+02	9.77E+01
f24	mean	2.98E+03	**2.91E+03**	2.93E+03	3.20E+03	3.15E+03	3.38E+03
	std	5.20E+01	**1.95E+01**	2.39E+01	6.19E+01	6.89E+01	1.47E+02
f25	mean	2.90E+03	2.90E+03	**2.89E+03**	2.94E+03	2.99E+03	3.05E+03
	std	1.32E+01	1.48E+01	**1.85E-01**	1.55E+01	1.36E+01	8.08E+01
f26	mean	5.44E+03	**4.36E+03**	4.91E+03	6.63E+03	5.32E+03	7.09E+03
	std	5.01E+02	1.41E+03	**1.08E+02**	4.73E+02	1.79E+03	6.47E+02
f27	mean	3.26E+03	3.25E+03	**3.21E+03**	3.37E+03	3.39E+03	3.31E+03
	std	1.50E+01	1.30E+01	**7.24E+00**	1.17E+02	1.08E+02	8.58E+01
f28	mean	**3.14E+03**	3.20E+03	3.14E+03	3.22E+03	3.28E+03	3.74E+03
	std	6.18E+01	9.09E+01	6.22E+01	2.18E+01	**1.12E+01**	2.90E+02
f29	mean	3.80E+03	3.57E+03	**3.56E+03**	5.29E+03	5.02E+03	4.49E+03
	std	1.16E+02	1.15E+02	**9.57E+01**	2.37E+02	4.51E+02	4.81E+02
f30	mean	8.85E+03	8.02E+03	**5.45E+03**	7.70E+05	5.94E+06	6.79E+05
	std	3.73E+03	1.85E+03	**4.41E+02**	5.40E+05	2.53E+06	4.10E+05
ROA-CS win		**21**	**16**	**-**	**27**	**30**	**30**

4.2 Convergence Analysis

In order to evaluate the performance of the ROA-CS algorithm and to make a more visual observation, we have selected the convergence curves of two unimodal functions F1, F3, two multimodal functions with simple structure F5, F6, six hybrid functions F13, F15, F16, F18, F19, F20 and two composite functions F23, F30 for demonstration. The convergence curves of the six algorithms on the twelve functions are shown in Figure 2. The convergence curves can be used to see more intuitively the difference in convergence speed between the ROA-CS algorithm and other algorithms. In the convergence curves, the horizontal axis represents the number of iterations and the maximum number of iterations. The vertical axis represents the value of the mean fitness value that the algorithm can fetch.

In the case of unimodal functions F1 and F3, the ROA-CS algorithm demonstrates suboptimal performance with slower convergence rates, where it is surpassed by AFDB-ARO. For multimodal functions F5 and F6, the initial convergence speed is relatively slow; however, the ROA-CS algorithm has the ability to break out of local optima and achieve better convergence in later stages. In multiple hybrid functions, ROA-CS consistently achieves the best mean fitness values. Regarding composite function F23, the early convergence performance of ROA-CS is not as effective as PSO, ROA but both of them converge and stop after 5000 evaluations, while ROA-CS continues to converge in search of the optimal fitness value. In the paper, the orthogonal strategy of Taguchi theory mixes ROA and CS. It enables ROA-CS to continue exploration beyond the local optimum in case of temporary stagnation of convergence.

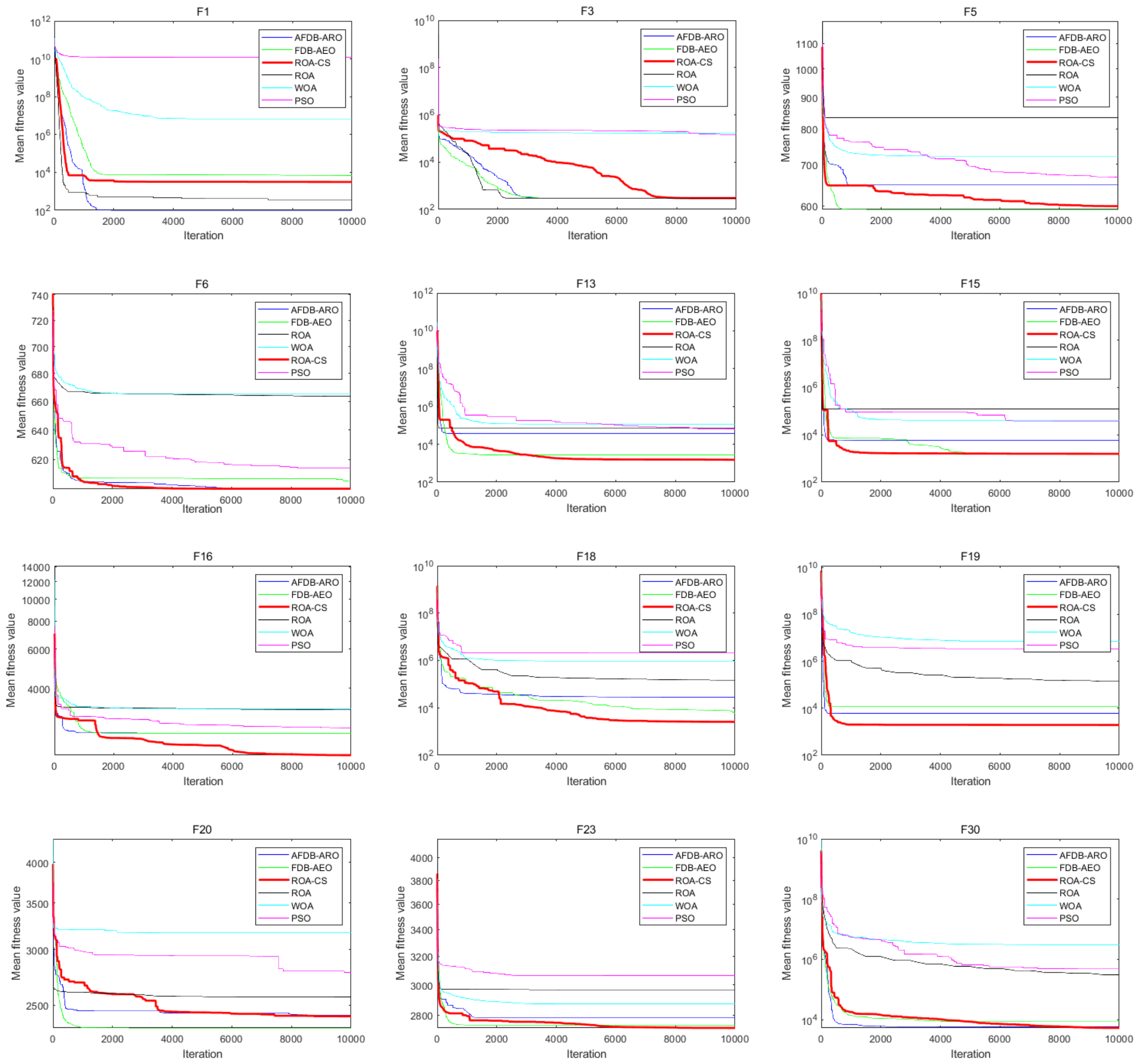


Figure 2. Convergence test results in 30 dimensions
(F1, F3, F5, F6, F13, F15, F16, F18, F19, F20 F23, F30)

4.3 Stability Analysis

This section's goal is to use benchmark test functions to illustrate and validate the ROA-CS algorithms' stability and effectiveness. Thirty functions from the CEC2017 test suite are used in this method to conduct simulation experiments. Using the 30 benchmark functions from CEC2017, this section of the study examines the computational complexity and stability of the suggested algorithms. Three evaluation metrics are presented, using the research in the literature [50-51] as a guide: success rate (SR), average number of fitness evaluations (AFEs), and average calculations durations (ACDs), defined as follows:

$$SR = \frac{n}{N} \times 100 \tag{12}$$

$$AFEs = \frac{1}{n} \sum_{i=1}^{n} FEs[i] \tag{13}$$

$$ACDs = \frac{1}{n} \sum_{i=1}^{n} Calcu\_Duration(i) \tag{14}$$

where n is the number of times a feasible solution that has been found through successful searches for the problem. The number of runs of the algorithm is N. The number of fitness evaluations and the amount of time the method takes to identify a workable solution are called FEs and Calcu_Duration, respectively.

In this study, we set feasibility solution criteria for each problem, using the mean value of each run of the ROA and ROA-CS algorithms as the benchmark, and taking the mean value of both as the solution. The feasibility solutions for the 30 test functions of CEC2017 are listed in Table 4. On this basis, we deeply analyze and compare the performance of ROA and ROA-CS on these 30 functions, including SR, AFEs and ACDs, details of which are presented in Table 5. The analysis results show that compared to the ROA algorithm, the ROA-CS algorithm has made significant progress in terms of stability, and this enhancement is not only statistically significant, but also strongly strengthens the robustness of our proposed algorithm.

Table 4. Possible solutions to 30 problems in the CEC 2017 benchmark suite

f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
2.80E+03	5.00E+09	3.00E+02	4.72E+02	7.15E+02	6.35E+02	9.25E+02	9.95E+02	6.15E+03	5.62E+03
f11	f12	f13	f14	f15	f16	f17	f18	f19	f20
1.24E+03	1.76E+06	6.01E+04	5.02E+03	3.59E+04	2.81E+03	2.30E+03	6.22E+04	6.96E+04	2.54E+03
f21	f22	f23	f24	f25	f26	f27	f28	f29	f30
2.50E+03	4.67E+03	2.83E+03	3.06E+03	2.92E+03	5.77E+03	3.29E+03	3.18E+03	4.43E+03	3.88E+05

Table 5. SR, AFEs and ACDs on ROA and ROA-CS algorithms

Fun.	SR		AFEs		ACDs (s)	
	ROA	ROA-CS	ROA	ROA-CS	ROA	ROA-CS
f1	87.73%	62.15%	1.10E+04	1.73E+04	6.43E+00	8.47E+00
f2	82.26%	62.78%	9.08E+03	1.21E+04	1.60E+00	2.32E+00
f3	70.98%	88.71%	3.06E+04	2.76E+04	2.69E+00	2.49E+00
f4	90.12%	95,29%	1.76E+04	1.06E+04	3.28E+00	3.02E+00
f5	20.13%	91.87%	3.67E+04	1.53E+04	1.28E+01	6.24E+00
f6	33.79%	88.92%	2.65E+04	1.30E+04	6.27E+00	4.16E+00
f7	62.52%	100%	1.65E+04	1.05E+04	2.68E+00	2.06E+00
f8	73.60%	92.18%	2.58E+04	1.59E+04	6.21E+00	3.18E+00
f9	69.23%	83.80%	3.52E+05	2.84E+05	1.23E+02	8.70E+01
f10	56.01%	88.56%	3.68E+04	2.59E+04	4.24E+00	3.02E+00
f11	90.85%	93.41%	1.65E+04	1.43E+04	3.43E+00	2.92E+00
f12	89.32%	82.71%	1.94E+04	1.64E+04	2.91E+00	3.36E+00
f13	72.10%	87.39%	2.48E+04	2.11E+04	3.38E+00	2.81E+00
f14	79.03%	97.28%	2.81E+04	2.08E+04	3.93E+00	3.16E+00
f15	57.90%	90.07%	5.28E+04	2.65E+04	8.36E+00	4.14E+00
f16	78.62%	89.06%	2.94E+04	2.39E+04	2.96E+00	2.09E+00
f17	92.35%	96.27%	1.32E+04	1.29E+04	1.45E+00	1.29E+00
f18	80.21%	89.20%	2.05E+04	1.83E+04	2.32E+00	2.15E+00
f19	62.17%	92.89%	3.72E+04	2.06E+04	6.25E+00	4.19E+00
f20	79.06%	86.77%	2.04E+05	1.85E+05	1.26E+02	1.14E+02
f21	53.12%	83.63%	3.60E+05	3.03E+05	3.21E+02	2.04E+02
f22	49.37%	98.45%	2.05E+04	1.63E+04	2.62E+00	2.29E+00

f23	73.56%	87.24%	3.95E+04	3.10E+04	4.25E+00	3.80E+00
f24	78.32%	90.73%	1.29E+05	9.86E+04	1.34E+01	9.89E+00
f25	94.53%	96.25%	1.21E+04	1.18E+04	1.62E+00	1.38E+00
f26	70.63%	86.52%	1.17E+05	9.84E+04	3.91E+02	2.83E+02
f27	77.25%	90.34%	3.79E+04	2.74E+04	5.07E+00	3.72E+00
f28	90.05%	92.68%	7.93E+04	6.74E+04	1.40E+01	9.08E+00
f29	74.62%	82.26%	9.04E+04	7.39E+04	2.33E+01	1.22E+01
f30	42.98%	83.90%	8.30E+04	4.82E+04	9.34E+00	5.92E+00

5 Application in the Image Enhancement

Image enhancement means that the brightness, contrast, resolution and other information of the image is improved so that the picture can be more in line with people's vision and looks like a higher quality picture [52]. Of course, it is also good in terms of the evaluation index of the image.

To study the night image quality evaluation model, an extensive database of natural night images (NNID) is established by Xiang et al. [53]. NNID includes images at low luminance and images at normal luminance. Twelve of these images are chosen for experiments and some evaluation metrics of image quality are used to detect the images after image enhancement by the improved Rafflesia optimization algorithm.

Among the evaluation metrics of images, the full-reference image quality evaluation metrics and the no-reference image quality evaluation metrics are included. The full-reference image quality evaluation is to judge the good or bad image quality by comparing the amount of information or feature similarity possessed by the original image and the distorted image. The no-reference image quality evaluation does not have a high-quality reference image and directly calculates the image quality of the distorted image. In image enhancement applications, the information entropy, edge information and standard deviation of the image are enhanced by optimizing the four optimal parameters in the solution to maximize the value of the fitness function.

5.1 Evaluation Indicators

The image quality evaluation indexes used in the experiments are MSE, PSNR, SSIM, and NIQE. The above four evaluation indicators are employed to assess the improved images through the application of various optimization algorithms. The evaluation findings are displayed in Table 6 - Table 9.

MSE (Mean Square Error): It is a full reference image quality evaluation index, which indicates the mean square error of the current image X and the reference image Y. The smaller value means the better image quality.

PSNR (Peak Signal to Noise Ratio): is the full reference image quality evaluation index. A larger value means less distortion, better image quality. The error between the respective pixel locations serves as the foundation for the most popular and extensively used objective evaluation index for images. The evaluation outcomes often do not align with human subjective perception because the visual characteristics of the human eye are not considered.

SSIM (Structural Similarity), a full reference image quality evaluation index, is a metric used to evaluate the similarity between the enhanced low luminance image and the high luminance image, and the larger the value, the better the quality of the enhanced image. It is a metric that matches people's visual perception, but in images with high PSNR or SSIM, the texture details of their images do not necessarily match the visual habits of the human eye.

Table 6. MSE values of 12 NNID images

	ROA-CS	ROA	PSO	CLAHE	Retinex
Img1	632.2145	541.5997	**464.3749**	676.7569	620.9642
Img2	**429.6633**	434.5089	431.5108	710.0540	712.4047
Img3	**452.0910**	475.8848	459.8762	718.9162	718.8108
Img4	84.3292	85.2034	**78.0278**	595.9507	599.7083
Img5	**229.0187**	366.6886	375.6510	701.7692	707.0509
Img6	127.2501	120.1999	**97.8378**	578.4577	581.1659
Img7	**27.3842**	31.9018	34.6842	245.9963	261.7859
Img8	**125.2035**	221.1993	247.3316	606.5937	610.5814
Img9	**337.9881**	413.7419	405.5153	598.0212	604.8308
Img10	451.5146	370.6894	**162.9001**	654.0573	655.6920
Img11	**350.4589**	353.5372	357.4512	592.2840	568.5324
Img12	**39.3602**	211.1451	46.0118	348.4718	357.9852

Table 7. PSNR values of 12 NNID images

	ROA-CS	ROA	PSO	CLAHE	Retinex
Img1	20.1222	20.7940	**21.4621**	19.8265	20.2001
Img2	**21.7995**	21.7508	21.7809	19.6179	19.6035
Img3	**21.5785**	21.3558	21.5044	19.5640	19.5647
Img4	28.8710	28.8262	**29.2083**	20.3787	20.3514
Img5	**24.5321**	22.4878	22.3830	19.6689	19.6363
Img6	27.0842	27.3318	**28.2257**	20.5081	20.4878
Img7	**33.7558**	33.0926	32.7295	24.2215	23.9513
Img8	**27.1546**	24.6830	24.1980	20.3018	20.2734
Img9	**22.8418**	21.9635	22.0507	20.3636	20.3145
Img10	21.5841	22.4407	**26.0116**	19.9746	19.9638
Img11	**22.6844**	22.6465	22.5986	20.4055	20.5833
Img12	**32.1802**	24.8850	31.5021	22.7091	22.5922

Table 8. SSIM values of 12 NNID images

	ROA-CS	ROA	PSO	CLAHE	Retinex
Img1	**0.5177**	0.4550	-0.0049	0.2639	0.4351
Img2	0.5962	**0.6316**	0.6303	0.4871	0.4823
Img3	**0.6658**	0.6634	0.6560	0.3471	0.3670
Img4	**0.5638**	0.3214	0.4812	0.4366	0.4320
Img5	**0.4987**	0.3515	0.4788	0.2449	0.2376
Img6	**0.4794**	0.4556	0.3306	0.3792	0.3712
Img7	0.3308	0.3780	-0.0062	0.4654	**0.4834**
Img8	0.4455	0.2301	**0.4464**	0.3860	0.3793
Img9	**0.3606**	0.3528	0.2970	0.2347	0.2355
Img10	**0.4184**	0.2344	0.2603	0.2056	0.1942
Img11	0.4325	0.4232	0.4318	0.4254	**0.4481**
Img12	0.2827	0.0105	0.2642	**0.4232**	0.4076

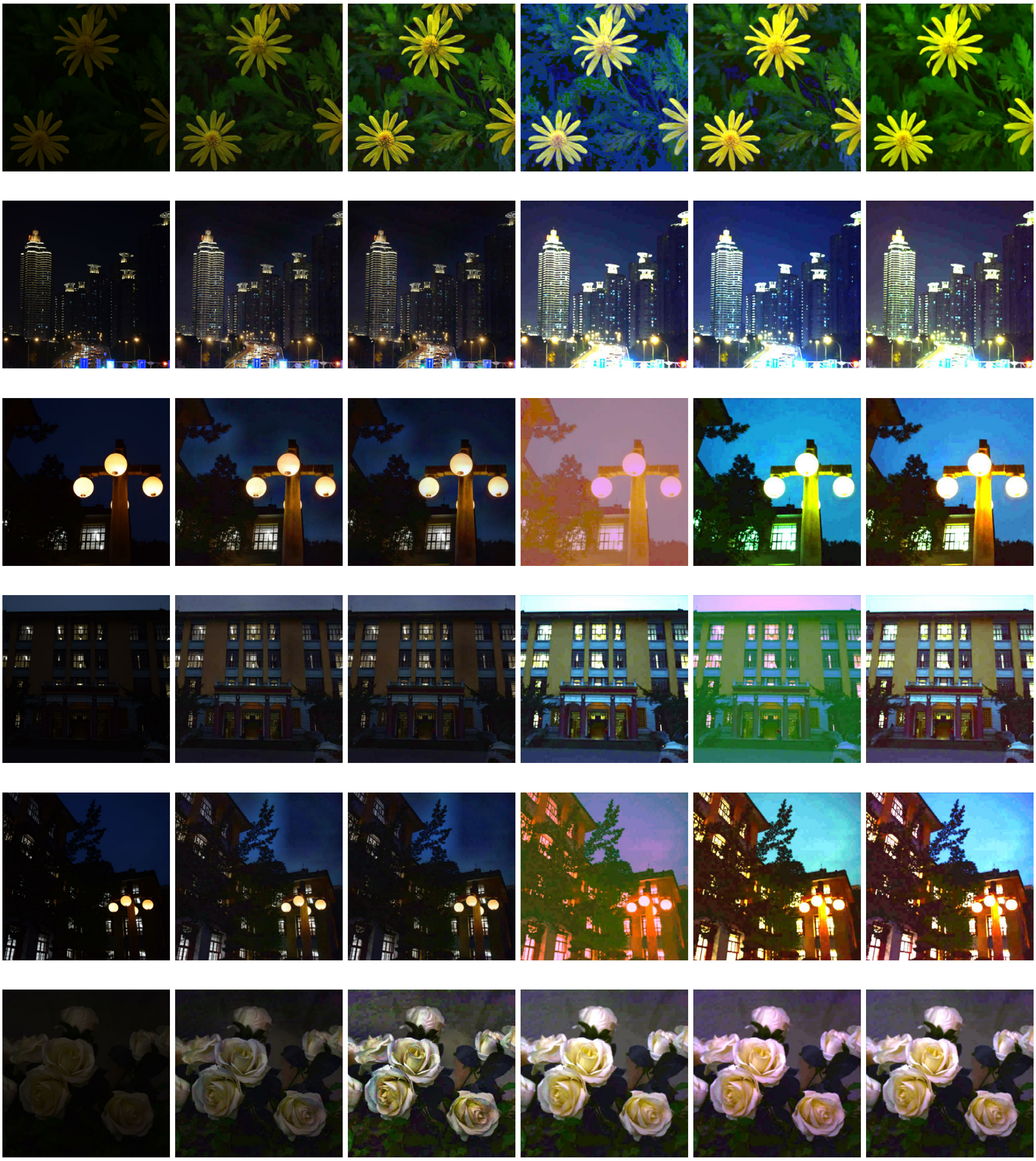
Table 9. NIQE values of 12 NNID images

	ROA-CS	ROA	PSO	CLAHE	Retinex
Img1	3.7613	3.5775	**3.2643**	4.3153	4.3778
Img2	4.9049	4.9683	4.8566	4.2500	**4.1755**
Img3	3.7355	3.6432	3.7815	3.9244	**3.6324**
Img4	**3.5858**	3.6089	3.5918	4.2296	4.1293
Img5	**4.5894**	4.7286	4.6594	5.8072	5.7994
Img6	**2.9421**	3.0147	2.9500	3.8605	3.8273
Img7	3.7492	**3.6951**	4.1239	4.5270	4.6695
Img8	**3.3131**	3.7161	3.3198	4.9032	4.8352
Img9	**3.0644**	3.2374	3.5213	3.9164	3.8652
Img10	**3.3653**	3.3968	4.8125	4.3338	4.4933
Img11	3.1695	3.1841	**3.1454**	3.5223	3.6124
Img12	**4.3787**	3.8824	3.9401	6.0979	5.9665

NIQE (Natural Image Quality Evaluator): It is a non-reference image quality evaluation index. Its smaller value means better image quality. As mentioned above, since SSIM alone may be biased in evaluating images, researchers have found this more effective image quality evaluation metric.

5.2 Visual Evaluation of Images

Figure 3 shows six images of the experiments, giving subjective evaluations of the low luminance images using the five enhancement algorithms. Among them, (a) is the original nighttime low brightness image. (b) is the image processed by the Constrained Contrast Adaptive Histogram Equalization (CLAHE) algorithm, which is a commonly used image enhancement algorithm. (c) is the image processed by Retinex algorithm, which is also a commonly used image enhancement method. (d) is the image enhanced by PSO algorithm. (e) is the image enhanced by the original ROA algorithm. And (f) is the image enhanced by ROA-CS algorithm. The image processed by ROA-CS algorithm, although in some image quality evaluation metrics are inferior to those of the images enhanced by other algorithms, but visually evaluated, they are more in line with the human aesthetics. The proposed ROA-CS algorithm effectively enhances the image details at low luminance, effectively reduces image artifacts, and maintains the naturalness of the original image.



(a) Original image (b) CLAHE (c) Retinex (d) PSO (e) ROA (f) ROA-CS

Figure 3. Comparison of enhancement results of 6 experimental images in NNID database

6 Conclusion

In this paper, we use Taguchi theory to hybridize the Rafflesia optimization algorithm and the cuckoo search algorithm to obtain a new hybrid algorithm. It is proved experimentally that the hybrid algorithm performs better than the original Rafflesia optimization algorithm. The improved algorithm is applied to nighttime image enhancement and compared with the images processed by other optimization algorithms and the original image enhancement algorithm. The evaluation indexes of image quality of the images are compared to prove the effectiveness and practicality of the ROA-CS algorithm. The simulation results show that ROA-CS obtains impressive results, optimizes the images' details and improves the images' recognition. There are many more ways to optimize the algorithm, which can incorporate parallel techniques, compact techniques, variable populations, etc. In the future, we can use many classical and advanced methods to optimize the ROA algorithm further and further enhance the algorithm's capability in practical applications. Through experimental comparisons, we found that the existing bootstrapping mechanisms FDB, FDC, and NSM have the powerful ability to improve the algorithms' ability to develop and explore. In the future, we can apply these methods to ROA-CS to further improve the performance of the algorithm. The algorithm proposed in this paper is only applied to the brightness enhancement aspect of image enhancement, and the processed images may become blurred and also have too much brightness.

Further improvements can be made in the future for potential uses in the field of image enhancement, such as Human-Aware Motion Deblurring (ICCV), and also ROA-CS can be applied to other fields, such as image segmentation, compression, feature selection, etc.

References

[1] J. Zhang, H. Nian, X. Ye, X. Ji, Y. He, A spatial correlation based partial coverage scheduling scheme in wireless sensor networks, *Journal of Network Intelligence*, Vol. 5, No. 2, pp. 34-43, May, 2020.

[2] A. Hatamlou, Black hole: A new heuristic optimization approach for data clustering, *Information sciences*, Vol. 222, pp. 175-184, February, 2013.

[3] D. E. Golberg, Genetic algorithms in search, optimization and machine learning, *Addion wesley*, 1989, p. 36.

[4] J. Kennedy, R. Eberhart, Particle swarm optimization, *Proceedings of ICNN'95-international conference on neural networks*, Perth, WA, Australia, 1995, pp. 1942-1948.

[5] E. Pashaei, E. Pashaei, A fusion approach based on black hole algorithm and particle swarm optimization for image enhancement, *Multimedia Tools and Applications*, Vol. 82, No. 1, pp. 297-325, January, 2023.

[6] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, *Journal of global optimization*, Vol. 11, No. 4, pp. 341-359, December, 1997.

[7] D. Karaboga, D. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, *Journal of global optimization*, Vol. 39, No. 3, pp. 459-471, November, 2007.

[8] C. Tian, X. Zhang, J.-C.-W. Lin, W. Zuo, Y. Zhang, Generative adversarial networks for image super-resolution: A survey, *arXiv preprint*, arXiv:2204.13260, April, 2022. https://arxiv.org/abs/2204.13620.

[9] X. Xue, Y. Wang, Using memetic algorithm for instance coreference resolution, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 28, No. 2, pp. 580-591, February, 2016.

[10] B. Zhao, T.-W. Sung, X. Zhang, A quasi-affine transformation artificial bee colony algorithm for global optimization, *Journal of Intelligent & Fuzzy Systems*, Vol. 40, No. 3, pp. 5527-5544, March, 2021.

[11] X. Xue, H. Yang, J. Zhang, Using population-based incremental learning algorithm for matching class diagrams, *Data Science and Pattern Recognition*, Vol. 3, No. 1, pp. 1-8, March, 2019.

[12] X.-S. Xue, A compact firefly algorithm for matching biomedical ontologies, *Knowledge and Information Systems*, Vol. 62, No. 7, pp. 2855-2871, July, 2020.

[13] J.-S. Pan, L. Li, S.-C. Chu, K.-K. Tseng, H.-A. Shehadeh, Martial art learning optimization: A novel metaheuristic algorithm for night image enhancement, *Journal of Internet Technology*, Vol. 24, No. 7, pp. 1415-1428, December, 2023.

[14] S.-C. Chu, P.-W. Tsai, J.-S. Pan, Cat swarm optimization, *Pacific Rim international conference on artificial intelligence*, Guilin, China, 2006, pp. 854-858.

[15] P.-W. Tsai, J.-S. Pan, S.-M. Chen, B.-Y. Liao, S.-P. Hao, Parallel cat swarm optimization, *international conference on machine learning and cybernetics*, Kunming, China, 2008, pp. 3328-3333.

[16] P.-W. Tsai, J.-S. Pan, P. Shi, B.-Y. Liao, A new framework for optimization based-on hybrid swarm intelligence, in: B. K. Panigrahi, Y. Shi, M. H. Lim (Eds.) *Handbook of Swarm Intelligence: Concepts, Principles and Applications*, Vol. 8, Springer, Berlin, Heidelberg, 2011, pp. 421-449.

[17] Z.-Y. Meng, J.-S. Pan, H.-R. Xu, Quasi-affine transformation evolutionary (quatre) algorithm: A cooperative swarm based algorithm for global optimization, *Knowledge-Based Systems*, Vol. 109, pp. 104-121, October, 2016.

[18] Z.-G. Du, J.-S. Pan, S.-C. Chu, H.-J. Luo, P. Hu, Quasi-affine transformation evolutionary algorithm with communication schemes for application of RSSI in wireless sensor networks, *IEEE Access*, Vol. 8, pp. 8583-8594, January, 2020.

[19] P. C. Song, S. C. Chu, J.-S. Pan, H. Yang, Simplified Phasmatodea population evolution algorithm for optimization, *Complex & Intelligent Systems*, Vol. 8, No. 4, pp. 2749-2767, August, 2022.

[20] J.-S. Pan, L. G. Zhang, R. B. Wang, V. Snášel, S. C. Chu, Gannet optimization algorithm: A new metaheuristic algorithm for solving engineering optimization problems, *Mathematics and Computers in Simulation*, Vol. 202, pp. 343-373, December, 2022.

[21] J.-S. Pan, Z.-L. Fu, C.-C. Hu, P.-W. Tsai, S.-C. Chu Rafflesia Optimization Algorithm Applied in the Logistics Distribution Centers Location Problem, *Journal of Internet Technology*, Vol. 23, No. 7, pp.1541-1555, December, 2022.

[22] X.-S. Yang, S. Deb, Cuckoo search via Lévy flights, *2009 World congress on nature & biologically inspired*

computing (NaBIC), Coimbatore, India, 2009, pp. 210-214.

[23] Ş. Kasman, Impact of parameters on the process response: A Taguchi orthogonal analysis for laser engraving, *Measurement*, Vol. 46, No. 8, pp. 2577-2584, October, 2013.

[24] S. Fan, W. Liang, D. Ding, H. Yu, LACN: A lightweight attention-guided ConvNeXt network for low-light image enhancement, *Engineering Applications of Artificial Intelligence*, Vol. 117, Article No. 105632, January, 2023.

[25] Q. Zhang, J. Xiao, C. Tian, J.-C.-W. Lin, S. Zhang, A robust deformed convolutional neural network (CNN) for image denoising, *CAAI Transactions on Intelligence Technology*, Vol. 8, No. 2, pp. 331-342, June, 2023.

[26] S. Suresh, S. Lal, Modified differential evolution algorithm for contrast and brightness enhancement of satellite images, *Applied soft computing*, Vol. 61, pp. 622-641, December, 2017.

[27] A. H. Khan, S. Ahmed, S. K. Bera, S. Mirjalili, D. Oliva, R. Sarkar, Enhancing the contrast of the grey-scale image based on meta-heuristic optimization algorithm, *Soft Computing*, Vol. 26, No. 13, pp. 6293-6315, July, 2022.

[28] G. Singh, A. Mittal, Various image enhancement techniques-a critical review, *International Journal of Innovation and Scientific Research*, Vol. 10, No. 2, pp. 267-274, October, 2014.

[29] K. G. Dhal, S. Ray, A. Das, S. Das, A survey on nature-inspired optimization algorithms and their application in image enhancement domain, *Archives of Computational Methods in Engineering*, Vol. 26, No. 5, pp. 1607-1638, November, 2019.

[30] S.-H. Yun, G. J. Tearney, J. F. de Boer, N. Iftimia, B. E. Bouma, High-speed optical frequency-domain imaging, *Optics express*, Vol. 11, No. 22, pp. 2953-2963, November, 2003.

[31] Y. Zeng, X. Wu, Medical Image Denoising Processing Application Technology Based on Combined Filtering, *2022 2nd International Conference on Bioinformatics and Intelligent Computing*, Harbin, China, 2022, pp. 447-451.

[32] F. Saitoh, Image contrast enhancement using genetic algorithm, *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*, Tokyo, Japan, 1999, pp. 899-904.

[33] M. Braik, A. F. Sheta, A. Ayesh, Image Enhancement Using Particle Swarm Optimization, *Proceedings of the World congress on engineering*, *Vol. 1*, London, UK, 2007, pp. 696-701.

[34] P. K. Mahapatra, S. Ganguli, A. Kumar, A hybrid particle swarm optimization and artificial immune system algorithm for image enhancement, *Soft computing*, Vol. 19, No. 8, pp. 2101-2109, August, 2015.

[35] T. Xiang, H. Wang, Y. Shi, Hybrid WSN node deployment optimization strategy based on CS algorithm, *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chengdu, China, 2019, pp. 621-625.

[36] Z.-S. Wang, J.-S. Pan, K.-C. Huang, T.-S. Pan, J.-P. Li, Hybrid Gray Wolf Optimization and Cuckoo Search Algorithm based on the Taguchi Theory, *Advances in Intelligent Information Hiding and Multimedia Signal Processing*, Kaohsiung, Taiwan, 2022, pp. 219-228.

[37] V. Wankhede, D. Jagetiya, A. Joshi, R. Chaudhari, Experimental investigation of FDM process parameters using Taguchi analysis, *Materials Today: Proceedings*, Vol. 27, No. 3, pp. 2117-2120, 2020.

[38] J.-T. Tsai, T.-K. Liu, J.-H. Chou, Hybrid Taguchi-genetic algorithm for global numerical optimization, *IEEE Transactions on evolutionary computation*, Vol. 8, No. 4, pp. 365-377, August, 2004.

[39] C. Zhao, Z. Liu, Z. Chen, Y. Ning, A novel krill herd algorithm with orthogonality and its application to data clustering, *Intelligent Data Analysis*, Vol. 25, No. 3, pp. 605-626, April, 2021.

[40] H. T. Kahraman, S. Aras, E. Gedikli, Fitness-distance balance (FDB): a new selection method for meta-heuristic search algorithms, *Knowledge-Based Systems*, Vol. 190, Article No. 105169, February, 2020.

[41] B. Ozkaya, H. T. Kahraman, S. Duman, U. Guvenc, Fitness-Distance-Constraint (FDC) based guide selection method for constrained optimization problems, *Applied Soft Computing*, Vol. 144, Article No. 110479, September, 2023.

[42] H. T. Kahraman, M. Katı, S. Aras, D. A. Taşci, Development of the Natural Survivor Method (NSM) for designing an updating mechanism in metaheuristic search algorithms, *Engineering Applications of Artificial Intelligence*, Vol. 122, Article No. 106121, June, 2023.

[43] B. Ozkaya, S. Duman, H. T. Kahraman, U. Guvenc, Optimal solution of the combined heat and power economic dispatch problem by adaptive fitness-distance balance based artificial rabbits optimization algorithm, *Expert Systems with Applications*, Vol. 238, Article No. 122272, March, 2024.

[44] Y. Sonmez, S. Duman, H. T. Kahraman, M. Kati, S. Aras, U. Guvenc, Fitness-distance balance based artificial ecosystem optimisation to solve transient stability constrained optimal power flow problem, *Journal of Experimental & Theoretical Artificial Intelligence*, Vol. 36, No. 5, pp. 745-784, 2024.

[45] S. Mirjalili, A. Lewis, The whale optimization algorithm, *Advances in engineering software*, Vol. 95, pp. 51-67, May, 2016.

[46] S. Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm, *Knowledge-based systems*, Vol. 89, pp. 228-249, November, 2015.

[47] X.-S. Yang, A. H. Gandomi, Bat algorithm: a novel approach for global engineering optimization, *Engineering computations*, Vol. 29, No. 5, pp. 464-483, July, 2012.

[48] X.-S. Yang, Firefly algorithm, stochastic test functions and design optimisation, *International journal of bio-inspired computation*, Vol. 2, No. 2, pp. 78-84, March, 2010.

[49] S. Arora, S. Singh, Butterfly optimization algorithm: a novel approach for global optimization, *Soft Computing*, Vol. 23, No. 3, pp. 715-734, February, 2019.

[50] S. Gürgen, H. T. Kahraman, S. Aras, İ. Altın, A comprehensive performance analysis of meta-heuristic optimization techniques for effective organic rankine cycle design, *Applied Thermal Engineering*, Vol. 213, Article No. 118687, August, 2022.

[51] H. T. Öztürk, H. T. Kahraman, Meta-heuristic search algorithms in truss optimization: Research on stability and complexity analyses, *Applied Soft Computing*, Vol. 145, Article No. 110573, September, 2023.

[52] K.-W. Huang, C.-C. Lin, Y.-M. Lee, Z.-X. Wu, A deep learning and image recognition system for image recognition, *Data Science and Pattern Recognition*, Vol. 3, No. 2, pp. 1-11, May, 2019.

[53] T. Xiang, Y. Yang, S. Guo, Blind night-time image quality assessment: Subjective and objective approaches, *IEEE*

Transactions on Multimedia, Vol. 22, No. 5, pp. 1259-1272, May, 2020.

Biographies

Jeng-Shyang Pan received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology in 1986, the M.S. degree in communication engineering from National Chiao Tung University, Taiwan, in 1988, and the Ph.D. degree in electrical engineering from the University of Edinburgh, U.K., in 1996. He is currently the Professor of Shandong University of Science and Technology. He is the IET Fellow, U.K., and has been the Vice Chair of the IEEE Tainan Section.

Yingying Wang received the B.S. degree from Shandong University of Finance and Economics, China, in 2022. She is currently pursuing the master degree with the Shandong University of Science and Technology, Qingdao, China. Her recent research interests are swarm intelligence and image enhancement.

Shu-Chuan Chu received the Ph.D. degree in 2004 from the School of Computer Science, Engineering and Mathematics, Flinders University of South Australia. She joined Flinders University in December 2009 after 9 years at the Cheng Shiu University, Taiwan. She is the Research Fellow in the College of Science and Engineering of Flinders University, Australia from December 2009. Currently, She is the Research Fellow with PhD advisor in the College of Computer Science and Engineering of Shandong University of Science and Technology from September 2019. Her research interests are mainly in swarm intelligence, intelligent computing and data mining.

Ling Li received the B.S. degree from Shandong University of Science and Technology, China, in 2020. She is currently pursuing the master degree with the Shandong University of Science and Technology, Qingdao, China. Her research interests are swarm intelligence and image enhancement.

Ning Zhong received the Ph.D. degree in computer application from University of Tokyo, Japan, in 1995. He is a major creator in the field of Web Intelligence (WI). Since 2001, he has been a professor at Maebashi University of Technology in Japan. His current research intersets include network intelligence, knowledge discovery and data mining, intelligent agent technology and applications, brain informatics.