Heliyon 10 (2024) e37814

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Efficient adaptive learning rate for convolutional neural network based on quadratic interpolation egret swarm optimization algorithm

Peiyang Wei [a,b,c,d,*], Mingsheng Shang [c,d], Jiesan Zhou [b], Xiaoyu Shi [c,d]

[a] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[b] School of Software Engineering, Chengdu University of Information Technology, Chengdu, 610225, China
[c] Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing, 400714, China
[d] Chongqing School, University of Chinese Academy of Sciences, Chongqing, 400714, China

A R T I C L E I N F O

Keywords:
Convolutional neural network
Multi-domain image classification
Adaptive learning rate
Egret swarm optimization algorithm
Quadratic interpolation

A B S T R A C T

Convolutional neural network (CNN) has recently become popular for addressing multi-domain image classification. However, most existing methods frequently suffer from poor performance, especially in performance and convergence for various datasets. Herein, we have proposed an algorithm for multi-domain image classification by introducing a novel adaptive learning rate rule to the conventional CNN. Specifically, we adopt the CNN to extract rich feature representations. Given that the hyperparameters of the learning rate have a positive effect on the prediction error, the Egret Swarm Optimization Algorithm (ESOA) is introduced to update the learning rate, which can jump out of local extrema during exploration. Therefore, combined with quadratic interpolation, the objective function can be approximated by a polynomial, thereby improving its prediction accuracy. To verify the robustness of the proposed algorithm, we conducted comprehensive experiments in five domain public datasets to fulfil the task of image classification. Meanwhile, the highest accuracy rate of 97.15 % was obtained on the test set. The performances of our method on 24 benchmark functions (CEC2017 and CEC2022) are compared with Particle Swarm Optimization (PSO), Genetic Algorithm(GA), Whale Optimization Algorithm (WOA), Catch Fish Optimization Algorithm(CFOA), GOOSE Algorithm(GO) and ESOA. In two benchmark sets, the performance metric values of our algorithm rank no. 1, especially in all unimodal functions in contrast with other baseline algorithms.

1. Introduction

Deep learning, especially convolutional neural network (CNN), has recently become popular for various image classification tasks ([1,2]; [3] [4–6]; [7]). These include, but are not limited to, facial recognition, intelligent video analysis, the transportation sector and image recognition in the medical and other fields. However, the hyperparameters of neural networks, particularly the learning rate, have important impacts on image classification results. If the learning rate is tiny, the convergence speed is not good and the training time is wasted. On the contrary, an excessive learning rate may cause the parameters to oscillate back and forth on both sides of the

* Corresponding author. School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China.
E-mail address: weipy@cuit.edu.cn (P. Wei).

https://doi.org/10.1016/j.heliyon.2024.e37814
Received 13 May 2024; Received in revised form 6 September 2024; Accepted 10 September 2024
Available online 13 September 2024
2405-8440/

optimal solution ([8]; [9]). Therefore, learning rate research is of great significance for optimizing and improving image classification tasks.

The framework of CNN consists of convolution, pooling and dense and output layers (X. [10]). In training, the step size of the gradient updating direction is the learning rate ([11–13]; [14]). Updating the learning rate, the minimum loss value can be found, and the converged loss value is guaranteed to be the global optimal solution of the neural network, subsequently improving the overall performance of the model [15].

The adaptive learning rate dynamically adjusts the strategy. Automatically adjust the learning rate using specific algorithms such as Adam, RMSprop, and so forth [16]. This strategy can be optimised by introducing various evolutionary algorithms. However, adaptive learning rate has problems such as difficulty in hyperparameter adjustment, too slow or too fast adjustment of learning rate and the acquisition of local minima. Meanwhile, we will introduce this situation in detail in section II about famous scholars' work around the world. Based on this strategy and the above problems, the ESOA algorithm combined with the quadratic interpolation is introduced to optimise the adaptive learning rate.

The ESOA is a heuristic optimization algorithm that is explained by Chen et al. [17] by the predation behaviour of snow egrets and great egrets. The algorithm combines the sit-and-wait and aggressive strategies, takes the objective function of the optimization problem as the fitness function and finds the optimal solution by simulating the predation behaviour of the egret group. In addition, the algorithm can effectively balance the relationship between exploitation and exploration. Although the algorithm can quickly obtain the optimal solution, it may also fail to search for the global optimal solution because the exploration space is not sufficiently large. Therefore, we introduce the quadratic interpolation method. Because this method can provide a smoother function approximation during the search process, it is helpful to jump out of local extreme points (Z, [1]; Z, [15]; C. [18]; [19,20]). Thus, the proposed quadratic interpolation model based on the Egret Swarm Optimization Algorithm (QIESOA) provides a new approach and method to solve the problem. The primary contributions of this paper are as follows.

a) We applied five public datasets covering e-commerce, characters, natural scenes, animals, weather, flowers and agricultural crops for image classification to explain the generality and accuracy of the presented method.
b) A novel adaptive learning rate rule is proposed based on QIESOA that can be embedded in a CNN for effective image classification. The proposed QIESOA can automatically find the optimal solution using sit-and-wait and aggressive strategies. In addition, the approximation characteristics of quadratic interpolation are applied to further improve the image classification efficiency.
c) Extensive experiments show that our model can trade off accuracy and be effective for multi-domain image classification. In addition, the presented QIESOA outperforms several state-of-the-art models when incorporating CNN in completing these tasks.

The remainder of this paper is organised as follows. Section 2 highlights a survey of related works and presents the preliminaries. Section 3 presents the proposed model. Section 4 describes the experimental results and analysis of its key information. Finally, Section 5 presents the conclusion and future work.

2. Background and related work

In recent years, to complete the CNN-based image classification task, numerous neural networks have been developed. The VGG model is a classic CNN model. It uses multiple 3x3 small convolution kernels to replace the large 5x5 convolution kernels, which reduces the number of parameters and computation. The GoogLeNet model introduced by Li Y et al. [21], also called the Inception model, is based on the VGG model and further adopts the idea of 'decomposition'. It decomposes the large convolution kernel into small convolution kernels, such as decomposing the 5x5 kernel into 1x5 and 5x1 kernels, thus reducing the number of model parameters and computation. The Inception-v3 model is a representative implementation of this model, which achieves more effective feature extraction through structures such as upsampling, 1x3 and 3x1 convolution kernels. Szegedy et al. [22] introduced a residual structure named the ResNet model, which effectively solves the gradient vanishing problem in deep neural networks so that the network can be designed deeper. Using 'skip connections', ResNet avoids the performance degradation caused by increasing depth in traditional networks. In addition, there are some similar improved models, such as DenseNet ([23]), MobileNet ([24]) and SegNet [25], that have achieved good results in all types of computing tasks. This improvement mainly involves network structure and model design, such as the use of a deeper network structure, a more effective connection mode and a more reasonable activation function. However, these models require a lot of parameter tuning work, especially the learning rate. The learning rate directly determines the convergence speed of the model and whether the optimal solution can be obtained.

The learning rate controls how much the model updates its weights each time. A constant value is the easiest way to set the learning rate, such as 0.01 [26]. However, this method has obvious immutability. Therefore, Park et al. ([27]) proposed a cost function method to search for the optimal value of the learning rate parameter. Yang et al. ([28]) presented a new metric to measure the degree of advantage between different sub-tasks, to reflect its influence on the learning rate during gradient update. In addition, based on these experimental findings, the original optimiser is divided into different tasks to have the effect of task adaptation. Lee et al. [29] proposed a general learning rate tuning framework and three key heuristics to address the problem of poor generalisability. Liu et al. [30] proposed a CNN model with channel-spatial attention for adaptive learning rate to surface damage detection of wind turbine blades. Moreover, Hu et al. (2022) presented an adaptive learning rate network based on the idea of randomness. Because the interests of users and items will shift over time in streaming media recommendation systems, Viniski et al. [31] proposed four special adaptive learning rate optimisers with strong generalisability based on the matrix factorisation method. It can be seen from the above literature that a great deal of work has been conducted on the research and application of adaptive learning rates; however, it still has optimization problems such as slow execution speed and long convergence period, which are more sensitive to metaheuristic algorithms, requiring further improvement.

Learning rate optimization based on a metaheuristic algorithm mainly considers finding the best learning rate in the feasible solution space using a heuristic method. Evolutionary computation follows Darwin's survival of the fittest and achieves the optimal solution through the overall optimization of the population [31,32]. However, it has several issues such as high computational complexity and the need to adjust multiple parameters. There is another type of metaheuristic algorithm that simulates the way humans solve problems to achieve optimal solutions ([33]; [34]). The main advantage is that it can deal with complex and nonlinear optimization problems; however, it may also get trapped in local optimal solutions. Physics-based and chemical-based algorithms obtain optimal solutions by simulating natural phenomena such as changes in molecular structure and energy transformation ([1,5]; X. [10–12]; [9]; [35]). However, these methods require specialised knowledge and have high computational complexity. The algorithm based on swarm intelligence has always been a research hotspot and trend. This type of method mainly simulates the collective behaviour of natural groups such as birds and fish schools and realises the overall optimal solution through interactions among individuals in the group, such as the Beetle Swarm–Based Butterfly Optimization ([34,36]), Grey Wolf Optimization ([7]; [37]), Particle Swarm Optimization (C. [18]; [38]), Whale Optimization [39] and Dwarf Mongoose Optimization [40]. ESOA is a new swarm intelligence algorithm proposed in 2022 [17]. This algorithm has been used in fraud detection ([3]), gas turbine cooling ([41]), medical image classification ([42]; [43]) and household integrated energy [30] and has achieved several good results. ESOA can obtain the most out of the information of individuals in the swarm and exhibits good parallelism; however, it may also fall into the problem of local optimal solutions. To address this issue, many researchers have used Gaussian process regression [1], sine cosine algorithm ([5]; [41,44] [32,45]; X. [46,46]) and quadratic interpolation (Z. [1,32]; [34]; [39]) to conduct more research and experiments. Among them, the quadratic interpolation method is the first choice for optimizing the ESOA algorithm because it can approximate the function by constructing quadratic polynomials. The quadratic interpolation method is a simple, efficient, practical and widely applicable numerical approximation method that is suitable for various types of functions and obtains high accuracy and computational efficiency. Therefore, it has a strong reference and application significance to build an adaptive learning rate model based on the ESOA.

3. Methodology

This section gives a detail of the CNN with its learning rate parameter and ESOA with a quadratic interpolation–based optimization method.

3.1. Convolutional neural network

The advantages of CNN, such as translation invariance, feature selection, parameter sharing, pooling operation, multi-level feature extraction and powerful computing power, allow it to be suitable for image classification tasks. The quality of the dataset, depth of the model, choice of parameters, regularisation method, choice of optimiser and loss function yield better classification results. Fig. 1 presents the structure of the CNN network.

The framework of the CNN mainly includes multiple convolutional layers, relying on layers, max pooling layers, dropout layers, flattened layers, dense layers, and so on. The combination and stacking method of these layers affects the image classification effect. As an important component of CNN, convolutional layers can extract image features, such as colour, texture and shape.

The max pooling layer decreases the dimensions of features, reduces the number of parameters and calculations and enhances the generalisability of the model. The location and selection strategy of the max pooling layer will also affect the image classification effect. The dense layer is typically located in the last part of the CNN and can classify the extracted features. The number of neurons and

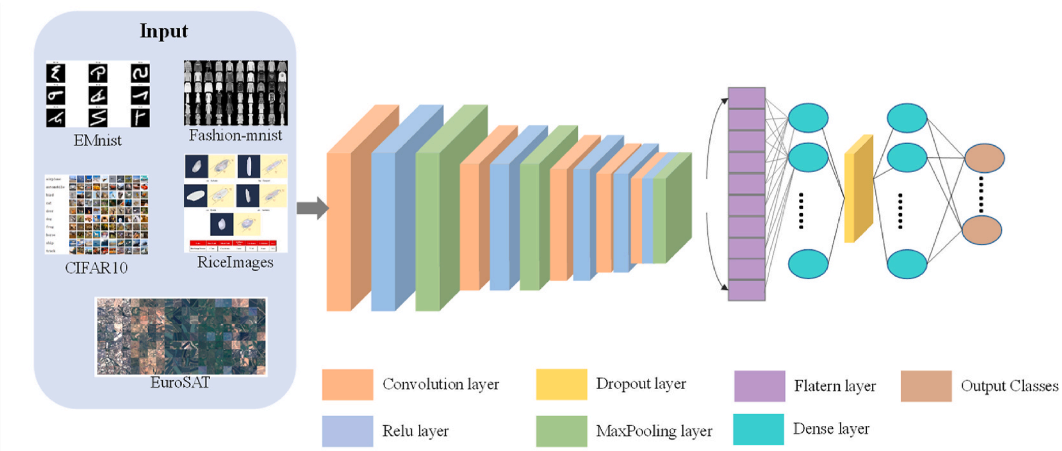


Fig. 1. CNN network.

parameter settings in the dense layer will affect the accuracy and robustness of classification. The dropout layer can mainly remove unimportant neurons in the network, which is conducive to the improvement of model efficiency. This alleviates the overfitting problem and reduces computational complexity. We chose Relu as the activation function of the model. The flattened layer is primarily used to convert the input multi-dimensional data into one-dimensional data. In particular, to accelerate the convergence of the model and avoid the influence of the affine transformation of the image, we do a normalisation pre-processing operation on the input image data to further improve the training speed. Therefore, choosing an appropriate structure allows CNN to efficiently process image data and extract more effective features, thereby improving the accuracy and robustness of image classification. The learning rate is an important parameter in CNN models that affects the model training speed and model performance. For image classification tasks, if the learning rate is adjusted dynamically and adaptively based on the progress of training, it can better adapt to the variation of training data and improve the performance of the model. Therefore, it is crucial to obtain a suitable learning rate strategy to complete the image classification task in the training process.

3.2. ESOA

ESOA is a heuristic optimization algorithm proposed by Chen et al. [17]. Inspired by the predatory behaviour of egrets, the algorithm combines the collaborative optimization of the sit-and-wait strategy and aggressive strategies. The outstanding features of ESOA are that it is simple and hospitable to implement, has global search ability, can handle complex and difficult-to-solve optimization problems with traditional methods and has adaptability and good robustness; thus, it is suitable for updating and optimizing the learning rate of CNN, as shown in Fig. 2.

Egrets adopt a sit-and-wait strategy will quietly wait for a prey to appear, whereas egrets that adopt an aggressive strategy will actively pursue prey. These two strategies have different characteristics in energy consumption and revenue acquisition. The energy consumption of the sit-and-wait strategy is low; however, the revenue is relatively stable. However, the aggressive strategy has higher energy consumption, but it is possible to achieve higher payoffs.

Inspired by these two strategies, the ESOA considers candidate solutions as individuals in the egret swarm and searches for the optimal solution by simulating the predation behaviour of the egret swarm. The algorithm is divided into three main parts: sit-and-wait strategy, aggressive strategy and discriminative condition. Each egret colony consists of n egret squads, each of which in turn contains three egrets. Among them, Egret A implements the sit-and-wait strategy, while Egrets B and Egret C adopt the random walk and bounding mechanism in the aggressive strategy, respectively.

With respect to location update, Egret A is updated according to the observation equation, Egret B adopts random walk, while Egret C uses boundary strategy. In each iteration, the fitness of each egret is separately calculated, and the egret position is updated based on its value. Specifically, if the new value of the fitness is better, it is updated; otherwise, it will only be updated with a certain probability.

1) **Sit-and-Wait Strategy**: The observation equation of the $i\,th$ Egret A is $\widehat{y}_i = F(x_i)$, where a function $F(*)$ represents the evaluation method of Egret A for possible targets at the current location, x_i is the position of group i and $\widehat{y}_i$ is the estimation of prey in current location by iteration. Moreover, the estimation method can be parameterised as;

$$\widehat{y}_i = w_i \cdot x_i \tag{1}$$

where w_i is the weight of the estimate method. The error e_i could be described as follows;

$$e_i = \frac{\|\widehat{y}_i - y_i\|^2}{2} \tag{2}$$

where g_i is the practical gradient of w_i, which can be calculated by Eq. (2), and finally the prediction of the direction $\widehat{d}_i$ is completed based on the gradient;

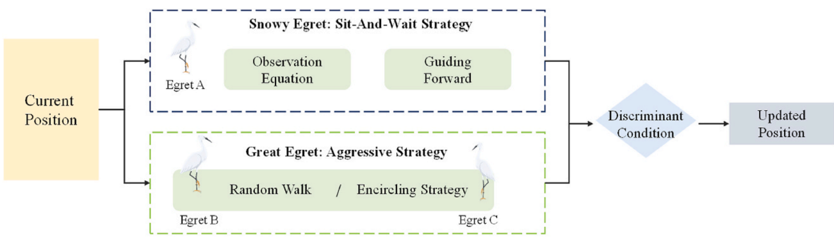


Fig. 2. The basic framework of ESOA.

$$\begin{aligned}\hat{g}_i &= \frac{\partial \hat{e}_i}{\partial w_i} \\ &= \frac{1}{2}\frac{\partial \|\hat{y}_i - y_i\|^2}{\partial w_i} \\ &= (\hat{y}_i - y_i)\cdot x_i\end{aligned} \tag{3}$$

$$\hat{d}_i = \frac{\hat{g}_i}{|\hat{g}_i|} \tag{4}$$

ESOA defines $d_{b,i}$ and $d_{g,i}$. $d_{b,i}$ denotes the change of direction made based on the optimal position of the same group of egrets. Meanwhile, $d_{g,i}$ denotes the change of direction made according to the optimal position of all egrets;

$$d_{b,i} = \frac{x_{ibest} - x_i}{|x_{ibest} - x_i|}\cdot\frac{f_{ibest} - f_i}{|x_{ibest} - x_i|} + d_{ibest} \tag{5}$$

$$d_{g,i} = \frac{x_{gbest} - x_i}{|x_{gbest} - x_i|}\cdot\frac{f_{gbest} - f_i}{|x_{gbest} - x_i|} + d_{gbest} \tag{6}$$

where d_{ibest} is the direction of the best flight individual in the egret swarm, d_{gbest} is the global historical optimal position of the best individual and the flight direction of the historical optimal position, respectively, f_{ibest} is the best fitness of the egret group and f_{gbest} is the best fitness of the population. The equation of integrated gradient g_i can be obtained according to Eqs. (4)–(6);

$$g_i = (1 - r_i - r_g)\cdot\hat{d}_i + r_i\cdot d_{b,i} + r_g\cdot d_{g,i} \tag{7}$$

where $r_i, r_g \in [0, 0.5)$. The position of Egret A is updated using the integral gradient in Eq. (7);

$$x_{a,i} = x_i + \exp\left(\frac{-t}{0.1\cdot t_{max}}\right)\cdot 0.1\cdot hop\cdot g_i \tag{8}$$

where t is the current iteration time and t_{max} is the maximum iteration time; meanwhile, hop represents the upper limit of the solution minus the lower limit of the solution, which is the D-value of the location boundary. Finally, the observation equation weights are updated based on the following equation;

$$m_i = \beta_1\cdot m_i + (1 - \beta_1)\cdot g_i \tag{9}$$

$$v_i = \beta_2\cdot v_i + (1 - \beta_2)\cdot g_i^2 \tag{10}$$

$$w_i = w_i - m_i/\sqrt{v_i} \tag{11}$$

where β_1 , β_2 are set to 0.9 and 0.99 and m_i and v_i are initialised to 0.

2) **Aggressive Strategy:** This strategy is adopted by Egret B, which is manifested as a random walk, and egret predatory behaviour can be expressed using the following equation;

$$x_{b,i} = x_i + \tan(r_{b,i})\cdot hop\,/\,(1 + t) \tag{12}$$

where $x_{b,i}$ is the desired location of Egret B next time and $r_{b,i}$ is a random number in $(-\pi/2, \pi/2)$.

Egret C tends to actively pursue prey, so it adopts the encircling mechanism as the update method of its position;

$$x_{c,i} = (1 - r_i - r_g)\cdot x_i + r_i\cdot(x_{ibest} - x_i) + r_g\cdot(x_{gbest} - x_i) \tag{13}$$

3) **Discriminant Condition:** In ESOA, each group of egrets must act in unison; therefore, the final destination of this group of egrets based on discriminant condition is determined. The solution matrix is;

$$x_{s,i} = [x_{a,i}, x_{b,i}, x_{c,i}] \tag{14}$$

The principle of discriminant condition is described as follows: The egret swarm compares the position and fitness of the three updated egrets with the position and fitness of the previous iteration and selects the smallest egret as the result of this iteration. If all the updated positions of the egrets are worse than the previous egret, the scheme with the worst updated position is adopted with a 30 % probability.

3.3. Quadratic interpolation

ESOA algorithm is divided into three strategies, among which the sitting strategy depends on the prey distance D, and the random strategy is apt to make the solution fall into the local minimum. Herein, the quadratic interpolation method is used to actively explore and predict the next position of prey to escape the local optimal solution.

Quadratic interpolation (Z, [15]), as a local search operator, can search for the best solution for the population in the known search area. The method uses the quadratic curve formed by three known nodes to approximate the objective function and makes the optimal solution of the objective function approach the extreme value of the quadratic function. The core idea and process of this method can be described as follows:

There are $A = (a_1, a_2, a_3, \ldots a_d)$, $B = (b_1, b_2, b_3, \ldots b_d)$, $C = (c_1, c_2, c_3, \ldots c_d)$ as three nodes. Then, according to the quadratic interpolation method, the position is updated using Eq. (15) to generate a new individual $\vec{x} = \left(\vec{x}_1, \vec{x}_2, \vec{x}_3, \ldots \vec{x}_d\right)$;

$$x_h = \frac{1}{2} \times \frac{(c_h^2 - b_h^2) \times f(A) + (a_h^2 - c_h^2) \times f(B) + (b_h^2 - a_h^2) \times f(C)}{(c_h - b_h) \times f(A) + (a_h - c_h) \times f(B) + (b_h - a_h) \times f(C)} \tag{15}$$

where the fitness values of the three nodes are $f(A)$, $f(B)$ and $f(C)$, respectively, and $h = 1, 2, 3, \ldots, d$ is the dimension of the problem to be solved. The newly generated individual is necessarily the minimum point of the conic.

After using the quadratic interpolation, the population will be updated into a new one. In the process of generating a new population position in each iteration, the egret group will be rearranged according to the fitness from high to low, and three individuals such as the best value x_l of the last iteration population, the best value x_c of the current iteration population and the global best value x_g are selected from the population in turn for quadratic interpolation to obtain a new individual x_b. The adaptive greedy strategy is applied to individuals x_b and $\vec{x}_b$ to re-evaluate their fitness as follows;

$$x_b = \begin{cases} \vec{x}_b f\left(\vec{x}_b\right) < f(x_b) \\ x_b, f\left(\vec{x}_b\right) \geq f(x_b) \end{cases} \tag{16}$$

The quadratic interpolation of the three known optimal solutions can obviously enhance the search capability and speed of the algorithm. On the contrary, if the three individuals are too far from the optimal solution, the quadratic interpolation can increase the population diversity, expand the search range and enhance the ability of the algorithm to jump out of the local optimal solution. Therefore, the quadratic interpolation is applied to the ESOA algorithm as a local search operator to enhance its local search ability and improve the convergence speed and accuracy of the population.

3.4. QIESOA model design

This section discusses a new adaptive learning rate optimization algorithm QIESOA designed for CNN. Fig. 3 presents the overall framework of our proposed model calculation (see Algorithm 1 for a pseudo-code description of the QIESOA). Meanwhile, the main

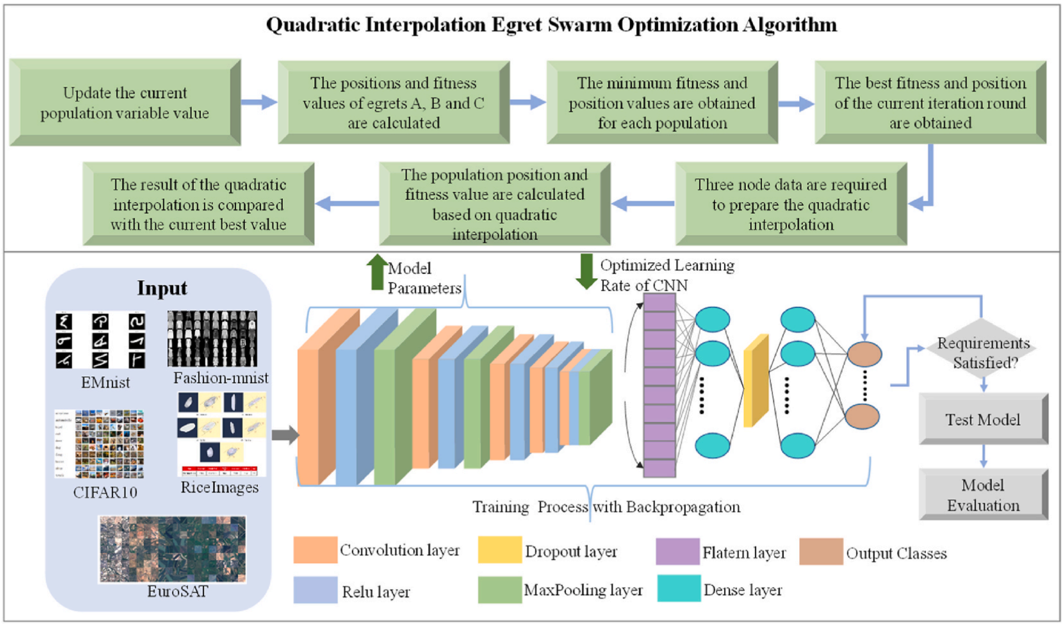


Fig. 3. Proposed QIESOA-based learning rate Optimised CNN model.

steps of the QIESOA process are as follows.

Step 1. The population parameters (*size_pop* is the number of populations, *max_iter* is the maximum number of iterations and *t* is the iteration counter) are set, initialising the population size and the position (the optimal position) x^0 at random.

Step 2. The basic information is calculated and updated according to the current position of the team, which mainly includes the current team fitness value, the best fitness value, the best position, the applied gradient direction, the best gradient direction, the global best fitness, the best gradient direction, the position and the parameter *w*.

Step 3. Strategy 1 updates the position of Egret A using Eqs. (7) and (8), strategy 2 updates Egret B usingEquation 12, and strategy 3 updates Egret C using Eq. (13).

Algorithm I: QIESOA

Input: *max_iter, size_pop*, $x^0 \in \mathbb{R}^{P\times N}$

Operation	**Cost**
/* Initialization */	
Initialize *max_iter=m, size_pop=n*, x^0	T_1
/* Training Start */	
/* Sitandwait Function */	
function Sitandwait(x)	T_2
The integrated gradient g is updated by Eq.(7)	
The weight ω of the observation equation is updated by Eq. (11)	
The desired position x_a of Egret A is calculated by Eq. (8)	
Search and obtain the fitness y_a of Egret A	
return x_a, y_a	
end function	
/* Aggressive Function */	
function aggressive(x)	T_3
The desired position x_b of Egret B is calculated by Eq. (12)	
The desired position x_c of Egret C is calculated by Eq. (13)	
Search and obtain the fitness y_b, y_c of Egret B and C	
Return x_b, x_c, y_b, y_c	
end function	
/* Discriminant Condition */	
while t < *max_iter* **do**	T_4
$x_a^t, y_a^t \leftarrow SitAndWait(x^t)$	
$x_b^t, x_c^t, y_b, y_c \leftarrow Aggressive(x^t)$	
Update new position by Eq.(15)	
Retrieve the fitness of the algorithm by Eq.(16)	
Get next position x^{t+1} via Eq.(14)	
end while	
/* Training Ends */	

Output: x_{best}, y_{best}

Step 4. The fitness of Egrets A, B and C were compared horizontally, and each group selected the one with the smallest fitness to record its position and fitness.

Step 5. Update the fitness *f* and position *p* of each current egret team.

Step 6. Update the best fitness *f* and position *p* of each group of egret squads.

Step 7. The best fitness *f* value and position *p* in this population in this iteration are calculated for the quadratic interpolation. Afterwards, three nodes of the quadratic interpolation are obtained: x_1 is the best population of the last iteration, x_2 is the best population in the current iteration and the global best is x_3.

Step 8. The predicted value of quadratic interpolation is compared with the optimal value of the population in this iteration, the optimal value is compared with the global optimal value and the global optimal value is updated.

Step 9. Determine whether the maximum iteration *max_iter* is reached at this time m; if not, return to *step 2*. Otherwise, the final iteration outputs the optimal position x_{best} and the corresponding fitness value y_{best}.

The computational complexity of the algorithm consists of four parts.

1) Initialisation

$$T_1 = \Theta(r_1), \tag{17}$$

where r_1 is the dimension of x^0.

2) The computational complexity of the sit-and-wait

$$T_2 = \Theta(size\_pop) \tag{18}$$

where size_pop is the population size.

3) The computational cost of the aggressive

$$T_3 = \Theta(2 \times size\_pop) \tag{19}$$

4) The computational cost of the discriminant condition

$$T_4 = \Theta(3 \times size\_pop \times \max\_iter) \tag{20}$$

From the above description, the computational cost of the QIESOA method is expressed as follows;

$$T_k = T_1 + T_2 + T_3 + T_4 \approx \Theta(r_1 + size\_pop \times \max\_iter) \tag{21}$$

4. Experiments

In this section, all experiments are based on CNN to verify the validity of the presented QIESOA algorithm.

4.1. General settings

This work adopts four evaluation metrics, namely, accuracy, precision, recall and f1-score, as follows;

$$\begin{aligned} accuracy &= \frac{TP + TN}{TP + TN + FP + FN} \\ precision &= \frac{TP}{TP + FP} \\ recall &= \frac{TP}{TP + FN} \\ f1 - score &= \frac{2 \times precision \times recall}{precision + recall} \end{aligned} \tag{22}$$

where *TP* indicates that the predicted value is 1, the actual value is 1 and the prediction is correct. *FP* indicates that the predicted value is 1, the actual value is 0 and the prediction is wrong. *FN* indicates that the predicted value is 0, the actual value is 1 and the prediction is wrong. *TN* indicates that the predicted value is 0, the actual value is 0 and the prediction is correct.

4.2. Datasets

There are five types of datasets: Fashion-MNIST [47], EMNIST (G. [48]), CIFAR10 (Moradi et al., 2019 [49]), EuroSAT (P. [50]) and Rice Image ([51]). We performed experiments using them and gave an overview of the case for datasets. Table 1 presents the details of the concerned datasets.

Fashion-MNIST: The fashion product dataset mainly contains 70,000 28 × 28 greyscale images of 10 categories, i.e. 7000 images per category. The 10 classes of images in the dataset comprised T-shirts, pants, pullovers, dresses, coats, sandals, shirts, sneakers, bags and ankle boots.

Extended MNIST: The dataset consists of a collection of images of handwritten digits at a 28 × 28 resolution, which is four times the size of the original MNIST dataset. The dataset has 62 classes; however, some letters are hard to distinguish between upper and

Table 1
Details of the concerned datasets.

Datasets	Categories	Sizes (M)	Image count
Fashion-MNIST	10	31	70,000
EMNIST	47	536	814,255
CIFAR10	10	163	60,000
EuroSAT	10	22,528	27,000
Rice Image	5	267	75,000

lower case handwriting; these letters are combined to form a new category, so the final 47 classes are left.

CIFAR10: This dataset contains 60,000 RGB colour images with a 32 × 32 resolution, divided into 10 classes with 6000 images in each category. The training and test sets of this dataset contain 50,000 and 10,000 images,

respectively. The images are labelled with 10 categories such as airplane, car (but not truck or pickup), bird, cat, deer, dog, frog, horse, boat and truck (but not pickup).

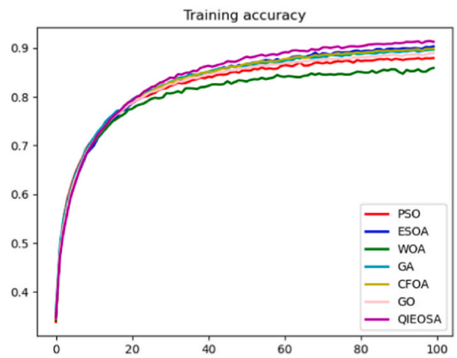
EuroSAT: The dataset is a Sentinel-2 satellite imagery, which consists of 27,000 labelled geographic images comprising 13 spectral bands and 10 categories. The image resolution is 64 × 64 pixels and has been verified. The 10 categories of this dataset are as follows: industrial buildings, residential buildings, annual crop, permanent crop, river, sea and lake, herbaceous vegetation, highway, pasture and forest.

Rice Image: The dataset typically includes images of rice plants at different growth stages, with various angles, lighting conditions and backgrounds. The dataset is often used to develop and evaluate algorithms for tasks such as plant classification, segmentation and feature extraction. The rice varieties used in the dataset are Arborio, Basmati, Ipsala, Jasmine and Karacadag.

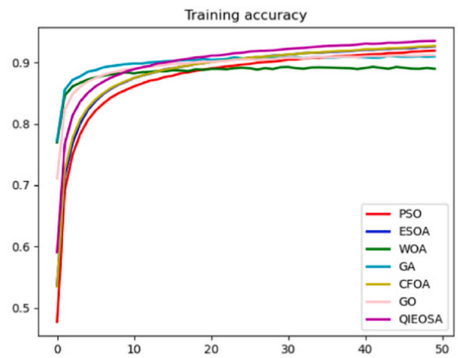
4.3. Baselines

In this paper, we mainly introduce several state-of-the-art image classification models that will be compared with the QIESOA algorithm. The compared algorithms are as follows.

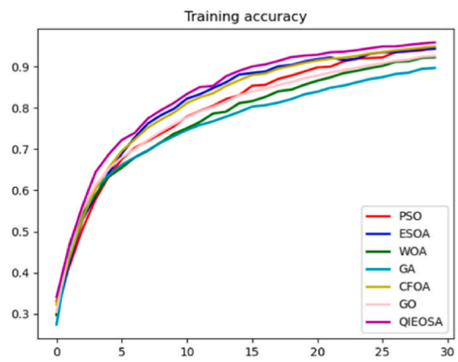
1) Particle Swarm Optimization (PSO): It is a stochastic search algorithm based on swarm cooperation, inspired by simulating the foraging behaviour of birds (C. [18]).
2) Genetic Algorithm (GA): It is a classical evolutionary computation algorithm ([34]).
3) Whale Optimization Algorithm (WOA): It is a swarm intelligence algorithm that is formed by simulating whale predation behaviour ([34]).



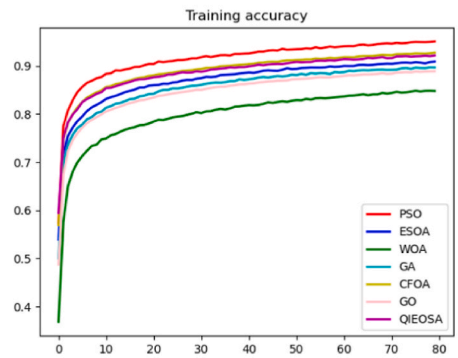
(a) Training CNN network on CFAR10



(b) Training CNN network on EMNIST



(c) Training CNN network on EuroSAT



(d) Training CNN network on Fashion-MNIST

Fig. 4. Accuracy comparison of different optimization datasets.

4) Catch Fish Optimization Algorithm (CFOA): It is a metaheuristic optimization algorithm that simulates the process of rural fishermen fishing in a pond [52].
5) GOOSE Algorithm (GO): Hamad R K et al. ([53]) proposed a novel metaheuristic algorithm based on the resting and foraging behaviour of geese. By adaptively adjusting the resolution of the search space and the search speed, the algorithm can find the optimal solution quickly and accurately.
6) Egret Swarm Optimization Algorithm (ESOA): It is a novel evolutionary computation method [17].

5. Results

5.1. Performance verification

For the image classification task of five datasets, different baseline algorithms are applied to compare the learning rate of our proposed QIESOA based on the CNN model. Fig. 4 presents the accuracy of the different models during the training phase. QIESOA method achieves the highest classification task accuracy in all four datasets presented. The CIFAR10, EMNIST, EuroSAT and Fashion-MNIST datasets undergo 100/50/30/80 epochs, respectively, and each baseline, such as the QIESOA algorithm, gradually stabilises. The Rice Image dataset is not listed in the figure because the relevant images in this dataset have obvious characteristics and clear classification. It is easy to achieve more than 95 % accuracy by running no more than five epochs, and the figure drawn is relatively monotone.

Tables 2–6 present the measurement information of various baseline models, such as QIESOA, on the five datasets. From the above tables, the training loss and test loss of the QIESOA algorithms yield the smallest and the best performance on other datasets except EMNIST. With respect to accuracy, the convergence during training wsas observed, as shown in Fig. 4, and Tables 2–6 show the accuracy for both the training and test sets. Similarly, except for the EMNIST dataset, the accuracy of QIESOA performs well on the other four datasets. In particular, on the Rice Image dataset, the test accuracy improved by 3.36

percentage points over the training accuracy. With respect to precision, recall and F1 score, we can see that similar good performance except for the EMNIST dataset is either in the best single metric performance, such as CIFAR10 and EuroSAT datasets, or in the best together with other model metrics, such as Fashion-MNIST, Rice Image datasets and even the slightly worse EMNIST data set are in the second level of three indicators. The algorithm sub-optimally performs on the EMNIST dataset, probably because letters and digits have a lot of overlap and repetition. In image classification tasks, the algorithm is a little less effective at learning and recognising patterns in images. Therefore, this warrants an optimization space for subsequent automatic learning or feature selection.

However, the algorithm has been trained and verified on different image classification datasets such as e-commerce, alphanum digital, nature and objects, remote sensing satellite images and agricultural crops and has achieved excellent performance and results in the evaluation indicators. The size of these datasets ranges from 31M to 22,528M, involving a wide range of fields. It shows that the algorithm not only has excellent generalisation ability but also has good results in stability and robustness.

5.2. Benchmark test function and results

Some representative unimodal functions, multi-modal functions, hybrid functions and combination functions are selected from the current popular CEC2017 and CEC2022 function libraries. QIESOA algorithm is compared with other state-of-the-art models using selected functions to evaluate the performance of each algorithm. Tables 7 and 8 present the benchmark functions of CEC2017 and CEC2022, respectively. Unimodal function can not only be used to test and compare the performance of the algorithm but also to compare the global optimal solution and convergence speed of the algorithm. Multi-modal functions allow us to evaluate how algorithms perform despite issues with multiple local optima and whether they can find global optima, which is used to verify the exploration ability of the algorithm.

For each model, we adopt the following metrics to measure its performance.

1) **Average value (*Ave*):** It represents the average of the solutions obtained by running the algorithm M times, which is computed according to;

Table 2
CIFAR10 dataset comparison model metrics data.

Baseline	Training Loss	Test Loss	Training Accuracy	Test Accuracy	Precision	Recall	F1 score
PSO	0.3572	0.3527	0.8789	0.8792	0.88	0.88	0.88
WOA	0.4444	0.4192	0.8514	0.8590	0.86	0.86	0.86
GA	0.3035	0.3001	0.8953	0.8965	0.9	0.9	0.9
CFOA	0.3084	0.2971	0.8982	0.9001	0.9	0.9	0.9
GO	0.3261	0.3640	0.8893	0.8705	0.87	0.87	0.87
ESOA	0.2762	0.2793	0.9020	0.9030	0.9	0.9	0.9
QIESOA	**0.2465**	**0.2464**	**0.9132**	**0.9125**	**0.91**	**0.91**	**0.91**

Table 3
EMNIST dataset comparison model metrics data.

Baseline	Training Loss	Test Loss	Training Accuracy	Test Accuracy	Precision	Recall	F1 score
PSO	0.2512	0.3396	0.9191	0.8953	**0.9**	**0.9**	**0.9**
WOA	0.2282	**0.3375**	0.9259	0.8983	**0.9**	**0.9**	**0.9**
GA	**0.1917**	0.3391	**0.9351**	**0.9012**	**0.9**	**0.9**	**0.9**
CFOA	0.2031	0.3477	0.9267	0.8944	0.89	0.89	0.89
GO	0.2340	0.3523	0.9224	0.8901	0.89	0.89	0.89
ESOA	0.3631	0.5356	0.8895	0.8675	0.87	0.87	0.87
QIESOA	0.2754	0.4283	0.9100	0.8866	0.89	0.89	0.89

Table 4
EuroSAT dataset comparison model metrics data.

Baseline	Training Loss	Test Loss	Training Accuracy	Test Accuracy	Precision	Recall	F1 score
PSO	0.2096	0.1548	0.9456	0.9453	0.95	0.94	0.94
WOA	0.2323	0.2184	0.9291	0.9226	0.93	0.92	0.93
GA	0.2171	0.2942	0.934	0.8968	0.93	0.93	0.93
GO	0.2333	0.2149	0.9260	0.9299	0.93	0.93	0.93
CFOA	0.2043	0.1897	0.9491	0.9345	0.93	0.93	0.93
ESOA	0.2023	0.1698	0.9485	0.9432	0.95	0.95	0.95
QIESOA	**0.1571**	**0.1220**	**0.9587**	**0.9587**	**0.96**	**0.96**	**0.96**

Table 5
Fashion-MNIST dataset comparison model metrics data.

Baseline	Training Loss	Test Loss	Training Accuracy	Test Accuracy	Precision	Recall	F1 score
PSO	0.2511	0.2304	0.9091	0.8953	0.92	**0.92**	**0.92**
WOA	0.4178	0.3478	0.8480	0.8983	0.88	0.88	0.88
GA	0.2854	0.2444	0.8966	**0.9012**	0.91	0.91	0.91
GO	0.3257	0.2641	0.9273	0.9068	0.91	0.90	0.91
CFOA	0.1956	0.2243	0.9217	0.9186	0.92	**0.92**	**0.92**
ESOA	0.2108	0.2243	0.9218	0.8675	0.92	**0.92**	**0.92**
QIESOA	**0.1278**	**0.2159**	**0.9509**	0.8866	**0.93**	**0.92**	**0.92**

Table 6
RiceImage dataset comparison model metrics data.

Baseline	Training Loss	Test Loss	Training Accuracy	Test Accuracy	Precision	Recall	F1 score
PSO	0.1860	0.0943	0.9326	0.9669	**0.97**	**0.97**	**0.97**
WOA	0.2193	0.1083	0.9210	0.9621	0.96	0.96	0.96
GA	0.2301	0.1140	0.9197	0.9614	0.96	0.96	0.96
GO	0.2512	0.1425	0.9191	0.9519	0.95	0.95	0.95
CFOA	0.1923	0.1101	0.9291	0.9612	0.96	0.96	0.96
ESOA	0.2164	0.0959	0.9246	0.9676	**0.97**	**0.97**	**0.97**
QIESOA	**0.1745**	**0.0831**	**0.9379**	**0.9715**	**0.97**	**0.97**	**0.97**

$$Ave = \frac{1}{M}\sum_{i=1}^{M} F_i \tag{23}$$

where F_i is the solution obtained by the ith time.

2) **Statistical standard deviation (*Std*):** It is used to measure the dispersion of the obtained solutions and is used to illustrate the stability and performance of the algorithm, which is calculated as follows;

$$Std = \sqrt{\frac{1}{M}\sum_{i-1}^{M} (\mathrm{F}_i - Ave)^2} \tag{24}$$

Table 7
CEC2017 test functions.

Type	ID	Range	Dimension	f_{min}
Unimodal	F1	[−100,100]	20	100
	F2	[−100,100]	20	200
Multi-modal	F3	[−100,100]	20	300
	F4	[−100,100]	20	400
	F5	[−100,100]	20	500
	F6	[−100,100]	20	600
	F7	[−100,100]	20	700
	F8	[−100,100]	20	800
	F9	[−100,100]	20	900
Hybrid	F10	[−100,100]	30	1000
	F11	[−100,100]	30	1100
	F12	[−100,100]	30	1200

Table 8
CEC2022 test functions.

Type	ID	Range	Dimension	f_{min}
Unimodal	F1	[−100,100]	10/20	300
Multi-modal	F2	[−100,100]	10/20	400
	F3	[−100,100]	10/20	600
	F4	[−100,100]	10/20	800
	F5	[−100,100]	10/20	900
Hybrid	F6	[−100,100]	10/20	1800
	F7	[−100,100]	10/20	2000
	F8	[−100,100]	10/20	2200
Composition	F9	[−100,100]	10/20	2300
	F10	[−100,100]	10/20	2400
	F11	[−100,100]	10/20	2500
	F12	[−100,100]	10/20	2700

3) **Best value (*Best*):** It is the best value selected from the *M* solutions, which is calculated as follows;

$$Best = \min_{1 \le i \le M} F_i \tag{25}$$

4) **Rank:** These algorithms are ranked according to their *Ave* and *Std.* The smaller the *Ave* and *Std*, the smaller the ranking value.

Hybrid functions absorb the advantages of unimodal and multi-modal functions and can be used to test and compare algorithms for different characteristic problems. Composite functions, which usually consist of several basic functions, have advantages in measuring the robustness and capability of an algorithm.

The dimensions of each benchmark function range from 20 to 30. Tables 9 and 10 present the consequences of the executions of CEC2017 and CEC2022, respectively. The convergence of different models is compared. Figs. 5 and 6 present the consequences.

6. Discussion and analysis

In this research, we propose an improved QIESOA algorithm and evaluate the impact of employing image classification outcomes. We choose six algorithms as benchmarks and perform multi-dimensional experiments to assessment accuracy, convergence and CEC function sets. Moreover, we conducte the following discussion and analysis.

As can be seen from relevant tables and figures of experimental results in Section 5, the QIESOA algorithm has shown excellent outcomes in terms of precision, accuracy and convergence. From the theoretical analysis, the algorithm innovation introduces quadratic interpolation in ESOA. This method can not only reduce the number of iterations, thereby make the algorithm converge quickly and achieve the goal of improving the prediction accuracy. The application of quadratic interpolation in (Z.B. [15]) is theoretically proved and analyzed about the Beetle Antennae search algorithm. Furthermore, on the grounds of the analysis of the specific experimental results, the metric data in Tables 2–6 show that QIESOA performs excellent except Table 3 in the EMNIST dataset. The training accuracy of QIESOA reaches the advantage of 2.36 in Table 5, while the testing accuracy achieves the consequence of 1.34 in Table 4. Simultaneously, the precision in Tables 2, 4 and 5 attains 1 improvement. We analyse from both theoretical and experimental perspectives, QIESOA plays a significant role in improving the results compared with baseline algorithms.

In the test metric values select by CEC2017 and CEC2022 function sets, QIESOA algorithm achieves the optimal value in at least one metric value among all comparison algorithms. For unimodal functions (CEC2017 F1, F2; CEC2023 F1), these results demonstrate that

Table 9
Comparison results for 12 benchmark test functions with CEC2017.

F		QIESOA	ESOA	CFOA	GO	PSO	WOA	GA
F1	*Ave*	**182026758**	223132557	847095658	498796343	741442029	456088400	977076306
	Std	**1997201907**	2213537690	4571242324	3117447267	4410793908	3049332268	3249383555
	Best	**16147.33369**	50084.1995	14988234.87	254716.0909	702079.607	738415.699	101527466
F2	*Ave*	**3954.182819**	7941.783927	9324.71335	10777.82066	10478.1089	14136.0214	9276.1742
	Std	**9445.022198**	15313.37494	19269.91978	15965.81853	10772.5743	17293.0916	11252.3512
	Best	**201.786262**	255.17314	484.0406052	280.080185	389.083148	2412.70141	1190.93233
F3	*Ave*	**382.2970846**	651.3883397	535.8137098	571.2777626	632.069447	495.554768	751.644757
	Std	**345.2924782**	3015.767342	393.6584906	1853.869639	543.880293	638.586443	581.162428
	Best	**304.7285309**	394.9745937	462.4838821	356.1803376	318.709904	400.142578	476.552001
F4	*Ave*	2184.388266	2320.625354	**1705.31245**	2935.51745	14747.7672	5084.47104	7495.69701
	Std	6672.192736	6859.606938	**3998.706707**	10974.19283	21821.5210	10694.9942	5324.85959
	Best	**473.1970224**	650.0535295	1030.607308	619.0842005	551.176639	1410.67479	1916.08499
F5	*Ave*	**500.0001234**	500.0021186	500.0094365	500.0027396	500.004771	500.008293	500.005378
	Std	**0.001140433**	0.003262539	0.001828762	0.006254644	0.00866510	0.0040725	0.00535858
	Best	**500.0000014**	500.0000071	500.00924	500.0000183	500.000087	500.005958	500.001910
F6	*Ave*	**4727.35426**	5518.761457	20892.6541	6318.771399	8624.71605	4054.70495	2577.46187
	Std	**7659.5123**	15024.01775	33516.56624	11054.89291	9816.94452	5364.53866	7803.93886
	Best	1857.685568	**727.2273394**	12271.47383	747.4054363	3575.33421	2744.91016	935.664118
F7	*Ave*	**700.0573184**	700.7843096	701.0663813	700.2878148	701.521508	700.986568	700.804796
	Std	**0.221845604**	0.813558447	0.402837541	0.63853062	1.83160654	0.5596701	1.0522844
	Best	**700.0157765**	700.1967714	700.4914364	700.0315444	700.105471	700.601877	700.472282
F8	*Ave*	**806.3375099**	806.8858218	815.3650234	806.4401112	817.2876587	813.214351	810.6761803
	Std	5.815702687	5.158390445	0.942654867	1.43681163	14.18820285	13.35239251	8.737068342
	Best	**802.0401449**	803.0527831	815.1756558	806.1227615	805.0043035	807.1589281	804.3473558
F9	*Ave*	**2239.502543**	2440.019524	2942.970593	2974.200652	3401.802194	2545.018321	2425.913024
	Std	992.7691992	960.1913247	1015.437543	**408.8007418**	1287.354224	797.5147531	840.1895802
	Best	**1280.414435**	1632.81028	1854.991328	2633.434144	2221.953381	1969.64354	1699.571192
F10	*Ave*	**1404287.905**	9915083.171	46324388.02	28669375.82	86828797.98	25958838.52	85291663.21
	Std	**283299.0051**	90909889.05	215756404.8	100003615.3	337342754.2	109184521.8	838960372
	Best	61902.74724	90945.76239	**60925.71474**	170118.5508	109652.9046	151326.1782	267671.7642
F11	*Ave*	**145969251.3**	222011261.2	187726373.8	188796272.4	790511121.2	201139591.3	688557059.1
	Std	1048575473	2256893696	**645360614.9**	1171927248	3067037803	1371321487	1884466904
	Best	**179599.8396**	217758.658	83312100.72	2650536.946	3887188.37	21524333.86	41739001.1
F12	*Ave*	**204241856.9**	203843457.9	675228972.5	206652274.7	962904284.7	325216566.1	375989427.6
	Std	4100285583	2074496947	519887018.2	**1873990804**	3837441298	1984844092	1334321259
	Best	**38966.36385**	82726.18384	473204272.3	148962.6108	187241.1752	55571494.65	51259544.03
Rank		1	4	2	3	5	5	5

the QIESOA algorithm has significant advantages over other algorithms. The QIESOA algorithm can obtain two optimal solutions. Moreover, it proves the good exploitation ability of the QIESOA algorithm. For multi-modal functions, the results of the QIESOA algorithm are also superior to other algorithms. The QIESOA algorithm finds 10 optimal solutions obtained in 11 functions, except F6 for CEC2017. The results indicate the exploration ability of our propose algorithm to other algorithms. For hybrid functions, QIESOA achieves the optimal value in two of five functions, and the performance fails to meet expectations. This result demonstrates that the algorithm is less competent in catching and processing features, which in part explains why our algorithm performs slightly worse than other algorithms on the EMNIST dataset.

We also perform some experiments on combination functions, and all of them produce optimal solutions. This proves that the QIESOA algorithm can handle complex problems and exhibits strong robustness. Based on the CEC2017 and CEC2022 function sets, the convergence of different models is compared, as shown in Figs. 5 and 6. From these figures, we could obtain the following findings. QIESOA algorithm acquires fast convergence rates, except for the F6 function of CEC2017, which are lower than those of GO, GA and ESOA algorithms. For multi-modal functions, the QIESOA algorithm can converge faster and find the global optimal solution expect F6 of CEC2017. In CEC2017, functions F5 and F7 converge quickly within 10 iterations, whereas functions in CEC2022 all converge quickly within 100 iterations. This result demonstrates again that QIESOA has a strong exploration ability and can optimise the learning rate of CNN to obtain the optimal solution as soon as possible. With respect to the hybrid function of the QIESOA algorithm, only F7 and F8 of CEC2022 rapidly converged after 50 iterations. This conclusion is consistent with the indicator data performance, as shown in Tables 9 and 10. As regards the combination function, the QIESOA algorithm quickly converged after 75 iterations; particularly, F10 and F11 of CEC2022 have completed convergence after 25 iterations. Moreover, this demonstrates that the ability of QIESOA ability to deal with complex problems can efficiently and promptly obtain the global optimal CNN model learning rate, thereby improving the accuracy of classification tasks.

7. Conclusions

Herein, we present a hyperparameter tuning method of the QIESOA algorithm for multi-domain image classification, and it is successfully applied to the complex image classification task. Furthermore, the existing optimal solution is used for quadratic

Table 10
Comparison results for 12 benchmark test functions with CEC2022.

F		QIESOA	ESOA	CFOA	GO	PSO	WOA	GA
F1	*Ave*	**2995.116023**	5409.239958	6310.289204	7399.184156	8364.215561	9995.243294	8735.041163
	Std	**6675.118439**	7480.848746	8534.728117	11465.04217	11279.25592	10183.3218	13311.74447
	Best	**302.3584249**	314.120087	1599.379506	836.4200032	987.8770137	1866.42524	1939.745993
F2	*Ave*	**1224.584197**	1466.56564	1842.566015	1378.80794	2224.652577	2199.627359	2763.698087
	Std	5268.333861	4130.151714	1996.964319	**916.3132871**	1825.114498	2417.503205	2341.519131
	Best	**610.3739276**	616.8365255	837.4310175	937.2152921	745.9963494	912.8719134	943.6134886
F3	*Ave*	**600.3762531**	600.5427594	600.3839179	600.4190004	602.1569585	600.5772773	601.1408014
	Std	0.680602363	0.673595313	**0.556341776**	0.577904911	0.633290781	0.590828357	0.544291831
	Best	**600.0000437**	600.1138412	600.1244553	600.1929153	601.4830114	600.2846528	600.456946
F4	*Ave*	**800.6740117**	803.4114761	801.7581253	803.6811167	805.5058841	804.2949442	805.2478535
	Std	0.65774571	1.520099131	0.473633757	1.330268726	0.854537004	0.6541494	0.685616052
	Best	**800.4686878**	802.2816275	801.613766	802.236901	804.3369701	803.8248166	804.1595662
F5	*Ave*	**901.8495836**	902.0838428	903.1435925	908.7267918	913.8773049	919.7731637	911.4621259
	Std	3.432309871	3.064651116	2.981053757	**2.241908936**	7.675882474	4.567529217	7.551524035
	Best	**900.0204357**	900.8113041	902.541623	908.1748112	906.3778477	917.2546545	906.7134706
F6	*Ave*	**111221670.4**	144079976.4	39144715.28	91342226.51	572630636.6	68399160.85	153197109.8
	Std	**891416644.5**	2470045717	245158323.5	2069729315	1847417320	561678576.2	756121243.9
	Best	**78741.12817**	96890.27857	2920685.978	76029.26302	105680.0023	155478.2557	8071691.487
F7	*Ave*	**2587.212528**	3075.175865	3799.734016	2612.976965	6021.182644	3371.65245	4130.946995
	Std	3561.951965	2992.097273	1812.824757	**1283.426544**	2082.85278	1800.582175	2282.351486
	Best	2116.616751	2288.57388	2552.483741	**2030.573651**	3445.732763	2711.575142	2463.338235
F8	*Ave*	4.01E + 17	1.20E + 17	**4816066033**	3.2877E + 11	9.48395E + 12	5.63243E + 12	3.04952E + 11
	Std	8.02E + 18	2.40E + 18	**32200476172**	2.8699E + 12	1.17076E + 14	6.76117E + 13	1.69469E + 12
	Best	2479.856042	2982.720339	**24275.03274**	4905.901501	41528.93421	3839.188703	5224783.365
F9	*Ave*	**2760.113334**	2864.486965	3138.312545	3146.103282	3334.06997	3127.570746	3225.076182
	Std	**549.126721**	697.0997489	1061.84333	1055.045231	1017.832234	464.1211144	966.9341616
	Best	**2608.809577**	2677.130766	2773.421023	2791.698676	2842.840088	2838.266656	2886.591941
F10	*Ave*	**3003.70818**	3004.798183	5562.763816	5494.847895	3266.77894	3445.064915	6396.560584
	Std	972.0107911	822.649193	1790.558446	2142.207689	1385.454885	1326.943133	**655.6118583**
	Best	**2775.009436**	2781.649178	2874.379757	2889.676739	2832.599443	2843.914581	5719.268198
F11	*Ave*	**2758.484366**	3098.506071	3454.486778	2950.70386	3194.209337	4766.575305	3863.409001
	Std	577.6713248	1630.351316	1530.56192	**502.6596555**	1182.166472	1710.397385	1862.315101
	Best	**2601.128799**	2611.396657	2700.841147	2671.489209	2682.000624	2901.038202	2697.728201
F12	*Ave*	**3076.753647**	3087.694074	3695.960448	3632.650802	4158.040093	3338.128529	3273.036066
	Std	304.0323297	219.5529385	**28.01424488**	163.8322992	266.011952	194.2467647	270.7003303
	Best	**2992.393029**	2996.451435	3689.279672	3527.547954	3751.832328	3135.929928	3109.251153
Rank		1	4	2	2	4	4	3

interpolation to promote the computational efficiency and precision of image classification tasks. The proposed algorithm is executed in the CNN model, and its learning rate is optimised. Afterwards, the optimised model is applied to the image classification task on five public multi-domain image datasets, and the accuracy rate is up to 97.15 %. However, the following issues remain.

a) The deep learning model has a variety of different hyperparameters; thus, it is extremely urgent and realistic to make full use of the swarm intelligence algorithm to optimise the multi-hyperparameters.
b) For problems and tasks in different fields, a variety of different networks are combined to construct a joint learning function, and the model is optimised to adapt to different situations, which has a great room for a wide application of the model.

Data availability statement

The authors confirm that the data supporting the findings of this study are available within the article or its supplementary materials.

Funding

This research was funded by Sichuan University students innovation and entrepreneurship training program (S202410621082), Chengdu University of Information Technology key project of education reform (JYJG2023212).

CRediT authorship contribution statement

Peiyang Wei: Writing – original draft, Resources, Methodology, Conceptualization. **Mingsheng Shang:** Project administration, Investigation. **Jiesan Zhou:** Validation, Software, Data curation. **Xiaoyu Shi:** Writing – review & editing, Supervision, Formal analysis.

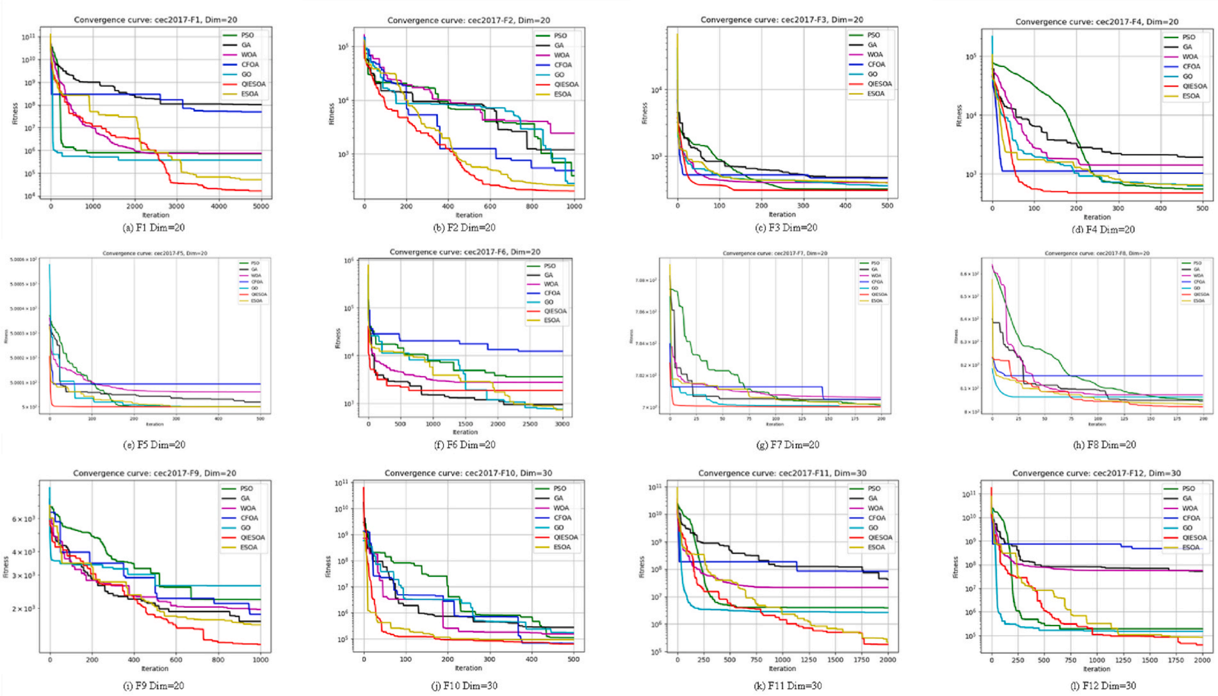


Fig. 5. Convergence curves for some typical functions with CEC2017.

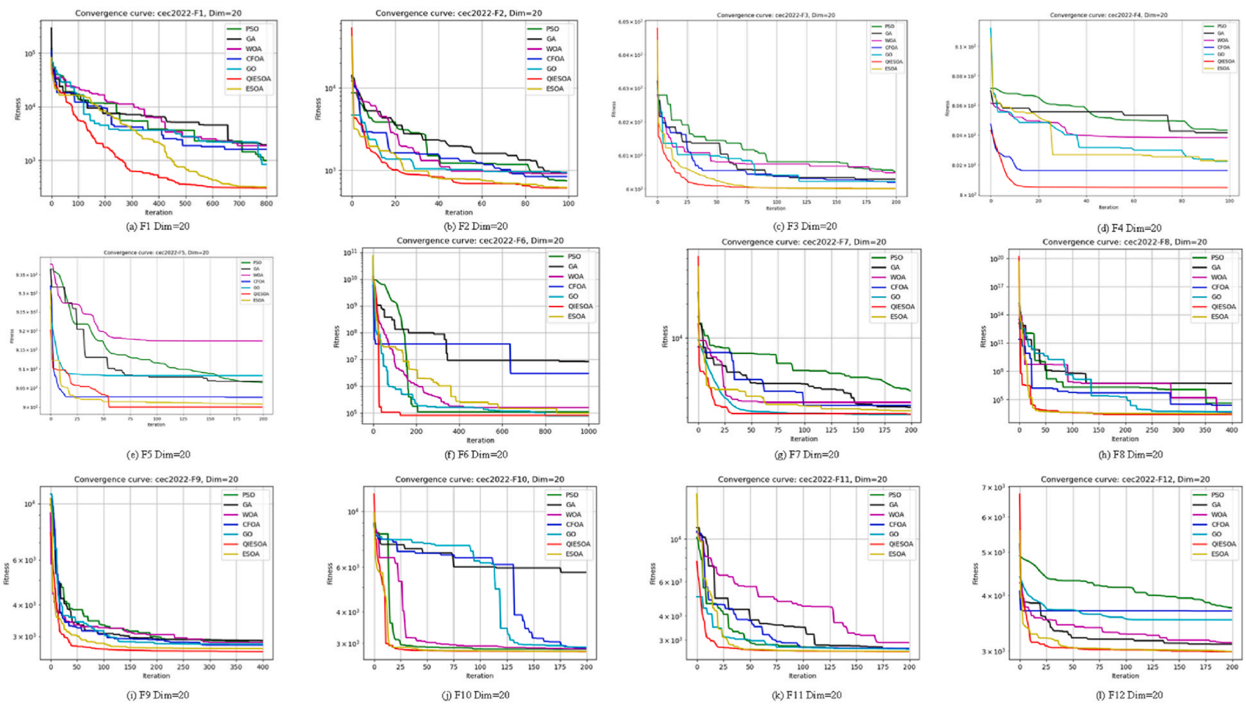


Fig. 6. Convergence curves for some typical functions with CEC2022.

Declaration of competing interest

We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled, "An efficient adaptive learning rate for Convolutional Neural Network based on Quadratic Interpolation Egret Swarm Optimization Algorithm".

References

[1] Yuanyuan Li, Q. Zhang, S.W. Yoon, Gaussian process regression-based learning rate optimization in convolutional neural networks for medical images classification, Expert Syst. Appl. 184 (12) (2021) 115357.1–115357.8.
[2] W. Wang, Y. Chen, P. Ghamisi, Transferring CNN with adaptive learning for remote sensing scene classification, IEEE Trans. Geosci. Rem. Sens. 60 (5533918) (2022) 1–18.
[3] Arunabha M. Roy, A multi-scale fusion CNN model based on adaptive transfer learning for multi-class MI-classification in BCI system, bioRxiv 17 (481909) (2022) 1–18.
[4] Z. Yi, X. Cao, X. Pu, et al., Fraud detection in capital markets: a novel machine learning approach, Expert Syst. Appl. 231 (120760) (2023) 1–10.
[5] M. Zhong, J. Wen, J. Ma, et al., A hierarchical multi-leadership sine cosine algorithm to dissolving global optimization and data classification: the COVID-19 case study, Comput. Biol. Med. 164 (107212) (2023) 1–48.
[6] Ali Mohammad Alqudah, Hiam Alquraan, Isam Abu Qasmieh, Segmented and non-segmented skin lesions classification using transfer learning and adaptive moment learning rate technique using pretrained convolutional neural network, Journal of biomimetics, biomaterials and biomedical engineering 42 (2019) 67–78.
[7] R. Mohakud, R. Dash, Designing a grey wolf optimization based hyper-parameter optimized convolutional neural network classifier for skin cancer detection, Journal of King Saud University-Computer and Information Sciences 34 (8) (2022) 6280–6291.
[8] Z.J. Wang, H.B. Gao, X.H. Wang, et al., Adaptive learning rate optimization algorithms with dynamic bound based on Barzilai-Borwein method, Inf. Sci. 634 (2023) 42–54.
[9] Haibo Huang, et al., Optimization of electric vehicle sound package based on LSTM with an adaptive learning rate forest and multiple-level multiple-object method, Mech. Syst. Signal Process. 187 (2023) 109932.
[10] X. Zhai, F. Qiao, Y. Ma, H. Lu, A novel fault diagnosis method under dynamic working conditions based on a CNN with an adaptive learning rate, IEEE Trans. Instrum. Meas. 71 (5013212) (2022) 1–12.
[11] S. Tang, Y. Zhu, S. Yuan, An improved convolutional neural network with an adaptable learning rate towards multi-signal fault diagnosis of hydraulic piston pump, Adv. Eng. Inf. 50 (101406) (2021) 1–19.
[12] M.G. Ragab, S.J. Abdulkadir, N. Aziz, Q. Al-Tashi, Y. Alyousifi, H. Alhussian, A. Alqushaibi, A novel one-dimensional CNN with exponential adaptive gradients for air pollution index prediction, Sustainability 12 (10090) (2020) 1–22.
[13] S. Alshmrany, Adaptive learning style prediction in e-learning environment using levy flight distribution based CNN model, Cluster Comput. 25 (1) (2021) 523–536.
[14] K. Chen, K. Zhu, M. Meng, Image-enhanced adaptive learning rate handwritten vision processing algorithm based on CNN, in: 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP), Wuxi, China, Oct. 2019, pp. 112–116.
[15] Tianyi Li, Yanmei Liu, Zhen Chen, Application of sine cosine egret swarm optimization algorithm in gas turbine cooling system, Systems 10 (6) (2022) 201.
[16] John Duchi, Elad Hazan, Yoram Singer, Adaptive subgradient methods for online learning and stochastic optimization, J. Mach. Learn. Res. 12 (7) (2011) 2121–2159.
[17] Z. Chen, A. Francis, S. Li, B. Liao, D. Xiao, T.T. Ha, J. Li, L. Ding, X. Cao, Egret swarm optimization algorithm: an evolutionary computation approach for model free optimization, Biomimetics 7 (144) (2022) 1–34.
[18] C. Mao, S. Li, Z. Chen, X. Zhang, C. Li, Robust kinematic calibration for improving collaboration accuracy of dual-arm manipulators with experimental validation, Measurement 155 (2020) 107524.
[19] Z.B. Li, S. Li, X. Luo, Using quadratic interpolated Beetle Antennae search to enhance robot arm calibration accuracy, IEEE Rob. Autom. Lett. 7 (4) (2022) 12046–12053.
[20] Z. Li, S. Li, X. Luo, An overview of calibration technology of industrial robots, IEEE/CAA J. Autom. Sinica. 8 (1) (2021) 23–36.
[21] Y. Li, J. Xiao, Y. Chen, et al., Evolving deep convolutional neural networks by quantum behaved particle swarm optimization with binary encoding for image classification, Neurocomputing 362 (2019) 156–165.
[22] Christian Szegedy, et al., Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA, Jun, 2015, pp. 1–9.
[23] G. Huang, Z. Liu, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017, pp. 4700–4708.
[24] Andrew G. Howard, et al., Mobilenets: efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv (2017).
[25] B. Ma, et al., Autonomous deep learning: a genetic DCNN designer for image classification, Neurocomputing 379 (2020) 152–161.
[26] T. Takase, S. Oyama, M. Kurihara, Effective neural network training with adaptive learning rate based on training loss, Neural Network. 101 (2018) 68–78.
[27] J. Park, D. Yi, S. Ji, A novel learning rate schedule in optimization for neural networks and it's convergence, Symmetry 12 (660) (2020) 1–16.
[28] E. Yang, et al., AdaTask: a task-aware adaptive learning rate approach to multi-task learning, Proc. AAAI Conf. Artif. Intell. 37 (9) (Jun.2023) 10745–10753. Washington DC, USA.
[29] Sunwoo Lee, Chaoyang He, Salman Avestimehr, Achieving small-batch accuracy with large-batch scalability via Hessian-aware learning rate adjustment, Neural Network. 158 (2023) 1–14.
[30] Z. Liu, et al., Channel-Spatial attention convolutional neural networks trained with adaptive learning rates for surface damage detection of wind turbine blades, Measurement 217 (2023) 113097.
[31] Antônio David Viniski, et al., Incremental specialized and specialized-generalized matrix factorization models based on adaptive learning rate optimizers, Neurocomputing 552 (126515) (2023).
[32] X. Yang, R. Wang, D. Zhao, et al., An adaptive quadratic interpolation and rounding mechanism sine cosine algorithm with application to constrained engineering optimization problems, Expert Syst. Appl. 213 (119041) (2022).
[33] Trojovská Eva, Mohammad Dehghani, Víctor Leiva, Drawer Algorithm, A new metaheuristic approach for solving optimization problems in engineering, Biomimetics 8 (239) (2023).
[34] S.R. Spea, Social network search algorithm for combined heat and power economic dispatch, Elec. Power Syst. Res. 221 (109400) (2023).
[35] X. Hu, S. Wen, H.K. Lam, Dynamic random distribution learning rate for neural networks training, Appl. Soft Comput. 124 (109058) (2022).
[36] J.I. Sankar, S. Subbaraman, Multi-converter UPQC optimization for power quality improvement using Beetle swarm-based butterfly optimization algorithm, Elec. Power Compon. Syst. 51 (4) (2023) 1–12.
[37] K. Mehmood, et al., Variants of chaotic grey wolf heuristic for robust identification of control autoregressive model, Biomimetics 8 (2) (2023) 141.
[38] M. Qaraad, S. Amjad, N.K. Hussein, et al., Quadratic interpolation and A new local search approach to improve particle swarm optimization: solar photovoltaic parameter estimation, Expert Syst. Appl. 236 (121417) (2024).

[39] Y. Sun, T. Yang, Z. Liu, A whale optimization algorithm based on quadratic interpolation for high-dimensional global optimization problems, Appl. Soft Comput. 85 (105744) (2019).
[40] G. Moustafa, et al., An enhanced Dwarf Mongoose optimization algorithm for solving engineering problems, Mathematics 11 (15) (2023) 3297.
[41] Y. Liu, et al., An optimization capacity design method of household integrated energy system based on multi-objective egret swarm optimization, IET Renew. Power Gener. 17 (8) (2023) 1993–2008.
[42] Abrar M. Alajlan, Abdul Razaque, ESOA-HGRU: egret swarm optimization algorithm-based hybrid gated recurrent unit for classification of diabetic retinopathy, Artif. Intell. Rev. 56 (Suppl 2) (2023) 1617–1646.
[43] K. Manivannan, S. Sathiamoorthy, Automated Tuberculosis classification using Egret Swarm Optimization with deep learning based fusion model on chest X-ray images, J. Intell. Fuzzy Syst. 45 (6) (2023) 10335–10347.
[44] X. Shi, et al., Relieving popularity bias in interactive recommendation: a diversity-novelty-aware reinforcement learning approach, ACM Trans. Inf. Syst. 42 (2) (2023) 1–30.
[45] W. Guo, et al., Improved sine cosine algorithm combined with optimal neighborhood and quadratic interpolation strategy, Eng. Appl. Artif. Intell. 94 (103779) (2020).
[46] X. Shi, Q. He, X. Luo, Y. Bai, M. Shang, Large-Scale and scalable latent factor analysis via distributed alternative stochastic gradient descent for recommender systems, IEEE Transactions on Big Data 8 (2) (2022) 420–431.
[47] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, arXiv preprint arXiv:1708.07747 (2017).
[48] G. Cohen, S. Afshar, J. Tapson, s A. van Schaik, EMNIST: extending MNIST to handwritten letters, 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA (2017) 2921–2926.
[49] Moradi Reza, Reza Berangi, Behrouz Minaei, Sparsemaps: convolutional networks with sparse feature maps for tiny image classification, Expert Syst. Appl. 119 (2019) 142–154.
[50] P. Helber, B. Bischke, A. Dengel, D. Borth, EuroSAT: a novel dataset and deep learning benchmark for land use and land cover classification, IEEE J. Sel. Top. Appl. Earth Obs. Rem. Sens. 12 (7) (2019) 2217–2226.
[51] Koklu Murat, Ilkay Cinar, Yavuz Selim Taspinar, Classification of rice varieties with deep learning methods, Comput. Electron. Agric. 187 (106285) (2021).
[52] H. Jia, Q. Wen, Y. Wang, et al., Catch Fish Optimization Algorithm: a New Human Behavior Algorithm for Solving Clustering Problems, Cluster Computing, 2024, pp. 1–38.
[53] R.K. Hamad, T.A. Rashid, GOOSE algorithm: a powerful optimization tool for real-world engineering challenges and beyond, Evolving Systems 15 (2024) 1249–1274.