JAISCR, 2025, Vol. 15, No. 2, pp. 167 – 195
10.2478/jaiscr-2025-0009

NOVEL METAHEURISTIC ALGORITHMS AND THEIR APPLICATIONS TO EFFICIENT DETECTION OF DIABETIC RETINOPATHY

Mahmoud Hassaballah[1,2,*] , Mohamed Abdel Hameed[3]

[1] *Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, AlKharj, 16278, Saudi Arabia*

[2] *Department of Computer Science, Faculty of Computers and Information, South Valley University, Qena, Egypt*

[3] *Department of Computer Science, Faculty of Computers and Information, Luxor University, Luxor, 85951, Egypt*

[*] *E-mail:mah.ali@psau.edu.sa*

Submitted: 12th September 2024; Accepted: 31st October 2024

Abstract

It is an extremely important to have an AI-based system that can assist specialties to correctly identify and diagnosis diabetic retinopathy (DR). In this study, we introduce an accurate approach for DR diagnosis using machine learning (ML) techniques and a modified golf optimization algorithm (mGOA). The mGOA optimizes ML classifiers through finding the best available parameters with respect to objective functions, hence decreases the number of features and increases the classifier's accuracy. A fitness function is employed to minimize the feature number of the medical dataset. The obtained results showed superiority of the mGOA with higher convergence speeds without extra processing costs across the datasets compared with several competitors. Also, the mGOA attained maximum accuracy and optimally reduced the number of features in the binary and multi-class datasets achieving the best CEC'2022 benchmark results compared with other metaheuristic algorithms. Based on this findings, three optimized ML classifiers called mGOA-SVM, mGOA-radial SVM,and mGOA-kNN were introduced as tools for classification of diabetic retinopathy disease and their performance was assessed on Messidor and EyePACS1 datasets. Experimental results demonstrated that mGOA-SVM and mGOA-radial SVM achieved remarkable accuracy in classification of DR disease with an average accuracy of 98.5% and precision of 97.4%.

Keywords: Machine learning, Metaheuristic optimization, Golf optimization algorithm, Diabetic retinopathy diagnosis

1 Introduction

Artificial Intelligence (AI) and Soft Computing are increasingly important in healthcare sector [1, 2, 3], especially for detecting diabetic retinopathy (DR), a major cause of vision impairment. AI, particularly through machine learning with some optimization can process retinal images to detect early signs of DR, such as microaneurysms and hemorrhages, with notable precision. These systems learn from extensive image datasets, enabling faster and more accurate diagnoses. Soft computing methods, including fuzzy logic and neural networks, are useful for managing the uncertainties often present in medical data. By combining AI and soft computing, healthcare professionals can identify DR earlier, allowing for prompt treatment and better patient care outcomes [4, 5, 6].

DR is a kind of serious diseases that impacts human eyes. Diabetes is becoming more common, with considerable variances across racial groups around the world [7, 8]. DR is defined by retinal damage in persons who have had diabetes mellitus for long time [9]. This disease is discovered with a systematic examination of the eye's fundus. DR begins with no symptoms, but as it progresses, it may result in gradual loss of vision acuity and even blindness. The first indications of DR is microaneurysms on the retina caused by irregular blood flow from retinal arteries. Microaneurysms are small, round, red dots that appear on the retina spots with sharp edges. When blood vessels are damaged due to aberrant swelling, bleeding results in hemorrhage, which is similar to microaneurysms but bigger and more noticeable on the retina picture. More capillary leakage can result in exudates, which are another characteristic feature of diabetic retinopathy. They are yellowish or white deposits that accumulate in the retina due to leakage from damaged blood vessels. Exudates result from the buildup of lipids (fats) and proteins that leak from blood vessels and collect in the retinal tissue. Figure 1 depicts the basic differences between the normal healthy retina and the retina with DR [10, 11].

Regular eye exams are crucial for people with diabetes because early detection and treatment can prevent or slow the progression of diabetic retinopathy. Thus, early diagnosis of DR is crucial for preventing irreversible damage and vision impairment [12]. Traditional diagnosing depends mainly on physicians' experiences, which might be subjective to some extent. Currently, diagnosing DR is a critical process that requires expert doctors to identify and analyze digital color fundus images of the retina. Typically, physicians provide their review results within a day or a few days. Delayed findings will cause missed clinical follow-up, inadequate communication, and delayed therapy. Early detection of this illness can prevent blindness and restore retinal function. The major problem in DR severity categorization is that fundus disease images have higher classification accuracy than other image categories. It demonstrates that the changes in DR lesion locations between the main neighboring groups are quite modest and difficult to differentiate between [13].

Existing manual approaches for recognize diabetic retinopathy are time-consuming because of involving large quantity of data and these conventional approaches have a high misclassification rate [14, 15]. In recent years, there are some AI-based approaches that have been developed for diagnosing DR. Those approaches are mainly utilized machine learning (ML) classifiers, such as K-Nearest Neighbor (KNN) and Support Vector Machines (SVM),...etc. to diagnose DR and its related symptoms [16]. Also, deep learning models were considered for predicting diabetic retinopathy (DR) in clinical endocrine practice [17]. Experiments show that machine learning based systems often achieve good accuracy rates. However, recall rates are modest, particularly in categories with limited training data. Machine learning-based research may lack the necessary network depth and scale to capture abstract medical aspects. Deep learning-based research can be time-consuming, particularly when training hyper-parameters for relevant networks. In this study, we propose a new approach for diagnosing and detecting DR by combining ML models and modified optimization technique.

Generally, gradient-based optimization algorithms are frequently unable to solve optimization problems with several local optimums of nonlinearity [18]. Meta-heuristic techniques have evolved to solve complicated optimization concerns with simple logic structures, good quality, efficiency, and low computing costs [19, 20]. A growing number of excellent meta-heuristic optimization techniques are available, such as Archimedes Optimization Algorithm (AOA) [21], Hunger games search algorithm [22], Grey Wolf Optimizer (GWO) [23], Harris Hawks Optimization[24], Marine Predators Algorithm [25], Runge Kutta optimization algorithm (RUN) [26], Golf Optimization Algorithm (GOA) [27] and so on. A wide range of issues have also been addressed by them, including dynamic multi-objective optimization [28], feature selection [29, 30], feed-forward neural networks [31], multi-objective optimization [32, 33], and large-scale complex optimization and scheduling optimization [34]. In this regard, a metaheuristic optimization model may even come close to precise optimal solutions for some particular situations [35].

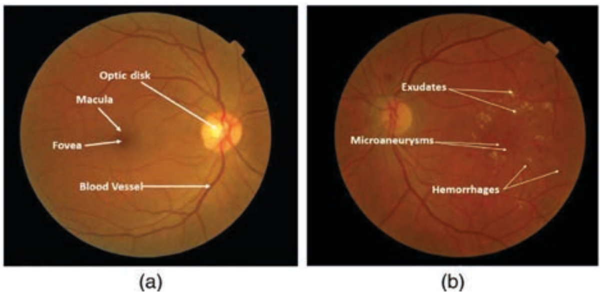

Figure 1. Illustration examples of (a) Healthy retina and (b) Retina affected with DR.

The innovation and contribution novelty of this work focus on developing a new metaheuristic algorithm called modified Golf Optimization Algorithm (mGOA) for optimizing searching parameters in ML classifiers as well as other optimization problems. The performance of mGOA algorithm is confirmed through several experiments and to the best of our knowledge there no similar algorithm inspired by the game of golf has been introduced so far. The main contributions of this study are as follows: The main contributions of this study are as follows:

1. A novel metaheuristic model, modified Golf Optimization Algorithm (mGOA) is introduced by combining genetic operators with the original GOA algorithm. The mGOA is hybridized with SVM to perform feature selection and classification. This reduces the total number of features required for the medical dataset while enhancing model's accuracy.

2. A fitness function is employed within mGOA to minimize selected number of features. So, the feature selection is optimized which result more accurate for mGOA model. The proposed mGOA is evaluated against the CEC'2022 benchmark and a medical dataset. So, the obtained results (statistical and visualization) demonstrated mGOA's superiority with faster convergence speeds compared to competitors without additional processing cost in both binary and multi-class datasets.

3. As an optimization algorithm, The proposed mGOA outperforms other metaheuristic algorithms including AOA, GWO, HHO, HGS, MPA, RUN, and GOA based on CEC'2022 benchmark results.

4. An Effective DR detection: The mGOA-SVM and mGOA-Radial SVM hybrid model demonstrates good performance in DR classification tasks, according to experiments conducted using detection DR datasets. More specifically, our approach performs optimally on levels 0,2,3, and 4.

5. For Diabetic retinopathy classification, high performance is achieved using the mGOA algorithm with ML classifiers on Messidor and EyePACS1 datasets for classification models SVM, Radial-SVM and K-NN, respectively.

The reminder parts of this work are organized as follows: The previous studies are discussed in literature review Section 2. The baseline golf optimization algorithm is presented in Section 3. After that, the proposed modified golf optimization algorithm (mGOA) and the optimized ML classifiers are introduced in Section 4. Experimental results, analysis and discussion are presented in Section 5. Finally, concluding remarks are given in Section 6.

2 Literature Review

Existing works on DR diagnosis are roughly classified into to two main categories: Machine learning based approaches and deep learning based approaches. The works which started in using metaheuristic optimization algorithms for the DR diagnosis are also discussed.

2.1 ML-based Diabetic Retinopathy Diagnosis

Several researchers have proposed machine learning (ML) algorithms as a way of detection and classification of DR. A few number of retinal datasets were used for the training and assessment of ML models. Early detection of DR can be realized through the automated AI approaches, including supervised and unsupervised ML approaches. Here are discussions of recent studies based on ML algorithm:

An approach for identifying diabetic retinopathy using machine learning techniques was demonstrated by Roychowdhury et al. [16]. The employed ML models are the K-NN, SVM, AdaBoost, and Gaussian mixture model (GMM). The MESSIDOR dataset's 1200 retinal pictures were used in the tests. The results revealed that the AUC was 0.9 and the specificity was 53.16% however, the main problem is the high rate of false positives in the results. While, a random forests (RF) classifier-based method for detecting DR was presented by Casanova et al. [36]. A retinal dataset consisting of 3443 eye-study photos was utilized. Their method yielded an accuracy rate of 90% without using of feature extraction or segmentation techniques. In addition, Kang et al. [37] proposed the Naïve Bayes (NB) classifier for detecting diabetic retinopathy. They employed feature extraction process using NB classifier including co-occurrence matrix, statistical texture features, and gray-level run-length texture analysis. The China dataset's fundus pictures of diabetic retinopathy (a total of 568 images) were classified using the trained model with accuracy of 93.44%.

Senapati et al. [38] suggested a method for detection diabetic retinopathy based on SVM classifier. A few geometric and statistical features were removed from the segmented pictures and the colorful lesions in the front are removed. Using the characteristics collected from the segmented retinal pictures, the SVM classifier was trained on DIARETDB1 and MESSIDOR datasets. There was just one kind of DR illness employed in their investigation achieved 96.66% accuracy. In [39], another method for DR detection was demonstrated based on the observation of red lesions known as microaneurysms located on the retinal facet. Because of this, binary classification uses the machine learning method Principle Component Analysis on multiple datasets with accuracy MESIDOR: 87.6%, DRIVE: 81.7%, DIARETDBI: 84.7%. Chetoui et al. [40] proposed a DR diagnosis method based on textual characteristics and SVM. From the segmented retinal pictures, it retrieves some handcraft features including the local ternary Pattern and local energy-based shape histogram. The SVM classifier was then trained using the characteristics derived from the histogram. Using the MESSIDOR dataset, it is able to differentiate between normal and pathological situations without classifying the other types of retinopathy with an accuracy of 90.4%.

Hardas et al. [41] illustrated a retinal fundus identification method based on SVM predictions. They employed SVM, K-means technique, the grey-level co-occurrence matrix, principle component Analysis, and the Gaussian mixture model. Their accuracy on the DIARETDB1 dataset was 77.3% which means the accuracy was insufficient. While, Yao et al. [42] suggested the decision tree (DT) models for early-stage classification of DR. The DT model was assessed on 241 eye of type 2 diabetes mellitus considering the area under the curve (AUC), sensitivity, and specificity. The findings showed that the method achieved AUC and sensitivity were 0.62 and 66%, respectively, while the specificity reached to 76%. Narayanan et al. [43] suggested hybrid ML architecture for automated detection and grading of retinal images using AlexNet, VGG16, ResNet, Inception-v3, NASNet, DenseNet, and GoogLeNet for DR detection with accuracy 98.2%.

In [44], an ensemble learning model considering decision trees, texture feature extraction, and gray-level intensity was introduced for DR diagnosis. Where several processes are considered in this method included preprocessing, texture features extracted from fundus images, feature selection, and ensemble learner training. The outcomes proved that 93.51% F-measure and an accuracy of 94.2% can be achieved. Another ensemble learning-based method for diagnosis DR was proposed in [45]. Four ML classifiers: K-NN, DT, SVM, and Naïve Bayes were included in the ensemble for microaneurysms in the eyes. The pre-processed photos of the E-ophtha and DIARETDB1 datasets were used to extract the form and intensity attributes. The AUC scores were 0.928 and 0.873 for the E-ophtha and DIARETDB1 datasets, respectively. The primary problem with such ML models is their low recall rates? performance caused by the similarities between diabetic retinopathy diseases with a small amount of training data and optimal parameters selection for ML classifier.

2.2 DL-based Diabetic Retinopathy Diagnosis

A recent development in deep learning (DL) approaches is based on deep neural networks. A large number of retinal datasets were used for the training and assessment of DL models. Apart from the structure of the retina image (brightness in the center and darkness at the edges) as shown in Fig. 1

The CNN model was utilized by Pratt et al. [46] to extract the retinal image characteristics from the 80,000 photos in the diabetic retinopathy Kaggle dataset. For specificity, accuracy, and sensitivity, they received scores of 95%, 75%, and 30%, respectively but there were too many false negatives in the outcomes. This means it missed a significant number of DR cases, raising concerns about its reliability for real-world applications. Lam et al. [47] suggested an automated approach for DR detection. They used 243 retinal pictures from the Kaggle EyePACS collection. The DL models that are employed were Inception-V3, VGG-16, AlexNet, GoogleLeNet-v1, and ResNet. The InceptionV3 model had the best-acquired accuracy, which came in at 98% but others have the accuracy was insufficient.

In [48], an early diagnosis system for non-proliferative DR based on DL algorithms is illustrated using CNN with two models, AlexNet and GoogleNet classifiers, with accuracy 89% and recall 97.11%. Birajdar et al. [49] suggested a method for the classification of DR using AlexNet architecture of convolutional neural networks (CNN) with accuracy 98% and precision 80.30%. Pan et al. [50] presented multi-label classification of retinal lesions in DR for automatic analysis of fundus based on VGGNet DL model with accuracy 86%. In [51], an adaptive ML classification for DR using U-Net model is proposed with accuracy 88%. Amalia et al. [52] demonstrated DL approach for DR detection and captioning based on lesion features using Inception-v3 model with accuracy 95% and recall 89%. In [53], the VGGNet ML model was proposed for DR detection using bright lesion in retinal fundus images with accuracy 89% and recall 97.8%. Butt et al. [54] introduced hybrid deep learning features for DR detection with accuracy and precision 86.34%, 91.7% respectively. In [55], the Uniform LBP Encoded Zeros features is used for detection and grading DR using SVM and CNN with accuracy 91.13% and 94.3% of precision. Sathwik et al. [56] presented DR classification with transfer learning method using DL models VGG16,VGG19, ResNet152, and DenseNet with accuracy and precision values of 78.14%, 77.68%, respectively.

Besides, many DL models including ResNet18, GoogleNet, SqueezeNet, VGG16, AlexNet and VGG19 were suggested to be used by Khalifa et al. [57]. For experimentation, the Asia Pacific Tele-Ophthalmology Society dataset was utilized with a total average accuracy of 96.3%, while the AlexNet model obtained the highest accuracy of 97.9%. A model known as the Hinge Attention Network (HA-Net) was presented by Shaik and Cherukuri [58]. They graded the severity of diabetic retinopathy using several attention modules extracted using the VGG-16 model. The IDRid dataset was used and the outcome accuracy was 66.4%. Also, four CNN models, namely InceptionV3, DenseNet-121, Xception, and ResNet-50 were utilized by Oulhadj et al. [59]. These models were used to capture the retinal pictures of Kaggle APTOS and score diabetic retinopathy. According to the data used, the best botained accuracy was 85.28%.

For feature extraction, Lahmar and Idri [60] examined a variety of DL models, including MobileNet-V2, VGG16, VGG19, DenseNet201, Inception ResNet-V2, and ResNet50. Using the collected features, four distinct ML classifiers (e.g., SVM, DT, MLP, and KNN) were tested after training. There were three datasets used: Messidor-2, Kaggle DR, and APTOS which were utilized in the accuracy results of 88.80%, 84.01%, and 84.05%, respectively. Abood et al. [61] proposed a hybrid image enhancement algorithm to improve diabetic retinopathy (DR) detection. They are pre-processed retinal images by cropping and applying a Gaussian blur. A deep learning network then extracts relevant features and classifies DR. On the Kaggle dataset, accuracy increased from 73% to 92% with enhanced images. Also, sensitivity improved to 82%. Similar improvements were observed on the MESSIDOR dataset, with accuracy reaching 93.6% and sensitivity reaching 84%. Mahmood et al. [62] proposed a method for DR detection using a combination of image processing and deep learning techniques to achieve high accuracy, sensitivity, and specificity.The obtained results proved that this method may accomplished in terms of DR vs non-DR 97.57% sensitivity, 95% specificity, and 96.36% accuracy. However, very poor-quality images can be a big problem in diagnosing DR which leads to high complexity in image processing before using the DL model in the DR detection.

Jabbar et al. [10] suggested a hybrid scheme for early DR detection using well-known DL models, ResNet-16 and GoogleNet combined with adaptive particle swarm optimization. They extracted features from fundus pictures and then fed them into several classifiers for multi-class DR categorization using the EyePACs dataset. The outcomes highlight how well CNN feature extraction and machine learning classifiers work together to provide an accuracy of 94%. Although these models are powerful, it might be challenging to understand how they operate on the inside [6, 63]. So, the lack of ability to interpret the results of diagnosis might be critical in the medical field, where it is important to know how the model determines the diagnosis of DR. While focusing on lesions is a valid approach, it might not capture all aspects of DR. Early stages can manifest through subtle signs beyond just lesions. The model's ability to detect these early stages could be an area for further investigation.

Consequently, the main problems with such DL models are similarities between DR classes (1,2,3,4), increasing model complexity, and generalization across populations which are caused by using large datasets that might contain noisy images or inconsistencies in labeling. So it can lead to wrong diagnoses and inaccurate assessments of DR severity.

2.3 Metaheuristic Optimization Algorithms

In the optimization problems, especially complex ones, finding the absolute best solution (global optimum) can be computationally expensive or even impossible. This is where metaheuristic algorithms come in. They are high-level strategies that guide a search process toward finding a "good enough" solution. They don't guarantee the absolute best solution but exceed at finding near-optimal solutions efficiently, especially for problems with large search spaces [64]. In the literature, there are a number of popular swarm-based algorithms that were introduced on the basis of swarming behaviors of some creatures such as animals, birds, ...etc. Among these swarm-based algorithms, Particle Swarm Optimization (PSO) [65], Firefly Algorithm (FA) [66], Moth-Flame Optimization Algorithm [67], Ant Colony Optimization (ACO) [68], and Crayfish optimization algorithm [69] are widely used in different fields. The concept behind PSO is moving the fish and flock of birds in searching for its food. The concept underlying ACO is the ants' endeavor to choose the best route from the nest to its preferred food supply. While, the Coati Optimization Algorithm (COA) [70] was created by modeling the tactics used by coatis to fight and flee from iguanas.

In another type of algorithms, the process by which males identify the location of their females during the period of mating and the hunting tactics of green anacondas form the basis of Green Anaconda Optimization (GAO) [71]. Various algorithms, including the Grey Wolf Optimizer (GWO) [23], Harris Hawks Optimization[24], and Hunger games search algorithm (HGS)[22] have been designed based on the concepts of food searching and hunting behavior. In this regard, there are several recent algorithms, such as Archimedes Optimization Algorithm (AOA) [72], Aquila optimizer (AO) [73], Marine Predators Algorithm (MPA) [25], and Golf Optimization Algorithm (GOA) [27]. Readers can find more details about these algorithms and others in [32, 74].

3 Golf Optimization Algorithm

This area is devoted to discuss the basics of Golf Optimization Algorithm (GOA) [27] and its model theory for usage in optimization related-applications.

3.1 Initialization of GOA

GOA as a population-based strategy aims to solve optimization issues by randomly searching its members in the space of problem solutions. Hence, the problem parameters are defined as the positions of the GOA's members within the entire search space. A matrix can be used to numerically describe the GOA member population in Eq. 1. These members are uniformly distributed over the space, like with earlier metaheuristic methods. Eq. 2 is used to randomize the location of the members in space of searching at the start of the algorithm's execution.

$$\Xi = \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_t \\ \vdots \\ \xi_n \end{bmatrix} = \begin{bmatrix} \xi_{1,1} & \cdots & \xi_{1,d} & \cdots & \xi_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \xi_{t,1} & \cdots & \xi_{1,d} & \cdots & \xi_{l,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \xi_{n,1} & \cdots & \xi_{n,d} & \cdots & \xi_{n,m} \end{bmatrix} \tag{1}$$

$$\Xi_t : \xi_{t,d} = lb_d + r \times (ub_d - lb_d) \tag{2}$$

Where, Ξ of $n \times m$ refers to GOA population matrix, ξ_i is the i^{th} GOA element, $\xi_{i,d}$ is the value of the d^{th} variable suggested by the i^{th} GOA member, $r \in [0-1]$ is any random number, lb_d and ub_d refer to the lower and upper bound of the d^{th}. While, n and m are the total number of GOA's members and number of variables, respectively. In this context, each GOA element is a potential solution to the on hand problem and specifies its variables. Thus, an objective function value F may be calculated using Eq. 3.

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_1 \\ \vdots \\ F_n \end{bmatrix} = \begin{bmatrix} F(\xi_1) \\ \vdots \\ F(\xi_1) \\ \vdots \\ F(\xi_n) \end{bmatrix}_{n\times 1} \tag{3}$$

In this case, F_i is the value of the objective function obtained based on the current GOA element. These values change with each iteration, thus the best member solution should be updated accordingly.

3.2 Model Theory of GOA

GOA updates the population members after finishing the initialization stages. In fact, the population of GOA is updated over the next two important stages: exploration and exploitation.

3.2.1 Stage 1: Exploration

Within the game of the golf, the grip is the part of the playground where the initial swing is made. Contributors try hitting a strong shot toward the best location "hole" in GOA. This approach covers a variety of search spaces, conforming the GOA's capacity for worldwide investigation. Eqs. 4 - 6 describe mathematically the updating process of the GOA's members based on the exploration phase. During this process, a new position is defined for each GOA's member using Eq. 4 based on simulation of the player's greatest shot to the ball. After that, the relevant member's prior location is replaced using Eq. 5 in the case the objective function increased in this new location. The players may hit shots in golf when the ball is close to or goes through the hole. This scenario is imitate by Eq. 4 according to parameter I. The ball becomes close to the hole if $I=1$. Simultaneously, if $I=2$, the algorithm may scan a large region of the searching space by raising the likelihood of change in the ball's location. So, the algorithm's exploratory power is enhanced in the global search.

$$\xi_1^{P_1} : \xi_{t,d}^{P1} = \xi_{t,d} + r \times (B_d - I \times \xi_{t,d}) \tag{4}$$

$$\xi_t = \begin{cases} \xi_l^{p_1}, & F_1^{P1} < F_1 \\ \\ \xi_1, & \text{Otherwise} \end{cases} \tag{5}$$

Based on the exploration stage, $\xi_l^{p_1}$ represents the new status of the i^{th} member and $\xi_{t,d}^{P1}$ is its d^{th} dimension with F_i^{P1} defines its objective function. While, B refers to the best member with B_d is its d^{th} dimension, r is a random number in $[0-1]$, and I is a random number of $[1,2]$.

3.2.2 Stage 2: Exploitation

The term "green" refers to the portion of playground where the hole is situated. Here, players attempt to use putts or kicks, to get the ball into the hole. Less force is applied to these accurate kicks to prevent the golf ball from straying away the hole. This tactic makes it possible to thoroughly explore the region where every GOA member is situated, demonstrating the GOA's potential for exploitation in local searches. Eqs. 6 and 7 describe mathematically the updating of GOA members based on the exploitation phase. At this stage of the GOA update, each GOA member's new location is determined using Eq. 6, which is based on the player's low-power strokes to the ball being mathematically modeled. Eq. 7 states that the subsequent location of the corresponding member is replaced by the new location if it raises the value of the goal function.

$$\xi_i^{P2} : \xi_{i,d}^{P2} = \xi_{i,d} + (1-2r) \times \frac{lb_d + r \times (ub_d - lb_d)}{t} \tag{6}$$

$$\xi_i = \begin{cases} \xi_i^{P2}, & F_i^{P2} < F_i \\ \\ \xi_i, & \text{Otherwise} \end{cases} \tag{7}$$

Based on the exploitation stage, ξ_i^{P2} refers to the new location of the i^{th} member, $\xi_{i,d}^{P2}$ is its d^{th} dimension with F_i^{P2} is the objective function value at t iteration.

The new options should be compared to the list of feasible alternatives at the end of each stage of updating the population's position. Firstly, a set of constraints is constrained by a permissible range of decision variables. All choice variables have their values put on the borderline if they are greater than those of the upper or lower band. Eqs. 8 and 9 are used to verify these constraints for considering the upper and lower bands of selected variables.

$$\xi_{i,d}^{P1} = \begin{cases} \xi_{i,d}^{P1}, & lb_d \le \xi_{i,d}^{P1} \le ub_d \\ ub_d, & \xi_{i,d}^{P1} > ub_d \\ lb_d, & \xi_{i,d}^{P1} < lb_d \end{cases} \tag{8}$$

$$\xi_{i,d}^{P2} = \begin{cases} \xi_{i,d}^{P2}, & lb_d \le \xi_{i,d}^{P2} \le ub_d \\ ub_d, & \xi_{i,d}^{P2} > ub_d \\ lb_d, & \xi_{i,d}^{P2} < lb_d \end{cases} \tag{9}$$

The second collection of constraints corresponds to the optimization problem's equal and unequal constraints. The penalty factor was employed to address these restrictions. If any of the equal or unequal restrictions are not fulfilled, the new solution may not fall into the possible solutions set. As a result, by appending a penalty coefficient to the value of the problem's objective function, the new solution is identified as unsuitable, and thus cannot be chosen as the solution to the problem. Eq. 10 can be utilized to check this collection of constraints as follows:

$$F_i = F_i + n_q \times PF_i \tag{10}$$

Mainly, the work of feature selection (FS) is significantly complicated by an exhaustive search space. As a result, meta-heuristic or optimization algorithms are frequently employed for considering the feature selection issues. In this regards, GOA has been modified to pick up the most significant features while searching adaptively in the feature space for picking up the optimal feature subset. Indeed, the ideal solution has the fewest characteristics picked and the highest possible classification accuracy as well as the lowest classification error rate. Consequently, we introduce a modified GOA version to achieve the main aim of this work for predicting the diabetic retinopathy disease.

Algorithm 1. The proposed modified golf opti-mization algorithm (mGOA).

Input: Details of the optimization problem.
Output: The best fitness of mGOA.
Set T: iterations number, N: members number of GOA
For k= 1 : T
Update the best member of GOA as hole
For n= 1 : N
Stage 1:.
Determine the new location of the i^{th} member considering the exploration stage of GOA (Eq. 4)
Stage 2:.
Apply genetic operators using Eqs. (14) - Eq. (17).
Determine new location of the i^{th} member considering the exploitation stage of GOA (Eq. 6).
Update the i^{th} member using Eq. (7).
End For.
Calculate the fitness of all Individuals.
Update X_i using wrapper feature selection is done by Eq. 13. Also, if there is an optimal solution compared to the previous one, then the fitness function is applied by Eq. 11.
Keep the best potential solution yet.
End For.
Return the best criteria
End

4 The Proposed mGOA

We have proposed a new approach inspired by golf club selection strategies to detect DR with respect to three ML models classification (K-NN, SVM, mGOA-radial SVM). The working procedure contained in the mGOA model is illustrated in Fig. 2, which incorporates five major steps containing (1) data Pre-processing, (2) wrapper feature selection with optimization, (3) the developing solution, (4) genetic operators, and (5) classification and validation.

The steps of the framework of mGOA is demonstrated in Algorithm 1. The mGOA addresses optimization problems by mimicking a golfer's decision-making process during shot execution. It considers factors like distance to the target, environmental conditions (e.g., wind speed), and golfer skill level. The core algorithm considers the leverage points of integration selection, crossover, and mutation operators to examine the solution space considering the converge towards the optimal solution. We utilize FS process to determine a subset of relevant features for improving the performance of the tested machine learning algorithms.

4.1 Data Pre-processing

In this phase, medical data from the Messidor image dataset [75] is used to predict whether an image contains signs of DR or not. This data is pre-processed in four stages: format conversion, missing value replacement, class labeling, and splitting into training and testing sets. Initially, the data is in the format *.arff*, which is changed into the pleasant *.csv* format. All lost values are occupied by the median model. The class labeling process is performed to assign a label based on the DR characteristics it finds. This labeled data is then crucial for training ML models to automate DR detection. At last, the dataset, which has 1152 instances, is divided into 70% and 30% as training and testing sets, respectively. Once the data is pre-processed, mGOA is applied for extracting a set of features from the dataset, which have 19 features to predict whether an image contains signs of DR or not. Also, EyePACS1 dataset [76] of the California health care foundation's DR detection competition is used in the experiments. It contains 35,126 fundus images in total. Precisely, ophthalmologists have labeled all images, then the images have been categorized into five different classes: 'normal', 'mild', 'moderate', 'severe', and 'prolifer-

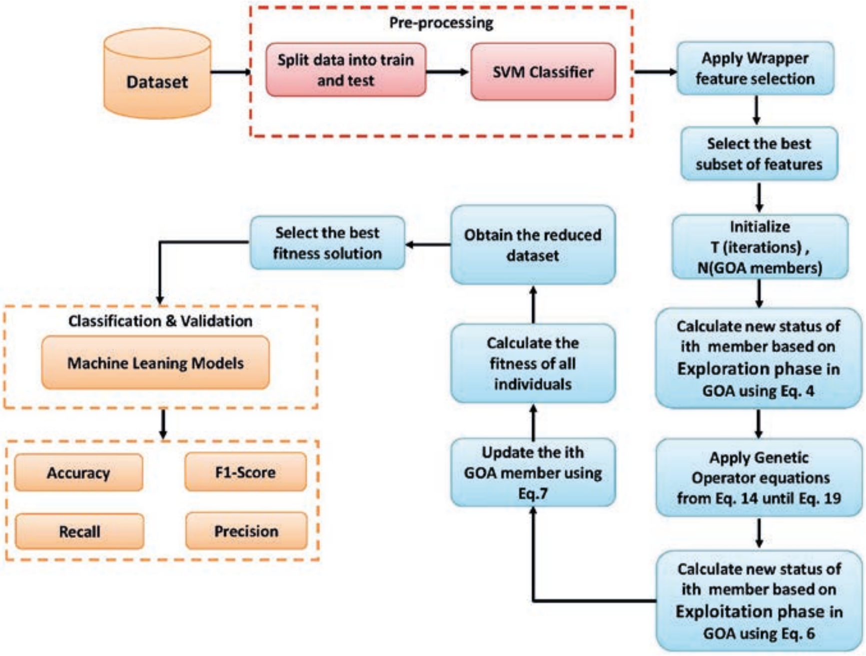

Figure 2. An illustration for the framework of the proposed mGOA method.

ative' DR with five labels (0,1,2,3,4), respectively. While, in the case of multi classification, the images have been classified into three classes only: no signs of diabetic retinopathy (NDR) (stage 0), moderate diabetic retinopathy MDR (stages 1 and 2), and proliferative diabetic retinopathy (PDR) (stages 3 and 4) on the ICDR severity scale. It is also divided into 70% of the fundus images for the training and 30% for testing set during the assessment of the proposed optimization model. Sample images from these datasets are shown in Figures 3 and 4.

4.2 Wrapper Feature Selection

FS techniques are divided into two categories, which are: filters and wrappers. The filters use independent criteria to assess the link between a collection of characteristics (e.g., distance, consistency measurements). While, the wrappers utilize a specialized learning method to determine the value of a collection of characteristics. Wrapper outperform filter approaches in terms of classification accuracy, where the extracted features is assessed with the learning algorithm's feedback. Consequentially, FS is applied after the pre-processing step which mGOA commonly uses for picking a set of features free of redundancies. In this context, the challenge with feature selection is the search space including all potential combinations of features, which can be huge and computationally expensive to explore. Therefore, meta-heuristic algorithms are widely utilized for solving such FS problems. In this regard, a wrapper FS scheme integrated with the SVM classifier is introduced in the proposed mGOA for selecting subsets of discriminating features as shown in Fig. 2. The fitness function fit (Eq. 11) has two key goals during the iterative process: evaluate accuracy, and check their performance, choose the amount of features.

$$fit = \gamma + \lambda \frac{|R|}{|C|} - G \tag{11}$$

Where C, R, G, γ and λ are the total feature number, classification error rate, a group of classifiers, controlling parameters with $\gamma = \lambda$ and $fit > T$, respectively. The mGOA-SVM begins the optimization process by initializing randomly the search agents considering a uniform random distribution using Eq. 12:

$$p\left(\xi_i^{\text{initial}}\right) = LB + \text{rand}_i(UB - LB), i = 1, 2, \ldots, n \tag{12}$$

$p_{\xi_i}^{initial}$ is a randomly initialized i^{th} solution vector, $rand_i \in [0, 1]$, and n represents the search agent. UB and LB refer to the upper and lower bounds, respectively. $p(\xi_i)$ goes through a binary transformation $p(\xi_i)^i_{bin}$ as follows:

$$p\left(\xi_i\right)_{\text{bin}}^i=\begin{cases}1 & \text{if } p(\xi)^i>0.5\\ \\ 0 & \text{Otherwise.}\end{cases} \tag{13}$$

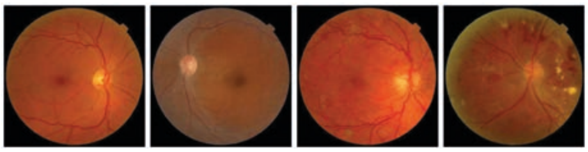

Figure 3. Sample fundus images from MESSIDOR dataset.

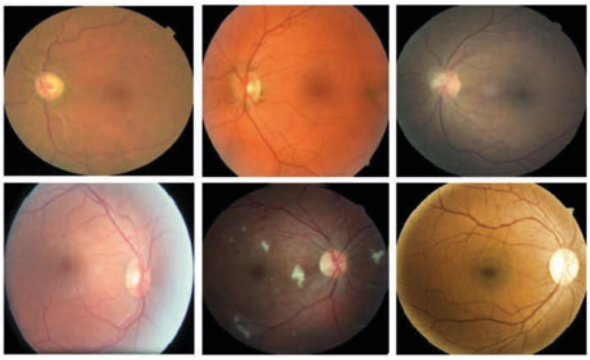

Figure 4. Sample fundus images in EyePACS1 dataset.

4.3 The developing solution and Genetic Operators

Any solution from exploration to exploitation stage is controlled via Eq. 4 and Eq. 5, then genetic operators are introduced through using Eq. 6 and Eq. 7 to improve the location while avoiding trapping in the local search. The results of mGOA tasks are used to determine which parameters should be mutated and by how much. By analyzing the performance of different solutions in the population, the algorithm can identify which parameters are most likely to lead to improvements in the overall fitness of the solutions. These parameters are then selected for mutation in order to examine new areas of the search space and discovering more optimum solutions. Overall, the developing operation is very important for maintaining genetic diversity within the population and driving exploration towards promising areas of the search space. Through using the results of optimization tasks to guide the mutation process, the scheme can efficaciously balance between exploration and exploitation to determine the best required solutions as given in Algorithm 1 and Figure 2.

On the other side, the genetic operators (i.e., mutation, crossover, and selection) control the impact of members on each other and speed up the convergence rate through generating population diversity by integrating two mutant vectors y_{Mut} and z_{Mut} to produce a new child w_{Cross} as given by Eq. 14. According to the selection operator, the offspring fitness value: y_{Mut}, z_{Mut} and w_{Cross} provided by Eq. 17 are compared to obtain the optimal solution F_i.

Mutation: In mGOA, the mutation operation is a key component that helps to introduce diversity in the population of solutions and prevent premature convergence to suboptimal solutions. The mutation operation works by randomly perturbing the values of certain parameters in a solution, creating a new candidate solution that is slightly different from its parent. The mutation operator is computed by using:

$$y_{\text{Mut}}=\begin{cases}p_{(\xi)} & \text{if} \quad \text{rand}_1\geq\zeta\\ y & \text{else}\end{cases}, \qquad z_{\text{Mut}}=\begin{cases}p_{(\xi)} & \text{if} \quad \text{rand}_2\geq\zeta\\ z & \text{else}\end{cases} \tag{14}$$

Such that, $\zeta=\frac{1}{T}$ *and* $z=y-r_v$. Also,

$$y_{\text{Mut}}=\begin{cases}p_{(\xi)} & \text{if } \rho_1\geq\zeta\\ y & \text{else}\end{cases}, \qquad z_{\text{Mut}}=\begin{cases}p_{(\xi)} & \text{if } \rho_2\geq\zeta\\ z & \text{else}\end{cases} \tag{15}$$

Such that, $\zeta=\frac{t}{T}$ *and* $z=y-r_v$

Crossover: combines two individuals in order to generate more variety. For generating a new offspring $wCross$, a linear combination with random integers μ and μ' is considered.

$$w_{Cross}=\mu*y_{Mut}+(1-\mu')*z_{Mut} \; and \; \mu\neq\mu' \tag{16}$$

Selection: The greedy selection process utilized in mGOA is based on differential evolution. After reaching to mutation and crossover functions evolution, the offspring are generating. Accordingly, performance of children and parents ought to be compared for deciding the best. The following formula defines this greedy selection:

$$p^i_{\xi+1} = \begin{cases} y_{Mut} & if\ fit(y_{Mut}) < fit(P(\xi)^i) \\ z_{Mut} & if\ fit(z_{Mut}) < fit(P(\xi)^i) \\ w_{Cross} & if\ fit(w_{Cross}) < fit(P(\xi)^i) \end{cases} \tag{17}$$

4.4 Classification and validation

The feature-reduced subset is fed into the ML model as an input, which executes the classification process to determine the proper class labels. The ML classifiers (SVM, radial SVM, and k-NN) are used in the proposed mGOA method, which comes under rule-based models with reliable benefits in terms of performance, design, and subsequent examination. Thus, ML classifiers modified versions are called mGOA-SVM, mGOA-radial SVM, and mGOA-K-NN, respectively focusing primarily on transforming data into a high-dimensional space.

4.4.1 Support vector machine (SVM)

SVM is a supervised max-margin machine learning classifier that can efficiently perform a non-linear classification utilizing kernel functions to get the optimal solution [77]. There are some tuning parameters which effect on the performance of SVM. In the proposed approach, we consider kernel SVM involving a non-linear transformation function that able to convert the complicated non-linear separable data into linear separable. Where, a hyperplane in the feature space can be expressed by

$$f(\mathbf{\Psi}) = \mathbf{\Psi}^T \mathbf{w} + b = \sum_{i=1}^{n} \Psi_i w_i + b = 0 \tag{18}$$

Dividing by $||\mathbf{w}||$, we get

$$\frac{\mathbf{\Psi}^T \mathbf{w}}{||\mathbf{w}||} = P_{\mathbf{w}}(\mathbf{\Psi}) = -\frac{b}{||\mathbf{w}||} \tag{19}$$

Each point $\mathbf{\Psi}$ on the plane is projected onto the vector $\mathbf{w}$ with $-b/||\mathbf{w}||$ distance from the origin to the plane. Also, the n-D space is divided by plane into two parts. Where,

$$y = sign[f(\mathbf{\Psi})] \quad \in \{1, -1\} \tag{20}$$

$$f(\mathbf{\Psi}) = \mathbf{\Psi}^T \mathbf{w} + b = \begin{cases} > 0, & y = 1,\ \mathbf{\Psi} \in P \\ < 0, & y = -1,\ \mathbf{\Psi} \in N \end{cases} \tag{21}$$

The points $\mathbf{\Psi} \in P$ on one side of the plane is assigned to 1, while points $\mathbf{\Psi} \in N$ on the other side is assigned to -1. For any point $\mathbf{\Psi}$ of unknown class is assigned to P if $f(\mathbf{\Psi}) > 0$ or N if $f(\mathbf{\Psi}) < 0$. The SVM algorithm 2 is illustrated in the following steps [77]:

Algorithm 2 SVM classification algorithm

Inputs:Splitting data(training and testing) is loaded.
Outputs: An accuracy is calculated.
The tuning parameters are chosen.
while (! stopping condition) **do**
  Train data is built for each data point.
  Testing data is classified
end while
Return *Accuracy*

4.4.2 Radial basis function SVM

The Radial basis function (RBF) kernel is a popular choice for SVM due to its ability to handle non-linearly separable data. It maps the input data into a higher-dimensional feature space where it becomes linearly separable, allowing the SVM to find a hyperplane that effectively separates the classes. Achieving the similarity between two input vectors x and y in a high-dimensional feature space is as

$$K(x,y) = \exp(-\gamma \times (x-y)^2) \tag{22}$$

where $K(x,y)$ is the kernel function, x and y are the input vectors and γ is a hyperparameter that controls the influence of each data point. While SVM and radial SVM are both powerful tools, the choice between them depends on the nature of the data and the desired level of flexibility. The RBF kernel offers a versatile approach for handling complex relationships making it a popular choice in many ML applications.

4.4.3 K-Nearest Neighbors (K-NN)

As a non-parametric, supervised learning ML algorithm, the k-NN can be used for classification or regression tasks [78]. k-NN is considered one of the most common classification and regression algorithm used in ML. It presents some advantages depending on the application such as, fast calculation times, few hyperparameters, and predictive ability as well as easy to implement. The main goal of the k-NN is to find the nearest neighbors of a given query x_q to define a class label to this query. To this end, the algorithm starts by calculating the distance between the query x_q and the other data points $x_i, i = 0, ...N$ using distance metrics such as the Euclidean distance, where

$$\begin{aligned} d(\mathbf{p},\mathbf{q}) &= \sqrt{(q_1 - p_1)^2 + \cdots + (q_n - p_n)^2} \\ &= \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \end{aligned} \tag{23}$$

Generally, k-NN provides a classification with suitable significance for the selected features. The value of k impacts the algorithm's behavior Thus, choosing a suitable value for k allows to avoid over-fitting and under-fitting problems. The k-NN steps are given in Algorithm 3.

4.4.4 Validation metrics

On the other hand, for validating the performance of these classifiers, we considered several validation metrics including accuracy, precision, recall, and F1 score. The formulas for these validation metrics are as follows [10]:

In these metrics, TP, TN, FP, FN attribute to True Positive, True Negative, False Positive, and False Negative, respectively.

Accuracy: The most basic metric, it represents the percentage of predictions correctly classified.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{24}$$

Precision (Specificity): Measures how many of the predicted positives actually belong to the positive class.

$$Precision = \frac{TP}{TP+FP} \tag{25}$$

Recall (Sensitivity): Measures how many of the actual positives were identified by the model.

$$Recall = \frac{TP}{TP+FN} \tag{26}$$

F1-Score: A harmonic mean of precision and recall, combining their importance into a single metric.

$$F1-Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{27}$$

Algorithm 3 K-NN classification algorithm.

Inputs:
  $S = s_i = (x_i, y_i)$
Initialize:
  $w_i^0 \leftarrow 0,\ i = 1, \ldots, n,\ S_0 \leftarrow S$
for l = 1 to L **do**
  $S_l \leftarrow S_{l-1}$
  for $s_q \in S_l$ **do**
    $N_q \leftarrow$ k nearest neighbors
    of s_q using $D(s_q, s_i)$
    label(s_q)= $argmax \sum_{s_i \in N_q} D(s_q, s_i)$;
    if label(s_q)$\neq y_q$ **then**
      for $s_i \in N_i$ **do**
        if $y_i \neq y_q$ **then**
          $w_i^t \leftarrow w_i^t - \lambda/d(x_q, x_i)$;
        else
          $w_i^t \leftarrow w_i^t + \lambda/d(x_q, x_i)$;
        end if
      end for
    end if
  end for
  if label(s_q)= $y_q \forall_{s_q}$ **then** break
  end if
end for
Return *Accuracy*

5 Experimental Results and Analysis

In this parts, we conduct quantitative comparisons for the proposed mGOA with its baseline GOA, as well as other traditional metaheuristics and state-of-the-art algorithms. The study compares benchmark functions, from IEEE CEC'2022 [79], [80] hybrid, compositions, shifts and rotations. The algorithms are evaluated using strict statistical analysis and benchmark performance measures. The experiments were tested on a PC with an Intel Xeon (R) CPU E5 (2.60 GHz), 32 GB, and Windows 11 OS with MATLAB R2018b.

5.1 CEC'2022 test suite description

This section offers an extensive overview of the 12 benchmark functions as displayed in Table 1. These functions fall into one of four categories: simple multimodal functions ($F1-F4$), complex multimodal functions ($F5-F8$), and hybrid functions ($F9-F12$). The purpose of these functions was to assess how well metaheuristic algorithms performed to confirm their effectiveness in asymmetric spaces and solutions. So, the majority of the test functions have been shifted and rotated.

Table 1. The CEC'2022 benchmark functions.

No.	Function description	Optimal Value
Simple Multimodal Functions		
F1	Simple Multimodal 1	100
F2	Simple Multimodal 2	300
F3	Simple Multimodal 3	400
F4	Simple Multimodal 4	500
Complex Multimodal Functions		
F5	Complex Multimodal 1	600
F6	Complex Multimodal 2	700
F7	Complex Multimodal 3	800
F8	Complex Multimodal 4	1000
Hybrid Functions		
F9	Hybrid Function 1 (N = 3)	2100
F10	Hybrid Function 2 (N = 3)	2200
F11	Hybrid Function 3 (N = 4)	2300
F12	Hybrid Function 4 (N = 4)	2400

5.2 Statistical results on the CEC'2022 test suite

Tables 2, 3 present the mean and Standard Deviation (STD) for the best minimum value obtained by the proposed mGOA method and all other compared metaheuristic algorithms, including (AOA, GWO, HHO, HGS, MPA, RUN, and GOA) for each function with 10 dimensions, the best results (i.e., small values) are highlighted. As it is clear, mean and STD values confirm the success of the proposed algorithm in solving the twelve functions compared to its baseline GOA algorithm. Further, mGOA attained first position for all the statistical results in mean and STD for most testing functions $F1$ to $F6$ and $F7$ to $F12$ in Tables 2, 3, respectively. So, the proposed mGOA method achieves a better quality with high effectiveness to reach the best solution found by the optimization algorithm on each problem instance.

5.3 Parameter space analysis of 2D visualization

As demonstrated in Figures 5, 6, 7, the 2D visualization of the CEC'2022 test suite for ($F1-F4$), ($F5-F8$), and ($F9-F12$) test functions, respectively, the obtained results of visualizing the parameter space for the proposed mGOA method in 2D provides a valuable tool for researchers to achieve the following aspects: compare functions to easily see the differences between various optimization problems in the CEC'2022 suite. Also, it analyzes algorithm performance by understanding how different optimization algorithms might perform on these functions based on their characteristics. Furthermore, it maintains new algorithms to gain insights into the challenges present in these functions and potentially inspire the development of more effective optimization techniques.

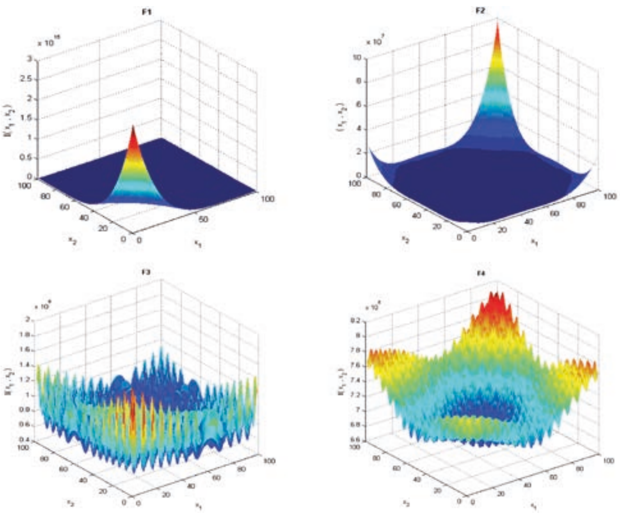

Figure 5. 2D visualization of the CEC'2022 test suite for simple multimodal $F1-F4$ functions.

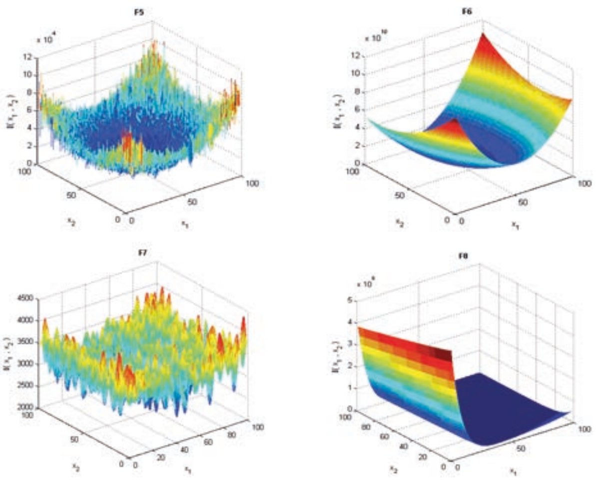

Figure 6. 2D visualization of the CEC'2022 test suite for complex multimodal $F5-F8$ functions.

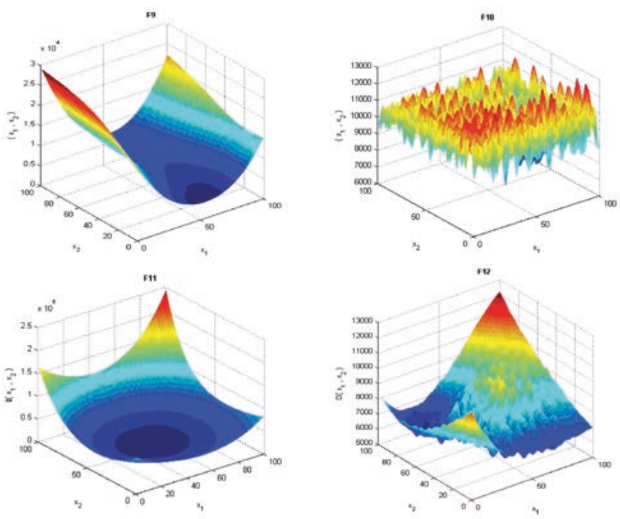

Figure 7. The 2D visualization of the CEC'2022 test suite for Hybrid Functions $F9-F12$ functions.

5.4 Convergence behavior analysis

The convergence speed refers to how quickly the algorithm reaches to the optimal solution or converge to a satisfactory solution. The relationship between convergence speed and optimization can vary according to the used algorithm. Generally, faster convergence speed is desirable as it allows for quicker optimization and reduces the computational time required. However, there are trade-offs to consider where some algorithms may have faster convergence speeds but are more prone to getting stuck in local optimal solutions. In comparison, others may have slower convergence speeds but are more likely to find global optimal analysis. In this regard, to identify the relation between the fitness function and features number, convergence curves were drawn. These curves also clarify the best-performed algorithm among the compared methods.

Figures 8, 9, 10 show the convergence curves of the mGOA variants on $(F1-F4)$, $(F5-F8)$, and $(F9-F12)$ test functions, respectively. The results demonstrate that the mGOA algorithm has improved search mechanisms that allow to exploration of the solution space more effectively and efficiently than other metaheuristic algorithms, including (AOA, GWO, HHO, HGS, MPA, RUN, and GOA). This can lead to faster convergence towards optimal solutions. Besides, the experimental outcomes of convergence curves show that mGOA, which incorporates genetic operators, enhances search capability and accuracy and outperforms other metaheuristic algorithms. Moreover, the proposed mGOA algorithm may have adequate balance between exploitation and exploration. Hence, the proposed mGOA can prevent premature convergence to suboptimal solutions.

5.5 Box plot behavior analysis

The box plot represents visually the distribution of the metric performance (e.g., fitness values, convergence rates) across multiple runs for each optimization algorithm [81, 82]. It allows for a visual comparison of the performance of counterpart algorithms across a set of benchmark functions used in the CEC'2022 competition. As shown in Figs. 11, 12, 13, the box plots of the proposed mGOA method and competitor algorithms on $(F1-F4)$, $(F5-F8)$, and $(F9-F12)$ test functions, respectively, the obtained results confirm that the box plot of mGOA is very narrow compared with other methods for all twelve CEC'2022 test functions possessing the most significant values. So, it offers precious insights into the good performance of different optimization algorithms and their effectiveness in solving a diverse set of optimization problems.

Table 2. Mean and STD of the optimum value obtained by the proposed mGOA algorithm and other metaheuristic algorithms on the CEC'2022 benchmark functions $F1$-$F6$ with $Dim = 10$.

Functions	Metric	AOA	GWO	HHO	HGS	MPA	RUN	GOA	mGOA
F1	Mean	3.0001E+02	1.7721E+03	3.1610E+02	3.0086E+02	3.0000E+02	3.0000E+02	7.7615E+03	7.5226E+03
	STD	5.2527E-02	1.4874E+03	2.8342E+01	3.6434E+00	2.5494E-10	2.2607E-03	2.7931E+03	2.4965E+03
F2	Mean	4.0161E+02	4.2201E+02	4.3527E+02	4.2237E+02	4.0027E+02	4.0510E+02	9.0448E+02	1.0611E+03
	STD	3.2047E+00	2.2534E+01	3.5778E+01	2.8839E+01	1.0114E+00	4.3048E+00	3.0284E+02	4.2756E+02
F3	Mean	6.0002E+02	6.0159E+02	6.3114E+02	6.0059E+02	6.0000E+02	6.1572E+02	6.3404E+02	6.3301E+02
	STD	4.2264E-02	2.4404E+00	1.0376E+01	6.3233E-01	6.7025E-04	1.1030E+01	8.7677E+00	7.5092E+00
F4	Mean	8.1131E+02	8.1719E+02	8.2692E+02	8.3617E+02	8.0882E+02	8.2232E+02	8.3605E+02	8.3488E+02
	STD	4.3368E+00	9.5191E+00	5.9398E+00	1.1287E+01	2.9415E+00	6.2200E+00	8.9750E+00	1.0478E+01
F5	Mean	9.0158E+02	9.1409E+02	1.4097E+03	1.1476E+03	9.0002E+02	9.9886E+02	1.1845E+03	1.1910E+03
	STD	3.3118E+00	2.8239E+01	1.5332E+02	2.4250E+02	8.2948E-02	5.3971E+01	1.2797E+02	1.1576E+02
F6	Mean	2.5647E+03	5.8489E+03	3.9704E+03	6.0157E+03	1.8003E+03	3.5183E+03	5.7191E+07	1.5623E+08
	STD	1.1650E+03	2.3736E+03	1.8236E+03	1.9420E+03	1.3124E-01	1.2463E+03	6.7501E+07	1.6515E+08

Table 3. Mean and STD of the optimum values obtained by the proposed mGOA and other metaheuristic algorithms on the CEC'2022 benchmark functions $F7$-$F12$ with $Dim = 10$.

Functions	Metric	AOA	GWO	HHO	HGS	MPA	RUN	GOA	mGOA
F7	Mean	2.0147E+03	2.0290E+03	2.0818E+03	2.0189E+03	2.0120E+03	2.0443E+03	2.0752E+03	2.0714E+03
	STD	1.1263E+01	8.34336566	3.2790E+01	5.04494005	9.3972E+00	1.1887E+01	1.8459E+01	1.8697E+01
F8	Mean	2.2191E+03	2.2247E+03	2.2360E+03	2.2218E+03	2.2042E+03	2.2238E+03	2.2475E+03	2.2521E+03
	STD	6.8263E+00	2.4644E+00	1.4072E+01	5.2383E+00	6.0088E+00	1.7800E+00	4.0591E+01	4.2483E+01
F9	Mean	2.4939E+03	2.5732E+03	2.5832E+03	2.5424E+03	2.5293E+03	2.5293E+03	2.7187E+03	2.7251E+03
	STD	2.0608E-07	3.8966E+01	3.5696E+01	3.8428E+01	2.7146E-11	5.1607E-04	3.0533E+01	4.3472E+01
F10	Mean	2.5622E+03	2.5731E+03	2.6021E+03	2.5411E+03	2.5003E+03	2.5583E+03	2.6283E+03	2.6224E+03
	STD	5.4977E+01	1.1764E+02	1.2885E+02	6.6619E+01	6.5171E-02	5.8708E+01	1.1692E+02	8.1255E+01
F11	Mean	2.7004E+03	2.7854E+03	2.7646E+03	2.8214E+03	2.6000E+03	2.6401E+03	3.3588E+03	3.4006E+03
	STD	1.4443E+02	1.6968E+02	1.4829E+02	1.3771E+02	5.1671E-05	6.7645E+01	3.9885E+02	4.4203E+02
F12	Mean	2.8885E+03	2.8666E+03	2.9191E+03	2.8637E+03	2.8601E+03	2.8641E+03	3.0488E+03	3.0195E+03
	STD	1.7840E+01	5.83863062	4.0446E+01	1.57116779	1.45858556	1.21619568	7.9129E+01	6.6997E+01

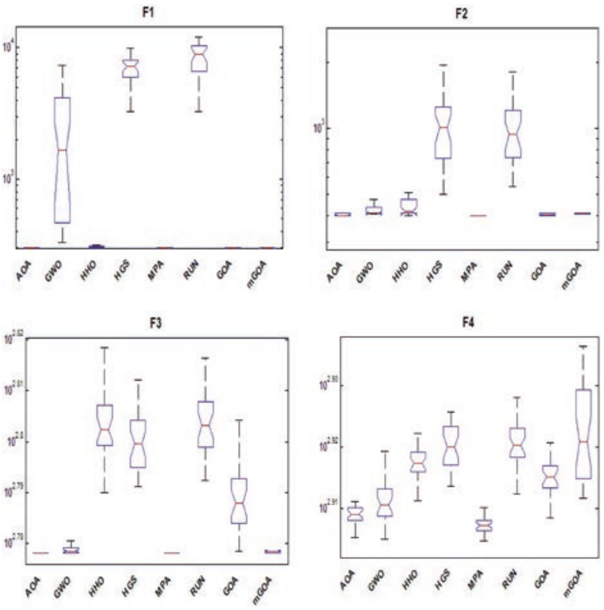

Figure 11. Box plots of the mGOA and competitor algorithms on $F1$ to $F4$ test functions.

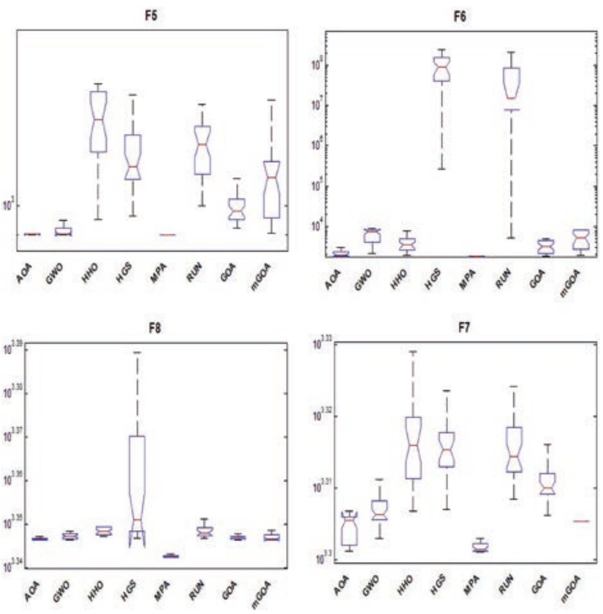

Figure 12. Box plots of the mGOA and competitor algorithms on $F5$ to $F8$ test functions.

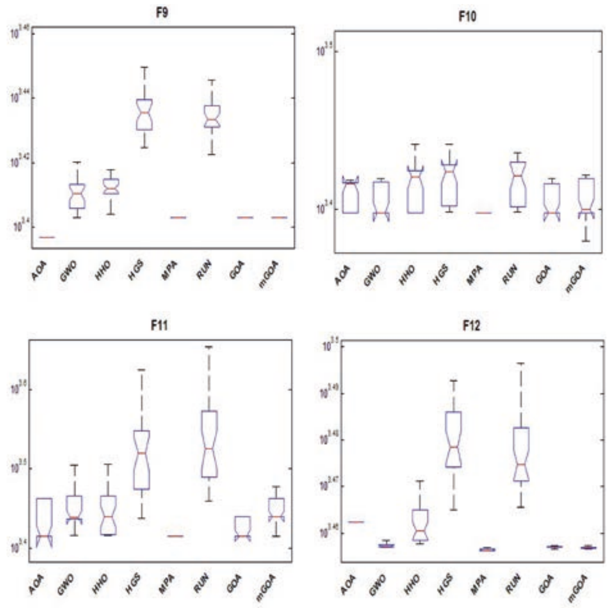

Figure 13. Box plots of the mGOA and competitor algorithms on $F9$ to $F12$ test functions.

5.6 Qualitative metrics analysis

Qualitative metrics refer to non-numeric measures used to evaluate the characteristics, behavior, and performance of optimization algorithms. While CEC'2022 competitions primarily focus on quantitative metrics such as convergence speed, accuracy, and computational efficiency, qualitative metrics provide additional insights into aspects of algorithm performance that cannot be easily quantified. Figs. 14, 15, 16, 17 depict the qualitative metrics on test functions ($F1-F3$), ($F4-F6$), ($F7-F9$), and ($F10, F12$), respectively. They contain search, average fitness, the optimization history achieved by the proposed mGOA method. So, these qualitative metrics point out the best conclusion of the optimization performance for related tasks confirming the high performance and analyze algorithm behavior under perturbations or variations in problem characteristics to determine the dependability and adaptability of the proposed mGOA method.

5.7 mGOA results with ML models

This section focused on supervised ML algorithms to improve the prediction of diabetic retinopathy (DR) disease on two datasets

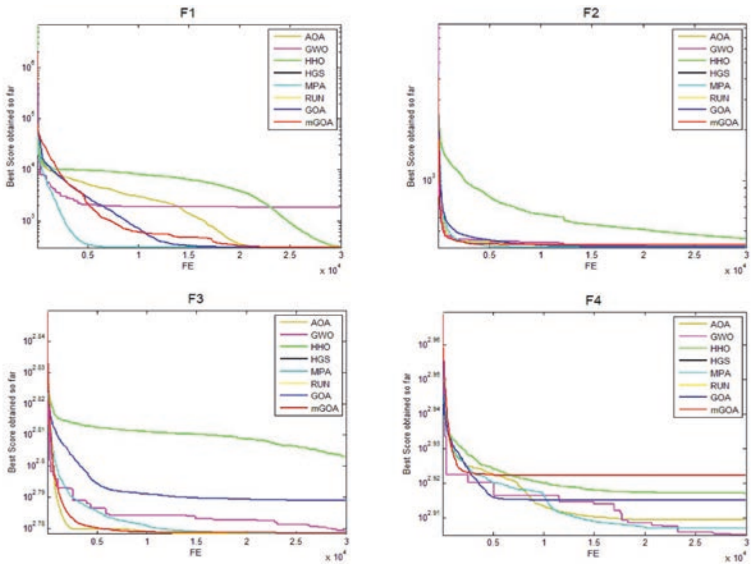

Figure 8. The convergence curves of the mGOA variants on $F1$ to $F4$ test functions.

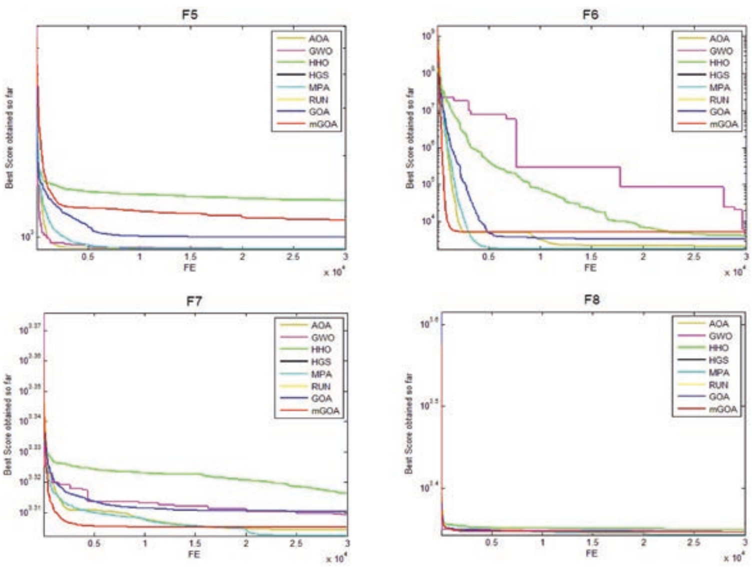

Figure 9. The convergence curves of the mGOA variants on $F5$ to $F8$ test functions.

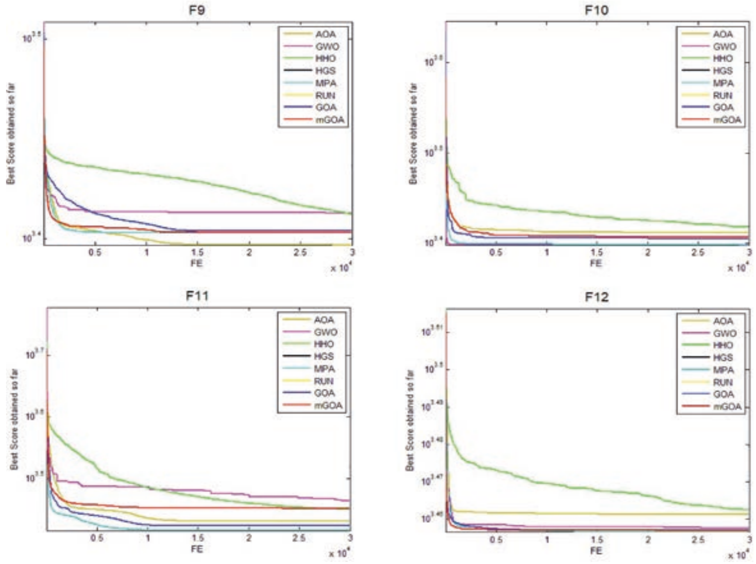

Figure 10. The convergence curves of the mGOA variants on $F9$ to $F12$ test functions.

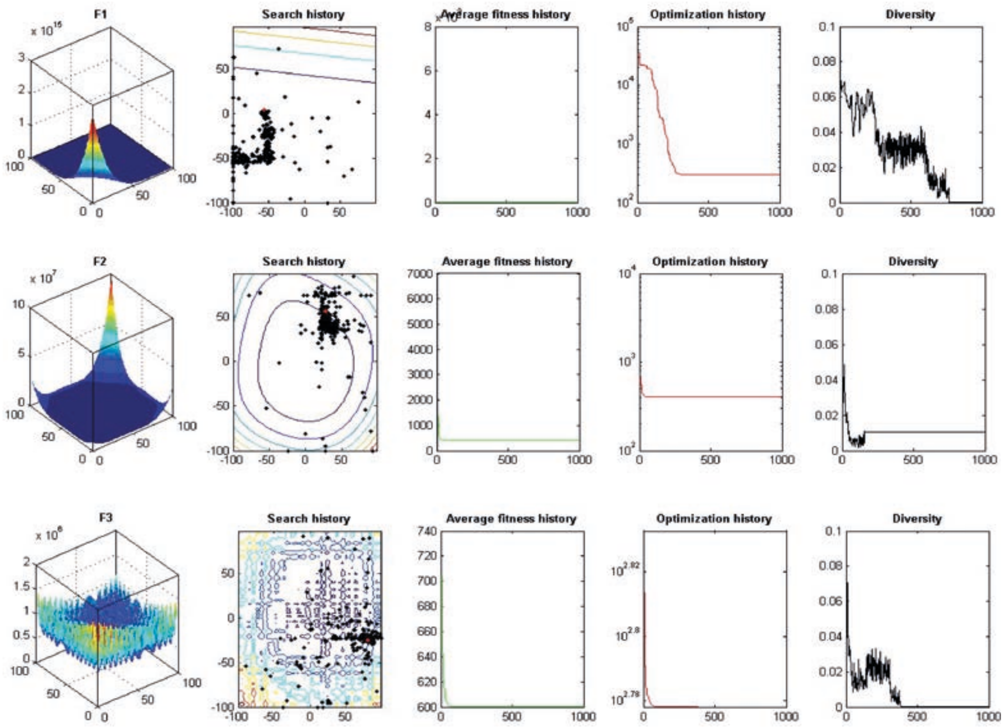

Figure 14. 2D view for the metrics on F1, F2, and F3 functions including: search history, average fitness history, optimization history and diversity.

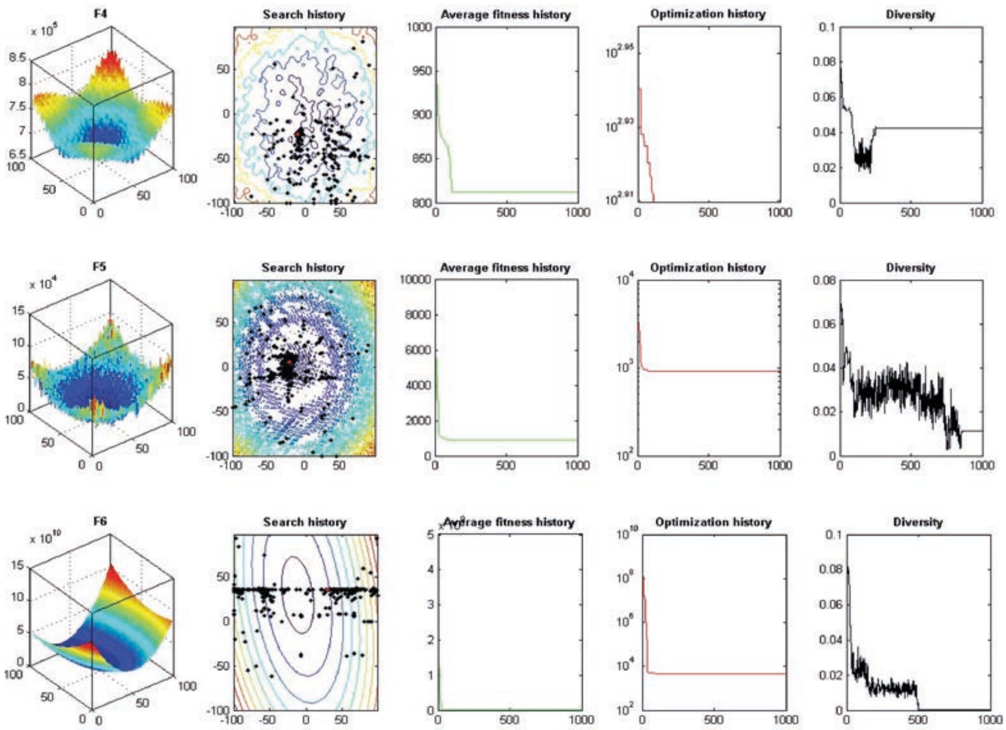

Figure 15. 2D view for the metrics on F4, F5, and F6 functions including: search history, average fitness history, optimization history and diversity.

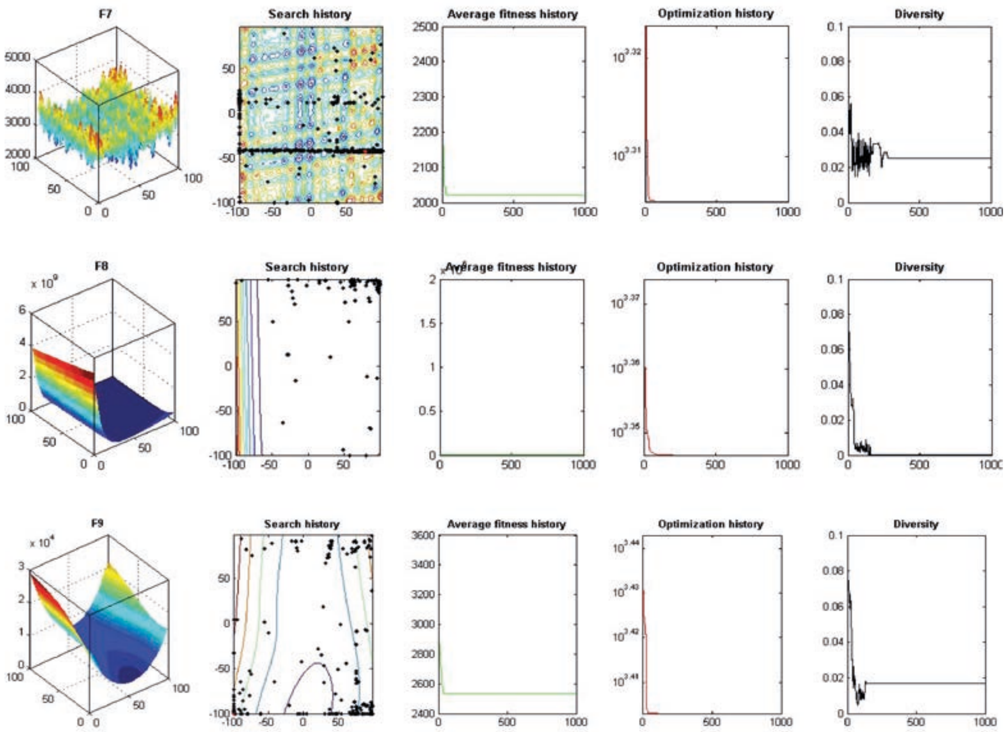

Figure 16. 2D view for the metrics on F7, F8, and F9 functions including: search history, average fitness history, optimization history and diversity.

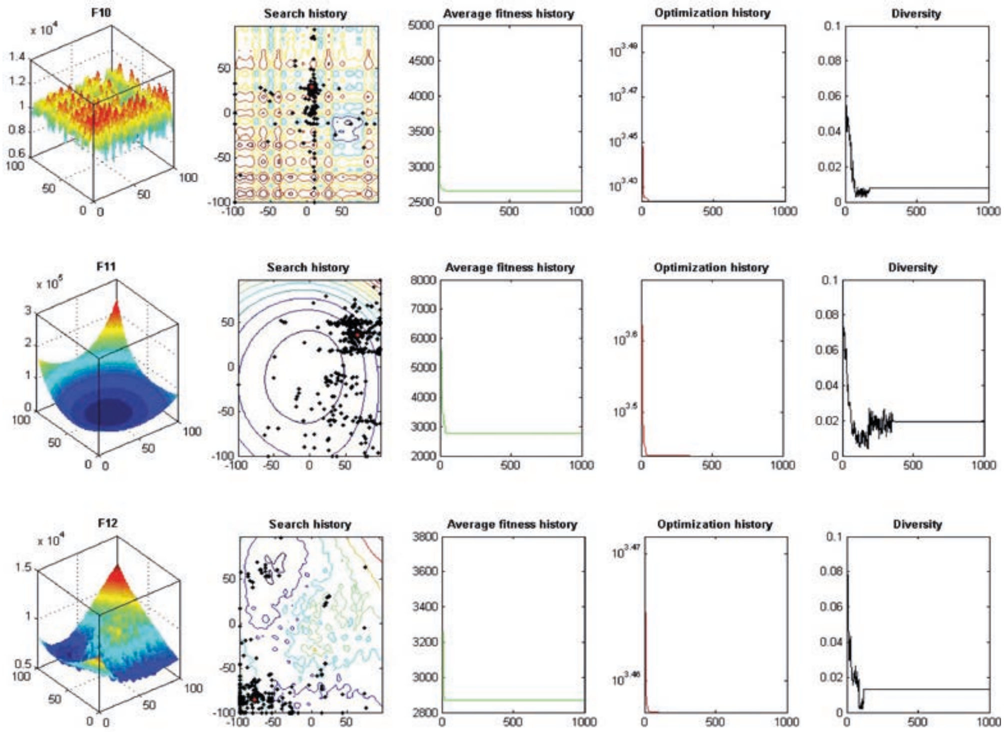

Figure 17. Qualitative metrics on F10, F11, and F12 functions in 2D view including: search history, average fitness history, optimization history and Diversity.

(MESSIDOR and EyePACS1) shown in Figures 3, 4. We discuss the results of classification using the models, mGOA-SVM, mGOA-radial SVM, and mGOA-KNN. A variety of classifiers are used to perform binary and multi-class classifications on the MESSIDOR [75] and EyePACS1 [76] datasets respectively. To assess classifier effectiveness, we used the well-known performance metrics: the accuracy, precision (specificity), recall (sensitivity) and F1-score. Here, precision is the percentage of right positive predictions in a class, whereas accuracy measures the percentage of correctly categorized predictions overall.

5.7.1 mGOA-SVM classification results

Table 4 evaluates the mGOA-SVM classification method using fund images from the Messidor dataset [75] in terms of assessment metrics according to two classes $(1, 0)$ for detecting DR or not. The obtained results for accuracy are 0.91, 0.90, for precision values are 0.90, 0.91, recall values are 0.92, 0.86, and F1-score values are 0.91, 0.88 for classes 0 and 1, respectively. On the other hand, Table 5 provides analysis of assessment metrics for the mGOA-SVM algorithm on the EyePACS1 dataset [76] with 5 classes $(0, 1, 2, 3, 4)$. Across these classes, the mGOA-SVM achieved an overall accuracy score ranging from 0.93 to 0.96, precision values from 0.91 to 0.98, recall values ranging from 0.87 to 0.95, and F1-scores ranging from 0.89 to 0.96. The mGOA-SVM is more accurate on the Messidor dataset in terms of classified diabetic retinopathy patients.

Table 4. Results of mGOA-SVM classification on Messidor dataset with two classes.

Class No.	Accuracy	Precision	Recall	F1-Score
Class 0	0.91	0.90	0.92	0.91
Class 1	0.90	0.91	0.86	0.88

Table 5. Results of mGOA-SVM classification on EyePACS1 dataset with five classes.

Class No.	Accuracy	Precision	Recall	F1-Score
Class 0	0.93	0.92	0.95	0.93
Class 1	0.94	0.91	0.87	0.89
Class 2	0.93	0.96	0.90	0.93
Class 3	0.95	0.95	0.91	0.93
Class 4	0.96	0.98	0.95	0.96

5.7.2 mGOA-radial SVM classification results

Table 6 evaluates the mGOA-radial SVM classification method using fund images from the Messidor dataset in terms of assessment metrics according to two classes $(1,0)$ for detecting DR or not. The obtained results for accuracy are 0.95, 0.94, where precision values are 0.94, 0.92, recall values are 0.93, 0.92, and F1-score values are 0.93, 0.92 for classes 0 and 1, respectively. Similarly, Table 7 provides analysis of assessment metrics for the mGOA-radial SVM model on the EyePACS1 dataset [76] with 5 classes $(0,1,2,3,4)$ respectively. Across these classes, the mGOA-radial SVM model achieved an overall accuracy score ranging from 0.94 to 0.98, precision values from 0.92 to 0.97, recall values ranging from 0.90 to 0.95, and F1-score ranging from 0.91 to 0.96.

Table 6. Results of mGOA-radial SVM classification on Messidor dataset with two classes.

Class No.	Accuracy	Precision	Recall	F1-score
Class 0	0.96	0.94	0.95	0.94
Class 1	0.95	0.93	0.94	0.93

Table 7. Results of mGOA-radial SVM classification on EyePACS1 dataset with five classes.

Class No.	Accuracy	Precision	Recall	F1-score
Class 0	0.95	0.93	0.92	0.92
Class 1	0.94	0.92	0.90	0.91
Class 2	0.96	0.94	0.93	0.93
Class 3	0.97	0.95	0.93	0.94
Class 4	0.98	0.97	0.95	0.96

5.7.3 mGOA-KNN classification results

Table 8 evaluates the mGOA-KNN classification performance using fund images from the Messidor dataset in terms of assessment metrics according to two classes $(1,0)$ for detecting DR or not. The obtained results for accuracy are 0.90, 0.88, precision values are 0.87, 0.86, recall values are 0.92, 0.83, and F1-score values are 0.89, 0.84 for classes 0 and 1, respectively. While, Table 9 lists assessment metrics for the mGOA-KNN model on the EyePACS1 dataset with 5 classes $(0,1,2,3,4)$. Across these classes, the mGOA-KNN model achieved an overall accuracy score ranging from 0.91 to 0.93, precision from 0.88 to 0.96, recall ranging from 0.85 to 0.94, and F1-scores ranging from 0.87 to 0.94.

Table 8. Results of mGOA-KNN classification on Messidor dataset with two classes.

Class No.	Accuracy	Precision	Recall	F1-Score
Class 0	0.90	0.87	0.92	0.89
Class 1	0.88	0.86	0.83	0.84

Table 9. Results of mGOA-KNN classification on EyePACS1 dataset with five classes.

Class No.	Accuracy	Precision	Recall	F1-Score
Class 0	0.91	0.88	0.94	0.91
Class 1	0.92	0.89	0.85	0.87
Class 2	0.93	0.96	0.92	0.94
Class 3	0.92	0.94	0.91	0.92
Class 4	0.92	0.90	0.93	0.91

Table 10. Comparative analysis with other DR classification methods on the EyePACS1 dataset.

Ref.	Method	Model	Accuracy	Precision	Recall	F1-score
[43]	Transfer Learning	AlexNet, GoogleNet, DenseNet	98.2%	N/A	N/A	N/A
[48]	CNN	AlexNet, GoogleNet	89%	86.03%	97.11	91%
[49]	CNN	VGGNet	98%	80.30%	85.5%	82%
[50]	CNN	AlexNet	86%	95.80%	87%	87%
[51]	CNN	U-Net	88%	87%	89%	**96%**
[52]	CNN	Inception-v3	95%	90.1%	**98.2%**	89%
[53]	CNN	VGGNet	89%	77.79%	97.8%	79%
[54]	CNN	Hybrid model structures	86.34%	91.7%	N/A	93.9%
[55]	ULBPEZ	SVM	91.13%	94.3%	84.70%	92.37%
[56]	Transfer Learning	VGG16, VGG19, ResNet152, DenseNet	78.14%	77.68%	78.14%	77.90%
[10]	Transfer Learning	GoogleNet+ResNet	94%	97%	89%	**96%**
Proposed model		SVM, Radial SVM	**98.5%**	**97.4%**	96%	95%

Overall, the mGOA-radial SVM model performs better than other models in evaluation metrics including accuracy, precision, recall, and F1-score greater in classes 2, 3, and 4. Besides in the Messidor dataset, the mGOA-radial SVM model outperforms other ML models in classes 0 and 1 with strong values of accuracy, precision, recall, and F1-score. While, mGOA-SVM and mGOA-KNN produce encouraging results, notably in accuracy, precision, and F1 measures in EyePACS1 and Messidor datasets.

5.8 Comparison with other DR classification methods

As reported in Table 10, a comparative analysis is demonstrated between the proposed model mGOA-radial SVM and recent methods for DR classification providing analysis of assessment metrics (accuracy, precision, recall, and F1-score). So, the suggested approach, mGOA-SVM, radial SVM, achieved maximum accuracy while reducing the number of features in the multi-class EyePACS1 dataset without requiring any image segmentation. The obtained outcomes of the proposed method outperform comparable approaches containing different deep learning models such as AlexNet [43] [50], GoogleNet [48], DenseNet, VGGNet [49] [53], U-Net [51], Inception-v3 [52], Hybrid model structures [54], ULBPEZ [55], VGG16, VGG19 [56] and GoogleNet+ResNet [10]. Considering the lower recall of 96% and F1-score 95%, the proposed mGOA-radial SVM achieves high accuracy of 99.6% and precision of 97.4%. On the other hand, Inception-v3 [52] achieves high recall of 98.2% with lower precision, F1-score, and accuracy of 90.1%, 89% and 95% respectively. Similarly, U-Net [51] and GoogleNet+ResNet [10] obtain a high F1-score of 96% better than the proposed approach. In the end, this work showed the highest accuracy and precision in the diagnosis of DR as compared to much recent research that could be accomplished for diabetics using an integration of ML models and modified GOA for the classification of DR, especially in class 4.

While the proposed method shows promising results, it has limitations that need to be addressed. The main issue is that the study relies heavily on the MESSIDOR and EyePACS1 datasets. This means that the model's performance might not be as good when used with other datasets, specially that contain a large number of features. This is because the MESSIDOR and EyePACS1 datasets might not represent a diverse enough population. Moreover, the performance of proposed method is strongly dependent on parameter adjustment. Finding optimal settings for this parameter may be difficult and cause time-consuming. Furthermore, Diabetic retinopathy datasets often exhibit class imbalance, with fewer cases of severe retinopathy compared to milder forms. This can bias the model towards the majority class, affecting its ability to detect critical conditions.

6 Conclusions

Diabetic retinopathy is a serious eye disease caused directly by high blood sugar in which the retina is injured as a result of diabetes. Early diagnosis or prediction of diabetic retinopathy represents the pivotal role in health monitoring of patients and the treatment stage. In this paper, we consider machine learning based techniques for detection and diagnosis of diabetic retinopathy from fundus images. To get of the traditional drawbacks of ML classifiers, we introduced a wrapper feature selection (FS) scheme that hybridizes the Golf Optimization Algorithm (GOA) with genetic operators added to control local, global search and avoid local solutions. Which assists in distinguishing the most relevant information from datasets and reducing its dimensionality. The proposed optimization algorithm called mGOA was tested on CEC'2022 optimization benchmark problems to evaluate its performance. According to the statistical results, convergence curves, and box plots, the mGOA achieved ideal maximum accuracy while reducing the number of features in the binary and multi-class datasets that represented our objective function. Then, three alternative ML models based on Support Vector Machine (SVM) and K-Nearest Neighbors (K-NN) called mGOA-SVM, mGOA-radial SVM, and mGOA-KNN optimized by the mGOA algorithm were introduced for premature detection and diagnosis of diabetic retinopathy. The performance of the proposed models was examined with the Messidor and EyePACS1 datasets. The obtained results proved that the proposed technique outperforms several other methods in terms of prediction accuracy according to mGOA-SVM and mGOA-radial-SVM models with the EyePACS1 dataset. Class-wise performance measures were also analyzed, and it was illustrated that class 0 and class 4 exceed all other classes in accuracy, recall, F1-score, and precision when testing the mGOA-radial SVM and mGOA-SVM in Messidor and EyePACS1 datasets, respectively. Remarkably, the optimized model mGOA-radial SVM outperforms existing binary and multi-class DR detection methods with an average accuracy of 98.5% and average precision of 97.4%.

Funding

This work was funded by Prince Sattam bin Abdulaziz University through the project number (PSAU/2023/01/26248)

Acknowledgments

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (PSAU/2023/01/26248)

References

[1] N. A. Wani, R. Kumar, J. Bedi, I. Rida et al., “Explainable AI-driven IoMT fusion: Unravelling techniques, opportunities, and challenges with explainable AI in healthcare,” Information Fusion, p. 102472, 2024.

[2] N. A. Wani, R. Kumar, and J. Bedi, “Harnessing fusion modeling for enhanced breast cancer classification through interpretable artificial intelligence and in-depth explanations,” Engineering Applications of Artificial Intelligence, vol. 136, p. 108939, 2024.

[3] L. Riaz, H. M. Qadir, G. Ali, M. Ali, M. A. Raza, A. D. Jurcut, and J. Ali, “A comprehensive joint learning system to detect skin cancer,” IEEE Access, vol. 11, pp. 79 434–79 444, 2023.

[4] N. Shakhovska, A. Shebeko, and Y. Prykarpatskyy, “A novel explainable AI model for medical data analysis,” Journal of Artificial Intelligence and Soft Computing Research, vol. 14, no. 2, pp. 121–137, 2024.

[5] Z. Wang, A. Ala, Z. Liu, W. Cui, H. Ding, G. Jin, and X. Lu, “A hybrid equilibrium optimizer based on moth flame optimization algorithm to solve global optimization problems,” Journal of Artificial Intelligence and Soft Computing Research, vol. 14, no. 3, pp. 207–235, 2024.

[6] A. Asif, F. Ahmed, J. A. Khan, E. Allogmani, N. El Rashidy, S. Manzoor, M. Anwar et al., “Machine learning based diagnostic paradigm in viral and non-viral hepatocellular carcinoma,” IEEE Access, 2024.

[7] P. Guleria, P. N. Srinivasu, and M. Hassaballah, “Diabetes prediction using shapley additive explanations and dsaas over machine learning classifiers: a novel healthcare paradigm,” Multimedia Tools and Applications, vol. 83, no. 14, pp. 40 677–40 712, 2024.

[8] P. N. Srinivasu, S. Ahmed, M. Hassaballah, and N. Almusallam, “An explainable artificial intelligence software system for predicting diabetes,” Heliyon, vol. 10, no. 16, p. e36112, 2024.

[9] A. Senapati, H. K. Tripathy, V. Sharma, and A. H. Gandomi, “Artificial intelligence for diabetic retinopathy detection: A systematic review,” Informatics in Medicine Unlocked, p. 101445, 2024.

[10] A. Jabbar, H. B. Liaqat, A. Akram, M. U. Sana, I. D. Azpíroz, I. D. L. T. Diez, and I. Ashraf, “A lesion-based diabetic retinopathy detection through hybrid deep learning model,” IEEE Access, vol. 12, pp. 40 019–40 036, 2024.

[11] M. Chetoui and M. A. Akhloufi, “Explainable end-to-end deep learning for diabetic retinopathy detection across multiple datasets,” Journal of Medical Imaging, vol. 7, no. 4, pp. 044 503–044 503, 2020.

[12] A.-U.-I. Rafid, S. Sanjana, M. B. Munir, and N. Sharmin, “An early-stage diagnosis of diabetic retinopathy based on ensemble framework,” Signal, Image and Video Processing, vol. 18, no. 1, pp. 735–749, 2024.

[13] G. Sivapriya, R. M. Devi, P. Keerthika, and V. Praveen, “Automated diagnostic classification of diabetic retinopathy with microvascular structure of fundus images using deep learning method,” Biomedical Signal Processing and Control, vol. 88, p. 105616, 2024.

[14] W. Alyoubi, W. Shalash, and M. Abulkhair, “Diabetic retinopathy detection through deep learning techniques: A review,” Informatics in Medicine Unlocked, vol. 20, p. 100377, 2020.

[15] H. Naz, N. Ahuja, and N. Rahul, “Diabetic retinopathy detection using supervised and unsupervised deep learning: a review study,” Artificial Intelligence Review, vol. 57, no. 5, pp. 1–66, 2024.

[16] S. Roychowdhury, D. D. Koozekanani, and K. K. Parhi, "DREAM: diabetic retinopathy analysis using machine learning," IEEE Journal of Biomedical and Health Informatics, vol. 18, no. 5, pp. 1717–1728, 2013.

[17] Y. Sun, "The neural network of one-dimensional convolution-an example of the diagnosis of diabetic retinopathy," IEEE Access, vol. 7, pp. 69 657–69 666, 2019.

[18] S. Kanarachos, A. Dizqah, G. Chrysakis, and M. Fitzpatrick, "Optimal design of a quadratic parameter varying vehicle suspension system using contrast-based fruit fly optimisation," Applied Soft Computing, vol. 62, pp. 463–477, 2018.

[19] A. Darvish and A. Ebrahimzadeh, "Improved fruit-fly optimization algorithm and its applications in antenna arrays synthesis," IEEE Transactions on Antennas and Propagation, vol. 66, no. 4, pp. 1756–1766, 2018.

[20] W. Deng, J. Xu, and H. Zhao, "An improved Ant colony optimization algorithm based on hybrid strategies for scheduling problem," IEEE Access, vol. 7, pp. 20 281–20 292, 2019.

[21] F. Hashim, H. Kashif, H. Essam, M. Mai, and A.-A. Walid, "Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems," Applied Intelligence, vol. 51, pp. 1531–1551, 2021.

[22] P. Mehta, B. Yildiz, S. Sait, and A. Yildiz, "Hunger games search algorithm for global optimization of engineering design problems," Materials Testing, vol. 64, no. 4, pp. 524–532, 2022.

[23] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in Engineering Software, vol. 69, pp. 46–61, 2014.

[24] A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," Future Generation Computer Systems, vol. 97, pp. 849–872, 2019.

[25] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine predators algorithm: A nature-inspired metaheuristic," Expert Systems with Applications, vol. 152, p. 113377, 2020.

[26] I. Ahmadianfar, A. A. Heidari, A. H. Gandomi, X. Chu, and H. Chen, "Run beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method," Expert Systems with Applications, vol. 181, p. 115079, 2021.

[27] Z. Montazeri, T. Niknam, J. Aghaei, O. P. Malik, M. Dehghani, and G. Dhiman, "Golf optimization algorithm: A new game-based metaheuristic algorithm and its application to energy commitment problem considering resilience," Biomimetics, vol. 8, no. 5, p. 386, 2023.

[28] C. Huang, X. Zhou, X. Ran, Y. Liu, W. Deng, and W. Deng, "Co-evolutionary competitive swarm optimizer with three-phase for large-scale complex optimization problem," Information Sciences, vol. 619, pp. 2–18, 2023.

[29] Y. Liu, A. A. Heidari, Z. Cai, G. Liang, H. Chen, Z. Pan, A. Alsufyani, and S. Bourouis, "Simulated annealing-based dynamic step shuffled frog leaping algorithm: Optimal performance design and feature selection," Neurocomputing, vol. 503, pp. 325–362, 2022.

[30] Y. Xue, X. Cai, and F. Neri, "A multi-objective evolutionary algorithm with interval based initialization and self-adaptive crossover operator for large-scale feature selection in classification," Applied Soft Computing, vol. 127, p. 109420, 2022.

[31] W. Deng, J. Xu, X.-Z. Gao, and H. Zhao, "An enhanced MSIQDE algorithm with novel multiple strategies for global optimization problems," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 52, no. 3, pp. 1578–1587, 2020.

[32] N. E. Chalabi, A. Attia, A. Bouziane, M. Hassaballah, A. Alanazi, and A. Binbusayyis, "An archive-guided equilibrium optimizer based on epsilon dominance for multi-objective optimization problems," Mathematics, vol. 11, no. 12, p. 2680, 2023.

[33] K. Yu, D. Zhang, J. Liang, K. Chen, C. Yue, K. Qiao, and L. Wang, "A correlation-guided layered prediction approach for evolutionary dynamic multiobjective optimization," IEEE Transactions on Evolutionary Computation, vol. 17, no. 5, pp. 1398–1412, 2023.

[34] S. Shi and H. Xiong, "Solving the multi-objective job shop scheduling problems with overtime consideration by an enhanced NSGA-II," Computers & Industrial Engineering, p. 110001, 2024.

[35] R. Salgotra, P. Sharma, S. Raju, and A. H. gandomi, "A contemporary systematic review

on meta-heuristic optimization algorithms with their MATLAB and Python code reference," Archives of Computational Methods in Engineering, vol. 31, pp. 1749–1822, 2024.

[36] R. Casanova, S. Saldana, E. Y. Chew, R. P. Danis, C. M. Greven, and W. T. Ambrosius, "Application of random forests methods to diabetic retinopathy classification analyses," PLOS One, vol. 9, no. 6, p. e98587, 2014.

[37] Y. Kang, Y. Fang, and X. Lai, "Automatic detection of diabetic retinopathy with statistical method and Bayesian classifier," Journal of Medical Imaging and Health Informatics, vol. 10, no. 5, pp. 1225–1233, 2020.

[38] B. V. Ratna and R. K. Senapati, "Bright lesion detection in color fundus images based on texture features," Bulletin of Electrical Engineering and Informatics, vol. 5, no. 1, pp. 92–100, 2016.

[39] W. Zhou, C. Wu, D. Chen, Y. Yi, and W. Du, "Automatic microaneurysm detection using the sparse principal component analysis-based unsupervised classification method," IEEE Access, vol. 5, pp. 2563–2572, 2017.

[40] M. Chetoui, M. A. Akhloufi, and M. Kardouchi, "Diabetic retinopathy detection using machine learning and texture features," in IEEE Canadian Conference on Electrical & Computer Engineering. IEEE, 2018, pp. 1–4.

[41] M. Hardas, S. Mathur, A. Bhaskar, and M. Kalla, "Retinal fundus image classification for diabetic retinopathy using svm predictions," Physical and Engineering Sciences in Medicine, vol. 45, no. 3, pp. 781–791, 2022.

[42] H. Yao, S. Wu, Z. Zhan, Z. Li et al., "A classification tree model with optical coherence tomography angiography variables to screen early-stage diabetic retinopathy in diabetic patients," Journal of Ophthalmology, vol. 2022, 2022.

[43] B. N. Narayanan, R. C. Hardie, M. S. De Silva, and N. K. Kueterman, "Hybrid machine learning architecture for automated detection and grading of retinal images for diabetic retinopathy," Journal of Medical Imaging, vol. 7, no. 3, pp. 034 501–034 501, 2020.

[44] N. Sikder, M. Masud, A. Bairagi, A. Arif, A. Nahid, and H. Alhumyani, "Severity classification of diabetic retinopathy using an ensemble learning algorithm through analyzing retinal images," Symmetry, vol. 13, no. 4, p. 670, 2021.

[45] M. Pendekal and S. Gupta, "An ensemble classifier based on individual features for detecting microaneurysms in diabetic retinopathy," Indonesian Journal of Electrical Engineering and Informatics, vol. 10, no. 1, pp. 60–71, 2022.

[46] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional neural networks for diabetic retinopathy," Procedia Computer Science, vol. 90, pp. 200–205, 2016.

[47] C. Lam, C. Yu, L. Huang, and D. Rubin, "Retinal lesion detection with deep learning using image patches," Investigative Ophthalmology & Visual Science, vol. 59, no. 1, pp. 590–596, 2018.

[48] L. Qiao, Y. Zhu, and H. Zhou, "Diabetic retinopathy detection using prognosis of microaneurysm and early diagnosis system for non-proliferative diabetic retinopathy based on deep learning algorithms," IEEE Access, vol. 8, pp. 104 292–104 302, 2020.

[49] U. Birajdar, S. Gadhave, S. Chikodikar, S. Dadhich, and S. Chiwhane, "Detection and classification of diabetic retinopathy using AlexNet architecture of convolutional neural networks," in International Conference on Computational Science and Applications. Springer, 2020, pp. 245–253.

[50] X. Pan, K. Jin, J. Cao, Z. Liu, and othes, "Multi-label classification of retinal lesions in diabetic retinopathy for automatic analysis of fundus fluorescein angiography based on deep learning," Graefe's Archive for Clinical and Experimental Ophthalmology, vol. 258, pp. 779–785, 2020.

[51] L. Math and R. Fatima, "Adaptive machine learning classification for diabetic retinopathy," Multimedia Tools and Applications, vol. 80, no. 4, pp. 5173–5186, 2021.

[52] R. Amalia, A. Bustamam, A. R. Yudantha, and A. A. Victor, "Diabetic retinopathy detection and captioning based on lesion features using deep learning approach," Commun. Math. Biol. Neurosci., vol. 2021:59, pp. 1–18, 2021.

[53] S. Rathore, A. Aswal, and P. Saranya, "Bright lesion detection in retinal fundus images for diabetic retinopathy detection using machine learning approach," Annals of the Romanian Society for Cell Biology, pp. 4360–4367, 2021.

[54] M. M. Butt, D. A. Iskandar, S. E. Abdelhamid, G. Latif, and R. Alghazo, "Diabetic retinopathy detection from fundus images of the eye

using hybrid deep learning features," Diagnostics, vol. 12, no. 7, p. 1607, 2022.

[55] M. Berbar, "Features extraction using encoded local binary pattern for detection and grading diabetic retinopathy," Health Information Science and Systems, vol. 10, no. 1, p. 14, 2022.

[56] A. S. Sathwik, R. Agarwal, S. S. Basa et al., "Diabetic retinopathy classification using deep learning," EAI Endorsed Transactions on Pervasive Health and Technology, vol. 9, 2023.

[57] N. Khalifa, M. Loey, M. Taha, and H. Mohamed, "Deep transfer learning models for medical diabetic retinopathy detection," Acta Informatica Medica, vol. 27, no. 5, p. 327, 2019.

[58] N. S. Shaik and T. K. Cherukuri, "Hinge attention network: A joint model for diabetic retinopathy severity grading," Applied Intelligence, vol. 52, no. 13, pp. 15 105–15 121, 2022.

[59] M. Oulhadj, J. Riffi, K. Chaimae, A. Mahraz et al., "Diabetic retinopathy prediction based on deep learning and deformable registration," Multimedia Tools and Applications, vol. 81, no. 20, pp. 28 709–28 727, 2022.

[60] C. Lahmar and A. Idri, "Deep hybrid architectures for diabetic retinopathy classification," Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, vol. 11, no. 2, pp. 166–184, 2023.

[61] S. Abbood, H. Hamed, M. Rahim, A. Rehman, T. Saba, and S. Bahaj, "Hybrid retinal image enhancement algorithm for diabetic retinopathy diagnostic using deep learning model," IEEE Access, vol. 10, pp. 73 079–73 086, 2022.

[62] M. A. I. Mahmood, N. Aktar, and M. F. Kader, "A hybrid approach for diagnosing diabetic retinopathy from fundus image exploiting deep features," Heliyon, vol. 9, no. 9, 2023.

[63] N. Ullah, M. S. Khan, J. A. Khan, A. Choi, and M. S. Anwar, "A robust end-to-end deep learning-based approach for effective and reliable BTD using MR images," Sensors, vol. 22, no. 19, p. 7575, 2022.

[64] B. Alatas, "Sports inspired computational intelligence algorithms for global optimization," Artificial Intelligence Review, vol. 52, pp. 1579–1627, 2019.

[65] A. Gad, "Particle swarm optimization algorithm and its applications: a systematic review," Archives of Computational Methods in Engineering, vol. 29, no. 5, pp. 2531–2561, 2022.

[66] V. Kumar and D. Kumar, "A systematic review on Firefly algorithm: past, present, and future," Archives of Computational Methods in Engineering, vol. 28, pp. 3269–3291, 2021.

[67] H. Zamani, M. Nadimi, S. Mirjalili, F. Soleimanian, and D. Oliva, "A critical review of Moth-Flame optimization algorithm and its variants: Structural reviewing, performance evaluation, and statistical analysis," Archives of Computational Methods in Engineering, pp. 1–49, 2024.

[68] M. Dorigo and T. Stützle, Ant colony optimization: overview and recent advances. Springer, 2019.

[69] H. Jia, H. Rao, C. Wen, and S. Mirjalili, "Crayfish optimization algorithm," Artificial Intelligence Review, vol. 56, no. Suppl 2, pp. 1919–1979, 2023.

[70] M. Dehghani, Z. Montazeri, E. Trojovská, and P. Trojovskỳ, "Coati optimization algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems," Knowledge-Based Systems, vol. 259, p. 110011, 2023.

[71] M. Dehghani, T. Pavel, and O. Malik, "Green anaconda optimization: a new bio-inspired metaheuristic algorithm for solving optimization problems," Biomimetics, vol. 8, no. 1, p. 121, 2023.

[72] M. Nurmuhammed, O. Akdağ, and T. Karadağ, "Modified archimedes optimization algorithm for global optimization problems: A comparative study," Neural Computing and Applications, vol. 36, no. 14, pp. 8007–8038, 2024.

[73] B. Sasmal, A. G. Hussien, A. Das, and K. G. Dhal, "A comprehensive survey on Aquila optimizer," Archives of Computational Methods in Engineering, vol. 30, no. 7, pp. 4449–4476, 2023.

[74] E. Houssein, M. Saad, K. Hussain, H. Shaban, and M. Hassaballah, "A review of metaheuristic optimization algorithms in wireless sensor networks," in Metaheuristics in Machine Learning: Theory and Applications. Springer, 2021, pp. 193–217.

[75] B. Antal and A. Hajdu, "Diabetic Retinopathy Debrecen," UCI Machine Learning Repository, 2014, DOI: https://doi.org/10.24432/C5XP4P.

[76] D. Emma, J. Jared, and C. Will, "Diabetic retinopathy detection," 2015. [Online]. Available: `https://kaggle.com/competitions/diabetic-retinopathy-detection`

[77] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, “A comprehensive survey on support vector machine classification: Applications, challenges and trends,” Neurocomputing, vol. 408, pp. 189–215, 2020.

[78] P. Cunningham and S. J. Delany, “K-nearest neighbour classifiers-a tutorial,” ACM Computing Surveys, vol. 54, no. 6, pp. 1–25, 2021.

[79] D. Yazdani, J. Branke, M. N. Omidvar, X. Li, C. Li, M. Mavrovouniotis, T. T. Nguyen, S. Yang, and X. Yao, “IEEE CEC 2022 competition on dynamic optimization problems generated by generalized moving peaks benchmark,” preprint arXiv:2106.06174, 2021.

[80] W. Luo, X. Lin, C. Li, S. Yang, and Y. Shi, “Benchmark functions for CEC 2022 competition on seeking multiple optima in dynamic environments,” preprint arXiv:2201.00523, 2022.

[81] A. Taheri, K. RahimiZadeh, A. Beheshti, J. Baumbach, R. V. Rao, S. Mirjalili, and A. H. Gandomi, “Partial reinforcement optimizer: an evolutionary optimization algorithm,” Expert Systems with Applications, vol. 238, p. 122070, 2024.

[82] N. E. Chalabi, A. Attia, A. Bouziane, and M. Hassaballah, “An improved marine predator algorithm based on Epsilon dominance and pareto archive for multi-objective optimization,” Engineering Applications of Artificial Intelligence, vol. 119, p. 105718, 2023.

Mahmoud Hassaballah received the Doctor of Engineering in Computer Science from Ehime University, Japan in 2011. He is currently a Professor of Computer Science at the Department of Computer Science, Prince Sattam Bin Abdulaziz University, Saudi Arabia. Also, he is a full Professor at the Department of Computer Science, South Valley University, Egypt. He serves as a reviewer for several Journals such as IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Industrial Informatics, IEEE Transactions on Fuzzy Systems. Also, he is a TPC member of many conferences. He is an Editorial Board member of Pattern Analysis and Applications, Real-Time Image Processing, IET Image Processing, and Imaging Science Journal. His research interests include human-centered artificial intelligence, machine learning, computer vision, biometrics, image processing, object detection/recognition, and data security.
https://orcid.org/0000-0001-5655-8511

Mohamed Abdel Hameed received the B.Sc. from the Faculty of Computers and Information, and the M.Sc. from the Faculty of Science both from Suez Canal University, Egypt in 2004 and 2012, respectively, and the Ph.D. in computer science from the Faculty of Science, South Valley University, Egypt, in 2019. He is currently an Associate Professor with the Department of Computer Science, Faculty of Computers and Information, Luxor University, Egypt. He is a reviewer for several journals of Elsevier and Springer. His research interests include information security, steganography, cryptography, Internet of Things security, Artificial Intelligence, cybersecurity, and big data.
https://orcid.org/0000-0002-2911-770X