Baghdad Science Journal

Volume 23 | Issue 4 Article 12

4-23-2026

Optimizing K-Nearest Neighbor Based on Dragonfly Algorithm for Diabetes Retinopathy Classification

Ahmed Subhi Abdalkafor
College of Computer Science and Information Technology, University of Anbar, Anbar, Iraq, ahmed.abdalkafor@uoanbar.edu.iq

Alaa Abdalqahar Jihad
Computer Center, University of Anbar, Anbar, Iraq, it.alaa.heety@uoanbar.edu.iq

Esam Taha Yassen
College of Computer Science and Information Technology, University of Anbar, Anbar, Iraq, co.esamtaha@uoanbar.edu.iq

How to Cite this Article

Abdalkafor, Ahmed Subhi; Jihad, Alaa Abdalqahar; and Yassen, Esam Taha (2026) "Optimizing K-Nearest Neighbor Based on Dragonfly Algorithm for Diabetes Retinopathy Classification," *Baghdad Science Journal*: Vol. 23: Iss. 4, Article 12.
DOI: https://doi.org/10.21123/2411-7986.5268

BAGHDAD SCIENCE JOURNAL 2026;23(4):1282–1298

RESEARCH ARTICLE

Optimizing K-Nearest Neighbor Based on Dragonfly Algorithm for Diabetes Retinopathy Classification

Ahmed Subhi Abdalkafor [1,*], Alaa Abdalqahar Jihad [2], Esam Taha Yassen [1]

[1] College of Computer Science and Information Technology, University of Anbar, Anbar, Iraq
[2] Computer Center, University of Anbar, Anbar, Iraq

ABSTRACT

The K-Nearest Neighbors (KNN) has been proven to be an effective method for addressing classification problems. The performance of the KNN algorithm is heavily dependent on the value of parameter K, which represents the number of nearest neighbors. Choosing an inappropriate value for K can affect the classification accuracy because a smaller chosen K can lead to overfitting and vice versa. So, the appropriate selection of the K value has a significant impact on the performance of KNN. Manually, adjusting the value of K is a very difficult process because the appropriate choices for this value depend on the status of the search. Accordingly, the need to utilize an on-line adjusting technique is still existing. One of the recent algorithms is Dragonfly Algorithm (DA), which solves several combinatorial problems. In this work, the DA is adopted to automatically determine the most appropriate value of K for the KNN algorithm. Additionally, the performance of the proposed model is enhanced via utilizing (PCA) for feature extraction. This integration produces the hybrid algorithm named (KNN-PCA-DA). Recently, diabetic retinopathy (DR), a chronic form of diabetes and the leading cause of blindness. Early and accurate diagnosis of DR is essential for early treatment and prevention of irreversible vision loss. Hence, the performance of the proposed model is evaluated using the Diabetic Retinopathy Debrecen dataset. The obtained experimental results demonstrate that the proposed model is an effective solution for the DR problem, achieving competitive results with an accuracy of 99.47% compared to other models.

Keywords: Diabetes retinopathy, Dragonfly algorithm, K-Nearest neighbors, Pattern recognition, Principal component analysis

Introduction

Artificial intelligence (AI) has shown great promise for accurate diagnosis in a variety of medical domains, including diabetic retinopathy Eye disease diagnostic tools have developed AI-based eye data analysis tools to improve diabetic retinopathy diagnosis. The quality and efficiency of care can be enhanced using deep learning algorithms, data segmentation, feature extraction, and clustering methods to diagnose diabetic retinopathy. These methods have shown promising results in the disease detection and diagnostic accuracy, improving patient outcomes.[1–3] Diabetic retinopathy is the leading cause of blindness worldwide, and its effective management and treatment depend on its early detection. Machine learning and artificial intelligence have the potential to dramatically increase the accuracy of diabetic retinopathy diagnosis. Using large-scale medical imaging, educators and clinicians can now rapidly identify and diagnose diabetic retinopathy.[4–6] One of the most important advantages of using artificial intelligence to diagnose diabetic retinopathy is the ability to provide a more accurate and rapid

Received 13 April 2025; revised 31 July 2025; accepted 14 August 2025.
Available online 23 April 2026

* Corresponding author.
E-mail addresses: ahmed.abdalkafor@uoanbar.edu.iq (A. S. Abdalkafor), it.alaa.heety@uoanbar.edu.iq (A. A. Jihad), co.esamtaha@uoanbar.edu.iq (E. T. Yassen).

https://doi.org/10.21123/2411-7986.5268

diagnosis. AI models make image analysis faster and more accurate than human models, facilitating diagnosis and treatment.[7–9] From this, patients can achieve better outcomes and reduced stress in healthcare settings.[10–12] Despite the tremendous potential of AI to improve the accuracy of diabetic retinopathy diagnosis, there are still problems to overcome. These challenges include uncertainty about its validity and prevalence, as well as dataset bias, and the quality of AI systems. These issues are still under investigation to improve the accuracy and reliability of diabetic retinopathy diagnosis.[13,14]

In this work, we developed a model to improve diabetic retinopathy classification using artificial intelligence (AI) to improve the accuracy of diabetic retinopathy diagnosis and prognosis, using several published methods. To identify important features and patterns in patient datasets that predict disease progression or severity, we use principal component analysis (PCA). Additionally, by applying preprocessing techniques, we intend to improve retinal quality and increase the diagnostic accuracy of any new data.

To reduce the risk of visual loss or visual problems and enhance the results of patients' tests, the main aim of this paper is to develop and improve the framework of the disease diagnosis to be more effective and efficient, so that can help it medical professionals can identify and distinguish diabetic retinopathy and try to treat it early before its exacerbation. This paper proposes a dragonfly algorithm to achieve the highest classification accuracy and improve K-nearest neighbor selection, thereby creating a framework for patient screening and follow-up that will enable prompt and accurate diagnosis of diabetic retinopathy.

Below are the most important contributions of this work.

- This work contributes to the application of the Dragonfly algorithm as an effective optimization tool for selecting the most important parameter (K-value) classifier.
- Involving PCA as a key step in the proposed model to enhance its efficiency because it extracts features that increase the model's classification ability.
- The obtained results demonstrated that the cooperation between the utilized techniques (KNN, PCA, DA) had a positive impact on the efficiency of the proposed model compared to the results of the traditional approach.

Related work

Diabetic retinopathy has been the focus of numerous scientific investigations and has been diagnosed and treated using a wide range of procedures and techniques in recent years. The pathogenesis, clinical manifestation, diagnosis, and treatment of this illness are just a few of the many topics that are covered in the literature review.

Herliana et al.[15] aimed to improve the classification method for early detection of diabetic retinopathy by using the data mining concept with particle swarm optimization (PSO) to select the best features from the diabetic retinopathy dataset, followed by classification using neural networks. This is due to medical experts facing difficulties in recognizing the early symptoms of this disease. The study results show that the proposed method can achieve a significant increase in accuracy of 76.11%, compared to using only the neural network classification. Additionally, using the feature selection method also results in a 4.35% increase in accuracy.

Harun et al.[16] sought to diagnose diabetic retinopathy by classifying fundus data using artificial neural networks, namely Multi-layered Perceptron (MLP) trained by Liebenberg-Marquardt (LM) and Bayesian Regularization (BR). The dataset contained fundus data with and without signs of DR, and 19 features were extracted from each data for use as neural network inputs. The MLP model trained with BR performed better than the use of LM, achieving 72.11% accuracy for training and 67.47% accuracy for testing. In addition to highlighting the value of precisely categorizing diabetic retinopathy for improved disease management, this study raises the possibility of using BR to other artificial neural network models. Accurate diabetic retinopathy classification and prognosis depend on appropriate data generation and dimensionality reduction, as in the study by Gadekallu et al.[17] Principal component analysis (PCA) and scalar normalization were two techniques utilized in this study to extract features from a dataset on diabetic retinopathy from the UCI Machine Learning Repository. Furthermore, the dataset's dimensionality was decreased through the application of the Firefly technique, and the resulting more manageable dataset was used to train a deep neural network model for classification. The results demonstrated that the suggested model performed better than other widely used machine learning models in terms of accuracy, precision, recall, sensitivity, and specificity.

Odeh et al.[18] suggested applying an ensemble-based learning strategy to combine various classification algorithms and produce a more sophisticated diagnostic model. The accuracy rates for this method were the highest among widely used classification algorithms in the industry. The investigation also yielded four sub-datasets containing the top 5 and top 10 features from the dataset, with accuracy rates of 70.7% and 75.1%, respectively. An integrated machine learning strategy for the early diagnosis of diabetic retinopathy, a condition that can result in permanent blindness, is proposed by Pragathi and Rao.[19] To identify the disease's symptoms, the author employs several machine learning algorithms, including SVM, PCA, and moth-flame optimization. When used on the dataset, the SVM algorithm outperforms better than the others. The performance of some algorithms is altered after the dataset's dimensions are reduced using the PCA technique. Finally, SVM and PCA are subjected to the moth-flame optimization technique to achieve an average performance of 85.61%, outperforming other ML algorithms in the study. The UCI ML repository contains the dataset that was used in the study.

Hou et al.[20] emphasized the significance of early retinopathy detection in diabetic patients to avoid blindness. Three machine learning algorithms—GBDT, KNN, and SVM—are used in the study to create an automatic diagnosis model using data from clinical patients with diabetic retinopathy. With the best accuracy (0.827) and AUC value (0.803) among the algorithms, the GBDT algorithm outperforms the competition. According to the study's findings, the GBDT-based automatic diagnosis model can be a useful tool for the clinical diagnosis of diabetic retinopathy. Numerous studies have used highly accurate retinal data analysis and diabetic retinopathy detection methods like PCA, K-means, and range filtering. As was mentioned above, traditional methods and algorithms were used in the earlier studies that were discussed, which ultimately resulted in an accuracy that may be medium to low. Accordingly, this work aimed to develop a feature extraction method using the PCA algorithm to improve the classification accuracy which will be proven when compared with those studies in the results and discussion section, identify vital biomarkers for the early classification and treatment of diabetic retinopathy by using K-means and KNN algorithms, and develop a framework for the quick and precise diagnosis of diabetic retinopathy that can be used during the patient's examination and follow-up.

Proposed methodology

The proposed work consists of several main stages, as shown in Fig. 1. The first stage (Preprocessing) aims to prepare the dataset for the classification stage. The second stage (feature extraction) works to extract features using the PCA algorithm. After that, the optimization process is applied. In this process, the DA algorithm is adopted to select the most suitable k value for the next stage (classification). The following subsections illustrate these stages in detail.

Preprocessing stage

Preprocessing includes dimensionality reduction, feature selection, normalization, and data cleaning. These methods contribute to the information needed for accurate and successful classification of diabetic retinopathy patterns.

Detect and replace outliers

Abnormalities in retinal data may affect the detection and diagnosis of diabetic retinopathy. By addressing the issue of outliers in retinal data, we propose a method to improve the diagnosis of diabetic retinopathy (fill-outliers) in this review. The proposed method is a regression filling in outlier pixels in retinal data. Using machine learning algorithms to detect diabetic retinopathy and then using the enriched pixels to improve the overall retinal data.[21] Below, we describe the formula for filling outlier values as follows:[22]

- Calculate the mean (μ) and standard deviation (σ) of the data pixels.
- Define the threshold value as a multiple of the standard deviation (e.g., 3σ).
- Identify outlier pixels as those that fall outside the threshold value (i.e., |pixel value - μ| > threshold).
- Replace outlier pixels with the average value of their neighboring pixels (e.g., use a medium filter).
- Repeat steps 1–4 several times until another distant pixel is detected.

Once the formula for fill-outliers is applied, the outlier pixels in the resulting data are replaced by nearby pixel values, resulting in a more complete and accurate representation of the retinal data. Then the processed data can be used to detect and diagnose diabetic retinopathy using machine learning algorithms.

Rotate factors

This technique is used in factor analysis to efficiently improve the interpretability of latent components by rotating the axes so that each characteristic is more closely related to a single component. It was used in our work to reduce the overlap between related characteristics and improve discrimination between them.

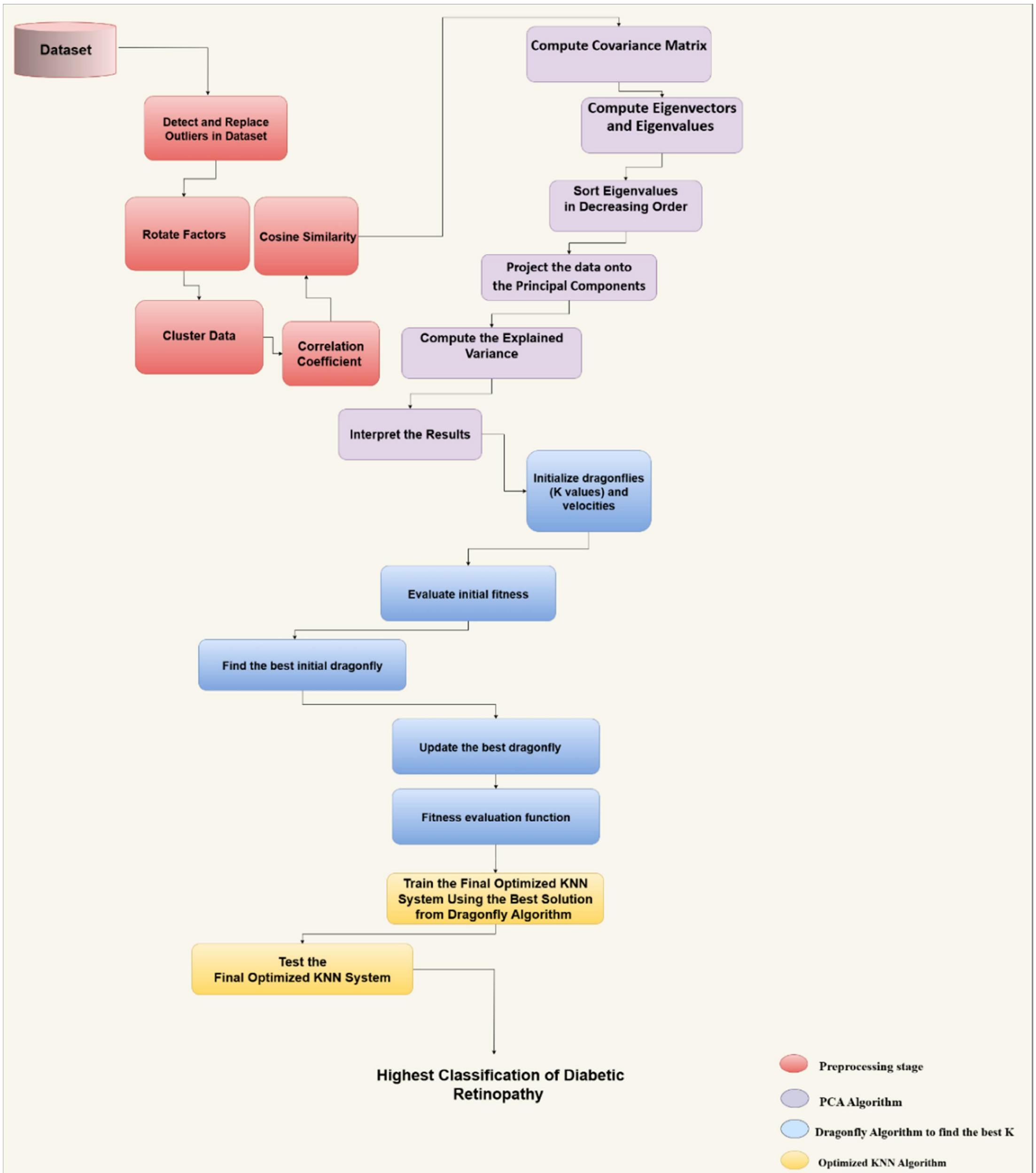

Fig. 1. Proposed model.

Hierarchical clustering

Cluster-data was applied to divide the samples into three groups to identify the internal structure based on the internal similarity of the samples, which may not be apparent given the individual dimensions, as well as to support the structural analysis of the data.

Correlation coefficient

The calculation of the correlation coefficient can identify data elements that are significantly associated with the severity of diabetic retinopathy; the data can be used to develop an accurate and reliable diagnostic model of the disease.[23] The formula for the correlation coefficient of two variables x and y is given in Eq. (1):

$$r = (\frac{1}{\mathcal{N}} * \frac{sum\left((x - mean\ \ (x)) * (y - mean\ \ (y))\right)}{(std\,(x)\ * std\ \ (y))} \tag{1}$$

where:

$\mathcal{N}$ is the number of observations. The means of the x and the y variables and the standard deviations of the x and y variables are mean (x) and the mean (y) and the $std(x)$ and std(y) respectively.

Cholesky decomposition

This technique is used to decompose the covariance matrix into an upper triangular form using Choley's decomposition. It is used in this work to verify the positivity of the matrix's semi-definition and to ensure the stability of mathematical operations based on the covariance matrix.

Min-Max normalization

This technique is one of the most widely applied methods in data processing. It aims to convert the digital value to a specific range between zero and one. This technique was used in our work for the purpose of homogenizing values and preparing data efficiently to feed the proposed model to choose the best value of K.

Cosine similarity

In the diagnosis of diabetic retinopathy, we used cosine similarity to measure the similarity of two feature vectors, where each feature corresponds to a specific symptom of the disease. The resulting similarity scores range from –1 to 1, where a score of 1 indicates that two vectors are parallel, a score of 0 indicates that two vectors are orthogonal, and a score of –1 indicates that the two vectors are opposite. The mathematical equation for the cosine similarity[24] is given in Eq. (2):

$$Cosine\_Similarity(x, y) = \frac{(x.y)}{(\| x \| \ \| y \|)} \tag{2}$$

where x and y are two vectors of equal length, "." represents the dot product of the two vectors, and " || || " represents the Euclidean norm, or the magnitude, of each vector.

Feature extraction stage

In the feature extraction stage, the Principal Component Analysis (PCA) algorithm is proposed. PCA is one of the most popular dimensionality reduction techniques, which transforms a large set of variables into a smaller number of unrelated principal components while preserving as much of the variance present in the raw data as possible. So, this technique generates new features that are more efficient than raw data patterns. By choosing the k eigenvectors with the largest eigenvalues, PCA selects the directions in which the data varies the most and discards the directions with little variability. The resulting matrix Y contains the principal components, which are linear combinations of the original features. PCA can be used to reduce the dimensionality of the data or to identify the most important features for Diabetic Retinopathy Diagnosis. Below are the mathematical equations for PCA applied to the diabetic retinopathy diagnostic dataset:[25]

Data preprocessing: Given a dataset of m observations (data) with n features (pixels in the data), compute the mean vector mu, and the centered data matrix X where each row represents an observation, as in Eqs. (3) and (4):

$$mu = \frac{1}{m} * sum(x_i | \ i = 1 \ to \ m \tag{3}$$

$$X = [+x1 - mu, x2 - mu, \ldots, xn - mu] \tag{4}$$

Computing the covariance matrix: Compute the sample covariance matrix S based on X as in Eq. (5):

$$S = \left(\frac{1}{m} - 1\right) * X * X' \tag{5}$$

Computing the eigenvectors and eigenvalues: Compute the eigenvectors v_1, v_2, ..., v_n and their corresponding eigenvalues lambda1, lambda2, ..., lambda of S as in Eq. (6):

$$S * vi = lambda_i * vi \tag{6}$$

where vi is an eigenvector and $lambda\_i$ is its corresponding eigenvalue.

Sorting the eigenvectors: According to their eigenvalues, order the eigenvectors in descending order, and then pick the k eigenvectors with the highest eigenvalues to create a new matrix V_K as shown in Eq. (7):

$$V_k = \left[+v1, v2, \ldots, vk\right] \tag{7}$$

Projecting the data: Project the data matrix X onto the k-dimensional subspace defined by V_K to obtain the new matrix Y, as in Eq. (8):

$$Y = v_k * X \tag{8}$$

where $V_K{}'$ is the transpose of V_K.

A preliminary set of features was extracted to ensure the clarity and efficiency of the data analysis process, resulting in a data matrix of (3662 * 8100). PCA was applied to achieve the desired goal of overcoming high dimensionality. A regular number of the most appropriate components was identified, as

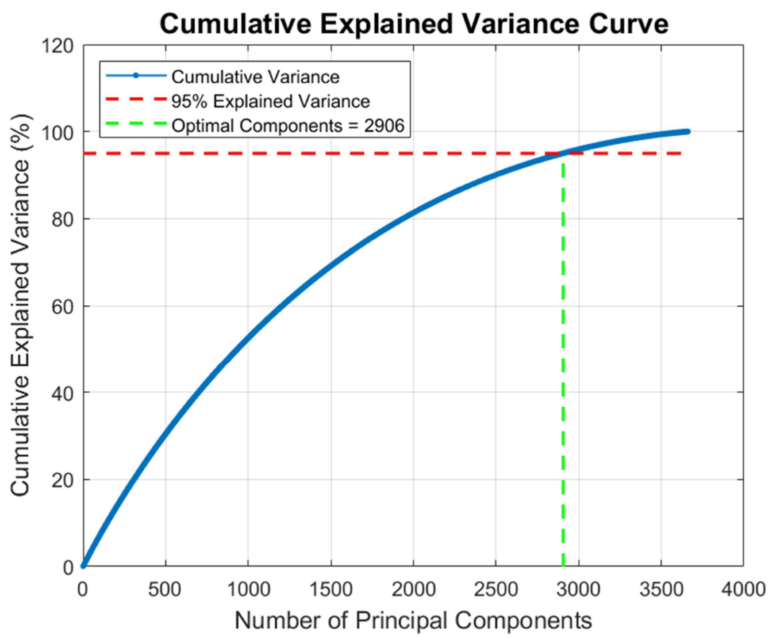

Fig. 2. Applying PCA for feature extraction.

shown in Fig. 2, by analyzing the explained cumulative variance curve, so that the number of principal components was not randomly selected.

Fig. 2 shows a reduction in the number of features from 8,100 to 2,906 after selecting components that explain 95% of the data variance. This approach resulted in a more efficient set of features for training the proposed model, enhancing the accuracy and reliability of the final results.

Optimization stage (Dragonfly algorithm)

Due to the importance of the current problem, on the one hand, and the inability of the existing methods to work well across all instances, on the other hand, the need to propose a new metaheuristic that has the ability to work well across all instances is still urgent. Recently, the dragonfly algorithm (DA)[26] managed to solve several real-world optimization problems in different fields such as networking, image processing, and machine learning. The DA mimics the natural swarming behavior of dragonflies in order to get food and avoid enemies.[27] Like any population-based meta-heuristic algorithm, the design of DA takes into account two important optimization standards: the search space exploration and its exploitation, and the way to balance them. Regarding optimization, the behavior of DA follows five main principles: (i) Separation, (ii) Alignment, (iii) Cohesion, (iv) Attraction, and (v) Distraction, as shown in Fig. 3.[28]

These principles enabled the Dragonfly algorithm to overcome the weakness of other metaheuristic algorithms such as GA, PSO, and GWO. Some of these weaknesses are the randomness and slow convergence, which lead to the dominance of exploration over exploitation and thus prevent it from achieving the required balance between them, which is the basis for improving the search and obtaining the best results. Given the positive characteristics of DA, this work investigates its performance in selecting the most appropriate number of nearest neighbors (K) for KNN, taking into consideration the search status. Although this type of integration results in some extra time (for the tuning process), it will hopefully enable the KNN to increase the accuracy of the model, as shown in Fig. 4. Below are the mathematical equations for this algorithm.

1. Initialize the dragonflies' positions $\mathcal{X}_i$ and velocities $\mathcal{V}_i$ at random, as in Eq. (9)

$$\mathcal{X}_i \sim Uniform\left(\left(Lower\_Bound, Upper\_Bound\right)\right.,$$
$$\mathcal{V} = 0, i = 1, 2, 3, \ldots, \mathcal{N} \tag{9}$$

2. Update equations for each iteration i

- Separation $(\mathcal{S}_i)$:

$$\mathcal{S}_i = -\sum_{j=1}^{\mathcal{N}} \mathcal{X}_i - \mathcal{X}_j,\ \forall \in Neighbor\,(i) \tag{10}$$

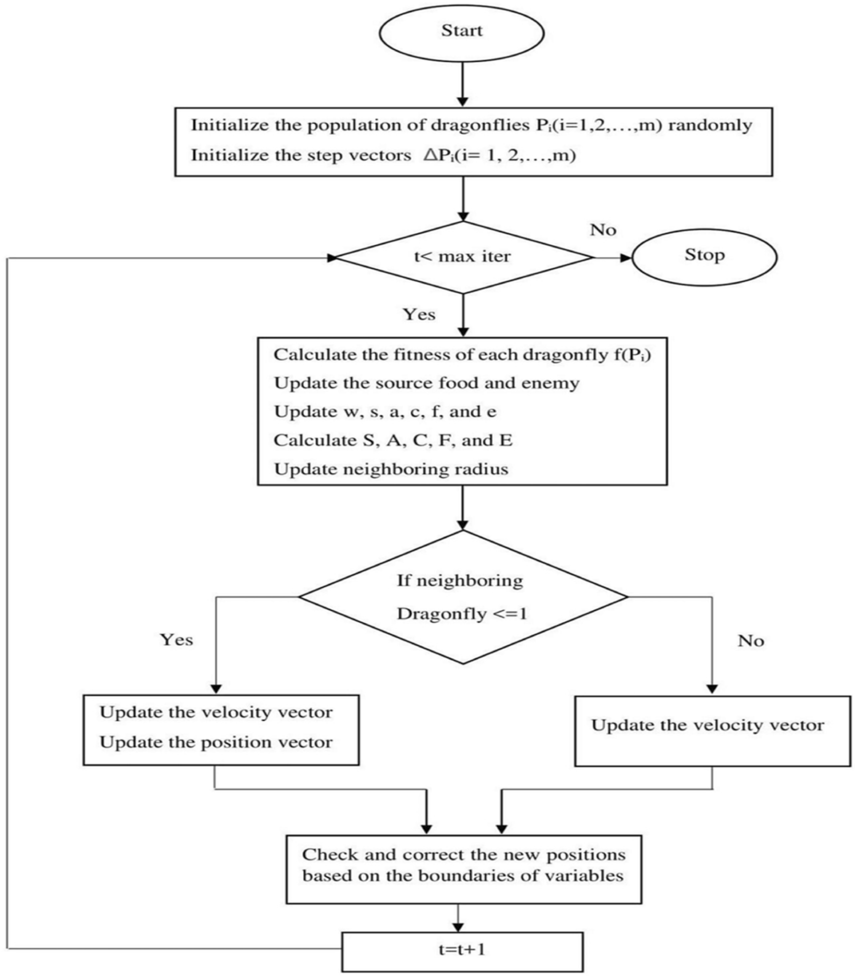

Fig. 3. Dragonfly optimization process.

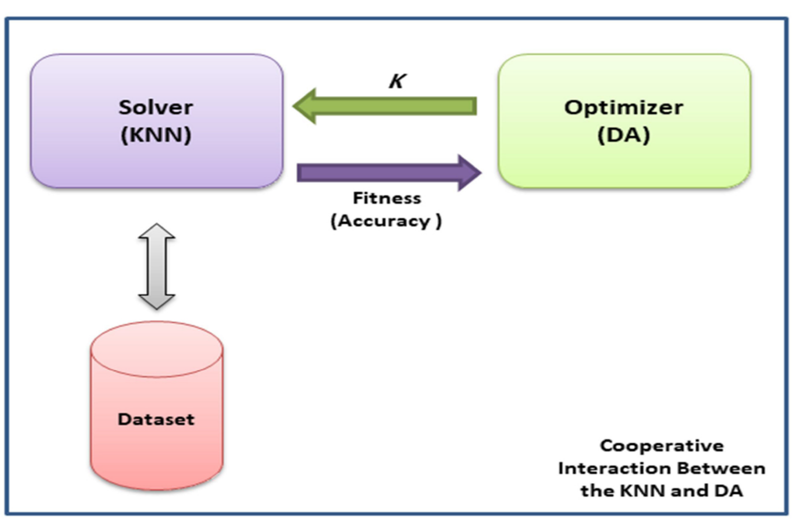

Fig. 4. Cooperative interaction between the KNN and DA.

Table 1. Dragonfly algorithm parameters.

Parameter	Value	Description
Population Size	20	Number of candidate solutions or "dragonflies"
Maximum Iterations = 50	50	Number of iterations in the optimization process
Inertia Weight	0.9	Controls the previous velocity's impact on the current one.
Cognitive Coefficient	2	Controls an individual's tendency to return to personal best.
Social Coefficient	2	Group influence coefficient (attraction to the best solution in the population)

- Alignment (AL_i):

$$AL_i = \frac{1}{\mathcal{N}_i}\sum_{j=1}^{\mathcal{N}} \mathcal{V}_i,\ \forall_j \in Neighbor\,(i) \tag{11}$$

Where $\mathcal{N}_i$ is the number of neighboring dragonflies?

- Cohesion ($\mathcal{C}_i$):

$$\mathcal{C}_i = \frac{1}{\mathcal{N}_i}\sum_{j=1}^{\mathcal{N}} \mathcal{X}_j - \mathcal{X}_i,\ \forall_j \in Neighbor\,(i) \tag{12}$$

- Attraction towards Food Source ($\mathcal{F}\_\mathcal{S}$):

$$\mathcal{F}\_\mathcal{S} : \mathcal{X}_{Food} - \mathcal{X}_i \tag{13}$$

- Distraction from Enemy ($\mathcal{DE}_i$):

$$\mathcal{DE}_i = \mathcal{X}_{\text{Enemy}} - \mathcal{X}_i \tag{14}$$

Update of Position and Velocity

Update velocity:

$$\mathcal{V}_i\,(\mathcal{T} = 1) = s\mathcal{S}_i + aAL_i + c\mathcal{C}_i + f\mathcal{F}_\mathcal{S} + d\mathcal{DE}_i + w\mathcal{V}_i\,(\mathcal{T}) \tag{15}$$

Where $s,\ a,\ c,\ f,\ d,\ and\ w$ are the weights for separation, alignment, cohesion, attraction, distraction, and inertia, respectively.

$$\mathcal{X}_i\,(\mathcal{T}+1) = \mathcal{X}_i\,(\mathcal{T}) + \mathcal{V}_i\,(\mathcal{T}+1) \tag{16}$$

Boundary Control:

Make sure the updated positions are within the previously defined search space, as in Eq. (17):

$$\mathcal{X}_i\,(\mathcal{T}+1) = \max\left(Lower\_Bound\right),\ \min\left(\mathcal{X}_i\,(\mathcal{T}+1),\ Upper\_Bound\right)) \tag{17}$$

Evaluation of Fitness:

Evaluate the fitness of each dragonfly position $\mathcal{X}_i(\mathcal{T}+1)$ based on the objective function as in Eq. (18).

$$fitness_i = f\,(\mathcal{X}_i\,(\mathcal{T}+1)) \tag{18}$$

Update the best solution:

If the current position is better than the best-known position, as in Eq. (19).

$$\mathcal{X}_{best} = \mathcal{X}_i\,(\mathcal{T}+1) \tag{19}$$

$$Best_{fitness} = fitness_i$$

Table 1 shows the DA parameters. Although these parameters are important and affect DA performance, they are less sensitive than the KNN classifier. This is because DA aims to fine-tune the value of k for KNN. This tuning process must be performed quickly (in less time) to avoid negatively impacting the model's running time. Thus, appropriate values for the parameters are determined relying on preliminary testing or as suggested in related works.

Classification stage (KNN algorithm)

The system first finds the k data points that are the closest to the given test point to classify the label or value of the test point based on the labels or value equation's k closest data points. In other words, the KNN algorithm is based on the hypothesis that data points that are close to one another are likely to share labels or values.

The KNN algorithm's mathematical representation:[29]

Assume we have a training dataset with n data points and a label for each data point (xi) in it. We want to predict a test point's label y, given a test point x. The following steps will help us accomplish this:

Step 1: Establish a distance metric between data points, as in Eq. (20).

$$d\,(xi, x) = sqrt\left((xi1 - x1)^2 + (xi2 - x2)^2 + \cdots + (xip - xp)^2\right) \tag{20}$$

where p is the number of features in each data point. Euclidean distance is also frequently used.

Step 2: Determine the separation between each data point xi in the training dataset and the test point x, as in Eq. (21).

$$D\,(xi)\ equals\ d\,(xi, x) \tag{21}$$

Step 3: Based on their distances, choose the k data points that are the closest to the test point x.

$$S = \{xi1,\ xi2, \ldots, xik\} \tag{22}$$

where S is the collection of k data points that are k closest to x.

Step 4: Based on the labeling of the k closest data points, predict the label y for the test point x. We can use voting by the majority to predict the label for classification tasks. In other words, the label that appears frequently between the k the nearest data points are the predicted label. The predicted value for regression tasks can be the mean of the results of the k nearest data points, as in Eq. (23).

$$y = argmax(sum(yij, j = 1 \; to \; k \tag{23}$$

where yij denotes the label of the data point that is j_{th} nearest to x.

Current search techniques to determine the best value of K often rely on traditional search methods, which require a long time and intensive calculations. The importance of using the Dragonfly Algorithm, which was carefully and intelligently proposed and inspired by the static and dynamic swarming behavior of dragonflies, comes to provide an effective and intelligent way to improve the value of K.

This proposed algorithm begins by initializing a set of candidate solutions (dragonflies) within a specific range, and then evaluating their quality based on the classification accuracy using KNN with cross-validation. Next, the algorithm updates the positions and speeds of these dragonflies, as shown in Algorithm 1.

Furthermore, a repeated k-fold cross-validation scheme is implemented in order to verify the generalizability of the obtained results and to reduce the risk of overfitting due to the small values of K. Specifically, cross-validation is applied with varying folds ranging from 2 to 38, and the evaluation is repeated 30 times to ensure robustness.

Results and discussion

To investigate the effectiveness of the proposed algorithm (KNN-PCA-DA), two sets of experiments are conducted. The first experiment was designed to verify the effect of PCA and DA on the performance of the standard KNN. The second experiment was designed to verify the performance of the KNN-PCA-DA in solving the current problem compared to the previous studies. Both dimensionality reduction, PCA, and KNN classifiers have shown promising results for improving the classification of diabetic retinopathy models. By mapping the dataset to a lower-dimensional space, PCA can reduce the dimensionality of the dataset. In addition, based on their nearest neighbors K, the KNN algorithm effectively classified additional unlabeled data points in the classification phase. Correct detection and identification of diabetic retinopathy patterns resulted in a significant improvement in overall classification accuracy.

Dataset

The Debrecen dataset contains data on diabetic retinopathy, a condition that damages blood vessels in the retina and can affect diabetics. Researchers at the University of Debrecen in Hungary developed the dataset, which was made accessible via the UCI Machine Learning Repository. Each sample in the dataset consists of 19 features or features describing different aspects of the patient's eye, such as optic disc diameter, macula height and width ratio, and presence of aqueous outflow. There are 1151 examples or samples in the entire dataset. According to the two class scores, zero (0) indicates the absence of the condition, while 1 indicates the presence of diabetic retinopathy in the patient. There are approximately 537 cases classified as category 1 (retinopathy) and 614 cases classified as category 0 (no retinopathy), and the distribution of categories is considered unbalanced.

Additionally, in order to prove the generality of the proposed model, another dataset is adopted. The APTOS 2019 database from the official Kaggle website was used, as it includes 3,662 images collected from various medical devices.

Comparison between the proposed variants

This section aims to investigate the effect of the PCA and DA on KNN performance. To achieve it, initially, the results obtained by these variants are tested and compared with each other based on the quality of the obtained solutions (in terms of accuracy). So, the best (Best), the average (Avr), and the standard deviation (Std) are reported. Table 2 demonstrates that the KNN-PCA-DA is more efficient than the standard KNN and the KNN-PCA as it obtained the best results in terms of the best, the average, and the standard deviation. The time taken by the Dragonfly algorithm to optimize the K parameter was 5.30405 seconds to find the optimal value for K.

Fig. 5 The accuracy differences between KNN, KNN-PCA, and KNN-PCA-DA.

Algorithm 1: Proposed Classification Algorithm

Input:
- *X:* Input features matrix
- *y:* Corresponding labels vector
- *N_Dragonflies:* dragonflies Number (solutions)
- *max_Iter:* Maximum number of iterations

Output:
- *Best_K*: Optimal number of neighbors K
- *Best_Accy*: Accuracy completed by the optimal K

1. Initialization:
- Randomly generate the initial positions of the dragonflies *{dragon_flies}*.
- Set the dragonfly's initial velocity to zero.
- Determine each dragonfly's level of fitness by applying a cross-validated KNN method.

2. Determine the Initial Best:
- Beginner: Select the dragonfly with the highest fitness score to perform at its best:

$Best\_Acc = \max(fitness),\ bestK = dragon\_flies\,[max\_fitness\_index].$

3. Loop of optimization:
- ***For*** *iter =1 to max_Iter*
  - ***For*** each dragonfly i.
    - ***For*** each dragonfly j from 1 to *N_Dragonflies* (where i≠j)
      - ***IF*** fitness[j] > fitness[i]
        - Dragonfly i velocity updated: velocities[i] ← w × velocities[i] + c1 × rand × (*dragon_flies*[j] − *dragon_flies* [i]) + c2 × rand × (bestK − dragon_flies[i])
        - Position of dragonfly i Updated: *dragon_flies* [i] ← *dragon_flies* [i] + velocities[i]
        - Round the dragonfly's position to the nearest integer: *dragon_flies* [i] ← round (*dragon_flies* [i])
        - Make sure the dragonfly's position remains within the range [2, 10]: *dragon_flies* [i] ← max (2, min (10, *dragon_flies* [i])) Evaluate the fitness of the updated position *fitness*[i] = *evaluate_fitness*(X,y, *dragon_flies* [i]).
    - ***End For***
    - Updated the best dragonfly
      - identify the dragonfly that has the highest fitness: *current_Best_Acc*=max(fitness)
      - IF *current_Best_Acc > best_Acc:*
        - Update *best_Acc* = *current_Best_Acc* and *best_K*= *dragon_flies* [*max_fitness_index*]
- ***End For***

4. End algorithm

Table 2. Comparison between the proposed variants.

Method	Best	Avr	Std	No. Runs
KNN	68.70	61.44	2.28	30
KNN-PCA	65.22	60.53	1.87	30
KNN-PCA-DA	99.47	98.95	0.13	30

Additionally, to verify that the results of these variants are statistically significant, a statistical test has been conducted. The Friedman test is employed to detect statistical differences between a set of algorithms when multiple comparisons are performed. Table 3 shows the ranking of the proposed variables.

The results presented in Table 3 showed that the Friedman test(p-value) is < 0.00001, and the result is significant at p-value < 0.05. The KNN-PCA-DA had the best results over 30 runs, so it is ranked first. Whilst KNN and KNN-PCA are in the 2nd and 3rd, respectively. Furthermore, it is worth noting that the p-value of the Friedman statistical tests proved that there is a considerable variation between the results of KNN, KNN-PCA, and KNN-PCA-DA (p-value $<$ 0.05, i.e., p-value = 0.00001). Fig. 6 shows the overall performance of the improved KNN model across 30 rounds. We observe that the proposed model maintained relatively stable performance across the 30

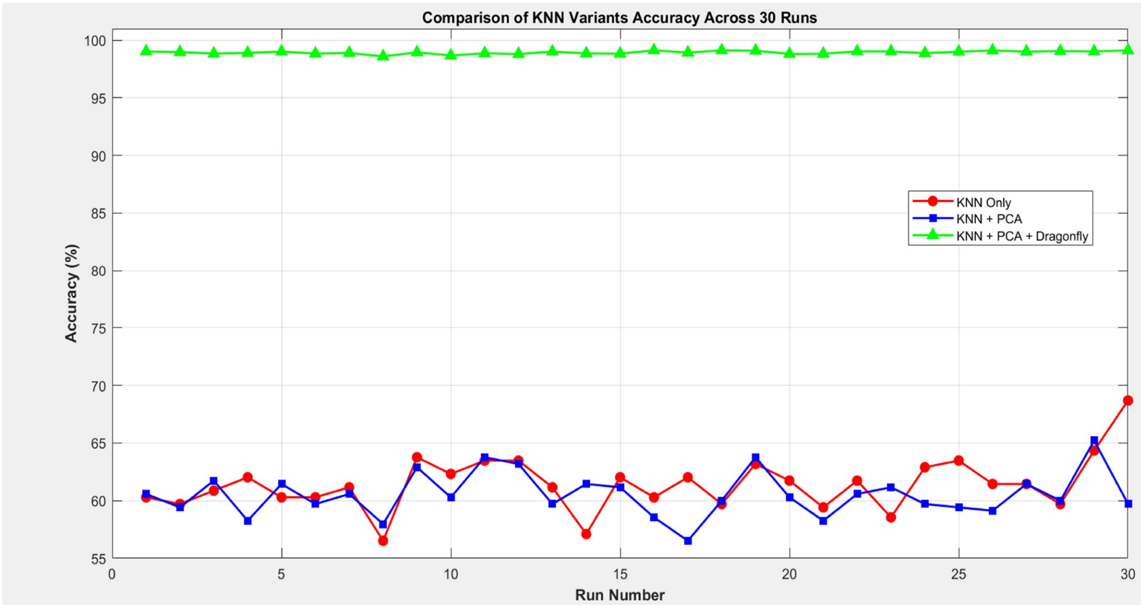

Fig. 5. Accuracy differences between KNN, KNN-PCA, and KNN-PCA-DA.

Table 3. Ranking of the proposed variants.

Method	Rank
KNN-PCA-DA	1
KNN	2
KNN-PCA	3

rounds, with minor fluctuations, demonstrating the reliability of the proposed model.

To evaluate the stability and generalizability of the proposed model, a sensitivity analysis was conducted using the number of iterations (maxIter) and the number of flies (numDragonflies) on the performance of the proposed model. The results showed the best performance when the number of flies was set to 20 and the maximum number of iterations to 50, with a classification accuracy of 99.47%. The experiment also demonstrated the ability of the Dragonfly algorithm to achieve an optimal configuration that balances simplicity and classification accuracy by setting the optimal number of neighbors in the optimized KNN

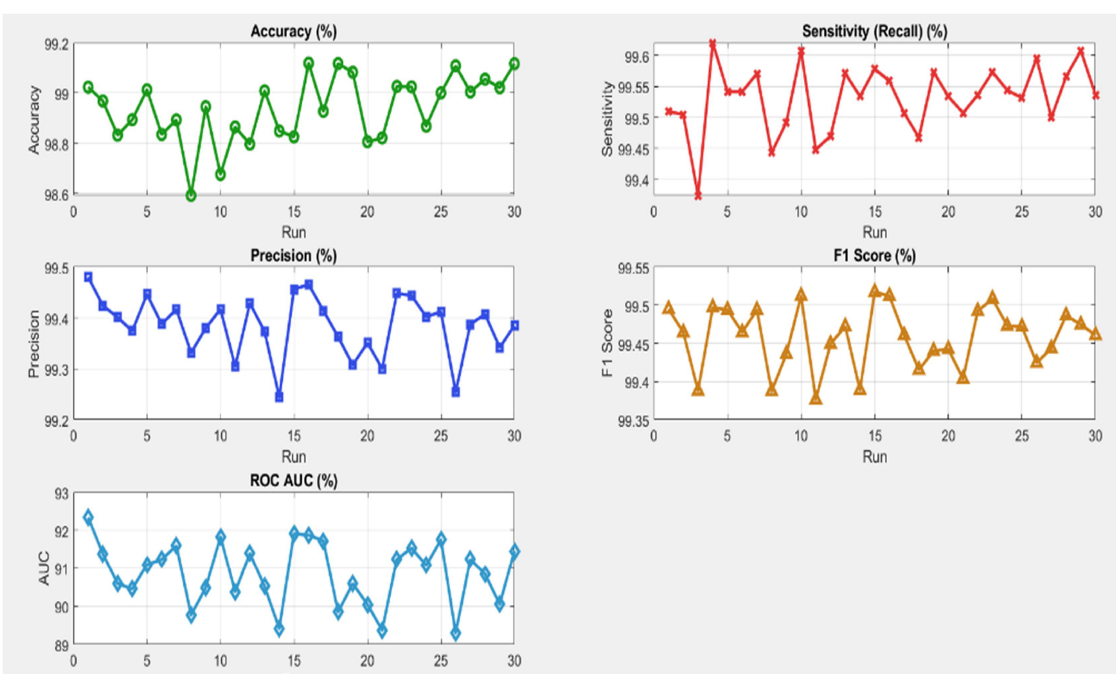

Fig. 6. Performance metrics over 30 rounds.

model to K = 2. This analysis strengthens the reliability of the proposed work and confirms that the selection of its parameters is based on an efficient exploration and exploitation mechanism rather than randomness. Therefore, the work demonstrates stability across different settings in addition to the high performance of the model, an important indicator in sensitive medical applications. Fig. 7 shows the confusion matrix, while Table 4 shows some cases that were misclassified as having retinopathy (1) when in fact they did not have retinopathy (0), and vice versa in Table 5.

The results shown in Tables 4 and 5 indicate that there were limited cases in which the proposed model failed to classify, compared to the large number of cases that were correctly classified by the model.

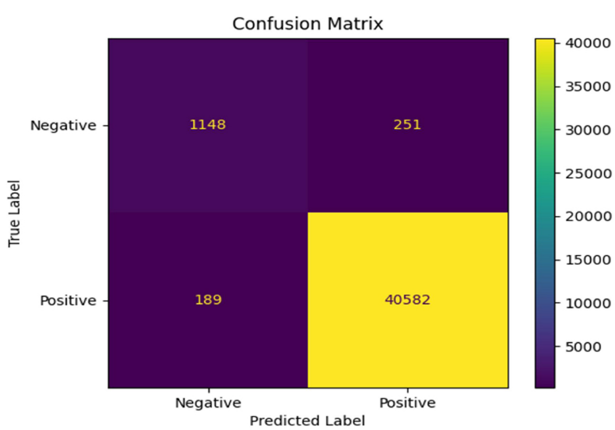

Fig. 7. Confusion matrix model performance.

Table 4. False positive errors (cases that should have been 0 but expected 1).

Case Number	True Label	Predicted Label	Error Type
17	0	1	False Positive
113	0	1	False Positive
258	0	1	False Positive
377	0	1	False Positive
406	0	1	False Positive
512	0	1	False Positive
527	0	1	False Positive
625	0	1	False Positive
653	0	1	False Positive
861	0	1	False Positive

Table 5. False negative errors (cases that should have been 1 but expected 0).

Case Number	True Label	Predicted Label	Error Type
78	1	0	False Negative
184	1	0	False Negative
415	1	0	False Negative
544	1	0	False Negative
575	1	0	False Negative

These cases resulted from problems with the original images in the database. Most of the results (false positives) showed visual distortions or unclear tissue patterns, as well as poor lighting, which led to similar pathological signs and, consequently, misclassification. The results (false negatives), on the other hand, occurred in the early stages, when these signs were very faint, making them difficult to detect accurately. However, the larger number was correctly and efficiently classified, achieving high competitive accuracy.

Comparison of the proposed model with the state-of-the-art methods

The results of this study are very encouraging for the diagnosis of diabetic retinopathy. High accuracy was achieved by applying various state-of-the-art methods and algorithms, such as PCA, the KNN algorithm, etc., in this diagnosis. A highly competitive accuracy of 99.47% was achieved in addition to the construction of a fit by modifying the hyperparameter settings using the dragonfly algorithm (K-fold). This is a major advance in medical imaging that could have a significant impact on the early diagnosis and treatment of diabetic retinopathy, thereby improving patient outcomes. We will go into details about the findings and how they relate to the future of diagnosing and treating diabetic retinopathy in this part. We investigated the studies that categorized the diabetes retinopathy pattern and found that the categorization accuracy varied considerably, as shown in Table 6 and Fig. 8.

When compared to the proposed work, Table 6 and Fig. 8 show that the proposed work improves the classification of diabetic retinopathy patterns by being more accurate than the various classifiers used in earlier studies. The findings demonstrated that the suggested method outperformed the accuracy obtained in earlier studies, which ranged from 67.47% to 86.3%, by achieving the highest accuracy of 99.47%. On the UCI machine learning repository dataset, for instance, Gadekallu et al. achieved an accuracy of 86.3% using the XGBoost-PCA Firefly algorithm, while Pragathi and Rao attained the same accuracy using the SVM, PC, and MF strategy. We can evaluate the relative advantages and disadvantages of these strategies using this comparison. The high accuracy found in the proposed work can be attributed to the effective use of PCA for dimensionality reduction and KNN for classification. The combination of these algorithms proved highly effective in enhancing the classification of diabetic retinopathy patterns, outperforming other approaches.

Table 6. Comparison between previous works.

Previous study	Year	Dataset	Classifier	Accuracy	Precision	Sensitivity	F1 Score	ROC-AUC
Herliana et al. [15]	2018	Retinopathy Debrecen dataset	neural network based particle swarm optimization (PSO)	76.11%	–	–	–	–
Harun et al. [16]	2019	UCI machine learning repository	MLP trained with BR	67.47%	–	–	–	–
Gadekallu et al. [17]	2020	UCI machine learning repository	XGBoost-PCA Firefly	86.3%	88.5	83.3%	–	–
Odeh et al. [18]	2021	InfoGainEval	Neural Networks	75.1%	77.8	–	–	82.7%
Pragathi and Rao [19]	2022	UCI ML repository	SVM + PC + MF	86.3%	86	94.2%	86	–
Hou et al. [20]	2022	Diabetic Retinopathy Dataset	GBDT	82.7 %	–	–	–	80.3%
Emon et al. [28]	2021	UCI machine learning repository	Logistic	75%	80		75	83%
Oladele et al. [30]	2019	Retinopathy Debrecen dataset	MLP	73.7%	–	–	–	–
Nagi et al. [31]	2021	Diabetic Retinopathy Debrecen dataset	Two Stage Classifier (multiple classifiers)	79.40%	–	–	–	–
Şentürk ZK [32]	2020	UCI machine learning repository	ANN + Rapid Miner	76.09%		88.52%	–	–
Rathi [33]	2022	UCI machine learning repository	Bagged Trees	80%	–	–	–	–
Prabhakar et al. [34]	2024	Diabetic Retinopathy Dataset	EGFOA	91.6%	–	92.2%	–	–
Ramani et al. [35]	2025	UCI machine learning repository	AKW-MRPK	92 %	–	–	–	–
Guefrachi et al. [36]	2025	Kaggle's diabetic retinopathy	refined Inceptio nResnet(V2)	96.61%	–	–	–	–
Sushith [37]	2025	DRIVE dataset	Hybrid (CNNs) and (RNNs)	97.5%	95.42%	–	97.48%	–
Hussain [38]		Diabetic Retinopathy Detection dataset	Enhanced (CNN) based P-EDR	93%	–	92%	–	97%
Lateef [39]	2025	UCI machine learning repository	Optimized KNN + PCA-Based Data Fusion and Cuckoo Search Optimization	98.05%	–	–	–	–
Proposed Work	2025	APTOS 2019	Optimized KNN based Dragonfly Algorithm	92.30%	92.74%	92.02%	92.38%	92.30%
Proposed Work	2025	UCI machine learning repository	Optimized KNN based Dragonfly Algorithm	99.47%	99.38%	99.53%	99.45%	92.99%

The reported specificity highlights the potential clinical relevance of the algorithm. Accurate classification of diabetic retinopathy may contribute to early detection and timely intervention, thereby improving patient outcomes. However, it should be noted that there is a need for clinical validation of these protocols and integration into broader healthcare practice. Understanding these factors will enable researchers to make informed decisions, identify areas for improvement, and pave the way for improvements in diabetic retinopathy classification using machine learning. Additionally, in order to prove the generality of the proposed model, the APTOS 2019 dataset was utilized by applying the KNN-PCA-DA model. As shown in Table 6, the proposed model obtained competitive results 92.30%, 92.74%, 92.02%, 92.38%, 92.30% for Accuracy, Precision, Sensitivity, F1 Score, and ROC-AUC, respectively. These results demonstrated that the KNN-PCA-DA has the ability to work well over different situations by obtaining competitive results compared with the current state-of-the-art studies.

Limitations

This work presents an innovative approach to improving the performance of the k-NN classifier using the Dragonfly algorithm, achieving highly competitive results. However, some fundamental limitations must be disclosed that may affect the generalizability of the results when applied to larger scales or across diverse fields, which will be discussed in detail. Despite achieving high levels of accuracy and performance overall, the graphs, particularly the ROC-AUC accuracy, show some noticeable instability across the 30 consecutive runs. For illustration, while the average accuracy is around 99%, significant fluctuations were recorded in some runs, such as runs 6, 15, and 25 for the area under the curve (ROC-AUC) and runs

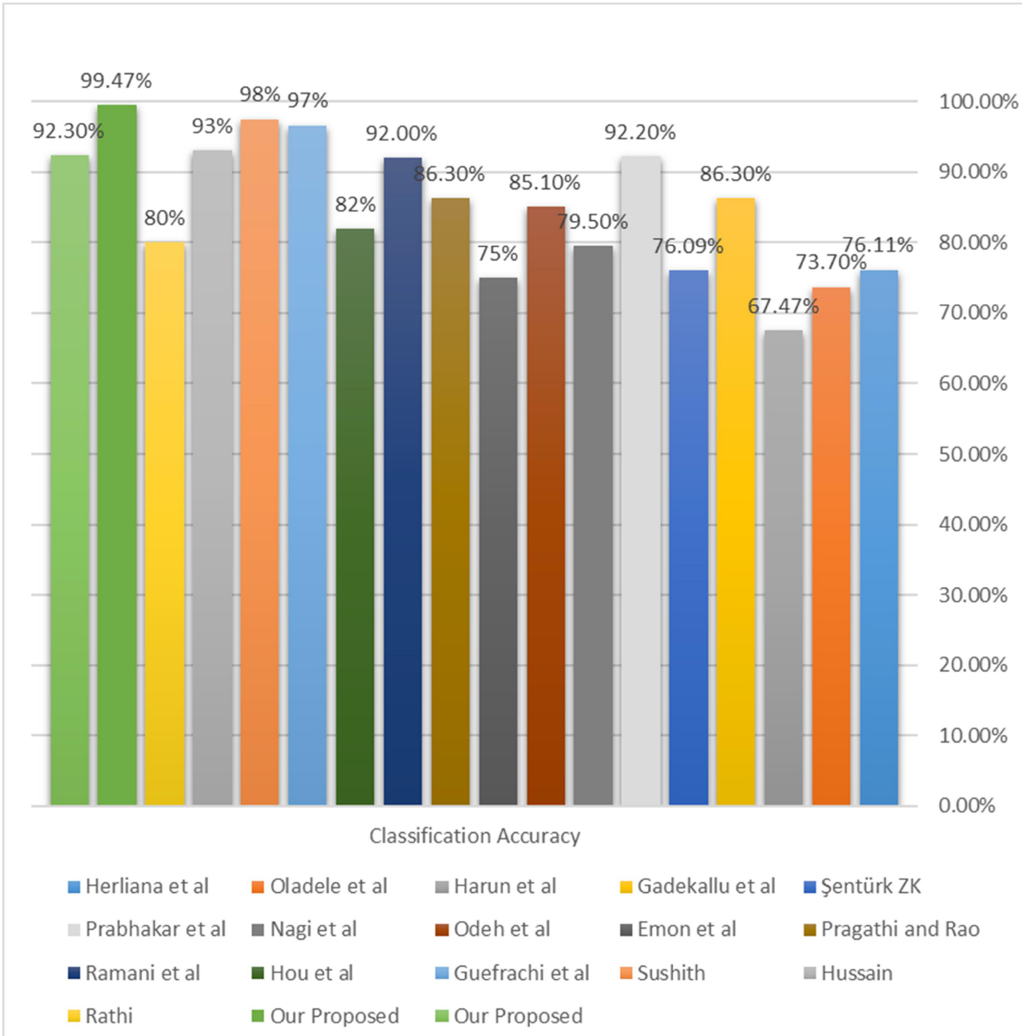

Fig. 8. Compare the results of the proposed work with previous works.

6, 15, and 25 for accuracy. This fluctuation suggests that the model's performance may not be completely stable against the effects of small environmental factors or random data splitting. This requires further robustness testing across larger runs or the application of more advanced data splitting techniques.

Conclusion

Diabetic retinopathy is considered a serious complication of diabetes, which can cause blindness and vision loss if it is not diagnosed and treated early. Consequently, solving this problem has a positive effect on healthcare systems. This work aimed to enhance the ability of K-Nearest Neighbors (KNN) for classifying the patterns of diabetic retinopathy. To achieve this, two enhancements for KNN are adopted based on the Principal Component Analysis (PCA) and Dragonfly algorithm (DA), resulting in two improved versions, namely KNN-PCA and KNN-PCA-DA. In these enhancements, the PCA works to reduce and simplify the datasets, taking into account the preservation of the most important data. The DA works to manage the KNN parameter (k), which has a great impact on the performance of KNN. To verify the effectiveness of the proposed model, two different Diabetic Retinopathy Debrecen datasets have been adopted. The experimental results demonstrated that the KNN-PCA-DA outperformed others, as it achieved superior performance in comparison to the traditional KNN and KNN-PCA (with an accuracy of 99.47%). Given the accuracy achieved by the proposed work, we recommend applying it to other diseases in future work. Furthermore, incorporating deep learning models may contribute to improved classification, especially in complex cases.

Acknowledgment

The authors express their gratitude to all the editors of this prestigious journal, especially the Editor-in-Chief.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images that are not ours have been included with the necessary permission for republication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- The authors state that no human samples or samples were obtained at the time of data collection; Instead, data from eliminated patients were collected through the Medical Dataset Website.
- Ethical Clearance: The project was approved by the local ethical committee at University of Anbar.

Authors' contributions statement

A.A contributed by proposing and implementing the idea. A.J contributed by designing and writing the study. E.T contributed by conducting the statistical analyses.

Data availability

Data was collected from patients through the Medical Data dataset website. (https://archive.ics.uci.edu/dataset/329/diabetic+retinopathy+debrecen).

References

1. Ismail HR, Hassan MM. Bayesian deep learning methods applied to diabetic retinopathy disease: a review. Indones J Electr Eng Comput Sci. 2023;30(2):1167–77. http://doi.org/10.11591/ijeecs.v30.i2.pp1167-1177.
2. Datta P, Das P, Kumar A. Hyper parameter tuning based gradient boosting algorithm for detection of diabetic retinopathy: an analytical review. Bull Electr Eng Informatics. 2022;11(2):814–824. https://doi.org/10.11591/eei.v11i2.3559.
3. Salih N, Ksantini M, Hussein N, Halima D Ben, Razzaq AA, Ahmed S. Deep learning models and fusion classification technique for accurate diagnosis of retinopathy of prematurity in preterm newborn. Baghdad Sci J. Publ online Oct. 2023;21(15):1729–1742. https://doi.org/10.21123/bsj.2023.8747.
4. Kangra K, Singh J. Comparative analysis of predictive machine learning algorithms for diabetes mellitus. Bull Electr Eng Informatics. 2023;12(3):1728–1737. https://doi.org/10.11591/eei.v12i3.4412.
5. Rosu LM, Prodan-Bărbulescu C, Maghiari AL, Bernad ES, Bernad RL, Iacob R, *et al.* Current trends in diagnosis and treatment approach of diabetic retinopathy during pregnancy: A narrative review. Diagnostics. 2024;14(4):1–16. https://doi.org/10.3390/diagnostics14040369.
6. Shrinivasan L, Verma R, Nandeesh MD. Early prediction of diabetes diagnosis using hybrid classification techniques. IAES Int J Artif Intell. 2023;12(3):1139–1148. http://doi.org/10.11591/ijai.v12.i3.pp1139-1148.
7. Ramesh J, Aburukba R, Sagahyroon A. A remote healthcare monitoring framework for diabetes prediction using machine learning. Healthc Technol Lett. 2021;8(3):45–57. https://doi.org/10.1049/htl2.12010.
8. Islam MM, Yang H-C, Poly TN, Jian W-S, Li Y-CJ. Deep learning algorithms for detection of diabetic retinopathy in retinal fundus photographs: A systematic review and meta-analysis. Comput Methods Programs Biomed. 2020;191:105320:1–17. https://doi.org/10.1016/j.cmpb.2020.105320.
9. Alwan JK, Jaafar DS, Ali IR. Diabetes diagnosis system using modified Naive Bayes classifier. Indones J Electr Eng Comput Sci. 2022;28(3):1766–1774. http://doi.org/10.11591/ijeecs.v28.i3.pp1766-1774.
10. Bahnam BS, Abd Dawwod S. A proposed model for diabetes mellitus classification using coyote optimization algorithm and least squares support vector machine. IAES Int J Artif Intell. 2022;11(3):1164–1174. http://doi.org/10.11591/ijai.v11.i3.pp1164-1174.
11. Abdulkareem SA, Radhi HY, Fadil YA, Mahdi H. Soft computing techniques for early diabetes prediction. Indones J Electr Eng Comput Sci. 2022;25(2):1167–1176. http://doi.org/10.11591/ijeecs.v25.i2.pp1167-1176.
12. Raghavendra S, Santosh KJ. Performance evaluation of random forest with feature selection methods in prediction of diabetes. Int J Electr Comput Eng. 2020;10(1):353–359 http://doi.org/10.11591/ijece.v10i1.pp353-359.
13. Wasekar UW, Bathla RK. A review on supervised learning methodologies for detection of exudates in diabetic retinopathy. Indones J Electr Eng Comput Sci. 2021;23(2):837–846. http://doi.org/10.11591/ijeecs.v23.i2.pp837-846.
14. Taifa IA, Setu DM, Islam T, Dey SK, Rahman T. A hybrid approach with customized machine learning classifiers and multiple feature extractors for enhancing diabetic retinopathy detection. Healthc Anal. 2024;100346:1–17. https://doi.org/10.1016/j.health.2024.100346.
15. HerlianaA, ArifinT, SusantiS, HikmahAB. Feature selection of diabetic retinopathy disease using particle swarm optimization and neural network. In: Proceedings of the 2018 6th International Conference on Cyber and IT Service Management (CITSM); 2018 Aug7–9; Parapat, Indonesia. Piscataway (NJ): IEEE. 2018. p. 1–4. https://doi.org/10.1109/CITSM.2018.8674295.
16. HarunNH, YusofY, HassanF, EmbongZ. Classification of fundus images for diabetic retinopathy using artificial neural network. In: Proceedings of the 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT); 2019 Apr9–11; Amman, Jordan. Piscataway (NJ): IEEE/ 2019. p. 498–501. https://doi.org/10.1109/JEEIT.2019.8717479.
17. Gadekallu TR, Khare N, Bhattacharya S, Singh S, Maddikunta PKR, Ra I-H, *et al.* Early detection of diabetic retinopathy using PCA-firefly based deep learning model. Electronics. 2020;9(2):1–16. https://doi.org/10.3390/electronics9020274.
18. OdehI, AlkasassbehM, AlauthmanM. Diabetic retinopathy detection using ensemble machine learning. In: Proceedings of the 2021 International Conference on Information

Technology (ICIT); 2021 Jul14–15; Amman, Jordan. Piscataway (NJ): IEEE. 2021. p. 173–178. https://doi.org/10.1109/ICIT52682.2021.9491645.
19. Pragathi P, Nagaraja Rao A. An effective integrated machine learning approach for detecting diabetic retinopathy. Open Comput Sci. 2022;12(1):83–91. https://doi.org/10.1515/comp-2020-0222.
20. Hou J, Jia N, Duan Y. Research on diagnostic model of diabetic retinopathy based on machine learning. In: International Conference on Automation Control, Algorithm, and Intelligent Bionics (ACAIB 2022). SPIE. 2022. p. 286–291. https://doi.org/10.1117/12.2639489.
21. Fraz MM, Remagnino P, Hoppe A, Uyyanonvara B, Rudnicka AR, Owen CG, *et al.* Blood vessel segmentation methodologies in retinal images–a survey. Comput Methods Programs Biomed. 2012;108(1):407–433. https://doi.org/10.1016/j.cmpb.2012.03.009.
22. Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, *et al.* A state-of-the-art survey on deep learning theory and architectures. electronics. 2019;8(3):1–66. https://doi.org/10.3390/electronics8030292.
23. Asuero AG, Sayago A, González AG. The correlation coefficient: An overview. Crit Rev Anal Chem. 2006;36(1):41–59. https://doi.org/10.1080/10408340500526766.
24. Xia P, Zhang L, Li F. Learning similarity with cosine similarity ensemble. Inf Sci (Ny). 2015;307:39–52. https://doi.org/10.1016/j.ins.2015.02.024.
25. Abdi H, Williams LJ. Principal component analysis. Wiley Interdiscip Rev Comput Stat. 2010;2(4):433–459. https://doi.org/10.1002/wics.101.
26. Mirjalili S. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput Appl. 2016;27:1053–1073. https://doi.org/10.1007/s00521-015-1920-1.
27. Rahman CM, Rashid TA, Alsadoon A, Bacanin N, Fattah P, Mirjalili S. A survey on dragonfly algorithm and its applications in engineering. Evol Intell. 2023;16(1):1–21. https://doi.org/10.1007/s12065-021-00659-x.
28. Meraihi Y, Ramdane-Cherif A, Acheli D, Mahseur M. Dragonfly algorithm: a comprehensive review and applications. Neural Comput Appl. 2020;32(21):16625–16646.
29. EmonMU, ZannatR, KhatunT, RahmanM, KeyaMS. Performance analysis of diabetic retinopathy prediction using machine learning models. In: Proceedings of the 2021 6th International Conference on Inventive Computation Technologies (ICICT); 2021 Jan20–22; Coimbatore, Tamilnadu, India. Piscataway (NJ): IEEE. 2021. p. 1048–1052. https://doi.org/10.1109/ICICT50816.2021.9358612.
30. Oladele TO, Ogundokun RO, Kayode AA, Adegun AA, Adebiyi MO. Application of data mining algorithms for feature selection and prediction of diabetic retinopathy. In: Computational Science and Its Applications – ICCSA 2019: 19th International Conference, Saint Petersburg, Russia, July 1–4, 2019. Proceedings, Part V. Cham: Springer International Publishing; 2019. p. 716–730. https://doi.org/10.1007/978-3-030-24308-1_56.
31. NagiAT, AwanMJ, JavedR, AyeshaN. A comparison of two-stage classifier algorithm with ensemble techniques on detection of diabetic retinopathy. In: Proceedings of the 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA); 2021 Apr6–7; Riyadh, Saudi Arabia. Piscataway (NJ): IEEE. 2021. p.212–215. https://doi.org/10.1109/CAIDA51941.2021.9425129.
32. Şentürk ZK. Artificial neural networks based decision support system for the detection of diabetic retinopathy. Sakarya Univ J Sci. 2020;24(2):424–431. https://doi.org/10.16984/saufenbilder.630482.
33. Rathi P, Rungta S, Ghosh SM. Automated diabetic retinopathy prediction using machine learning classifiers. NeuroQuantology. 2022;20(20):2303–2317. https://doi.org/10.48047/NQ.2022.20.20.NQ109232.
34. Prabhakar T, Rao TVM, Maram B, Chigurukota D. Exponential gannet firefly optimization algorithm enabled deep learning for diabetic retinopathy detection. Biomed Signal Process Control. 2024;87:105376. https://doi.org/10.1016/j.bspc.2023.105376.
35. Ramani R, Raja SE, Dhinakaran D, Jagan S, Prabaharan G. MapReduce based big data framework using associative kruskal poly kernel classifier for diabetic disease prediction. MethodsX. 2025;103210. https://doi.org/10.1016/j.mex.2025.103210.
36. Guefrachi S, Echtioui A, Hamam H. Diabetic retinopathy detection using deep learning multistage training method. Arab J Sci Eng. 2025;50(2):1079–1096. https://doi.org/10.1007/s13369-024-09137-9.
37. Sushith M, Sathiya A, Kalaipoonguzhali V, Sathya V. A hybrid deep learning framework for early detection of diabetic retinopathy using retinal fundus images. Sci Rep. 2025;15(1):15166. https://doi.org/10.1038/s41598-025-99309-w.
38. Hussain M, Ahmed HA, Babar MZ, Ali A, Shahzad HM, Rehman SU, *et al.* An Enhanced Convolutional Neural Network (CNN) based P-EDR Mechanism for Diagnosis of Diabetic Retinopathy (DR) using Machine Learning. Engineering, Technology and Applied Science Research. 2025;15(1):19062–19067. https://doi.org/10.48084/etasr.8854.
39. Lateef AAA, Abdalkafor AS, Nafea AA. Optimized KNN algorithm for diabetic retinopathy classification with PCA-based data fusion and cuckoo search optimization. J Intell Syst Internet Things. 2025;15(1):122–32. https://doi.org/10.54216/JISIoT.150110.

تحسين أقرب الجيران K على أساس خوارزمية اليعسوب لتصنيف اعتلال الشبكية السكري

احمد صبحي عبد الغفور [1]، علاء عبد القهار جهاد[2]، عصام طه ياسين[1]

[1]كلية علوم الحاسوب وتكنولوجيا المعلومات ، جامعة الانبار، الانبار، العراق.
[2]مركز الحاسبة، جامعة الانبار، الانبار، العراق.

الخلاصة

لقد ثبت أن أقرب الجيران (KNN) طريقة فعالة لمعالجة مشاكل التصنيف. يعتمد أداء خوارزمية KNN بشكل كبير على قيمة المعلمة K، والتي تمثل عدد أقرب الجيران. يمكن أن يؤثر اختيار قيمة غير مناسبة لـ K على دقة التصنيف لأن اختيار K الأصغر يمكن أن يؤدي إلى الإفراط في التجهيز والعكس صحيح. لذا، فإن الاختيار المناسب لقيمة K له تأثير كبير على أداء KNN. يعد تعديل قيمة K يدويًا عملية صعبة للغاية لأن الخيارات المناسبة لهذه القيمة تعتمد على حالة البحث. وفقًا لذلك، لا تزال الحاجة إلى استخدام تقنية الضبط عبر الإنترنت موجودة. إحدى الخوارزميات الحديثة هي خوارزمية (Dragonfly (DA، والتي تمكنت من حل العديد من المشكلات التوافقية. في هذا العمل، تم اعتماد DA لتحديد القيمة الأنسب لـ K لخوارزمية KNN تلقائيًا. بالإضافة إلى ذلك، تم تحسين أداء النموذج المقترح من خلال استخدام تحليل المكونات الرئيسية (PCA) لاستخراج الميزات. يُنتج هذا التكامل خوارزمية هجينة تُسمى (-KNN-PCA DA) . يُعد اعتلال الشبكية السكري(DR) أحد الأمراض المزمنة التي تُسبب العمى، وهو أحد الأسباب الرئيسية للعمى. يُعد التشخيص المبكر والدقيق لاعتلال الشبكية السكري ضروريًا للعلاج المبكر والوقاية من فقدان البصر غير القابل للعكس. لذلك، يُقيّم أداء النموذج المقترح باستخدام مجموعة بيانات اعتلال الشبكية السكري في ديبريسين. تُظهر النتائج التجريبية المُحصلة أن النموذج المقترح يُعد حلاً فعالاً لمشكلة اعتلال الشبكية السكري، حيث حقق نتائج جيدة وتنافسية، بدقة بلغت (99.47%) مقارنةً بالنماذج الأخرى.

الكلمات المفتاحية: اعتلال الشبكية السكري، خوارزمية اليعسوب، أقرب الجيران (K)، التعرف على الأنماط، تحليل المكونات الرئيسية.