DOI: 10.15849/IJASCA.250730.10

Int. J. Advance Soft Compu. Appl, Vol. 17, No. 2, July 2025
Print ISSN: 2710-1274, Online ISSN: 2074-8523

Intensified Diabetic Retinopathy Severity Grading: Leveraging Weighted Prominence Learning KNN System

[*1a,b]I. S. Hephzi Punithavathi, [2]Martin Margala, [3]Siva Shankar S, [4]Prasun Chakrabarti

[1a]Post Doctoral Researcher, University of Louisiana at Lafayette,USA.
[1b]Associate Professor, Department of CSE (AI & ML), Vidya Jyothi Institute of Technology, Hyderabad, India.
[2]Director School of Computing & Informatics, University of Louisiana at Lafayette,USA
[3]Associate Professor, & IPR Head, Department of Computer Science & Engineering, K. G. Reddy College of Engineering, Hyderabad, India.
[4]Professor, CSE, Sir Padampat Singhania University, Udaipur, Rajasthan, India.
[*]Corresponding mail ID: hephzi21@gmail.com

Abstract

Diabetic retinopathy (DR) represents a significant complication of diabetes that threatens vision due to excess glucose damaging retinal blood vessels. Without timely treatment, this condition can lead to complete vision loss. Early screening and monitoring are crucial for prevention. Several, traditional diagnostic methods employed by ophthalmologists are often labor-intensive, costly, and prone to errors. To enhance DR detection and severity assessment from retinal images, various Artificial Intelligence (AI) methods utilizing computer vision are currently being developed. Despite the promise of AI, many machine learning approaches encounter difficulties with the complex features inherent in retinal images, resulting in low accuracy, particularly in the early stages of DR. In response to these challenges, the proposed work introduces a Weighted Prominence Learning K-Nearest Neighbors (WPL-KNN) method integrated with VGG16 and ResNet50 classifiers, labelled as VGG16-WPL-KNN and ResNet50-WPL-KNN. This new system of assigning weight helps proposed model better capture the intricate aspects, particularly during the initial phases of the illness, resulting in enhanced effectiveness. It utilizes the Indian DR Image Dataset (IDRID), which comprises high-quality retinal images, to improve diabetic retinopathy classification outcomes through effective feature extraction using VGG-16 and ResNet-50.To further enhancing performance, the Seagull Optimization Algorithm (SOA) and Pelican Optimization Algorithm (POA) are applied for hyper parameter tuning. The VGG16-WPL-KNN model achieves an impressive accuracy of 0.99, while the ResNet50-WPL-KNN model achieves an accuracy of 0.96, showcasing the VGG16-WPL-KNN model's accuracy in classifying DR classes. This research contributes to the fields of ophthalmology, pathology, and diabetology by providing a more efficient and accurate means of diagnosing diabetic retinopathy, thereby offering hope for saving the vision of countless people globally.

Keywords: *Deep Learning, Artificial Intelligence, Computer vision, Diabetic retinopathy, Hyper parameter Tuning, K-Nearest Neighbors*

Received 24 February 2025; Accepted 4 April 2025

1 Introduction

Diabetic retinopathy (DR) is a severe complication of diabetes, causing significant vision impairment due to damage from excess sugar in the body, particularly the eyes [1]. Micro aneurysms develop in the retina, and if untreated, the condition may worsen, leading to more severe damage and potential total vision loss. Early screening and monitoring can reduce this risk, especially for high-risk patients. Traditional diagnosis is labor-intensive and error-prone, involving ophthalmologists examining digitized retinal fundus images. To address these challenges, deep learning (Kodakandla) approaches are being explore [2]. However, promising results in medical image analysis and classification have highlighted the importance of computer vision techniques for automated diabetic retinopathy (DR) detection from retinal images. Similarly, many methods struggle with pre-processing and identifying intricate features, resulting in reduced accuracy, especially in early DR stages. So computer-assisted diagnosis could enhance large-scale diabetes screenings and improve diagnostic efficiency. Recent advancements in computational power, communication systems, and ML methodologies have boosted the application of AI in the medical [3]. On the other hands, feature descriptors [4] from multiple pre-trained DCNN are utilized to process retinal images. Jyostna and her team used Deep CNN composite learns independently from these filtered representations through several channels, enhancing model generalization. When tested on the APTOS-2019 blindness detection task from Kaggle, the DCNN method outperforms other prior [5]. Similarly, collected digital fundus images from common optometric hospitals in Pakistan and framed a DR-Insight dataset and developed the Residue Dense System for DR classification. To enhance the RDS-DR DNN model, residual and dense blocks have been incorporated, along with a transition layer, into a DNN [6]. The DNN system has been trained on fundus images of retina. The prevailing DNN classification method showed better accuracy when compared with other systems. However, the model should use other augmentation techniques, analysis of new datasets using NASNet, Mobile Net, or Efficient Net [7].

In recent times, proposed hybrid CNN-VGG approach combining a deep CNN [8] system and 2 VGG systems namely VGG19 and VGG16 utilized for the detection and categorization of DR based on the severity of DR. However, the hybrid CNN-VGG [9] approach was used to perform multiclass [10] classification utilizing 13,673 images from 9598 patients, categorized into six classes by seven graders based on quality of the images and DR severity. Additionally, 757 DR images were chosen to annotate four kinds of DR-related lesions. The hybrid CNN-VGG method achieved a DR classification has achieved high accuracy [11]. On the other hand, a novel automatic approach based on DL has been suggested for severity detection using a single Color Fundus Photograph (CFP) with enhanced DenseNet169 model developed [12]. Which has been featured with the encoder for embedding. Convolutional Block Attention Module (CBAM) is integrated into the encoder to improve its classifying capability [13]. It is then trained using cross-entropy loss on the APTOS Kaggle dataset. As a result, improvements in severity grading are necessary for the system to be favorable in the medical [12]. Subsequently, the traditional models have achieved successful outcomes without introducing new elements.

The classification of diabetic retinopathy (DR) has been a challenging area due to the limitations of existing methods. Classical classifiers have primarily focused on binary classification, which fails to address the complexities of effective severity classification [14]. This shortcoming results in systems that often lack the necessary accuracy for medical applications, revealing a pressing need for improvements in severity classification. A significant limitation of current approaches is the insufficient emphasis on dataset pre-processing and the utilization of diverse, region-specific datasets. These elements are

critical for enhancing model accuracy and reliability in clinical applications [15-18]. Additionally, traditional K-nearest neighbors (KNN) classifiers treat all neighbors equally, regardless of their distance from the query point. This characteristic makes KNN sensitive to noise and outliers, further complicating the classification process.
In response to these challenges, the proposed research introduces Weighted Prominence Learning KNN (WPL-KNN), which enhances the traditional KNN approach by assigning weights to neighbors. This method typically gives more influence to closer points through a weighting scheme based on inverse distance. By doing so, WPL-KNN becomes more robust against noise and has the potential to improve classification accuracy by emphasizing the most relevant data points during prediction. The research also proposes two advanced classification systems namely, VGG16-WPL-KNN and ResNet50-WPL-KNN, designed for multiclass classification of DR. Both ResNet-50 and VGG-16 are utilized for feature extraction, while (SOA) and (POA) are applied for hyper parameter tuning. The IDRID dataset serves as the foundation for these models, aiming to enhance both accuracy and computational efficiency in DR diagnosis. Ultimately, this research seeks to leverage Weighted Prominence Learning within the classifier system to improve its efficiency, accuracy, and precision. The performance of the proposed model will be evaluated using appropriate metrics to assess its effectiveness in clinical applications, thereby contributing significantly to ophthalmological, pathological, and diabetological research related to DR diagnosis and treatment. Similarly, the novel weighting mechanism enhances the influence of closer neighbors, thus improving prediction accuracy, especially in cases where class distributions are imbalanced. Unlike standard methods that apply uniform weights, this proposed approach dynamically adjusts weights based on neighbor distances and class prominence, allowing for more nuanced decision-making.

The structure of the research paper is organized around the effective techniques used in the classification of DR. Section II presents an analysis of research conducted this field. The methodology implemented in the projected research is outlined in Section III. It is followed by Section IV that illustrates the results achieved by the respective model and a detailed discussion. Finally, Section V describes the conclusion and future work of the proposed method

2 Related Work

In the early stages of diabetic retinopathy (DR), symptoms are often not noticeable, making timely diagnosis challenging. A key characteristic of this condition is the development of swollen structures known as microaneurysms in the retina[16]. If left untreated, the condition can worsen, leading to the formation of severe blotches and damage to the retinal vessels. This deterioration can ultimately result in total vision loss, underscoring the critical importance of early detection and intervention in managing the disease effectively [17]. Early screening and monitoring of diabetic retinopathy (DR) are crucial for mitigating the risk of vision loss in high-risk patients. However, the human detection and classification of DR severity present significant challenges due to the complexity of images captured through color fundus photography. Traditional machine learning algorithms, when combined with specific feature extraction techniques, have been employed to detect and classify DR levels; however, these methods often fall short of achieving satisfactory accuracy. The advent of deep learning (Kodakandla) methodologies has revolutionized the field of computer vision, enabling remarkable precision levels in image analysis. DL technologies mimic the human brain and visual system, significantly enhancing the detection and categorization of DR with the integration of DL techniques into computer vision, achieving high accuracy in diagnosing DR has become feasible. These deep learning models, which effectively represent the functionalities of the human brain and eyes, have greatly improved the accuracy of DR detection and classification, facilitating earlier intervention and better patient outcomes [15]. A recent research focused to detect DR at early stages using Ensemble Learning Decision Tree (ELDT) study is the APTOS 2019 BD dataset. The study outlines a new approach for DR diagnosis, which is based on the extraction of texture features and intensity of grey-level of fundus images utilizing ELDT method. The ELDT method achieves higher accuracy [19].

Vision loss due to DR can be controlled by DL aided detection systems that aid ophthalmologists, which rely on observational image analysis that are laborious and tedious. The prior DL work classifies DR into five severity levels, from 0 to 4. Grade 0 represents the non-DR category, grade 1 indicates mild symptoms, grade 2 signifies moderate NPDR severity, grade 3 denotes severe DR symptoms, and Proliferative DR symptoms are represented by grade 4. The corresponding research for screening DR has utilized the SVM model which has demonstrated promising results with DRiD dataset. Also, SVM system [19] has employed a Gaussian kernel and achieved better accuracy. These results indicate that the system could serve as an effective tool for automated DR screening, as it has shown a significant enhancement in the outcomes achieved [20].

DR involved in varying degrees of severity, from moderate to severe can be detected with SqueezeNet DCNN methods [21]. The study in focus proposes a unique approach to the multilevel severity classification of DR, leveraging SqueezeNet and DCNN in two stages. In the SqueezeNet DCNN methodology SqueezeNet is utilized to categorize the images. Subsequently, DCNN is used in the second level decomposition to ascertain the severity level for the abnormal images. The SqueezeNet DCNN method has shown greater accuracy, sensitivity, and specificity. However, the SqueezeNet DCNN system needs to further improve the performance metrics to be suitable for clinical application. [22] Also DR treatment focuses on preserving the vision level from deteriorating further, facilitation of the early detection is crucial for effective vision maintenance. The difficulty of DR detection is the manual diagnosis is costly, and labor-intensive, requiring an ophthalmologist to examine retinal fundus [23] images of the eye, hence the existed model has pre-trained DenseNet-121and ResNet 50, serving as the main feature extractors and Random Forest (RF) [24] and AdaBoost (AB) which ensemble methods to classify DR. The research reveals that other prior methods achieved high accuracy. However, the RF-AB ensemble system requires an improved dataset and accuracy [25].

The risk of developing DR is around 18% in India and approximately 28.5% in the US. So the existed model that utilizes ResNet-50 for feature extraction and feeds it into a Random Forest [RF] for classification using two divisions of the Messidor-2 dataset [26] and achieved better accuracy. However, it is evident that the ResNet50-RF model underperforms when applied to different datasets [27]. It demonstrated that the ResNet-CNN framework, trained using the IDRID which comprises three levels of DR can perform significantly better than most other methods [28]. The ResNet-CNN method starts with pre-processing of dataset images, including augmentation and normalization of the intensity. For grading, the pre-processed images were fed into the ResNet-CNN [29] system to extract a compact feature vector for identifying the DR lesions. However, the accuracy is still not sufficient and needs improvement for effective use in clinical procedures [30]. Similarly, an existed model utilized to classify DR into five stages of no DR, mild DR, moderate DR, severe DR, and PDR, the model have been trained with three types, back propagation NN, Deep Neural Network (DNN) and Convolutional Neural Network (CNN) the target class thresholds weighted Fuzzy C-means algorithm has been used. The model will be helpful to identify the proper class of severity of diabetic retinopathy images [31]. On the other hand another study combined through VGG16 architecture for feature extraction with XGBoost classifier for enhanced performance. Which has Explored DenseNet121 for its efficient feature extraction capabilities in DR detection and classification. Also, focused on creating a hybrid model to leverage the strengths of both deep learning and gradient boosting techniques can significantly improve the efficiency and accuracy of DR diagnosis [32]. Consequently, the other DL model which has utilize three stages of deep learning for DR classification namely, Image Quality Assessment Sub-network which evaluates image quality for gradability, Lesion-Aware Sub-network which detects and segments retinal lesions such as microaneurysms and haemorrhages, and DR Grading Sub-network for classifying images into non-DR, mild NPDR, moderate NPDR, severe NPDR, or PDR [33]. Accordingly, a multistage framework integrating a Hybrid Fuzzy-KNN (HF-KNN) classifier was developed for grading diabetic retinopathy (DR). The framework utilized pre-processing techniques to enhance retinal image quality and employed region-specific feature extraction to identify critical lesions. Evaluation on a benchmark dataset demonstrated high classification accuracy and proven outperformance than traditional methods, showcased its effectiveness in managing DR has implemented [34]. Another work has implemented hybrid model that integrated Hierarchical Block Attention [35] and HBA-U-Net architectures. This approach enhanced image processing by focusing on pixel details and spatial relationships while employing a multi-stage strategy for data pre-processing, feature extraction, and classification using Improved SVM-Radial Basis Function (ISVM-RBF), DT and (KNN). The model has been tested on the IDRiD dataset, the model achieved satisfied accuracy [36]. Likewise, in study [37] the method involved employing pre-trained networks (VGG19, ResNet101, Shuffle Net) for feature extraction through transfer learning, followed by classification using a KNN algorithm with PCA for feature reduction. The results have showed that the ResNet101-based feature extraction combined with the KNN classifier achieved a better classification accuracy, indicated the effectiveness of the hybrid method in identifying DR abnormalities.

From the review of the recent researches, the problems concerned are specified as follows,

- Classical classifiers have primarily focused on utilizing various architectural configurations to enhance the performance of binary classification, which indicates the presence or absence of diabetic retinopathy (DR) [22, 29].
- However, the effective classification of the disease's severity remains an unresolved task, as the accuracy of existing severity classification systems has not been sufficient to meet the requirements of the medical industry [22, 29]
- The performance in severity classification has shown to be quite low and necessitates significant improvement [15- 18].
- Furthermore, there has been a lack of emphasis on dataset pre-processing and the efficient utilization of newly available datasets that are relevant to specific regions, which are critical for enhancing model accuracy and reliability in clinical applications [27-29].

3 The Proposed Model

The methodology integrates advanced DL architectures, such as VGG-16 and ResNet-50, with innovative feature optimization techniques and WPL-KNN classifiers for the precise classification of DR severity from retinal images. ResNet-50 and VGG-16 DL algorithms are used for feature extraction which identifies and isolates crucial patterns and information from raw data, facilitating better understanding and analysis by the classifier. The method involves data pre-processing and feature extraction, the system utilizes algorithms such as Seagull Optimization Algorithm (SOA) and Pelican Optimization Algorithm (POA) to tune the model hyper parameters. Utilizing natural behaviour of seagull and pelican, the number of features in a model are optimized by selecting the most relevant and informative features for classification tasks. To achieve better DR Severity grading the VGG16-WPL-KNN, ResNet50-WPL-KNN model utilizes Leveraging Weighted Prominence Learning KNN system which assigns weights (numerical value) to data points based on their frequency, addressing class imbalance and enhancing model performance in classification tasks. Therefore, the suggested model is effectively trained in a seamless manner. By using pre-trained CNN architectures like VGG16 and ResNet50 for extracting features, and then training the K-NN classifier with these features. The CNN models remain fixed during feature extraction to make use of their learned representations, while the K-NN classifier is trained using the extracted feature vectors. This combination approach enables to take advantage of the benefits of both deep learning and traditional machine learning methods. The figure 1. Illustrates the overall flow of the proposed work.

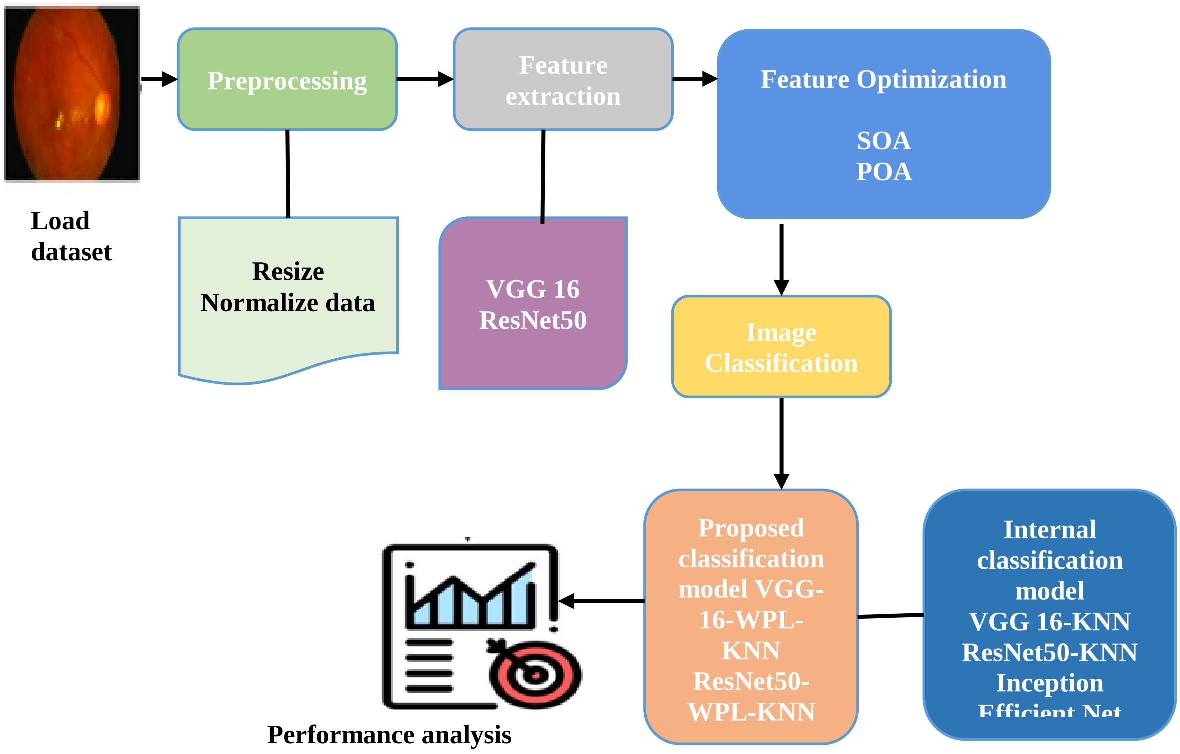


Figure.1 Overall flow of proposed work

Figure.1 signifies the proposed research comprises of the data loading, pre-processing, feature extraction using VGG 16 and ResNet-50, data splitting, feature optimization using SOA and POA Hyper Parameter Tuning, train test split, classification, prediction phase using Resnet50-KNN otherwise VGG16-KNN, results of which are compared with internally developed Efficient Net, Inception, unmodified VGG16-KNN and ResNet50-KNN. Finally, the performance metrics such as Accuracy, Recall, precision and F1 score are calculated. The elaborate description of every phase in the proposed research is given in the following sections.

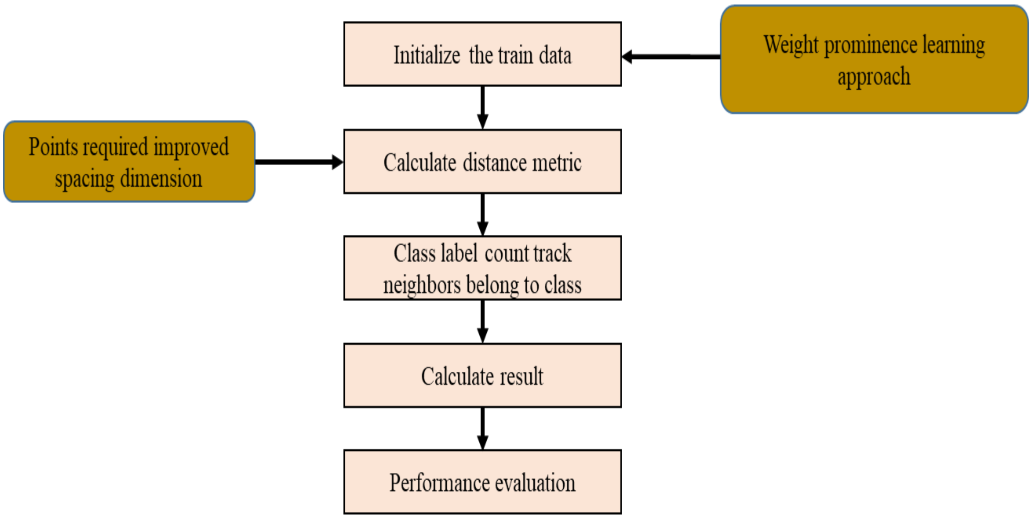


Figure.2 WPL-KNN Model Mechanism

The ResNet50-WPL-KNN and VGG16-WPL-KNN model initiates with attributing weighted prominence to the points in the training data. As given in the Figure.2, the model undergoes training for efficient feature extraction (ResNet-50 and VGG-16) and classification (modified VGG16-WPL-KNN and ResNet50-WPL-KNN). In the datasets, each sample is given a weight that is inversely proportional to its frequency. This approach neutralizes the impact of class imbalance. As a result, the semantic distance between samples determines the number of samples from different classes that will be classified into the same category as the test sample. The distance metric to measure the similarity between data points in the feature space are computed. The distance measures it uses may not be equipped to handle datasets that contain uncertain attribute values. The function assigns higher weights to neighbors with shorter distances and lower weights to those with longer distances. Weight assignment makes sure that the sum of weights for all neighbors is normalized to 1 Closer neighbors contribute more significantly to the final classification decision. WPL-KNN models utilize distance metrics and weight assignment techniques to classify data points accurately based on their feature representations. The model then pinpoints data points that need improved spacing for enhanced classification. During the classification process, the model keeps track of the counts of class labels among the nearest neighbors. Finally, performance metrics are calculated to evaluate the model's accuracy in classifying data points and to assess its ability to generalize.

3.1 Data Selection

The IDRID is a collection of high-quality retinal images. These images have been captured using various imaging techniques such as fundus photography. The primary purpose of the selected dataset is to facilitate the study and prediction of the severity grade of diabetic retinopathy in Indian patients. The dataset is publicly available and consists of 516 retinal fundus images. These images are categorized as follows:

- Retinal images exhibiting signs of DR
- Normal retinal images without signs of DR
- Disease Grading: It includes original color fundus images, with a total of 516 images. These images are divided into a training set (413 images) and a test set (103 images).

3.2 Data Pre-processing

Pre-processing enhances image quality for better classification by reducing distortions and emphasizing key features. Techniques like resizing to 224px×224px and normalization (e.g., Min-Max Scaling) improve dataset cleanliness, facilitating faster training and better model performance through rescaling pixel values.

3.3 Feature Extraction-VGG 16 and ResNet-50 Neural Network

Feature Extraction is a process that identifies and isolates crucial patterns and information from visual data, enabling the network to understand the input.

- ***VGG 16***: The architecture of the VGG 16 model is characterized by a sequence of convolutional layers followed by max-pooling layers, with an increasing depth. In the suggested method, the VGG-16 model is utilized to extract valuable data from the dataset, thereby enhancing the system's efficiency in the classification process.

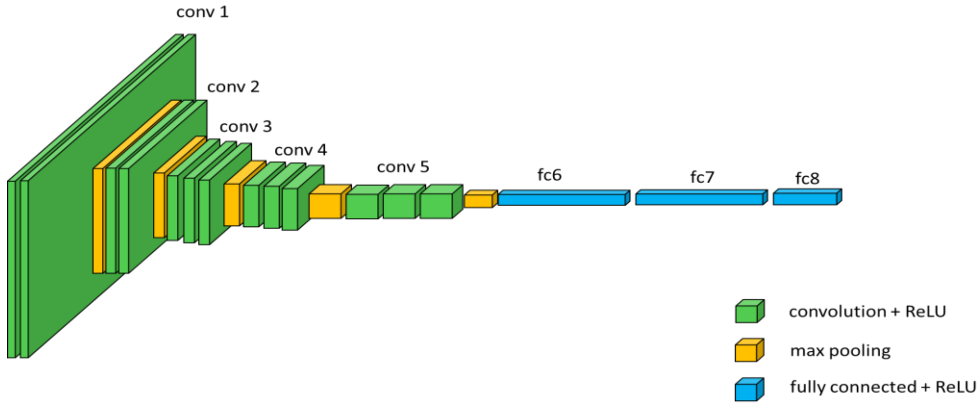


Figure.3 Architecture of VGG 16 System

Figure.3 illustrates the architecture of the VGG-16 model, a type of CNN architecture, known for its depth, comprising 16 layers, which include 13 convolutional layers and 3 fully connected layers. Max pooling layers down-samples convolved features to reduce their spatial size, thereby decreasing computational requirements and enhancing feature abstraction. It operates by selecting the maximum pixel value from a kernel-covered image portion. The VGG-16 model is noted for its simplicity, effectiveness, and its capability to deliver robust performance on a variety of computer vision tasks, such as image pre-processing and classification.

- ***ResNet-50***: The ResNet-50 employs bottleneck architecture for its building block. A bottleneck residual block utilizes 1x1 convolutions, which is a bottleneck that decreases matrix multiplications and parameters. This allows training to be fast for each layer. Instead of two layers, it uses a stack of three layers.

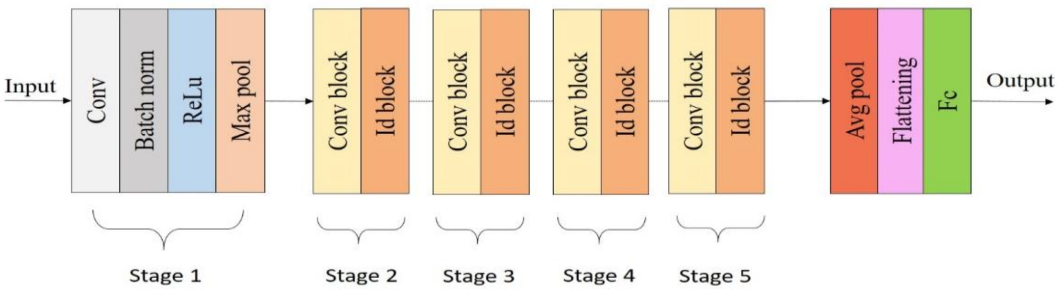


Figure.4 Architecture of ResNet-50 System

Figure.4 illustrates the architecture of ResNet-50, a fifty-layer deep convolutional neural network that begins with a convolutional layer and a max pooling layer. It then includes a series of convolutional layers with varying kernel sizes and quantities, repeated a specific number of times. The Identity block in ResNet-50 comprises three convolutional layers, each succeeded by batch normalization and ReLU activation. It features a skip connection that facilitates direct data flow, helping to address the vanishing gradient problem and ensuring the retention of crucial information across the network. The architecture concludes with average pooling and a fully connected layer comprising 1000 nodes, employing the softmax activation function. This structure allows ResNet-50 to effectively solve the issue of vanishing gradient, improving the training of deeper networks and achieving better accuracy in image classification tasks. However, the latent space analysis from the latent vector method from the feature extractor has utilized for dimensionality reduction, which involves reducing the high-dimensional retinal image data to enable visualization and interpretation of complex features related to DR. also, analysis of the latent space helps in identifying the most influential features for classifying various stages of DR. on the other hand, the latent space will show groups of similar images, showing how various stages of DR are depicted. By examining these groups, researchers can gain a clearer understanding of the advancement of the condition and the significant differences between stages.

3.4 Feature Optimization-SOA and POA Hyper Parameter Tuning

Hyper parameters are configuration settings explicitly defined by the user to control the learning process of machine learning algorithms before training begins. They play a crucial role in determining the performance and behaviour of models, as their appropriate selection can significantly impact outcomes such as convergence speed, accuracy, and generalization to unseen data. The importance of hyper parameters lies in their ability to influence model complexity and effectiveness. For instance, a well-tuned learning rate can prevent issues like under fitting or over fitting, ultimately leading to better model performance**.** So feature optimization is a process that aims to reduce the number of features in a model, thereby decreasing computational complexity and enhancing the model's performance. Despite the existence of numerous optimization algorithms, the model uses the seagull and pelican algorithms separately. The key hyperparameters for the Seagull Optimization Algorithm (SOA) include the number of seagulls (agents), iteration count, and the exploration-exploitation balance factor. The number of agents has influenced the diversity of solutions explored, while the iteration count has determined how thoroughly the search space has been examined. Adjusting the exploration-exploitation balance has proven crucial for avoiding local minima and ensuring comprehensive search behavior. In the case of the Pelican Optimization Algorithm (POA), important hyperparameters encompass the number of pelicans and the learning rate. The number of pelicans dictates solution diversity, similar to SOA, while the learning rate controls how quickly the algorithm adapts to new information. Fine-tuning these parameters has been essential for achieving optimal performance in both algorithms. as they have demonstrated better performance in resolving optimization problems. The hyperparameters, including the number of training epochs, were determined through a combination of empirical testing and cross-validation using the training dataset.

The seagull Optimization algorithm (SOA) [27] mimics the migration behaviour of a seagull population, moving from one location to another. During this phase, each individual seagull must meet three conditions:

- Collision Avoidance: To prevent collisions with other seagulls, the algorithm determines the new position of each seagull using an additional variable, AA. This variable helps control and prevents collisions between neighboring seagulls.
- Best Neighbor Directional Movement: To prevent overlap with other seagulls, each seagull moves in the direction of their optimal neighbor.
- Best Seagull's Location Approach: After moving towards the best neighbor, each seagull then moves in the direction of the global optimum, eventually reaching a new location.

Attacking Behavior: Seagulls are highly intelligent creatures that use their past experiences during the search process. During migration, they often engage in attacking behavior due to the need for prolonged hunting periods. When seagulls attack their prey, they first maintain a high level of stability using their wings and weight. This is followed by a specific spiral motion behavior in flight, where they continuously change their flight speed and attack angle.

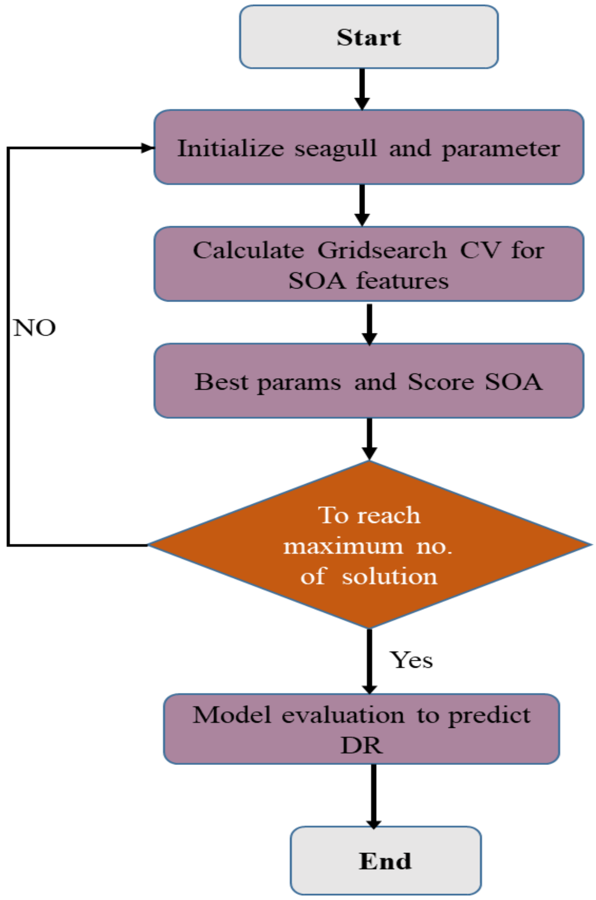


Figure 5. Flow Diagram for SOA optimizer

Figure 5 depicts the flow maintains its foundational structure but introduces key enhancements significantly improving its functionality. It begins with the Start point and progresses to incorporate a Seagull component, marking the initiation of the SOA, essential for optimizing the process. The calculation of GridSearchCV now focuses on SOA features, highlighting the relevance of specific data for this algorithm. The Best Parameters and Score SOA step maintains its function but is specifically tied to SOA features, optimizing model parameters more effectively. The Decision Node remains unchanged, ensuring an iterative exploration of parameter combinations through necessary loops back to the

GridSearchCV step. Lastly, the Model Evaluation phase to Predict Diabetic Retinopathy (DR) persists without alteration. The process culminates in an End marker, signaling the optimized workflow's completion, thus showcasing a more specialized approach to diabetic retinopathy classification through SOA integration.

The POA (Pelican Optimization Algorithm) [28] is designed by mimicking the behavior of pelicans and their predators in the ocean. The fundamental inspiration for the POA comes from the defence behaviour of pelicans with respect to attacks.

POA is a type of population-based algorithm where each member of the population, in this case, pelicans, represents a potential solution to the optimization problem. In these types of algorithms, the proposed values for the problem's variables are determined by the position of each member in the search space. At the start of the process, the members of the population are randomly initialized within the problem's lower and upper bounds.

The POA mimics the tactics and strategies pelicans use when hunting and attacking prey in order to update potential solutions. This hunting strategy is replicated in two phases.

- Exploration Phase: The main goal of using POA is to simulate the natural hunting behavior of a pelican. The hunting strategy in POA is divided into two stages: exploration and exploitation. During the exploration phase, the pelicans initially identify the location of the prey and move towards the detected area. This pelican strategy results in scanning the search space. A key aspect of POA is that the prey's location is randomly generated in the search space, enhancing the exploration power and aiding in achieving an accurate search of the problem-solving space.
- Exploitation Phase: In the exploitation phase, when the pelican reaches the water's surface, it spreads its wings on the water surface to push the fish upwards, and then the prey are gathered in the throat pouch. This strategy results in a higher catch rate in the attacked area. Furthermore, displaying this behaviour of pelicans can lead POA to converge to better points in the hunting zone. Therefore, this process amplifies the power of local search and contributes to the better exploitation capability of Pelican optimization.

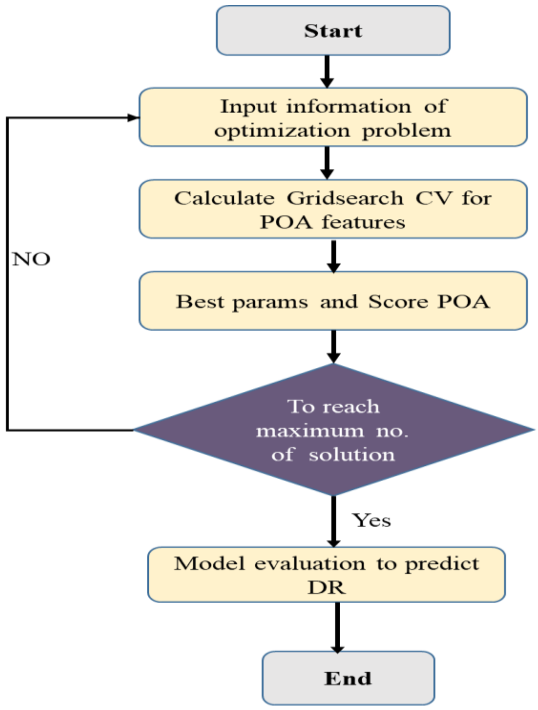


Figure 6. Flow Architecture of the POA algorithm

Figure 6 outlined structure presents a thorough approach to solving an optimization problem, starting with the initiation at the Start point. Next, all relevant data is collected in the Input Information of Optimization Problem, which includes parameters, constraints, and features necessary for the optimization strategy. The process proceeds to Calculate GridSearchCV for POA Features, where hyperparameter tuning is conducted using the GridSearchCV function, with POA referring to specific features for this tuning. The best params & score POA phase then identifies the optimal hyperparameters and scores for the POA features, enhancing the model's performance. A crucial step occurs at the decision node, which checks if conditions regarding the best parameters or scores are met; if not, it loops back for further tuning. Once satisfactory parameters are found, the workflow progresses to model evaluation to predict DR, utilizing the optimized model for performance assessment. The process concludes with an End marker, indicating the completion of this iterative optimization journey focused on effective outcome predictions. POA algorithms have been found to yield superior results in addressing optimization problems compared to other optimization algorithms. POA in particular, is used due to its unique approach.

SOA Hyper parameter Tuning: Hyper parameters are the configurations established for a model before it starts learning from data. They are not derived from the data itself but are set externally. SOA is used extensively for the tuning of the hyperparameters. These hyperparameters play a critical role in directing the learning process and shaping the model behavior during training and predictions. A SOA model is trained to adjust these hyperparameters, enhancing the model's performance and utilizing the data effectively. SOA hyperparameter tuning in the model significantly affects the various aspects of the performance, including prevent over-fitting, and reduce training time.

Table.1: Outcome of SOA Hyperparameter Tuning

Parameter	**Possible values**	**Value**
metric	euclidean, manhattan, minkowski.	euclidean
weights	uniform, distance	Uniform
n_neighbors	3, 5, 7, 9, 11	11

POA Hyper Parameter Tuning: POA is also extensively used for adjusting these hyperparameters. These hyperparameters play a pivotal role in guiding the learning process and influencing the model's behavior during training and prediction phases. The POA models are trained to fine-tune these hyperparameters, which in turn enhances the model's performance and its ability to effectively utilize the data. The tuning of hyperparameters using POA in the model has a significant impact on various performance aspects, including prevention of over-fitting, and reduction in training time.

Table.2 Outcome of POA Hyperparameter Tuning

Parameter	**Possible values**	**Value**
metric	euclidean, manhattan, minkowski.	euclidean
weights	uniform, distance	distance
n_neighbors	3, 5, 7, 9, 11	11

3.5 Data Splitting

Data splitting involves dividing the pre-processed dataset into 2 categories. One category is utilized to train the model and the other is utilized to test or evaluate the system performance. In the current method, 80% of the data is utilized for the model training with the WPL-KNN model, and the remaining 20% is used for testing the efficiency of the proposed research during the prediction phase. Additionally, techniques like cross-validation ensure that each fold maintains the same proportion of classes as the original dataset, further enhancing the robustness of the validation process. Thus, cross-validation is integral to effectively evaluating model performance during data splitting has shown in Table 3.

Table.3 5 fold and 10 fold Cross validation for various metrics.

Technique	Accuracy	Precision	Recall	F1-Score	Quadratic Weighted Kappa
5-Fold Stratified Cross-Validation					
ResNet50-WPL-KNN	0.94 ± 0.02	0.93 ± 0.02	0.94 ± 0.02	0.92 ± 0.02	0.937 ± 0.016
VGG16-WPL-KNN	0.97 ± 0.01	0.96 ± 0.01	0.94 ± 0.02	0.95 ± 0.01	0.975 ± 0.009
10-Fold Stratified Cross-Validation					
ResNet50-WPL-KNN	0.93 ± 0.01	0.92 ± 0.02	0.93 ± 0.01	0.91 ± 0.01	0.925 ± 0.013
VGG16-WPL-KNN	0.96 ± 0.01	0.95 ± 0.01	0.93 ± 0.01	0.94 ± 0.01	0.967 ± 0.007

Table 3 represents the 5 fold and 10 fold cross validation of various metrics namely, accuracy, precision, recall, F1-score and Quadratic weighted kappa of ResNet50- WPL-KNN, VGG 16-WPL-KNN techniques respectively.

3.6 Classification- WPL-KNN classifier –Enhancing Spacing Determination: Leveraging Weighted Prominence Learning

KNN system is a supervised learning algorithm extensively utilized in pattern classification since its development. The core principle of the rule is to maintain the entire training set during the learning phase. For each query, it assigns a class based on the majority label of its k-nearest neighbors within the training set. The KNN algorithm is a kind of non-parametric, supervised learning classifier. KNN carries out classifications concerning the categorization of a specific data point based on its proximity to other points. Similarly, the CNN classifiers VGG16 and ResNet50 were pre-trained on a dataset to extract features using cross-entropy loss. After feature extraction, these models were fine-tuned with a Weighted Prominence Learning K-Nearest Neighbors (WPL-KNN) head, enhancing classification accuracy. Various feature engineering techniques improved KNN performance, utilizing cosine similarity as the distance metric. Also distance Metric cosine similarity is utilized to optimize the KNN classifier. By converting raw data into meaningful features, the model's ability to detect patterns and make accurate predictions increased. Cosine similarity prioritizes data point orientation over magnitude, benefiting high-dimensional spaces. This integration improved predictive performance and overall KNN efficiency. Additionally, hyperparameters such as the number of neighbors (k) were tuned from the set [3, 5, 7, 9, 11], with weights set to 'uniform' or 'distance,' using 'euclidean,' 'manhattan,' or 'minkowski' distance metrics.

Pseudocode.1 KNN
Input: The training set S, test object i, category labels set C **Output**: The class c_i of test object i, c_i belongs to the C 1. Start 2. for each i belongs to S do 3. compute the distance S(j, i) between y and i 4. end for 5. select the subset N in the dataset S, the N includes k samples for training which are the k neighbors nearest to the test object i 6. compute the class of i: $C_i = \sum_{j \in N} \quad I \quad (c = class(j))$ 7. end

Pseudocode.1 describes the KNN algorithm, which is a type of instance-based learning used for classification. The loop iterates over each instance `i` in the training set `S`. For each instance `i`, it calculates the distance between the test object `y` and the current instance `i`. The distance can be calculated using various methods like Euclidean distance, Manhattan distance, etc. This marks the end of the loop. After calculating the distances, it selects a subset `N` from the training set `S`. This subset contains `k` training samples which are the nearest neighbors of the test object `i`. It then determines the category `C_i` of the test object `i`. For each category, it counts the instances of `N` belong to that category and assigns the test object `i` to the category with the most instances. The output of this algorithm is the category `C_i` of the test object `i`. The category is the one that has the most representatives within the `k` nearest neighbors of `i` in the training set.

Weighted Prominence Learning-KNN: The ability of the system to handle uncertain attributes enhances its applicability to DR detection where uncertainties are common. This robustness leads to more reliable predictions, even when dealing with unrelated data. The method assigns weights to neighbors based on their distances, giving more importance to those that are closer. This approach helps capture more relevant information from the training data, potentially leading to more accurate classifications. The WPL-KNN algorithm's flexibility in choosing distance metrics, including the Euclidean distance for certain attributes and the supermom metric for uncertain attributes allows it to better capture the relationships between data points in the feature space. This can lead to improved classification performance. When combined with feature extraction models like ResNet50 and VGG16, the k-NN algorithm can leverage the capability of DL feature representations and its own robustness in handling uncertainties and capturing local structures in the data. This can enhance classification accuracy. The WPL-KNN algorithm demonstrates robustness against data uncertainty and produces satisfactory results.

Pseudocode.2 WPL- KNN
Input: The training set S, test object i, category labels set C **Output: The category c_i of test object i, c_i ∈ C** 1. begin 2. initialize an empty list distances 3. for each j ∈ S do 4. calculate the distance d(j, i) between j and i 5. append (j, d(j, i)) to distances 6. end for 7. sort distances by the distance d(j, i) in ascending order

```
select the subset N from the sorted distances, where N contains the k training samples with the smallest distances
initialize a weight vector W of length k
for each j ∈ N do
    calculate the weight w_j   based on the distance d(j, i)
    store w_j in W
end for
initialize a dictionary category_scores with keys from C and values set to 0
for each j ∈ N do
    let class_j be the class label of j
    increment category_scores[class_j] by w_j
end for
let c_i be the category with the highest score in category_scores
end
```

Pseudocode.2 provides the step by step process of WPL-KNN model, in which a weighted KNN algorithm is used to classify medical images into its severity category. It begins by calculating the distance between a test object and each object in the training set. These distances are stored, sorted, and a subset of the 'k' nearest neighbors is selected. For each of these neighbors, a weight is calculated based on their distance to the test object. These weights are used to increment scores in a category score dictionary, which holds the sum of weights for each category based on the nearest neighbors. The category with the highest score is then selected as the predicted category for the test object. This approach ensures that closer neighbors have a greater influence on the final classification.

The weight assigned to the i^{th} nearest neighbor of the query u' is determined using a distance-weighted function. This function assigns more weight to neighbors that are closer and less weight to those that are farther away.

$$w_i' = \begin{cases} \frac{d(u',u_k^{NN}) - d(u',u_i^{NN})}{d(u',u_k^{NN}) - d(u',u_1^{NN})}, & if\ d(u',u_k^{NN}) \neq d(u',u), \\ 1 \quad , & if\ d(u',u_k^{NN}) = d(u',u_1^{NN}). \end{cases} \tag{1}$$

The neighbor with a shorter distance is given more weight than one with a longer distance: the closest neighbor is assigned a weight of 1, the farthest neighbor a weight of 0, and the weights of other neighbors are scaled linearly within this range.

$$w' = \arg max_w \quad \sum_{(u_i^{NN},w_i^{NN})\in T'} \quad w_i' \times \delta(w = w_i^{NN}) \tag{2}$$

$$w' = d(u',u) \tag{3}$$

The distance of uncertain attributes was computed using Euclidean distance and the uncertain attributes using supermom metric. Then the pairwise distance is modelled as (4)

$$d(u',u) = d1(u',u) + d2(u',u)\ .w' \tag{4}$$

Where, the total distance between u and u' is represented by $d(u',u)$, the distance between certain values are represented by $d1(u',u)$ and the distance between uncertain values are represented by $d2(u',u)$.

Supermom metric in the metric space is defined as (5)

$$d(f,r) = \sup\backslash f\ (t) - r(t)\backslash \tag{5}$$

$$t \in a,b \tag{6}$$

$$for\ all\ f, r\ \in C_{a,b} \quad (7)$$

Here, the supermom metric is utilized to calculate the length across the uncertain attributes known as Tehebyshev metric or uniform metric.

- ResNet50-WPL-KNN Classifier and VGG 16-WPL-KNN Classifiers:

The ResNet50-KNN algorithm is used classification of DR from retinal images. ResNet-50 is a CNN with 50 deep layers, and it's pre-trained on the Image Net dataset. Following pre-processing and features extraction, it is used as input for the KNN algorithm. The ResNet50-KNN algorithm then classifies the test object based on its 'k' nearest neighbors in the feature space. The distance between the test object and the training samples is calculated, and the 'k' training samples with the smallest distances would be selected. Each of these 'k' samples would be assigned a weight based on their distance to the test object, and these weights would be used to determine the category of the test object. This combination of ResNet-50 and KNN potentially leads to improved performance in classifying DR stages, as it combines the feature extraction capabilities with the effectiveness of a ResNet50KNN classifier. The ResNet50-WPL-KNN model was trained on from the IDRID dataset that provides the model with advantages of rich feature representations. This combination can lead to improved classification performance, especially in scenarios involving image data with uncertain attributes.

VGG16 is a CNN that is 16 layers deep, and it's pre-trained on the ImageNet dataset. This is also coupled with modified KNN and is used as the classifier.

The implementation of the WPL-KNN algorithm with ResNet50 and VGG16 potentially leads to better performance in the severity classification of DR. Further section deals with the results and discusses the significant achievements of WPL-KNN models with respect to unmodified methods.

4 Results and Discussion

This section presents the results attained by the suggested research and recent models. It includes Exploratory Data Analysis (EDA), performance metrics, performance analysis and a comparative analysis of the WPL-KNN model.

4.1 Exploratory Data Analysis

The analysis of data is used by the researchers to examine and explore the datasets to summarize the significant characteristics of the datasets.

Figure.7 Input Sample from IDRID Dataset

Figure.7 represents an image of the retina taken from the IDRID dataset which are utilized in the respective research. The dataset contains class imbalance to rectify that normalization method is used.

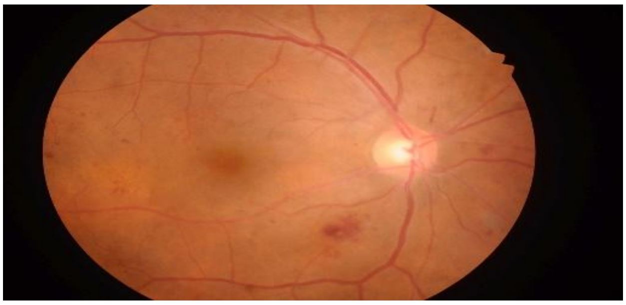


Figure.8 Pre-processed Sample from IDRID Dataset

Figure.8 shows the Pre-processed sample with the trained model which makes the data suitable for high efficiency classification. Images from the IDRID dataset utilised in the respective research is pre-processed which involves resizing the image to the 224px×224px size and normalization technique. This sample is utilized to train the model for the multiclass classification. The training labels of the dataset are shown in the figure.8.

4.2 Performance Metrics

The performances of the VGG16-WPL-KNN and ResNet50-WPL-KNN models are assessed using specific performance metrics to evaluate the effectiveness of the related research. The metrics employed to assess the performance of the system including precision, accuracy, f1-score, and recall are given below,

1. **Precision**: It defines the amount of the positive predictions that are true. It is examined with the true positives that are actually true to the overall positive predictions. It is represented in equation (8),

   $$Precision = \frac{TP}{TP+FP} \quad (8)$$

   Where, TP, FP are True Positive and False Positive.

2. **Sensitivity**: Sensitivity is calculated as the proportion of TP to the sum of TP and False Negatives [38]. Sensitivity is expressed by (9)

   $$Sensitivity = \frac{TP}{TP+FN} \quad (9)$$

   Where, FN is the False Negative.

3. **Accuracy**: It is the significant metric used to determine the correct prediction of the present research to the total amount of predictions. The formula for accuracy is depicted in equation (10),

   $$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (10)$$

4. **Recall**: It is targeted to compute the amount of actual positive that are predicted incorrectly in the system, which is signified in equation (11),

   $$Recall = \frac{TP}{TP+FN} \quad (11)$$

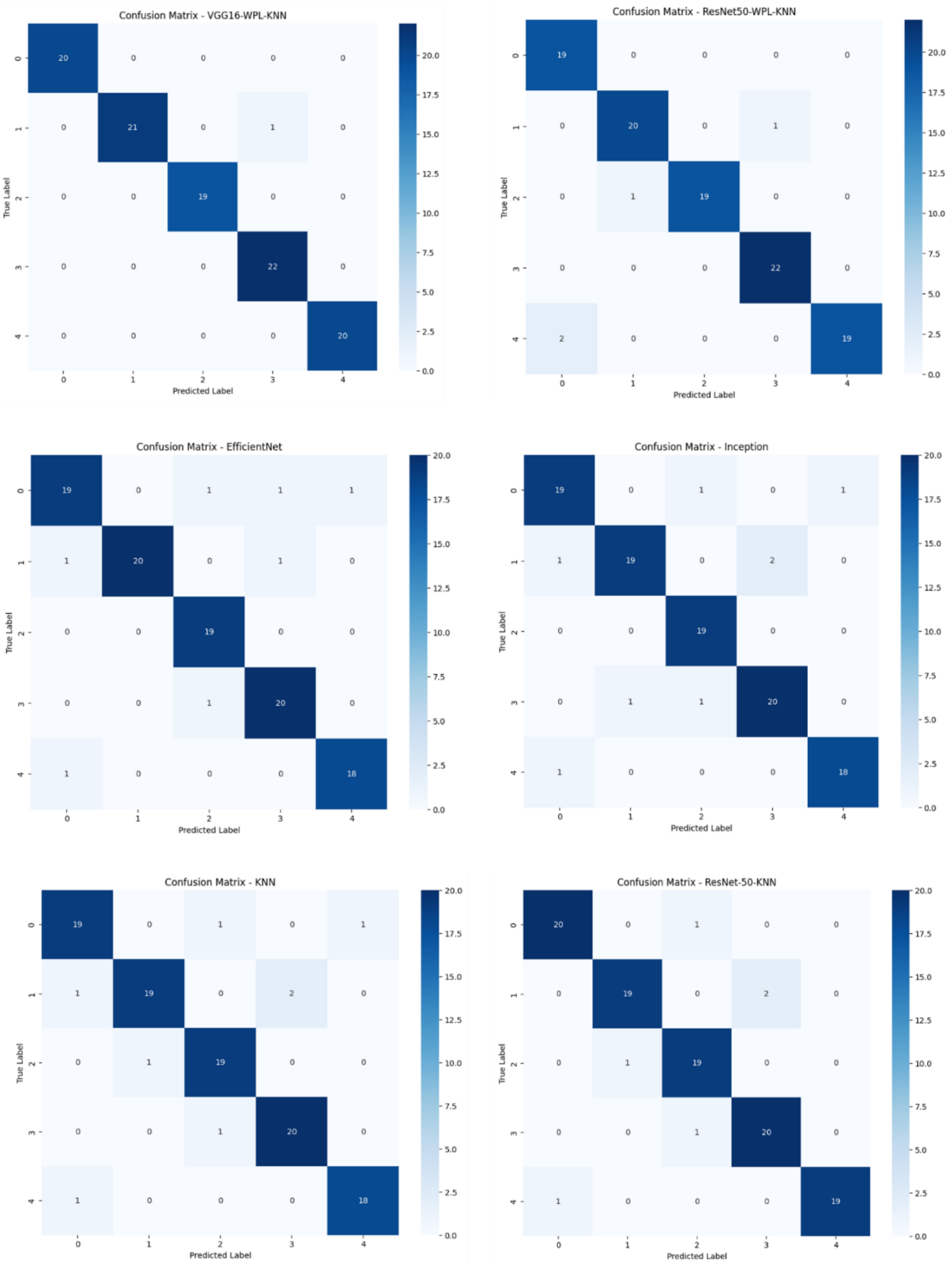
   Where, TP, FN are True positive and False Negative.

5. **F1-Score**: It is used to examine the binary classification through the predictions made for the positive class. Essentially, it is the mean of the recall and precision. The formula for the f1-score is shown in equation (12),

$$F1 - Score = 2 * \frac{Recall*Precision}{Recall+Precision} \tag{12}$$

4.3 Performance Analysis

The performance of the ResNet50-WPL-KNN and VGG16-WPL-KNN models of the respective research is analysed in this section through confusion matrix and ROC curve.



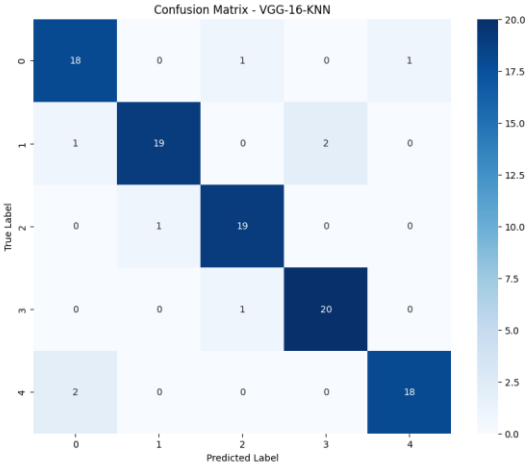
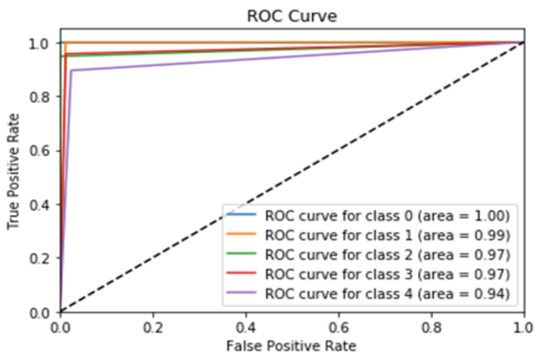


Figure.9 Confusion Matrix of Proposed Models and Other Models

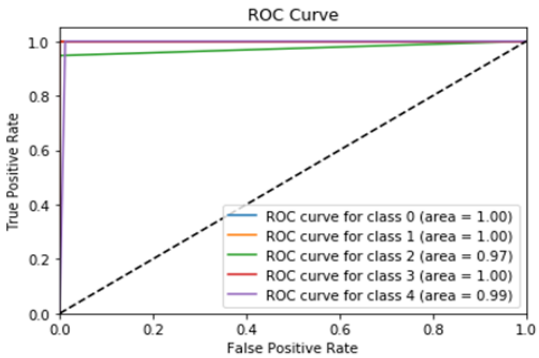
Figure 9 depicts the Confusion matrix of ResNet50-WPL-KNN, VGG 16-WPL-KNN and other classifiers such as Efficient Net, VGG16, ResNet50, KNN, and Inception for multiclass classification of attacks with 5 rows and 5 columns; it shows the improved ability to classify the correct and incorrect classes by the ResNet50-WPL-KNN and VGG 16-WPL-KNN classifiers. It correctly classified a significant number of better than most other models, but overall, it performed well across all classes. The ResNet50-WPL-KNN performed slightly less than VGG16-WPL-KNN. The Efficient Net model has the lowest accuracy among the four. Inception performed given similar results with Efficient Net, Inception has the lowest overall accuracy. It has the weakest performance across all classes. Overall, VGG16-WPL-KNN and ResNet50-WPL-KNN performed better when compared with other models. The following figure.9 presents the ROC curve of the WPL-KNN and comparable models. VGG16-WPL-KNN model shows the best overall performance.

4.4 ROC Curve Comparison

The Receiver Operating Characteristic (ROC) curve is a graphical tool used in binary classification to represent the comparison between the True Positive Rate (TPR) and False Positive Rate (FPR),



ResNet50-WPL-KNN



VGG16-WPL-KNN

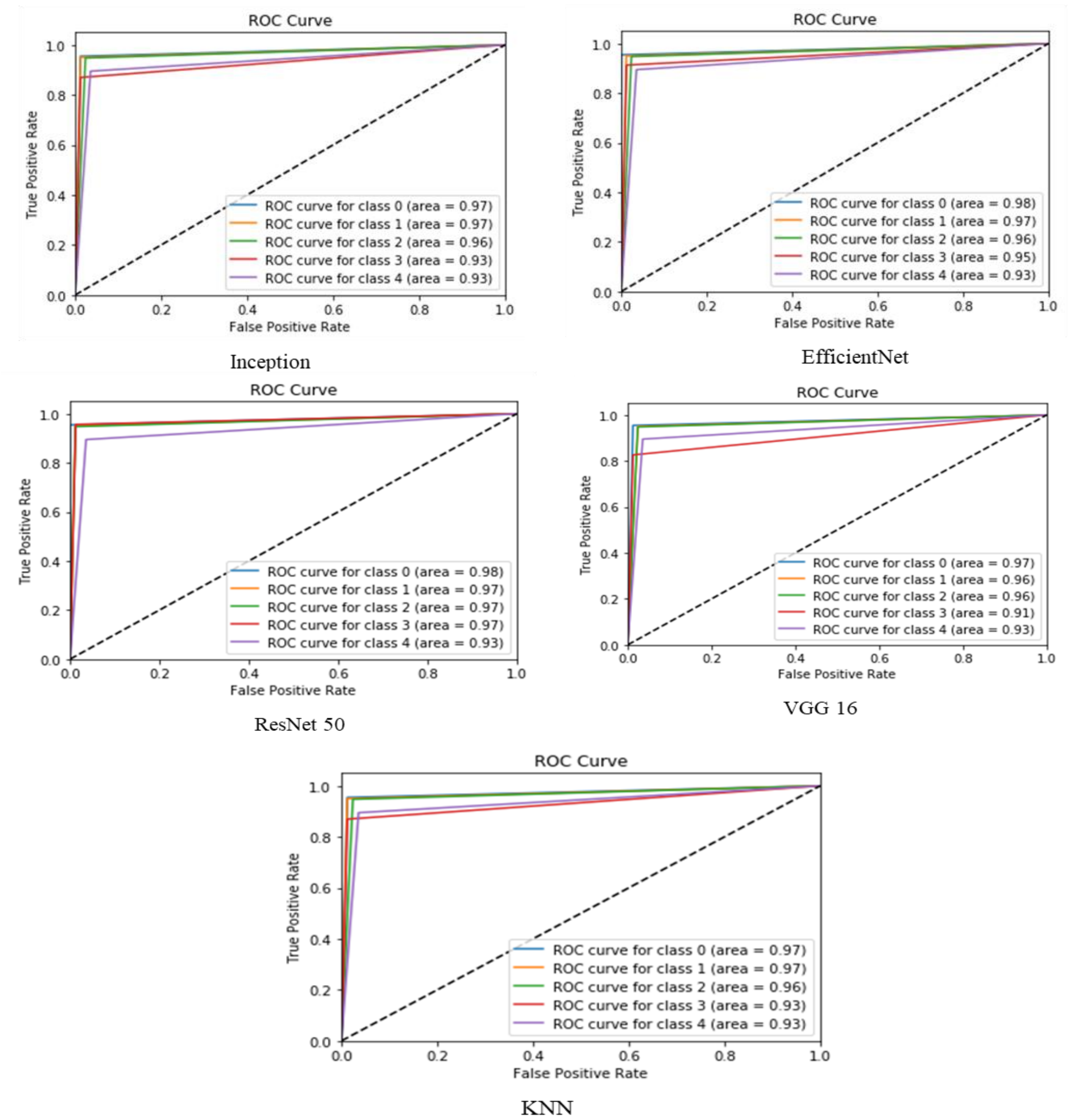


Figure.10 ROC Graph of Proposed Models and Other models

Figure 10 shows the Receiver Operating Characteristics (ROC) curve of the ResNet50-WPL-KNN and VGG 16-WPL-KNN models and comparable models such as Efficient Net, VGG16, ResNet50, Inception, and KNN for class 0, class 1, class 2, class 3 and class 4. ROC Curves plot the TPR on the Y-axis against the FPR on the X-axis. TPR is the proportion of positive cases that were correctly identified, while FPR is the proportion of negative cases that were incorrectly classified as positive. Ideally, a good model will have a ROC curve that goes towards the top-left corner of the graph, indicating both high TPR and low FPR. Both models have similar performance for class 0 with an AUC of nearly 0.99. For the consecutive models VGG16 maintains the consistency at 0.99 except for class 2. The gap in performance between the two models widens for class 2 and class 3. Overall, VGG16-WPL-KNN consistently performs better than VGG16-KNN across all the classes. Table.3 provides a comparison of performance metrics in WPL-KNN and other classifiers.

Table.4 Performance Metrics of Proposed Models and other Classifiers

Model	Accuracy	Precision	Recall	F1-Score	Quadratic Weighted Kappa
VGG-16-KNN	0.91	0.93	0.91	0.92	0.902
ResNet-50-KNN	0.94	0.93	0.94	0.93	0.927
KNN	0.92	0.94	0.91	0.92	0.901
Inception	0.92	0.94	0.91	0.92	0.903
EfficientNet	0.93	0.94	0.92	0.91	0.915
ResNet50-WPL-KNN	0.96	0.95	0.96	0.94	0.951
VGG16-WPL-KNN	0.99	0.98	0.96	0.97	0.988

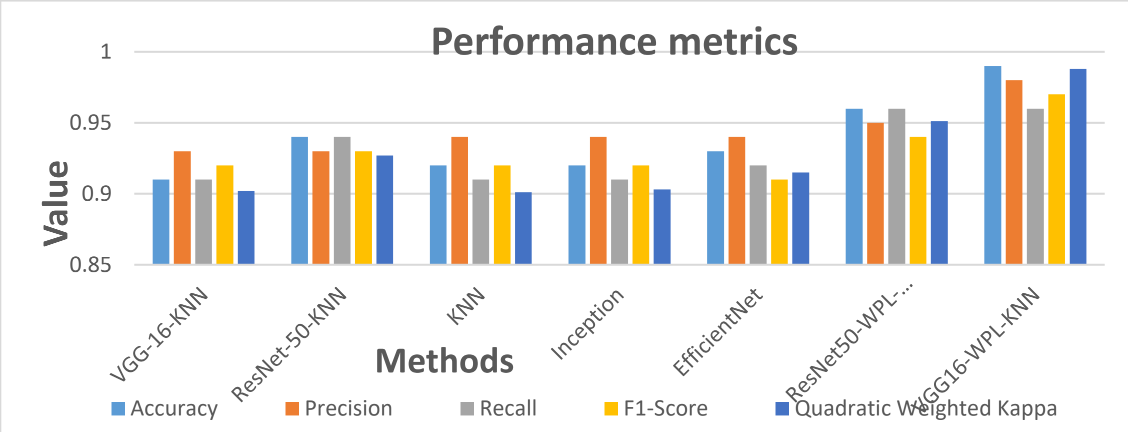


Figure.11 Performance Metrics of VGG16-WPL-KNN, ResNet50-WPL-KNN and Unmodified Models

Table.4 and Figure.11 presents the performance metrics including accuracy, precision, recall, and F1 score. Unmodified VGG-16 – KNN model has an accuracy of 0.91, precision of 0.93, recall of 0.91, F1-Score of 0.92, and quadratic weighted kappa of 0.902. The Unmodified ResNet-50-KNN model shows an accuracy of 0.94, precision of 0.93, recall of 0.94, F1-Score of 0.93, and quadratic weighted kappa of 0.972. KNN model has an accuracy of 0.92, precision of 0.94, recall of 0.91, F1-Score of 0.92, and quadratic weighted kappa of 0.901. VGG16-WPL-KNN model shows the highest accuracy of 0.99 among all models, with a precision of 0.98, recall of 0.96, F1-Score of 0.97 and quadratic weighted kappa of 0.988. The metrics suggest that the VGG16-WPL-KNN model performs exceptionally well in both identifying positive samples and limiting false positives. The ResNet50-WPL-KNN model has an accuracy of 0.96, precision of 0.95, recall of 0.96, F1-Score of 0.94, and quadratic weighted kappa of 0.951. On the other hand, the Inception model has an accuracy of 0.92, precision of 0.94, recall of 0.91, F1-Score of 0.92 and quadratic weighted kappa of 0.903. The Efficient Net model shows an accuracy of 0.93, precision of 0.94, recall of 0.92, F1-Score of 0.9, and quadratic weighted kappa of 0.915. On comparing all the mentioned model's performance, the -VGG16_WPL-KNN model appears to perform the best among all models based on the given metrics. A detailed comparison of the suggested models with external systems on the basis of Accuracy, Precision, Recall, F1-Score and Quadratic Weighted Kappa are given in the following section.

4.5 Comparative Analysis

This section delves into the outcomes achieved through the comparative study of the WPL-KNN models and unmodified systems, assessing the effectiveness of each method. Table 5 and Figure.12 illustrate the comparative analysis of the VGG16-WPL-KNN and ResNet50-WPL-KNN systems in relation to the Efficient Net model.

External Comparison:

The VGG16-WPL-KNN and ResNet50-WPL-KNN Models are compared with EfficientNet, J48 and KNN methods to assess the performance of the respective system.

Table.5: Accuracy Comparison of Proposed models with EfficientNet Model [40]

Model	Accuracy
EfficientNet	0.79
VGG16-WPL-KNN	0.99
ResNet50-WPL-KNN	0.96

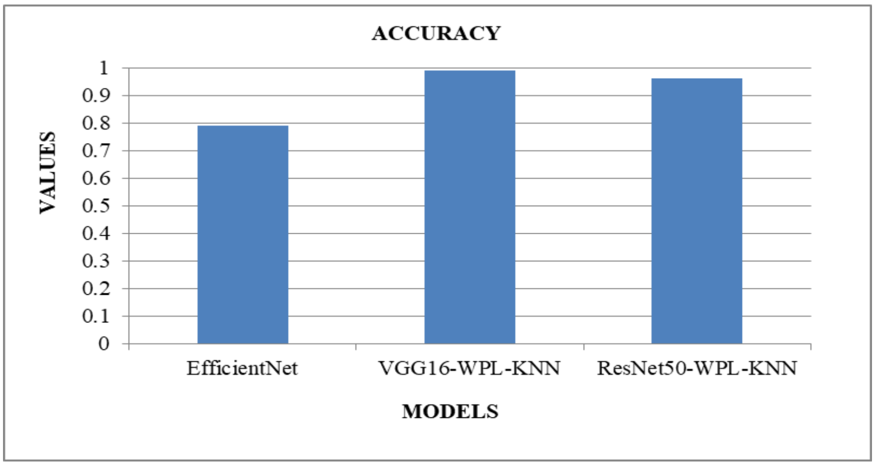


Figure.12 Accuracy analysis of Proposed models with EfficientNet Model

Table.5 and Figure.12 illustrates the performance of the ResNet50-WPL-KNN, VGG16-WPL-KNN and EfficientNet model. It can be observed that the model EfficientNet has an accuracy of 0.79 which is less than the accuracy provided by the VGG16-WPL-KNN model with the high accuracy of 0.99 outperforming the EfficientNet model. Also the ResNet50-WPl-KNN model performed almost near to the VGG16-WPL-KNN. It can be concluded that both VGG16-WPL-KNN and ResNet50-WPL-KNN models perform significantly better than the EfficientNet model.

Table.6: Accuracy comparison of VGG16-WPL-KNN, ResNet50-WPL-KNN, J48 and KNN Models [39]

Model	Accuracy
J48	0.79
KNN	0.81
VGG16 WPL-KNN	0.99

ResNet50- WPL-KNN	0.96

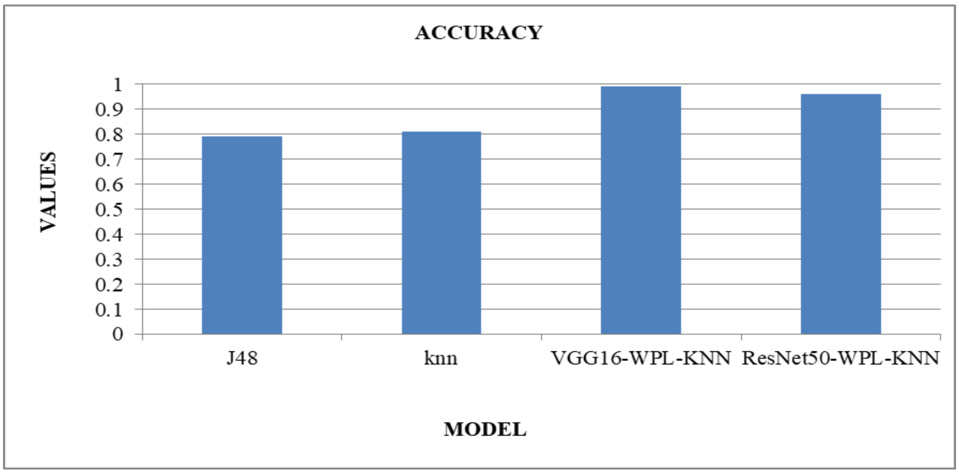


Figure.13 Accuracy comparison of Proposed Models with, J48 and KNN Models

Table.6 and Figure.13 illustrate the comparison of performance provided by ResNet50-WPL-KNN, VGG16-WPL-KNN and KNN and J48. KNN model has an accuracy of 0.81 which is better than the J48 model (0.79). On the other hand, the VGG16-WPL-KNN model has an accuracy of 0.99 outperforming the models. Also ResNet50-WPL-KNN performed similarly with an accuracy of 0.96. From these analyses it can be concluded that both the ResNet50-WPL-KNN and VGG16-WPL-KNN model performs significantly better than other prevailing models.

Table.7 AUC-ROC comparison of Proposed Models with Retfound and EyeFound [41]

Model	AUC-ROC
EyeFound	0.82
RetFound	0.82
VGG-16-KNN	0.92
ResNet-50-KNN	0.94
KNN	0.92
Inception	0.92
EfficientNet	0.93
ResNet50-WPL-KNN	0.95
VGG16-WPL-KNN	0.98

Table.7 illustrates AUC-ROC comparison of VGG16-WPL-KNN, ResNet50-WPL-KNN with Ret found and Eye Found. Correspondingly the conventional models like Retfound, Eyefound, VGG16-KNN, ResNet-50-KNN,KNN Inception, and EfficientNet has established the AUC-ROC score of 0.82, 0.82, 0.92, 0.94, 0.92, 0.92, 0.93, but the proposed model has achieved better than other conventional models with score of 0.95 and 0.98 in ResNet50- WPL-KNN and VGG16-WPL-KNN respectively.

4.6 Discussion

In recent years, several image classification models have been put forward for the automatic detection of DR and its different stages from retinal images often encounter challenges in identifying complex hidden features resulting in low performance. The systems are designed to overcome these challenges with the use of effective pre-processing, optimization and classification techniques. The results from the hyperparameter tuning process indicate that for the given model, SOA optimization leads to the selection of the Euclidean metric, uniform weights, and 11 neighbors, while POA optimization selects the Euclidean metric, distance weights, and also 11 neighbors. This comparison highlights the nuanced differences in the hyperparameter configurations chosen by each optimization approach and underscores their significance in optimizing model performance and effectively utilizing available data. A comparison with unmodified methods such as VGG-16–KNN, ResNet-50-KNN, KNN, Inception, and EfficientNet reveals better adeptness of the methods in handling the DR detection task. The -VGG16-WPL-KNN method exhibits an accuracy of 0.99, a significant improvement over the accuracy of 0.91 by the unmodified VGG16-KNN method. Similarly, the ResNet50-WPL-KNN method shows an accuracy of 0.96, which is better than the unmodified ResNet50-KNN method. In terms of precision, the VGG16-WPL-KNN method stands out with a score of 0.98, indicating high precision in predicting positive instances. The ResNet50-WPL-KNN method also shows a high precision of 0.95, better than the highest precision of 0.94 by KNN, Inception and EfficientNet models. The recall and F1-Score metrics further reinforce the better performance of the methods. Both the VGG16-WPL-KNN and ResNet50-WPL-KNN methods show high recall and F1-Score values, indicating effectiveness in identifying positive instances and maintaining a balance between precision and recall. This enhances usability and potential for real-world clinical applications. These findings suggest that the VGG16-WPL-KNN, ResNet50-WPL-KNN could contribute significantly to the fields of ophthalmology, pathology, and diabetology, particularly in assisting the diagnosis of DR. However, further research and testing are needed to validate the effectiveness of the VGG16-WPL-KNN, ResNet50-WPL-KNN and to explore potential ways to further improve their performance. This involves testing the method on different datasets, refining the algorithm, or exploring the integration of other DL techniques.

5 Conclusion

Diabetic retinopathy is a serious disorder prevailing mainly in working and older age groups. Due to the potential risk of causing blindness, treatment of DR at the initial stage is considered crucial to control the global increase in blindness due to DR. Therefore, prior detection and severity classification of DR comes as major problems since the availability of skilled ophthalmologists are limited. Also the cost of the manual procedure makes it unaffordable to the common public. The advent of AI technology has opened up a new opportunity for the automation of the aforementioned tasks. Although DL technologies performed better than previous works, it still needs to improve to reach a high level of precision and reliability. The VGG16-WPL-KNN, ResNet50-WPL-KNN system with SOA and POA optimization algorithm for hyperparameter tuning and a modern IDRID dataset potentially achieved the development of more effective diagnostic techniques. The presented model outperforms the similar approaches in the performance analysis, achieving significant improvement in the performance metrics in the detection of DR. The system, utilizes WPL-KNN, which leads to better performance with an accuracy of 0.99 with VGG16-WPL-KNN and 0.96 with ResNet50-WPL-KNN which depicts the ability of the VGG16-WPL-KNN model to classify the DR classes accurately. The WPL-KNN system outperforms other DL systems in terms of accuracy and other performance metrics such as precision, F1 score, and recall. The future prospects of the current research include mobile app based interface for seamless detection and classification of DR severity.

References

[1] Salma, A., Bustamam, A., Yudantha, A. R., Victor, A. A., & Mangunwardoyo, W. (2021). Artificial Intelligence Approach in Multiclass Diabetic Retinopathy Detection Using Convolutional Neural Network and Attention Mechanism. International Journal of Advances in Soft Computing & Its Applications, 13(3).

[2] Huang, X., Wang, H., She, C., Feng, J., Liu, X., Hu, X.,. Tao, Y. (2022). Artificial intelligence promotes the diagnosis and screening of diabetic retinopathy. Frontiers in Endocrinology, 13, 946915.

[3] Kodakandla, N. (2024). Hybrid Cloud Strategies: Optimizing Resource Allocation for Competitive Advantage in US Enterprises. Journal of Current Science and Research Review, 2(01), 01-17.

[4] Al-Kasasbeh, B. (2022). Artificial Intelligence Scheme for Medical Images Classification and Prediction. International Journal of Advances in Soft Computing & Its Applications, 14(2).

[5] Ayala, A., Ortiz Figueroa, T., Fernandes, B., & Cruz, F. (2021). Diabetic retinopathy improved detection using deep learning. Applied Sciences, 11(24), 11970.

[6] Bodapati, J. D., Shaik, N. S., & Naralasetti, V. (2021). Composite deep neural network with gated-attention mechanism for diabetic retinopathy severity classification. Journal of Ambient Intelligence and Humanized Computing, 12(10), 9825-9839.

[7] Alyoubi, W. L., Abulkhair, M. F., & Shalash, W. M. (2021). Diabetic retinopathy fundus image classification and lesions localization system using deep learning. Sensors, 21(11), 3704.

[8] Bashir, I., Sajid, M. Z., Kalsoom, R., Ali Khan, N., Qureshi, I., Abbas, F., & Abbas, Q. (2023). RDS-DR: An Improved Deep Learning Model for Classifying Severity Levels of Diabetic Retinopathy. Diagnostics, 13(19), 3116.

[9] Samanta, A., Saha, A., Satapathy, S. C., Fernandes, S. L., & Zhang, Y.-D. (2020). Automated detection of diabetic retinopathy using convolutional neural networks on a small dataset. Pattern Recognition Letters, 135, 293-298.

[10] Menaouer, B., Dermane, Z., El Houda Kebir, N., & Matta, N. (2022). Diabetic retinopathy classification using hybrid deep learning approach. SN Computer Science, 3(5), 357.

[11] Li, F., Wang, Y., Xu, T., Dong, L., Yan, L., Jiang, M., . . . Zou, H. (2022). Deep learning-based automated detection for diabetic retinopathy and diabetic macular oedema in retinal fundus photographs. Eye, 36(7), 1433-1441.

[12] Li, T., Gao, Y., Wang, K., Guo, S., Liu, H., & Kang, H. (2019). Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. Information Sciences, 501, 511-522.

[13] Farag, M. M., Fouad, M., & Abdel-Hamid, A. T. (2022). Automatic severity classification of diabetic retinopathy based on densenet and convolutional block attention module. IEEE Access, 10, 38299-38308.

[14] Kumar, G., Chatterjee, S., & Chattopadhyay, C. (2021). DRISTI: a hybrid deep neural network for diabetic retinopathy diagnosis. Signal, Image and Video Processing, 15(8), 1679-1686.

[15] Dao, L., & Ly, N. Q. (2024). Recent Advances in Medical Image Classification. International Journal of Advanced Computer Science & Applications, 15(7).

[16] Goel, S., Gupta, S., Panwar, A., Kumar, S., Verma, M., Bourouis, S., & Ullah, M. A. (2021). Deep learning approach for stages of severity classification in diabetic retinopathy using color fundus retinal images. Mathematical Problems in Engineering, 2021(1), 7627566.

[17] Lahmar, C., & Idri, A. (2023). Deep hybrid architectures for diabetic retinopathy classification. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, 11(2), 166-184.

[18] Math, L., & Fatima, R. (2021). Adaptive machine learning classification for diabetic retinopathy. Multimedia Tools and Applications, 80(4), 5173-5186.

[19] Sikder, N., Masud, M., Bairagi, A. K., Arif, A. S. M., Nahid, A.-A., & Alhumyani, H. A. (2021). Severity classification of diabetic retinopathy using an ensemble learning algorithm through analyzing retinal images. Symmetry, 13(4), 670.

[20] Zago, G. T., Andreão, R. V., Dorizzi, B., & Salles, E. O. T. (2020). Diabetic retinopathy detection using red lesion localization and convolutional neural networks. Computers in biology and medicine, 116, 103537.

[21] Bhardwaj, C., Jain, S., & Sood, M. (2021). Hierarchical severity grade classification of non-proliferative diabetic retinopathy. Journal of Ambient Intelligence and Humanized Computing, 12(2), 2649-2670.

[22] Gadekallu, T. R., Khare, N., Bhattacharya, S., Singh, S., Maddikunta, P. K. R., & Srivastava, G. (2023). Deep neural networks to predict diabetic retinopathy. Journal of Ambient Intelligence and Humanized Computing, 1-14.

[23] Beevi, S. Z. (2023). Multi-Level severity classification for diabetic retinopathy based on hybrid optimization enabled deep learning. Biomedical Signal Processing and Control, 84, 104736.

[24] Das, S., Kharbanda, K., Suchetha, M., Raman, R., & Dhas, E. (2021). Deep learning architecture based on segmented fundus image features for classification of diabetic retinopathy. Biomedical Signal Processing and Control, 68, 102600.

[25] Malerbi, F. K., Andrade, R. E., Morales, P. H., Stuchi, J. A., Lencione, D., de Paulo, J. V., . . . Ferraz, D. A. (2022). Diabetic retinopathy screening using artificial intelligence and handheld smartphone-based retinal camera. Journal of diabetes science and technology, 16(3), 716-723.

[26] Mustafa, H., Ali, S. F., Bilal, M., & Hanif, M. S. (2022). Multi-stream deep neural network for diabetic retinopathy severity classification under a boosting framework. IEEE Access, 10, 113172-113183.

[27] Shankar, K., Sait, A. R. W., Gupta, D., Lakshmanaprabu, S. K., Khanna, A., & Pandey, H. M. (2020). Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model. Pattern Recognition Letters, 133, 210-216.

[28] Yaqoob, M. K., Ali, S. F., Bilal, M., Hanif, M. S., & Al-Saggaf, U. M. (2021). ResNet based deep features and random forest classifier for diabetic retinopathy detection. Sensors, 21(11), 3883.

[29] Dhinakaran, D., Srinivasan, L., Selvaraj, D., & Sankar, S. (2023). Leveraging Semi-Supervised Graph Learning for Enhanced Diabetic Retinopathy Detection. arXiv preprint arXiv:2309.00824.

[30] Patil, M. S., Chickerur, S., Abhimalya, C., Naik, A., Kumari, N., & Maurya, S. (2023). Effective deep learning data augmentation techniques for diabetic retinopathy classification. Procedia Computer Science, 218, 1156-1165.

[31] Gadekallu, T. R., Khare, N., Bhattacharya, S., Singh, S., Maddikunta, P. K. R., Ra, I.-H., & Alazab, M. (2020). Early detection of diabetic retinopathy using PCA-firefly based deep learning model. Electronics, 9(2), 274.

[32] Dutta, S., Manideep, B., Basha, S. M., Caytiles, R. D., & Iyengar, N. (2018). Classification of diabetic retinopathy images by using deep learning models. International Journal of Grid and Distributed Computing, 11(1), 89-106.

[33] Mohanty, C., Mahapatra, S., Acharya, B., Kokkoras, F., Gerogiannis, V. C., Karamitsos, I., & Kanavos, A. (2023). Using deep learning architectures for detection and classification of diabetic retinopathy. Sensors, 23(12), 5726.

[34] Dai, L., Wu, L., Li, H., Cai, C., Wu, Q., Kong, H., . . . Liu, Y. (2021). A deep learning system for detecting diabetic retinopathy across the disease spectrum. Nature communications, 12(1), 3242.

[35] Abirami, N., & Andal, K. Framework for diabetic retinopathy grading using a hybrid fuzzy-knn classifier. Ictact journal on image and video processing, february 2025, volume: 15, issue: 03.

[36] Nassour, N., Akhbari, B., Ranganathan, N., Shin, D., Ghaednia, H., Ashkani-Esfahani, S., . . . Guss, D. (2024). Using machine learning in the prediction of symptomatic venous thromboembolism following ankle fracture. Foot and Ankle Surgery, 30(2), 110-116

[37] Bilal, A., Imran, A., Baig, T. I., Liu, X., Long, H., Alzahrani, A., & Shafiq, M. (2024). Improved Support Vector Machine based on CNN-SVD for vision-threatening diabetic retinopathy detection and classification. Plos one, 19(1), e0295951.

[38] Saproo, D., Mahajan, A. N., Narwal, S., & Yadav, N. (2025). Deep Feature Extraction and Classification of Diabetic Retinopathy Images using a Hybrid Approach. Engineering, Technology & Applied Science Research, 15(2), 21475-21481.

[39] Biswas, S., Mostafiz, R., Paul, B. K., Uddin, K. M. M., Rahman, M. M., & Shariful, F. (2023). DFU_MultiNet: A deep neural network approach for detecting diabetic foot ulcers through multi-scale feature fusion using the DFU dataset. Intelligence-Based Medicine, 8, 100128.

[40] Vijayan, T., Sangeetha, M., Kumaravel, A., & Karthik, B. (2023). Feature selection for simple color histogram filter based on retinal fundus images for diabetic retinopathy recognition. IETE Journal of Research, 69(2), 987-994.

[41] Shi, L., Wang, B., & Zhang, J. (2023). A Multi-stage Transfer Learning Framework for Diabetic Retinopathy Grading on Small Data. Paper presented at the ICC 2023-IEEE International Conference on Communications.

[42] Shi, D., Zhang, W., Chen, X., Liu, Y., Yang, J., Huang, S, .He, M. (2024). Eyefound: A multimodal generalist foundation model for ophthalmic imaging. arXiv preprint arXiv:2405.11338.

Notes on contributors

I. S. Hephzi Punithavathi is currently working as Associate Professor in the Department of CSE (Artificial Intelligence & Machine Learning), Vidya Jyothi Institute of Technology, Hyderabad and pursuing Post Doctoral degree in University of Louisiana at Layfette. Her areas of research interests include AI, Machine Learning and Image Processing. She has also published books about Artificial Intelligence and also served as reviewer for IEEE international conferences and other Scopus Indexed Journals. She is also a member of ISTE and ACM.

Margala, Martin is working as a Professor of Computer Science and Director of School of Computing and Informatics at University of Louisiana at Lafayette. He is a senior member of ACM, IEEE, and SPIE with more than 50 journal and 200 peer reviewed conference publications in the areas of Design for Testability for Energy Efficient Architectures and Systems, High-Performance Reliable Low-Power Architectures and Reconfigurable Secure Architectures and Systems. He has directed many Ph. D and M. S Students, many of whom now hold leading positions in academia and industry. He has served on numerous program committees of international conferences and on workgroups (such as the International Technology Roadmap for Semiconductors) that have a great impact on the future direction of academia and industry.

Dr.Siva Shankar S is working as an Associate Professor in the Department of Computer Science and Engineering, KG Reddy College of Engineering and Technology, Hyderabad, Telangana, India and is also IPR Head. He completed his B.Tech in Anna University, M.Tech in MS University and PhD in BHARATH University. He has received the National Excellence Award for SCS IPR firm.

Prof Dr. Prasun Chakrabarti has received his Ph.D. (Engg) from Jadavpur University, India in 2009. He is working as CSE, Sir Padampat Singhania University, Udaipur, Rajasthan, India. He has more than 230 publications, 11 books and 42 filed Indian patents in his credit. He has supervised 10 Ph.D. candidates successfully. On various research assignments, he has visited Waseda University Japan (2012 availing prestigious INSA-CICS travel grant), University of Mauritius (2015), Nanyang Technological University Singapore (2015,2016,2019), Lincoln University College Malaysia (2018), National University of Singapore (2019), Asian Institute of Technology Bangkok Thailand (2019) and ISI Delhi (2019). He is a Fellow of Royal Society of Arts London, IET(UK), IETE, ISRD(UK), IAER(London), AE(I), CET(I) , Nikhil Bharat Shiksha Parisad and a Senior member of the IEEE(USA).