IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 April 2023, accepted 25 April 2023, date of publication 1 May 2023, date of current version 4 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3272228

RESEARCH ARTICLE

A Lightweight Robust Deep Learning Model Gained High Accuracy in Classifying a Wide Range of Diabetic Retinopathy Images

MOHAIMENUL AZAM KHAN RAIAAN[1], KANIZ FATEMA[2], INAM ULLAH KHAN[2], SAMI AZAM[3], MD. RAFI UR RASHID[4], MD. SADDAM HOSSAIN MUKTA[1], MIRJAM JONKMAN[3], (Member, IEEE), AND FRISO DE BOER[3]

[1]Department of Computer Science and Engineering, United International University, United City, Dhaka 1212, Bangladesh
[2]Health Informatics Research Laboratory, Department of Computer Science and Engineering, Daffodil International University, Dhaka 1207, Bangladesh
[3]Faculty of Science and Technology, Charles Darwin University, Casuarina, NT 0909, Australia
[4]Department of Computer Science and Engineering, The Pennsylvania State University, State College, PA 16801, USA

Corresponding author: Sami Azam (sami.azam@cdu.edu.au)

ABSTRACT Diabetic retinopathy (DR) is a common complication of diabetes mellitus, and retinal blood vessel damage can lead to vision loss and blindness if not recognized at an early stage. Manual DR detection using large fundus image data is time-consuming and error-prone. An effective automatic DR detection system can be significantly faster and potentially more accurate. This study aims to classify fundus images into five DR classes, using deep learning methods, with the highest possible accuracy and the lowest possible computational time. Three distinct DR datasets, APTOS, Messidor2, and IDRiD, are merged, resulting in 5,819 raw images. Before training the model, various image preprocessing techniques are applied to remove artifacts and noise from the images and improve their quality. Three augmentation techniques: geometric, photometric, and elastic deformation, are used to create a balanced dataset. A shallow convolutional neural network (CNN) is developed using three blocks of convolutional layers and maxpool layers with a categorical cross-entropy loss function, Adam optimizer, 0.0001 learning rate, and 64 batch size as a base model, and this is also employed to determine the best data augmentation method for further processing. A study to optimize the performance is then conducted by changing different components and hyperparameters of the base model, resulting in our proposed RetNet-10 model. Six cutting-edge models are employed for comparison. Our proposed RetNet-10 model performed the best, with a testing accuracy of 98.65%. MobileNetV2, VGG16, Xception, VGG19, InceptionV3 and ResNet50 achieved testing accuracies of 91.42%, 90.16%,89.57%, 88.21%, 87.68% and 87.23%, respectively. The model is also trained with several k values to assess its robustness. After image processing and data augmentation, using the combined dataset, and fine-tuning the base model, our proposed RetNet-10 model outperformed other automated methods for DR diagnosis.

INDEX TERMS Diabetic retinopathy, retinal fundus images, multi-class classification, image preprocessing, augmentation, convolutional neural network, transfer learning models, model optimization, k-fold cross validation.

I. INTRODUCTION

Diabetic Retinopathy (DR) is a common and serious complication of diabetes mellitus. It is a leading cause of vision loss and blindness among people of working-age [1], [2]. DR causes changes in retinal blood vessels, utilized to transport oxygenated blood and nutrients to different parts of the retina [3]. It is a complication of diabetics causing retinal blood vessels to swell and leak fluid and blood in the posterior part of the eye [4], [5]. Abnormal growth of blood vessels with vascular blockage and blood leakage in healthy parts of the retina can also occur [6], [7], [8]. Generally, DR is

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang.

diagnosed based on the presence of various lesions, including microaneurysms (MAs), haemorrhages (HMs), and soft and hard exudates (EX), visible in images of the retina [9]. There are two major types of DR: Non- Proliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR) [10]. NPDR is the early stage of DR and can be further divided into mild, moderate, and severe stages. PDR refers to the advanced stages of DR [11]. Based on the presence of lesions, DR can be classified into five grades: no DR (Grade 0), mild non proliferative DR (NPDR) (Grade 1), moderate NPDR (Grade 2), severe NPDR (Grade 3), and proliferative DR (PDR) (Grade 4) [9], [12]. Worldwide DR causes 2.6% of blindness [13]. More than 239 million people were badly affected in 2010. The International Diabetics Foundation (IDF) estimated that there were approximately 451 million diabetes patients in 2017, of which more than a-third had DR, representing a large population at risk of optical disability or blindness [14]. It is expected that by 2025, the prevalence of DR will be 592 million [15]. People living with diabetes often remain undiagnosed for years [14]. Patients who are at risk of DR are often asymptomatic in the early stage. However, they suffer from floaters, distortion, blurred vision, and loss of vision in later stages.

Early identification of DR is of utmost importance for preventing progression to a more severe stage. DR can be identified and classified by using color fundus images [16]. Manual analysis can only be performed by highly trained experts and can be time-consuming and error-prone. An automated computer vision method to classify retinal fundus images and assist clinicians could be saving time and money [16], [17].

In previous studies, researchers applied different deep learning and machine learning techniques to retinal fundus images. Some performed binary classification [16], [18], [19], [20], while others also performed multi-class classification [16], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28] in their work. Using a single fundus image data set, satisfactory results were achieved in both binary and multi-class classification. There are also several studies [29], [30], [31] where multi-class classification with a large dataset of images did not lead to satisfactory results. Obtaining optimal results for multi-class classification using a combined dataset with images of different resolutions is difficult, however, this is a challenge that should be overcome to make automatic classification systems useful in practice. In this work, we focus on classifying retinal images into five classes: no DR (Grade 0), mild NPDR (Grade 1), moderate NPDR (Grade 2), severe NPDR (Grade 3), and PDR (Grade 4) by utilizing a large dataset (combining three different datasets) and experimenting with different computer vision techniques. All the processing steps are explained in the following sections.

The following list highlights the significant contributions of this study:

1) Since different datasets have different resolutions and image quality, achieving optimal accuracy is challenging. Nevertheless, in this research, three different retinal fundus image datasets: APTOS [32], Messidor2 [33], and IDRiD [34] are combined. After merging, 5,819 raw fundus images of different qualities are obtained.
2) Black image background and speckles are considered fundus artifacts and noise, respectively. To remove the artifacts, the Otsu thresholding and contour finding functions are applied, preserving the essential regions of interest (ROI) of the images. Morphological opening and non-local meaning denoising algorithms (NLMD) are applied to remove the noise.
3) The YUV color space format allows us to process only the Y channel instead of the U and V channels to get the overall illuminance component, which can help us achieve acceptable accuracy. In this study, the Contrast Limited Adaptive Histogram Equalization (CLAHE) enhancement technique is applied in an image's YUV color space to emphasize blood vessels and enhance image quality and contrast.
4) Three different datasets are created by using elastic deformation, geometric, and photometric augmentation methods to increase the number of images. A base model is constructed and applied to these three datasets to determine the optimal augmentation technique for multi-class classification.
5) The model is optimized by changing the model architecture and hyperparameters. During this model optimization, the time complexity is considered without compromising accuracy.
6) To evaluate the robustness of our proposed network, the model is trained using the k-fold cross-validation method with k values of 1, 3, 5, 7, and 9. In addition, the proposed model is trained with 75%, 50%, and 25% images of the pre-processed dataset achieving 96.88%, 95.67%, and 90.43% test accuracy, respectively. Although the number of images was reduced, our proposed model still provides values close to the best test accuracy, even when using only 50% of the images. Thus, it can be confirmed that this model can achieve optimal accuracy even with fewer images.

The paper is organized as follows: Section II provides an overview of the relevant literature on the prediction of diabetic retinopathy using different classifiers and hybrid approaches. A brief overview of diabetic retinopathy is given in section III. The research methodology is discussed in detail in section IV. Section V describes the results and analysis of our proposed model utilizing various performance metrics. Comparison with transfer learning models and state-of-the-art works, including the analysis of the robustness of the proposed model. Section VI concludes the study, and section VII provides a brief overview of the limitations and its future scope.

II. LITERATURE REVIEW

Researchers have proposed various deep-learning and machine-learning methods to classify eye images. In this section, we present studies which used common datasets.

Gayathri et al. [16] used three different DR datasets: IDRid, Messidor, and Kaggle, for binary and multi-class classification. They proposed a novel CNN model to extract features from retinal images. The output features of the CNN model are used as input to six distinct machine learning classifiers: Support Vector Machine (SVM), AdaBoost, Naive Bayes, Random Forest (RF), and J48. The J48 classifier achieved 99.89% and 99.59% accuracy for binary and multiclass classification, respectively. Their CNN feature extractor reduced computation time and complexity by using the entire image without leaving any ROIs that may be affected by DR. However, to attain good accuracy for both multiclass and binary classification, machine learning classifiers were required. While a CNN was utilized for feature extraction, it was not assessed how their proposed CNN would perform for classification. In another study, [18] a hybrid SVM, Naïve-Bayes classifier was developed to detect bright lesions from the fundus images reliably. The experiment utilized the image-Ret dataset. They employed image pre-processing, blood vessel segmentation and extraction, and optic disc identification. The suggested classifier provides an accuracy of 98.60% for optic disc localization and classification. However, the classification accuracy for DR of different severity was not described. Kaushik et al. [19] proposed a stacked deep learning technique where the weights of three CNNs are passed to a singular meta-learner to diagnose diabetic retinopathy. They used a dataset of 2,471 images named EyePACS, which contains two classes: diabetic retinopathy and no diabetic retinopathy. Transfer learning models such as VGG16 and ResNet50 were introduced for comparison with their proposed method. Performance analysis showed that stacked CNNs scored 97.92% accuracy, outperforming other models. For multiclass classification, their model attained an accuracy of 87.45%. Their stack CNN did well in binary classification due to a fusion technique to include the optimum weights from many neural networks into a single model. However, the absence of image enhancement and noise removal strategies is a limitation of their method. Yaqoob et al. [20] introduced a deep learning-based approach to classify and grade DR images. In this approach, the ResNet-50 models' features are used and passed to the RF for classification. The proposed approach was compared with five state-of-the-art models, VGG16, VGG19, MobileNet, Inception-v3, Xception, and ResNet50, utilizing two categories of the Messidor2 dataset (No Referable Diabetic Macular Edema Grade (DME) and Referable DME), and five categories of EyePACS dataset (no DR, mild, moderate, severe, and PDR). The proposed approach acquired accuracies of 96% and 75.09% for the Messidor2 and the EyePACS datasets, respectively. Utilizing ResNet-50's deep features in combination with a Random Forest classifier, their proposed architecture results in an accuracy of 96% for the two-category Messidor-2 dataset. However, this is reduced to 75.09% for the five-category EyePACS dataset due to their highly imbalanced dataset and lack of appropriate preprocessing strategies. Gen et al. [21] introduced a CNN model to compare the detection of severe DR based on original fundus images and based on entropy images. They utilized the "Kaggle diabetic Retinopathy" dataset, which contains 35,126 images of five DR grades (Grade 0- Grade 4), selected 21,123 images, and expanded this to 33,000 images by flipping and rotating before further processing. A block size of 9 was used to convert normal images into entropy images. They used 30,000 and 3,000 images for training and testing, respectively. Accuracies of 81.80% and 86.10% for the detection of referable DR (grade 2–4) were obtained based on the original fundus image data set and the entropy images, respectively. The study has several limitations, including a lack of model fine-tuning and a lack of image enhancement and noise removal. The proposed model could not provide optimal results in multi-class classification. In another study [22], a deep CNN model, using 18 convolutional layers and three fully connected (FC) layers, was proposed for fundus image classification. The model distinguished no DR, moderate DR (a combination of the mild and moderate NPDR classes), and severe DR (a combination of severe NPDR and PDR) class images. They worked with the APTOS 2019 Kaggle dataset (3,661 images) and generated additional images from these original images with augmentation methods. 4,600 images were used to train and test their model. The validation accuracy was in the range of 88%–89%. However, there were multiple ways in which the accuracy could have been improved, such as image processing (noise removal and contrast enhancement) and model fine-tuning. Shankar et al. [23] proposed a technique for automated hyperparameter optimization of Inception-v4 using Bayesian optimization to extract the features from fundus images. Their study used a feed-forward artificial neural network, multi-layer perceptron (MLP), to classify diabetic retinopathy. The Messidor dataset, which contains 1200 images and four classes, was used and the authors gave the diabetic retinopathy stages moderate NPDR and severe NPDR the same label. The proposed HPTI-v4 model achieved an accuracy of 99.43% after they optimized their model and extracted essential features from images. Although the researchers obtained satisfactory results, they did not experiment with a combined dataset. Al-Hazaimeh et al. [24] investigated a new approach to detect diabetic retinopathy in fundus images. The study used a large Kaggle dataset and compared different datasets. The proposed technique involved two phases, extraction of diabetic retinopathy features and classification, and they adopted the color space conversion method for this purpose. Detection and removal of the optic disc and blood vessel segmentation and disposal strategies were sequentially performed. A DCNN model extracted the features, and SVMGA was utilized as a classifier, achieving 98.80% accuracy. Integrating DCNN and SVMGA helped to classify the fundus image markers appropriately. The study has some limitations as the researchers did not classify the severity according to the grade. Wejdan et al. [25] proposed two different methods: CNN512 (first scenario) and an adopted YOLOv3 model (second scenario) to classify fundus images into five

grades, no-DR, mild, moderate, severe, and proliferative DR. They used the DDR and the APTOS Kaggle 2019 public datasets. The first CNN model (CNN512) consisted of one zero padding layer with a value of 2, six convolutional layers each followed by max pool layers, eight batch normalization layers, two fully connected layers, and one SoftMax layer for classification. The input image size was $512 \times 512 \times 3$. The second model was utilized to detect and localize the DR lesions, achieving a 0.216 mean average precision (MAP) in lesion localization for the DDR dataset. In classification, CNN512 achieved 88.6% and 84.1% accuracy for the DDR and the APTOS datasets, respectively. YOLOv3 acquired a classification accuracy of 89%. The accuracy could be enhanced by exploring more image processing techniques and model fine-tuning. In this case, researchers [26] focused on developing a CNN model using three blocks with convolution layers, batch normalization, the ReLU activation function, and max Pooling layers. A dropout layer was also added. They utilized two retinal image datasets, MESSIDOR and IDRiD, to classify images into four categories: No DR, Mild DR, Moderate DR, and Severe DR. They used several image processing techniques, canny edge detection, resizing, interpolation, and normalization to the optic diskless images, before feeding the images to the model. The proposed CNN model acquired an accuracy of 90.89% utilizing the MESSIDOR images and could effectively detect and grade the NPDR images. However, there were some limitations in their work, such as a lack of proper image enhancement techniques and model fine-tuning. Gharaibeh et al. [27] proposed an image processing method to identify diabetic retinopathy using the publicly available benchmark DARETDB1 dataset. They applied preprocessing, segmentation, and noise reduction techniques. The features were extracted, and feature selection techniques were applied to select significant features. The efficiency of the proposed two-phase image processing method was validated using performance metrics and resulted in accurate retinopathy diagnosis from fundus images. They combined the SVM classifier with the genetic algorithm and obtained an accuracy of 98.4%. In another study [28], a deep learning -based automated detection and classification model for fundus DR images was proposed. The Messidor dataset was used for multiclass classification into four classes. A Synergic Deep Learning (SDL) model was proposed and compared with several CNN models: M-AlexNet, Alex Net, VGGNet-16, VGGNet-19, GoogleNet and ResetNet. The SDL model uses two convolutional neural networks (DCNNs) that learn from each other via a process of reciprocal learning, and the proposed model performed well with 99.28% accuracy. Sehrish et al. [29] introduced a CNN ensemble-based framework for detecting and classifying the DR's different stages using color fundus images. Stacking accentuates each base model's best performance and deprecates each model's worst performance. The stacking method works best when the basic models are drastically different. The Kaggle DR dataset was used, and they applied up-and-down sampling to balance the dataset for multiclass classification. They integrated several deep learning models: Reset50, inceptionV3, Xception, Dense121, and Dense 169, and trained with the balanced and imbalanced dataset. A test dataset was created for testing the model, and results were compared with existing literature using a similar dataset. However, their proposed ensemble model acquired only 80.8% accuracy for the imbalanced dataset.

Wu et al. [30] developed a Coarse-to-fine network (CF-DRNet) architecture for automated DR classification using a CNN-based approach. It contains two different networks, the coarse and fine networks. The coarse network was designed for two-class classification (No DR, DR), and the fine network was used for four-class classification (mild NPDR, moderate NPDR, severe NPDR, and PDR). The publicly available IDRiD and the Kaggle fundus image datasets were used, and augmentation techniques were introduced to overcome the class imbalance problem. A transfer learning model, ResNet, was used for comparison. Their model outperformed ResNet with an overall accuracy of 83.10% and 56.19% for the Kaggle and IDRiD data sets, respectively. Rubina et al. [31] focused on the multi-class classification of mild diabetic eye diseases (normal, mild DR, mild DME, and mild Glaucoma) and multi-class diabetic eye diseases (normal, DR, DME, Glaucoma, and Cataract). They utilized four different datasets, namely the Messidor, Messidor-2, DRISHTI-GS, and retina datasets. The researchers fine-tuned two different CNN models, VGG16 and InceptionV3, by altering the optimizer (RMSprop, SGD, Adagrad, and Adam) and replacing the fully connected modules. The VGG16 model acquired an accuracy of 88.3% and 85.95% for multi-class classification and mild multi-class classification, respectively. After reviewing all the studies, it could be concluded that in most cases, researchers used publicly available datasets, such as eyepieces, DR, IDRiS, Messidor, Messidor 2, DARETDB1, and Kaggle. However, they did not try to combine multiple datasets to create a larger dataset of different-quality images. In some cases, researchers performed only binary classification, while others performed binary and multi-class classifications. Since the datasets were imbalanced, researchers balanced them using different techniques, including down-sampling and up-sampling. In most cases, image pre-processing methods and proper model fine-tuning were lacking. Although some models achieved excellent accuracy in binary or multi-class classification with a small number of images, the results may degrade when using a larger dataset. Digital image processing can enhance the accuracy of diagnosis and prediction [35]. Image preprocessing and a fine-tuned CNN model may help to achieve good classification results. The limitations of the studies described above are summarized in table 10.

III. MEDICAL ANALYSIS OF FUNDUS IMAGES

Before providing any type of medical images to an automated system for diagnosis, it is important to examine the images. This helps us to understand the markers indicated by radiologists during the diagnostic process. In this study, these

TABLE 1. DR severity classification based on markers.

DR Grades Based on Severity	Lesions Classification
No DR (Grade 0)	No lesions.
Mild DR (Grade 1)	Presence of Microaneurysms only
Moderate DR (Grade 2)	Markers for severe DR exist but are less prominent compared to severe DR and include more than Microaneurysms only
Severe DR (Grade 3)	Any of the following:
	(i) Over 20 intraretinal haemorrhages in each of the four quadrants
	(ii) Obvious venous beading in two or more quadrants
	(iii) Prominent intraretinal microvascular anomalies in one or more quadrants
Proliferative DR (Grade 4)	At least one or more of the following:
	(i) Neovascularization
	(ii) Pre-retinal haemorrhages

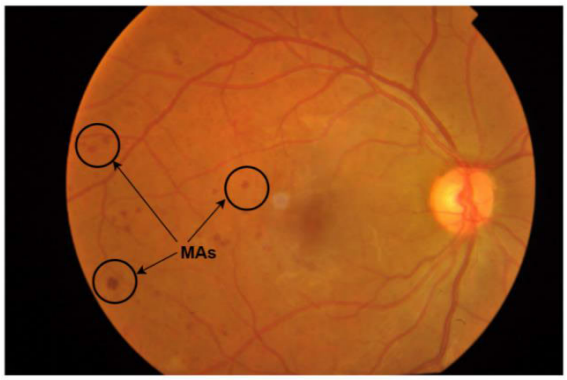

FIGURE 1. MAs of a fundus image.

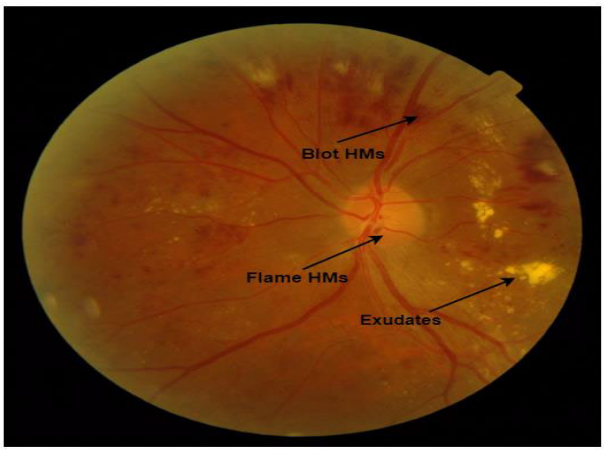

FIGURE 2. HMs of various types.

public datasets are accurately labelled with five different grades based on the characteristics that the specialists usually consider. The purpose of this section is to provide some understanding of the anomalies of these fundus images for to each class.

Grade 0 is the class of retinal fundus image of eyes without any presence of DR. These fundus images show structures like the Fovea, Macula, Optic Disc, and Blood Vessels. From Grade 1 (mild NPDR) onwards the fundus image shows markers that indicate the presence of DR. High sugar levels in the blood as a result of diabetes can cause severe damage to the retina's blood vessels, which is the primary cause of DR. This condition causes the blood vessels to swell and leak, leading to retinal damage [36]. The leaking blood and fluids appear as lesions on the fundus images, and these lesions can be identified as bright or red lesions. Red lesions are associated with microaneurysms and haemorrhage, whereas bright lesions are associated with soft and hard exudates. The small dark red dots are microaneurysms, while the more prominent spots are haemorrhages. Soft exudates, also called cotton wool spots, appear as yellowish-white and fluffy spots and are related to nerve fiber damage, whereas hard exudates appear as bright yellow spots. DR can be detected using these markers, and the severity of DR can be understood from the accumulation of these markers [25], [36]. Table 1 illustrates how the markers identify the severity of DR.

Generally, the optic disc and blood vessels are very clear in a normal retinal image. In contrast, in diseased eye images, there are abnormalities due to which the optic disc and blood vessels are not visible. As mentioned above, DR is diagnosed based on different lesions, such as soft exudate, hard exudate, MAs, and HMs. While MAs and HMs are red lesions, soft and hard exudates are bright white lesions [37]. DR stages can be classified depending on the presence of these lesions [36], [38], [39]. This section describes the different DR stages and the lesions.

1) MICROANEURYSMS (MAS)

Microaneurysms are due to brittleness of the blood vessel's walls. They are a first indication of DR, manifesting as tiny red circular dots on the retina. They are less than 125 μm in size and have a variety of sharp pointed margins. Fig 1 illustrates MAs in a fundus image.

2) HAEMORRHAGES (HMS)

Haemorrhages present as large spots on the retina. Their size is greater than 125 μm and they have an irregular margin. There are two types: flame superficial HMs and blot deep HMs [40]. Fig 2 shows the various types of HMs in a fundus image.

3) HARD EXUDATES

Hard exudates appear as a bright-yellow spots on the retina due to plasma leakage. They have sharp margins and can

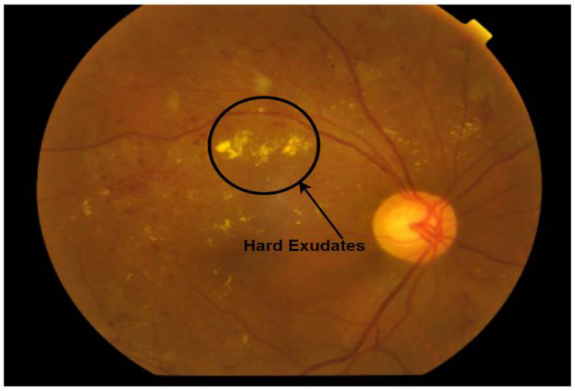

FIGURE 3. Fundus image with hard exudates.

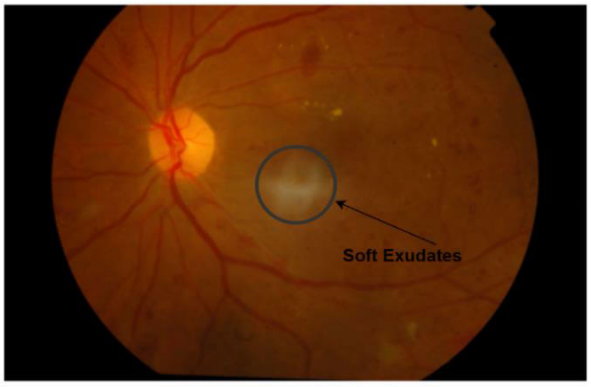

FIGURE 4. Soft exudates of fundus image.

be found in the outer layers of the retina. Fig 3 shows hard exudates in a fundus image.

4) SOFT EXUDATES

A soft exudate resembles a white spot on the retina and is due to swelling of the nerve fibers. Its shape is oval or round. Fig 4 shows soft exudates in a fundus image.

Examples of typical fundus images for the different DR grades are shown in Fig 5. These show the structure of the retina and the presence of lesions related to different stages of DR [41], [42], [43].

No DR: No lesions or abnormalities are visible at this stage. Thus, the retinal blood vessels and optic disc are visible without obstruction (See Fig 5a).

Mild NPDR: There are small areas of weakening and swelling of the blood vessels called MAs. This can allow fluid to leak into the retina (See Fig 5b).

Moderate NPDR: The blood vessels become swollen and distorted at this stage. They may also become blocked, interfering with the blood circulation of the eye. A small number of MAs with or without soft exudate and scarring are present, more than for mild NPDR and less than for severe NPDR (See Fig 5c).

Severe NPDR: At this stage, more than 20 intraretinal HMs are present in each of the four quadrants, definite venous beading is present in at least two quadrants, and prominent intraretinal microvascular abnormalities can be seen in at least one quadrant while no signs of PDR are detected. Many blood vessels are ruptured or blocked. This leads to a shortage of blood supply to the eye. When this happens, areas without a blood supply start signaling that new blood vessels should be established. However, these new blood vessels will grow weakly and may have a more malignant outlook (See Fig 5d).

PDR: This grade includes the presence of preretinal haemorrhage or neovascularization. Since this is an advanced stage of DR, additional scar tissue can detach the retina. PDR can cause permanent blindness. (See Fig 5e).

TABLE 2. Description of the number of original images in the datasets.

Name of the classes	Name of the datasets			After combining total number of images
	APTOS Dataset [32]	Messidor 2 Dataset [33]	IDRiD Dataset [34]	
No DR (Grade 0)	1,805	1,017	134	2,956
Mild NPDR (Grade 1)	370	270	20	660
Moderate NPDR (Grade 2)	999	347	136	1,482
Severe NPDR (Grade 3)	193	75	74	342
PDR (Grade 4)	295	35	49	379
Total number of images	3,662	1,744	413	5,819

IV. METHODOLOGY

This research can be divided into six stages: 1) combining three datasets, 2) preprocessing of fundus images, 3) using three types of augmentation techniques, 4) building a base model, 5) performing a model optimization, and 6) performance and result analysis. Fig 6 depicts the workflow of this study. In this section, all the steps are described in detail.

A. DATASET

Training with a single dataset can limit the robustness of an automated system. Images of different datasets are often collected using different scanners, cameras, etc., and these images differ in color intensity, number of pixels, etc.

Training an automated system using various images can result in a more effective diagnostic process. As various types of images are used in the medical field to diagnose disorders,

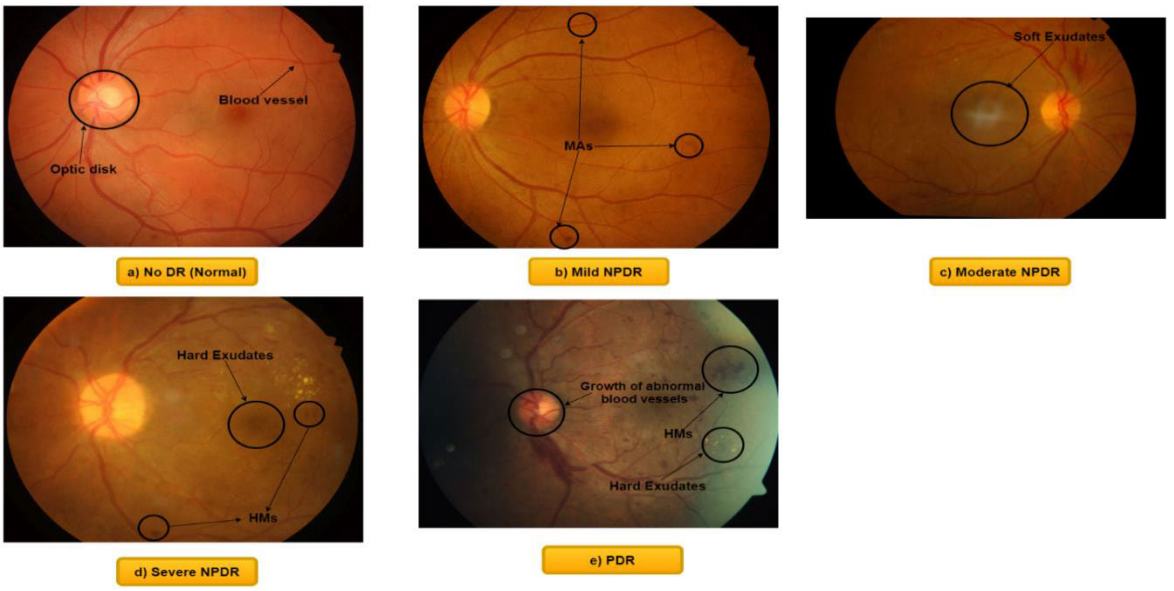

FIGURE 5. Difference stages of DR in fundus retinal image.

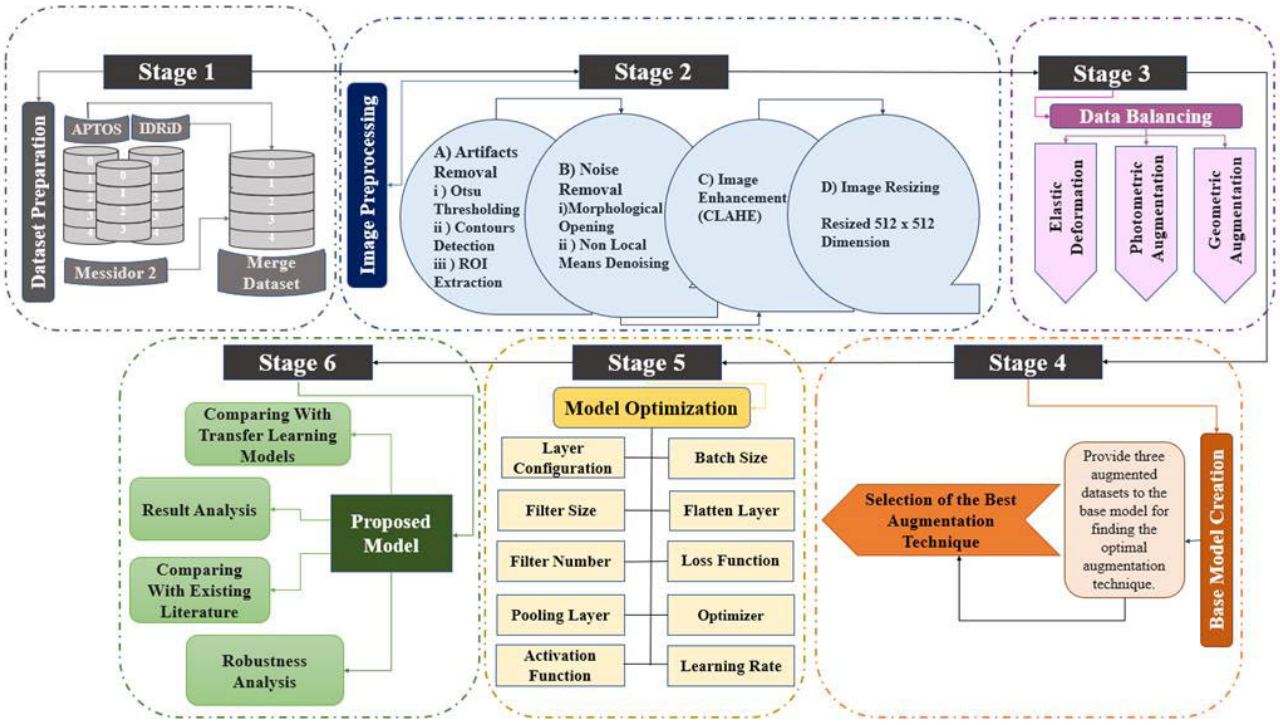

FIGURE 6. Total workflow diagram.

training models with different images is crucial. As previously stated, three different DR fundus image datasets: APTOS, Messidor2, and IDRiD, are combined to form one large dataset. Each dataset contains five classes: no DR (Grade 0), mild NPDR (Grade 1), moderate NPDR (Grade 2), severe NPDR (Grade 3), and PDR (Grade 4). Grade 0 comprises the fundus images of patients with no DR. From Grade 1 to Grade 5, the severity of the DR increases. The fundus images contain several features, including optic disc, arteries, veins, and blood vessels, and fundus images and several diabetic retinopathy markers, such as haemorrhages, soft exudates, hard exudates, and microaneurysms, as well.

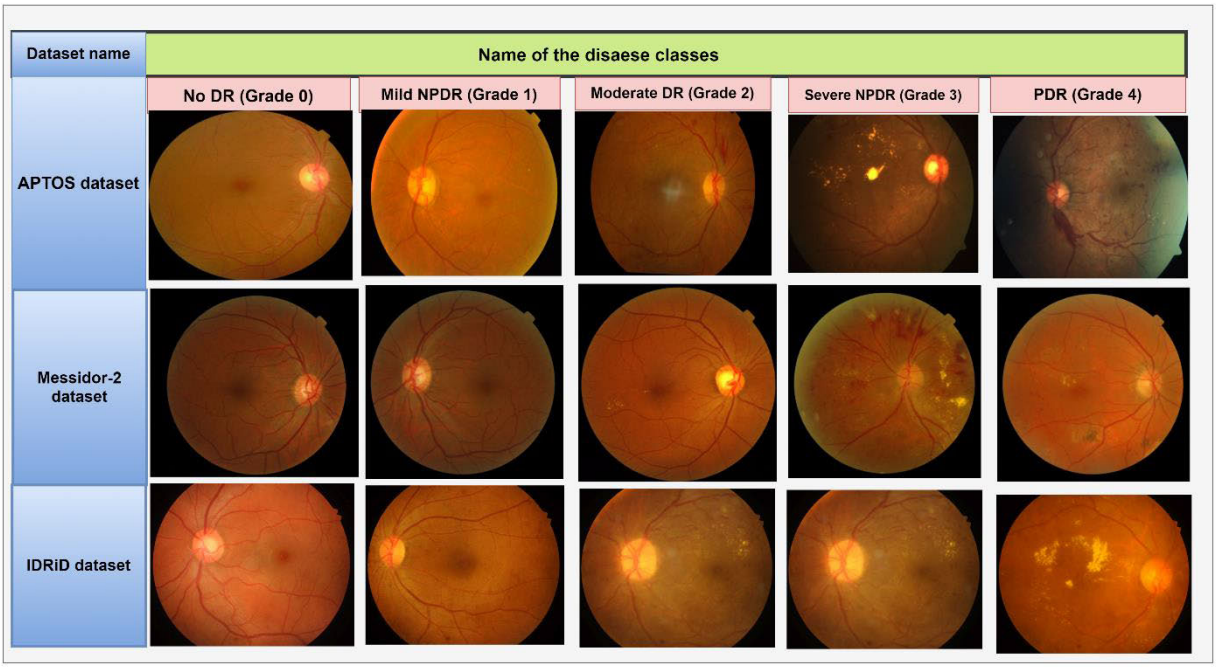

FIGURE 7. Sample images of the three datasets.

For this study, we have utilized the APTOS dataset provided by Kaggle, which contains 3,662 images [32], and the Messidor2 dataset, which has 1,744 retinal fundus images [33]. In addition, the IDRiD dataset is also employed in this study, and it contains a total of 413 fundus images [34]. Including the IDRiD dataset increased the number of raw images in Grade 3 and Grade 4 classes, which show more severe diabetic retinopathy. The APTOS collection contains photos obtained from individuals from rural areas in India. The Aravind Eye Hospital in India compiled the images of the APTOS dataset [32]. The Messidor-2 dataset comprises parts of the Messidor-Original and Messidor-Extension. Messidor-Original consists of 1058 images, and Messidor-Extension has 690 images [44]. Diabetic patients of the Ophthalmology department of Brest University Hospital (France) were recruited to compile the Messidor-2 dataset. The IDRiD dataset includes fundus images originating from an Eye Clinic in Nanded, Maharashtra, India [34].

After integrating these three distinct datasets, we had raw images for each class and could train our model using fundus images of different quality. Another advantage of combining the three datasets is that there are more images in the severe classes (Grade 3 and Grade 4). These two classes are required for a system to diagnose the severity of the DR.

Details of the datasets are given in Table 2 and an example of each of the five classes of the three-fundus datasets used in this study is shown in Fig 7.

Fundus image acquisition is accomplished using a fundus camera, where images are taken using in different lighting conditions from different angles. Images can have insufficient brightness and contrast, resulting in poor classification

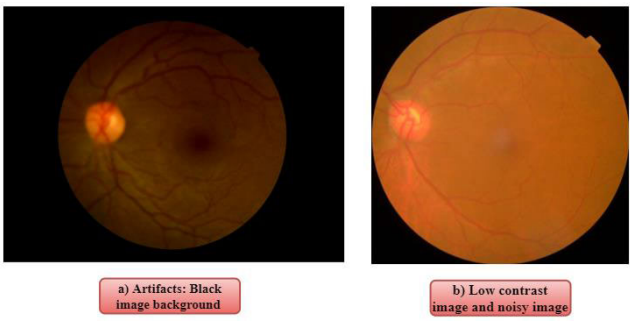

FIGURE 8. Challenges of the DR image dataset.

results [45]. This is a challenge when using fundus image datasets. Fig 8 shows some of the challenges of this retinal dataset including black image background (see Fig 8a) and noisy and low-contrast images (see Fig 8b) [46].

B. IMAGE PREPROCESSING

Image preprocessing, prior to utilizing images as a neural network input, is considered one of the most important steps in achieving good accuracy. It involves several steps, such as artifact removal, removal of undesired noise, and enhancing unclear but meaningful objects. Accurate and prompt classification of DR requires high-quality color retinal images [47]. Publicly available retinal fundus image datasets have been generated with various resolutions and compression formats and can contain background noise [48]. It is challenging to classify DR without using preprocessing techniques, as the neural network model that is used for classification often appears to require clean, enhanced, and moderately

symmetrical data. In the image preprocessing stage of this study, we first reviewed the retinal fundus images. We then applied Otsu thresholding, contour detection, and ROI extraction to eliminate artifacts. The output of this is used as the input for the noise removal step. In the noise removal step, the ROI image is converted into a binary image followed by morphological opening. A binary mask, along with the bitwise_AND function helps to convert the binary image into an RGB image. The fastNlMeansDenoisingColored function is then used to denoise the image. The noise-free image is converted into a YUV image, and CLAHE is applied to the Y channel of the image only. In addition, the enhanced image is resized to a 512 × 512 image. The preprocessing techniques used in this study are shown in Fig 9.

1) ARTIFACT REMOVAL

Undesirable regions or objects can inadvertently show up in the images. The removal of predominant artifacts is essential for the classification of DR. In our dataset of retinal fundus images, a black background is seen, which is not necessary for the classification task and can be considered an artifact. We apply Otsu thresholding, contour detection and sorting, boundary box finding, and region of interest extraction, to remove these artifacts.

a: OTSU THRESHOLDING

The Otsu method is a type of image thresholding to separate related data. The Otsu approach uses an image histogram to determine the ideal global threshold value [49]. Here, the Otsu thresholding method is performed on retinal fundus images to distinguish the background and the ROI of an image. This nonlinear operation transforms a grayscale image into a binary image. The input of the algorithm is usually a grayscale image, and its output is a binary image based on the original image's pixel intensity. If the intensity of a pixel is greater than the threshold, the corresponding output pixel is white. If the intensity of an input pixel is equal to or lower than the threshold, the output pixel is 0 or black. The threshold value can be calculated according to (1).

$$T = \frac{1}{2}\mu_1 + \mu_2 \tag{1}$$

In the above equation, T represents the value of the threshold and μ_1 *and* μ_2 represent the mean intensity. In our study, the cv2.threshold() method is used for the Otsu binarization process. The grayscale version of the retinal fundus image passes as a source parameter and cv2.THRESH_OTSU passes as an extra flag to indicate the initialization of Otsu's method. The parameter cv2.THRESH_OTSU selects the optimal threshold value using the Otsu algorithm, and the thresholding method cv2.THRESH_BINARY finds the pixel intensity. Fig 9B(i) shows the Otsu threshold mask.

b: CONTOUR DETECTION AND SORTING

A contour is an outline that represents the shape or form of an object, and contour detection extracts the curves that correspond to the shapes of objects in images [50]. The contours of the retinal fundus image can be detected using the cv2.findContours() function, where the binary image from Otsu's thresholding is used as a source image. After finding the contours, a sort of function is used to order them according to their area from the largest to the smallest contour. Two arguments are passed to the function, where the first is the contour list, and the second is the area found by cv2.contourArea.

c: EXTRACTION OF REGIONS OF INTEREST

The region of interest area is the target area in the retinal fundus image, which can be used for classifying diabetic retinopathy. To separate this region, we use the cv2.boundingRect() function. The sorted contours list is used as input. The cv2.boundingRect() function returns numbers corresponding to x, y, w, and h, respectively. These values represent the x coordinates, the y coordinates, the width, and the height. The region of interest can be cropped based on these pixel coordinates. Fig 9B (iii) illustrates the extraction of the ROI part, removing unnecessary black background padding.

2) NOISE REMOVAL

Fundus images are generally affected by noise and difficulties can also arise due to low contrast. These concerns make it challenging to identify and interpret diseases from retinal fundus images [51]. To eradicate the noise of the dataset's images, we first use morphological opening and then perform non-local means denoising. Fig 9C provides a visualization of the noise removal steps.

a: MORPHOLOGICAL OPENING

Morphological opening is used to smooth the optic discs and bright lesions. It can also aid in detecting microaneurysms and exudates [52]. We perform morphological opening on the previously acquired image of the extracted ROI. Before morphological opening is applied, the image is binarized using the cv2.threshold function (Fig. 9C(i)). Morphological opening is applied to the binary image using a kernel. We experimented with multiple kernel sizes, and the kernel size (10, 10) yielded the best results. The kernel size and cv2.MORPH OPEN are passed as parameters to the cv2.morphologyEx function to obtain a mask using the morphological opening algorithm. Using the bitwise AND function, the ROI image is integrated with the mask to produce a retinal fundus image with reduced noise. An overview of the morphological opening method is provided in Fig 9C(i).

b: NON-LOCAL MEANS DENOISING (NLMD)

Denoising the retinal fundus image is critical. However, we would like to keep features such as lesions and exudates for our classification task [53]. NLMD [54] is used to eliminate noise without removing essential features. The denoising of an image $x = (x1; x2; x3)$ in channel i to the pixel j is

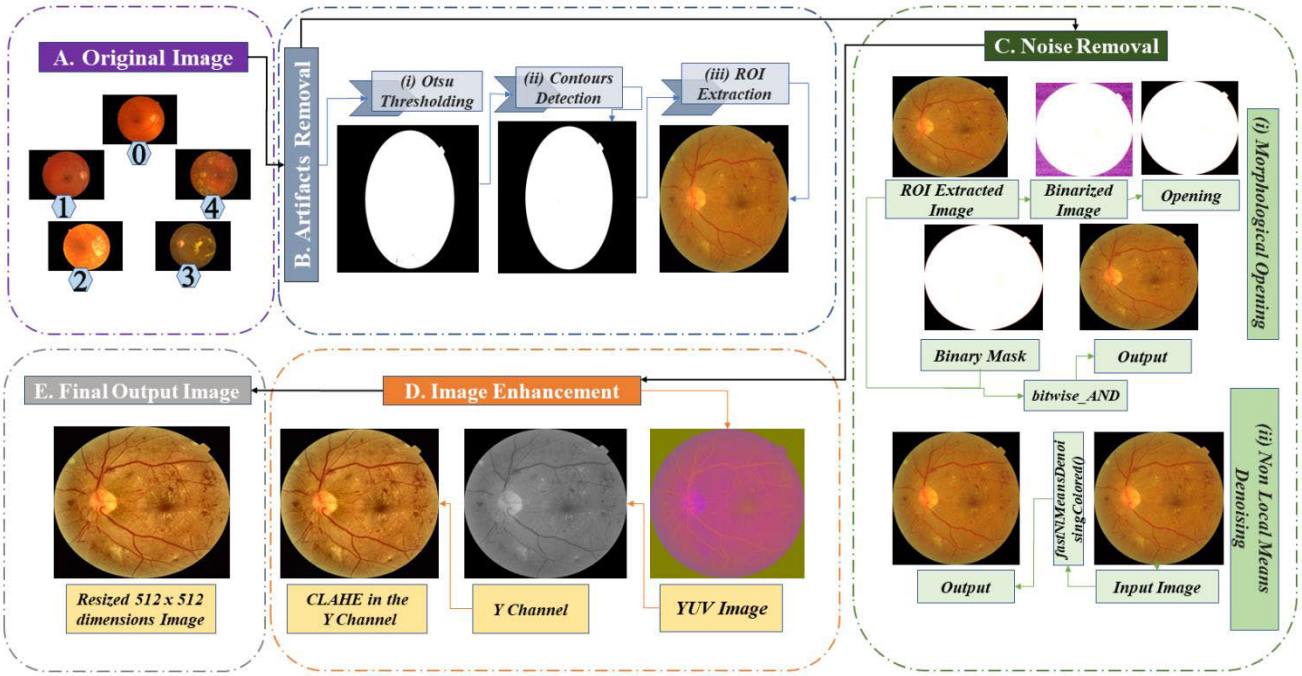

FIGURE 9. Preprocessing process.

executed according to (2) and (3) [53]:

$$\hat{x}_i(j) = \frac{1}{C(j)} \sum_{k \in B(j,r)} x_i(j)\, \omega(j,k) \tag{2}$$

$$C(j) = \sum_{k \in B(j,r)} \omega(j,k) \tag{3}$$

Here, $B(j, r)$ denotes a neighborhood of radius r surrounding pixel j. The weight w is determined by the squared Frobenius norm distance (or another induced norm distance) between color patches with centers at j and k which decay under a Gaussian kernel (j, k). To perform the NLMD, we use the cv2.fastNlMeansDenoisingColored() function of OpenCV. The image obtained after morphological opening is used as the source image. Other parameters are h and hColor, where h is the filter strength tuning parameter for the luminance component, and hColor is used for the color components. After experimenting with different values for the h and hColor parameters, we assign 1 to both parameters, considering that a large value for h removes noise perfectly, but also eliminates image details. A smaller h value preserves detail more accurately. For the templateWindowSize and searchWindowSize parameters, we use the recommended values 7 and 21, respectively. Fig 9C(ii) shows a noise-free image after NLMD.

3) IMAGE ENHANCEMENT (CLAHE)

When NLMD is employed, the image noise is eliminated. CLAHE is subsequently applied to the denoised images of the retinal fundus in order to improve contrast. CLAHE is a sophisticated variant of adaptive histogram equalization (AHE). Illumination can be distorted in three ways: darkness, brightness, and uneven illumination [55]. To address this, RGB color channels can be transformed into the YUV channels where the Y channel (Fig 9D) represents illumination components [47], [55], [56]. In our study, we converted the images of NLMD output into YUV color space format and separated the Y Channel. CLAHE is implemented utilizing cv2.createCLAHE function, which involves two parameters: clipLimit and tileGridSize. We selected the YUV color space format to get the overall illumination components. Several parameters' configurations are investigated, such as the clipLimit with values (0.5, 1.0, 2.0, 3.0) and tileGridSize with values ((3,3), (8,8), (15,15), (20,20)). We found that clipLimit with a value of 0.5 and tileGridSize with a value of (8,8) produced the best results. We select these values and employ CLAHE on the Y channel. Fig 9D shows the retinal fundus image of YUV color space, then the Y channel, and finally, the output image after applying CLAHE.

4) IMAGE RESIZING

After obtaining the preprocessed image, it is essential to feed the same-sized images to the neural network to reduce the computational overhead of the model. Therefore, we have resized the enhanced images to the dimensions of (512,512).

After completing all image preprocessing steps, we obtain enhanced images without artifacts and noise. It can be seen from Fig 10(A) that the original image contains unnecessary black background regions, considered artifacts, and that different features of the fundus images are not clearly visible. To select the fundus image part, we have cropped the black background and eliminated the small noises in the fundus images, as shown in Fig 10(B) and Fig 10(C), respectively.

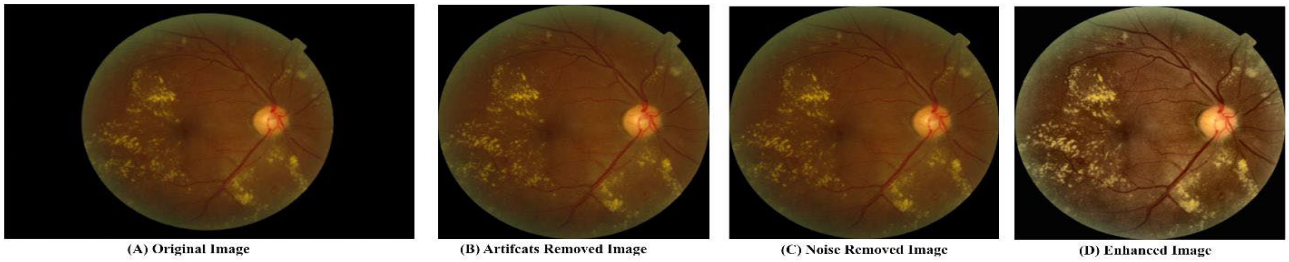

FIGURE 10. Visualization of the images after different stages of image preprocessing.

We have enhanced our fundus images to better visualize the features in the fundus image. Fig 10(D) depicts an enhanced image where the blood lesions and other attributes of the images are more visible than in the original image.

C. DATA AUGMENTATION

Data Augmentation [57] was employed in our study to balance the dataset. Multiple techniques can be used to prevent overfitting issues, but data augmentation is considered a core task. Augmentation-based oversampling techniques are frequently used to increase diversity and to mitigate possible overfitting [58]. Increasing the amount of data should be done in an efficient way so that it can generate samples similar to all possible images and balance the datasets. Another concern is not to reduce the quality of the images, especially in the medical domain where preserving image features may be essential [59]. Therefore, the augmentation method should generate the images without affecting the quality of the images.

In our study we utilize three augmentation techniques: geometric augmentation, photometric augmentation and elastic deformation to generate data. We obtain the same number of augmented images with all methods.

1) GEOMETRIC AUGMENTATION

Geometric transformation is the most prevalent data-balancing augmentation technique [60]. The procedure of changing the geometric shape of an image by altering its values to corresponding new values is called geometric augmentation. It is an efficient enhancement technique that does not affect the image quality and only transforms it into a new shape [61]. In our study, we conducted four augmentation strategies for geometric augmentation: vertical flipping, horizontal flipping and rotation (30°, −30°, 60°, −60°, 120°, −120°, 150° and −150°).

a: VERTICAL FLIPPING

Flipping the image across to its vertical axis is known as vertical flipping. In vertical flipping, the lower sides are flipped to the upper side and vice versa [62]. Equation (4)

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{4}$$

Here, f_x and f_y are the transformed coordinate values, and x and y refer to the original image's pixel value.

b: HORIZONTAL FLIPPING

The image can be horizontally flipped, and we have employed this strategy in our study. The original pixel coordinate values are changed horizontally as (5) [62]:

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{5}$$

Here, f_x and f_y are the transformed coordinate values, and x and y refer to the original image's pixel value.

c: VERTICAL AND HORIZONTAL FLIPPING

This method preserves horizontal and vertical columns and rotates the image horizontally and vertically. Equation (6) [62] appears below:

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{6}$$

Here, f_x and f_y are the transformed coordinate values, and x and y refer to the original image's pixel coordinates.

d: ROTATION

Rotation is a well-known augmentation technique in image augmentation [63]. Depending on the need, the image can be rotated at any angle. Even if we rotate the image at any orientation, the image information remains the same. Equation (7) is used for the rotation technique [62].

$$\begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{7}$$

In the equation 7, f_x and f_y two variables represent the new position of each pixel after the rotation operation, and x and y represent pixel of the original image. And $cos\theta$ and $sin\theta$ are used to determine the angles.

2) ELASTIC DEFORMATION

The applied forces induce a stress field within a continuous body, which leads to deformation. If the original shape is recovered when the stress field is removed, the deformation is elastic [64]. Using this method, the retinal fundus will be visible but in a deformed manner in which the image will

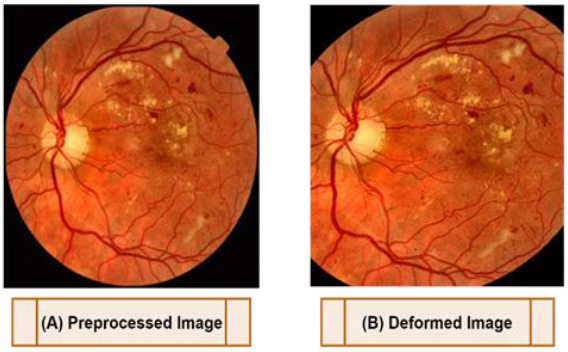

FIGURE 11. Elastic deformation of the retinal fundus image.

appear in a stretched position without the loss of valuable information. Fig 11 illustrates the deformed version of the preprocessed retinal fundus image. From the image, it can be seen that the image has been deformed. However, all of its essential features remain intact.

3) PHOTOMETRIC AUGMENTATION

The photometric augmentation technique modifies the RGB channels by shifting each pixel's (r, g, and b) value to a new (r′, g′, and b′) value according to predefined heuristics. It only modifies the images' color and lighting while preserving their geometry [65], [66]. The primary techniques include color jittering, gray scaling, filtering, light perturbation, noise addition, vignetting, contrast modification, and random erasing [65]. Increasing the number of samples is a necessary endeavor. However, it needs to be done in such a way that essential pixel information is preserved and will not cause overfitting issues. Several photometric approaches have been tried in the preliminary study, including HE, saturation, Gaussian noise, hue, altering brightness, contrast, color, and sharpness. However, adjusting brightness, contrast, color and sharpness provided the best results, and these four approaches are applied as photometric augmentation in this paper.

a: BRIGHTNESS ALTERING

The concept of brightness refers to an image's overall level of lightness or darkness. Equation (8) has been utilized to adjust the brightness of images and increase the number of images.

$$Brightness(x) = Source(x) + factor \tag{8}$$

Here, *Source (x)* refers to the input pixels, and *Brightness (x)* signifies output pixels after changing the brightness level employing factor values. A factor value less than 1 indicates a darker image and a factor value greater than 1 indicates a brighter image. An augmented image after altering the brightness is shown in Fig 12B.

Contrast Altering: Contrast is defined as the variation in brightness between pixels that make up the ROI and those that make up the background in a picture. When the contrast is increased, the light regions become lighter, and the dark ones become darker. The equation for modifying an image's contrast level can be written as follows:

$$Contrast(x) = Source(x) + factor \tag{9}$$

Factor values of more than 1 increase the contrast, and values less than 1 decrease it. Fig 12C shows an augmented image with the contrast adjusted.

b: COLOR ALTERING

Changing the level of the color of an image is also an efficient way to enhance the image. Using equation (10), we changed the color balance of an image, where *color (x)* represents the resultant image and *Source (x)* is the original image. A factor greater than 1 makes the color stronger, whereas a factor less than 1 reduces the colors. Fig 12D displays the altered image produced by adjusting the level of color.

$$Color(x) = Source(x) + factor \tag{10}$$

c: SHARPNESS ALTERING

Sharpness defines the details the imaging system can retain. It is an important factor in image quality. The borders between tonal zones are what define sharpness. The following (11) is the formula for sharpness:

$$Sharpness(x) = Source(x) + factor \tag{11}$$

In this case, applying sharpening with a factor of more than 1 enhances the image's edges and makes them appear more defined. A factor lower than 1 blurs and softens the image. Fig 12E depicts the augmented image after adjusting the sharpness.

For our study, we have applied several factor values to enhance the number of images without losing important pixel information. After experimentation, a minimum factor value of 0.5 and a maximum value of 1.8 produced good, augmented images. Fig 12 depicts all the photometric augmented images of the study.

Concise explanation of generating augmented datasets.

As shown in Table 3, we utilized a merged dataset, and the number of images in each class of the resulting dataset is highly unbalanced. Severe NPDR (grade 3), PDR (grade 4), and mild NPDR (grade 1) contain the lowest numbers of images. Compared to severe NPDR, moderate NPDR has 4.33 times, and No DR has 8.64 times the number of images, which indicates the imbalance is significant. Thus, we have taken steps to balance the images of each class. The number of images in the No DR class is increased by a factor of 2. To balance the dataset, we increase the number of images for NPDR, PDR, mild NPDR, and moderate NPDR by a factor of 16, 14, 8, and 3, respectively. To create three separate augmented datasets, we balance the dataset with three augmentation approaches (elastic deformation, geometric augmentation, and photometric augmentation). After that, we run our base model on these three datasets. The base model performance is based on the model's test accuracy, and the

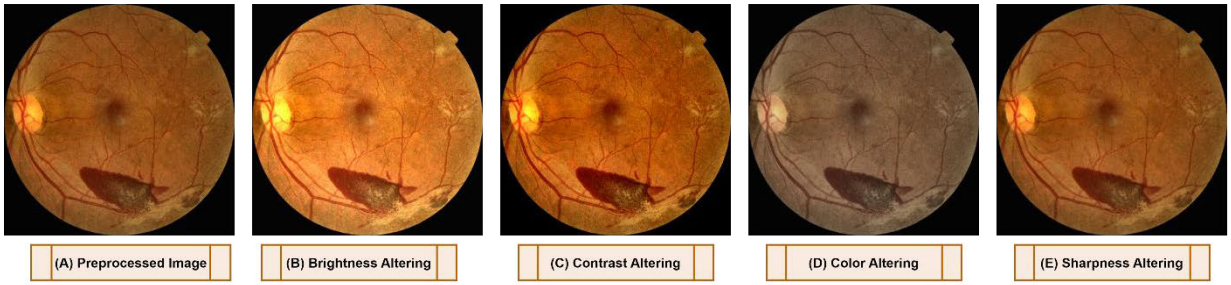

FIGURE 12. Photometric augmentation of the retinal fundus image.

TABLE 3. Image counts details of original and augmented datasets.

Dataset: Without Augmentation					Merge Dataset: With Augmentation		
Datasets	Messidor2 [33]	APTOS [32]	IDRiD [34]	Merge	Geometric	Photometric	Elastic Deformation
No DR (0)	1017	1805	134	2956	5912	5912	5912
Mild NPDR (1)	270	370	20	660	5940	5940	5940
Moderate NPDR (2)	347	999	136	1482	5928	5928	5928
Severe NPDR (3)	75	193	74	342	5814	5814	5814
PDR (4)	35	295	49	379	5685	5685	5685
Total Images	1744	3662	413	5819	29279	29279	29279

augmented photometric dataset performs significantly better. Hence, we continued our further implementation using the photometric augmented merge datasets. The detailed result of the augmented dataset is described in section IV (F (2)).

D. DATASET SPLIT

Splitting the dataset is the last step before feeding the images into the proposed model. We split the augmented dataset using an 80:10:10 ratio for training, validation, and testing, respectively. The training dataset contains 23,471 images, whereas the validation and testing datasets have 2,902 and 2,906 images, respectively.

E. EXPERIMENTAL SETUP

This study uses an AMD Ryzen 5 5600X 6-core Central Processing Unit (CPU) and 16 GB of RAM for all the experiments. It is paired with Graphical Processing Unit (GPU) named ZOTAC GAMING GeForce RTX 3060 Twin Edge OC GDDR6 with 12 GB video ram (VRAM). Jupyter Notebook version 6.4.12 has been utilized as the IDE.

F. PERFORMANCE OF OUR PROPOSED MODEL

We propose a computer-aided diagnosis system for classifying diabetic retinopathy using retinal fundus images in this study. CNN-based architectures have previously been applied in classifying and detecting diabetic retinopathy [67]. We discuss our base model's performance on the three augmented datasets. After getting the best-performing augmented dataset, we conduct ten case studies to propose the optimal model structure.

1) BASE CONVOLUTION NEURAL NETWORK MODEL

Our experiment begins with a base CNN model. We implemented the base model architecture from scratch. It consists of three convolutional layers, each accompanied by a maximum pool layer. The base model's loss function was categorical cross-entropy, and the initial learning rate was set at 0.0001 with Adam as the optimizer and a batch size of 64. Fig 13 illustrates the base model.

The base model contains three blocks, each consisting of one convolution and one maxpool layer. The input layer of block 1 is connected to the first Conv2D layer, which has a kernel size of 64. The dimensions of the input image are $224 \times 224 \times 3$. The first maxpool layer scales down the output of the first Conv2D layer to 111×111. Block 2 and 3 have similar configurations except for their kernel size. The kernel sizes for the second and third blocks are 64 and 128, respectively. The Conv2D layers of these blocks are followed by their corresponding maxpool layer. After that, a flattened layer with 86,528 parameters is added, followed by a dense layer and a drop-out layer with a value of 0.5. Since there are five classes in our dataset, the classification dense layer,

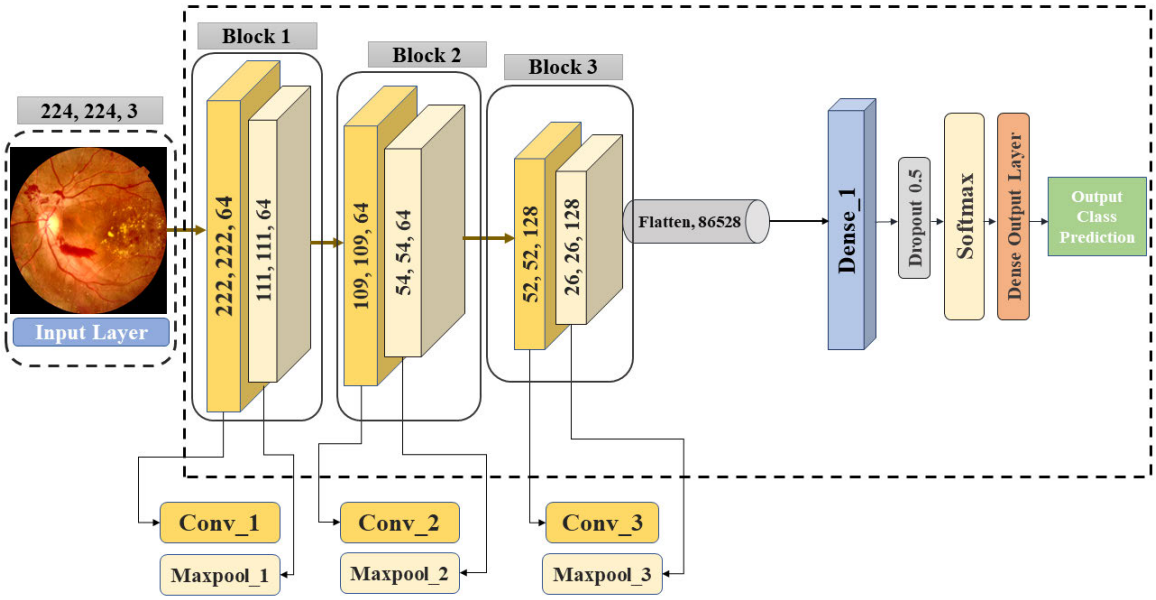

FIGURE 13. Framework of base CNN model.

the fully connected layer (FC), contains five neurons. It is equipped with the Softmax activation function.

The first two convolution layers have 64 filters, and the last one has 128 filters. The convolutional part is a significant part of our model since it helps to extract the features [68]. The formula of the convolution operation can be expressed according to (12) [69]:

$$Conv\,(i,j) = (x * w)\,[i,j] \sum_m \sum_n x\,[m,n]\,w[i-m, j-n] \tag{12}$$

Here, $Conv\,(i,j)$ is the output of the next layer, the input image or feature map is represented as x while w is the kernel, m, n represent the size of the kernel, and $*$ is the convolution operation. The convolution operation formulates a feature map. Feature maps show the result of the previous layer following the application of the filter. Equation (13) is the formula to calculate the feature map [70].

$$h^k = f\left(w^k * x + b^k\right) \tag{13}$$

where h^k represents the output of the feature map, w is the weight, and b^k is the bias.

Filter sizes of 3 × 3 and the activation function rectified linear unit (ReLU) extract the information efficiently from each convolution layer. Therefore, a ReLU activation function is added to every convolutional layer. The pooling technique reduces unnecessary details from the feature map area. Equation (14) is the mathematical expression for the pooling is given below [70]:

$$y_{kij} = max_{(p,q)\in R_{ij}} x_{kpq} \tag{14}$$

TABLE 4. Base model accuracies based on three types of augmentation technique.

Augmentation Technique	Validation Accuracy (%)	Test Accuracy (%)
Geometric	87.32	86.19
Photometric	**90.12**	**89.23**
Elastic Deformation	81.54	80.76

Here y_{kij} is the output of the pooling operator to the corresponding kth feature map and x_{kpk} is the element at (p, q) within the pooling region and R_{ij} are the local neighbourhood coordinates. Finally, we flatten the matrix using two dense layers with the fully connected layer. The softmax activation function is used for the classifier. After setting these hyperparameters and building the model, it runs for 100 epochs.

2) ACCURACY OF THE BASE MODEL

Three different augmentation methods are applied to the merged dataset, and the base model is used to identify the optimal augmentation technique. Table 4 illustrates that our base model achieved a validation accuracy of 87.32%, 90.12%, and 81.54% for geometric, photometric, and elastic deformation augmentation methods, respectively. The base model also acquired a test accuracy of 86.19% for geometric, 89.23% for photometric, and 80.76% for elastic deformation augmentation. As can be seen from Table 4, the base model acquired the highest validation and test accuracy after the photometric augmentation technique was applied. We,

therefore, proceed with the photometric augmentation technique to further study.

3) MODEL OPTIMIZATION

The nature and characteristics of a task and any potentially related issues should be considered to determine the optimal layer architecture and configuration of a CNN model. A model optimization strategy is a set of experiments in which hyperparameters of CNN architecture are tuned to evaluate these parameters' impact on the model's performance [71]. The purpose is to gain a complete understanding of the model's performance by studying the effects of modifying specific hyperparameters [72]. This method can recognize possible model performance issues, which may be addressed by updating and altering the network. Thus, we trained our base CNN model various times by varying layer numbers, filter sizes, filter numbers, hyper-parameters, and parameter values in order to achieve optimal performance while minimizing computational complexity.

a: RESULTS OF THE MODEL OPTIMIZATION

In section IV (E (1)), we discuss the configuration of our base model. The base CNN model is altered, and the results are recorded to determine the optimal architecture configurations utilizing ten case studies. We present the time complexity and training time per epoch to evaluate the performance compared to other configurations and test accuracy to select the best configuration for our model. We evaluate the training time considering that in real-time applications in remote areas, resource constraints can sometimes be very challenging. The need to utilize limited resources for a long time for training purposes can sometimes be problematic. We therefore consider the training time to evaluate the model, where an optimized model can provide fast-training accuracy along with fast testing accuracy.

The convolutional layers consume a large proportion of the computational, whereas the time pooling and fully connected layers only consume 5 to 10 percent of the computational time [73], [74]. We therefore focus on the time complexity of the convolutional layers; see Table 5 and 6 [74], [75], [76]. We compute the theoretical time complexity, which is defined in [73] as follows:

$$O = \sum_{j=1}^{k} n_{j-1}.s_w.s_h.n_j.m_w.m_h \tag{15}$$

In the (15), the index of the convolutional layer is denoted by j and the number of convolutional layers is denoted by k. n_{j-1} represents the number of filters (input channels) in the j-1th layer, whereas n_j represents the number of filters (output channels) in the jth layer. Lastly, s_w and s_h represent the width and height of the filters, and m_w and m_h the width and height of the output feature map. Table 5 and 6 give an overview of the results of the comprehensive model optimization. The results of the model's layer configurations and activation functions are presented in Table 5, while the outcomes of tuning hyper-parameters, the loss function, and the flatten layer are shown in Table 6. The time complexity is expressed in Millions (M).

i) CASE STUDY 1: CHANGING CONVOLUTIONAL AND MAX-POOL LAYERS

In case study 1, the number of convolutional and max-pool layers is altered, but the configuration for the base model is kept as it is. We commenced with three convolution layers, followed by three max-pool layers. Table 5 shows the accuracy of various model configurations along with the time complexity and training time. Configuration 5 produced the best result with an accuracy of 92.70%, which is almost 2% better than the second-best result. We achieve the maximum accuracy with the lowest training time: 84 epochs with a time of 141s per epoch. Configuration 5 contains five convolutional layers and three max-pool layers. This configuration of convolutional layers and max-pool layers was used for the remaining studies. The time complexity was reduced from 63.73M (time complexity of Base Model) to 57.67 M (time complexity of Configuration 5). Fig 14 (Layer Configuration) shows a rise in test accuracy and a decrease in time complexity.

ii) CASE STUDY 2: CHANGING FILTER SIZE

In case study 2, we investigated various filter sizes, 2×2 and 4×4. A filter size of 3×3 was used previously. Filter size 2×2 provided the lowest test accuracy of 89.76 %, while 4×4 resulted in an accuracy of 90.15%. In both cases, the accuracy dropped from the best previous accuracy. As shown in Table 5, filter size 3×3 resulted in the best accuracy of 92.70%, while the time complexity, epoch number and training time were lower than for the 4×4 filter size. Hence, configuration 1, a filter size of 3×3, is employed for additional case studies.

iii) CASE STUDY 3: CHANGING THE NUMBER OF FILTERS

Altering the number of filters in different layers can affect the performance. We initially used a fixed number of filters, 64, in all the convolution layers. The performance is decreased when the number of filters is reduced to 32. In configurations 3 and 4 (Table 5), we tried different numbers of filters in separate layers. Configuration 4, with filter numbers in a sequence of 16, 32, 64, 32, and 64 for the five convolutional layers, obtained the best performance with a test accuracy of 95.34%. This configuration also provided the lowest model training time and the second lowest time complexity, although the time complexity was higher than for configuration 2. We selected configuration 4 for its accuracy and proceeded with this configuration. The time complexity solely depends on filter size, feature map size, and kernel size, and these parameters remain the same for the rest of the case studies. Therefore, the time complexity for configuration 4 remained at 35.80 M for the subsequent case studies.

TABLE 5. Investigation of layer configurations and activation functions for model optimization.

Case Study 1: altering convolution and maxpool layer					
Configuration No.	No. of convolution layers	No. of pooling layers	Time complexity	Epoch x training time	Test accuracy (%)
1	3	3	63.73 M	91 x 141s	89.23
2	4	3	58.01 M	89 x 141s	90.02
3	4	4	72.19 M	91 x 141s	89.74
4	5	5	56.26 M	88 x 141s	90.87
5	**5**	**3**	**57.67 M**	**84 x 141s**	**92.70**
6	6	6	56.30 M	95 x 145s	90.45
Case Study 2: altering filter size					
Configuration No.	Filter size	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	**3 x 3**	**57.67 M**	**84 x 141s**	**92.70**	**Previous accuracy**
2	2 x 2	26.73 M	88 x 138s	89.76	Accuracy dropped
3	4 x 4	97.48 M	96 x 149s	90.15	Accuracy dropped
Case Study 3: altering the number of filter					
Configuration No.	No. of kernel	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	64→64→64→64→64	57.67 M	84 x 141s	92.70	Previous accuracy
2	32→32→32→32→32	26.11 M	91 x 141s	90.47	Accuracy dropped
3	32→32→64→64→128	59.34 M	85 x 141s	94.86	Near highest accuracy
4	**16→32→64→32→64**	**35.80 M**	**82 x 141s**	**95.34**	**Highest accuracy**
Case Study 4: altering type of pooling layer					
Configuration No.	Type of pooling layer	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	**Max**	**35.80 M**	**82 x 141s**	**95.34**	**Previous accuracy**
2	Average	35.80 M	85 x 142s	93.78	Modest accuracy
Case Study 5: altering activation function					
Configuration No.	Activation function	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	**PReLU**	**35.80 M**	**81 x 141s**	**96.12**	**Highest accuracy**
2	ReLU	35.80 M	82 x 141s	95.34	Previous accuracy
3	Leaky ReLU	35.80 M	96 x 149s	93.71	Accuracy dropped
4	Tanh	35.80 M	94 x 152s	92.35	Accuracy dropped

iv) CASE STUDY 4: CHANGING THE TYPE OF POOLING LAYER

Max pool and Average pool, two pooling layers, are assessed for case study 4, as shown in Table 5. The max pooling layer provided the best accuracy of 95.34%, with less training time for each epoch, and it took fewer epochs to gain the highest accuracy than average pooling. We, therefore, selected the max pool for further investigation.

v) CASE STUDY 5: CHANGING THE ACTIVATION FUNCTION

Selecting the best activation function for a model is an essential task in model building, as different activation functions are performed in different ways. We experimented with four activation functions: PReLU, ReLU, Leaky ReLU, and Tanh. From Table 5, we can see that utilizing PReLU increased the accuracy compared to ReLU and provided the highest accuracy of 96.12%. Further investigations of model optimization, therefore, employed the PReLU activation function.

vi) CASE STUDY 6: CHANGING THE BATCH SIZE

Batch size altering can also improve the performance of the model. A large batch size might result in the model taking a long time to converge [75], [76]. Some studies [77], [78], [79] suggest that reducing the batch size enables the network to train more effectively, whereas increasing the batch size degrades the test performance. Three batch sizes (Table 6) were employed for this experiment, and it was found that a batch size of 32 resulted in the highest test accuracy of 97.47%. Although batch sizes 64 and 128 required less time per epoch to complete, a batch size of 32 obtained better accuracy and required fewer epochs. Therefore, a batch size of 32 was selected for further case studies.

vii) CASE STUDY 7: CHANGING THE FLATTEN LAYER

A flatten layer transforms the previous layer's output into a single one-dimensional vector, which can be used as an input for a dense layer. The results of experiments with Global Max pooling and Global Average pooling demonstrate that the previously employed flatten layer generates the highest test accuracy of 97.47% while maintaining the minimum training time (Table 6). Hence the flatten layer remains as in the base model.

viii) CASE STUDY 8: CHANGING THE LOSS FUNCTIONS

Experiments were conducted with various loss functions, including Binary Crossentropy, Categorical Crossentropy, Mean Squared Error, and Mean Absolute Error, to find the best loss function for our network. The model achieved the best test accuracy of 97.47% (Table 6) when integrated with Categorical Crossentropy. As a result, this is chosen for the model's loss function.

ix) CASE STUDY 9: CHANGING THE OPTIMIZER

Experiments were conducted using a variety of optimizers, including Adam, Nadam, SGD, Adamax, and RMSprop, to find the best optimizer. The Adam optimizer attained the

TABLE 6. Investigation of model hyper-parameters, loss function and flatten layers for model optimization.

Case Study 6: altering batch size					
Configuration No.	Batch size	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	**32**	**35.80 M**	**80 x 142s**	**97.47**	**Highest accuracy**
2	64	35.80 M	81 x 141s	96.12	Previous accuracy
3	128	35.80M	92 x 139s	89.75	Accuracy dropped
Case Study 7: altering flatten layer					
Configuration No.	Flatten layer type	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	**Flatten**	**35.80 M**	**80 x 142s**	**97.47**	**Previous accuracy**
2	Global Max pooling	35.80 M	89 x 146s	94.31	Accuracy dropped
3	Global Average pooling	35.80 M	93 x 149s	94.47	Accuracy dropped
Case Study 8: altering loss functions					
Configuration No.	Loss Function	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	Binary Crossentropy	35.80 M	94 x 148s	91.32	Accuracy dropped
2	**Categorical Crossentropy**	**35.80 M**	**80 x 142s**	**97.47**	**Previous accuracy**
3	Mean Squared Error	35.80 M	88 x 146s	93.74	Accuracy dropped
4	Mean absolute error	35.80 M	98 x 147s	93.68	Accuracy dropped
Case Study 9: altering optimizer					
Configuration No.	Optimizer	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	**Adam**	**35.80 M**	**80 x 142s**	**97.47**	**Previous accuracy**
2	Nadam	35.80 M	85 x 143s	96.73	Accuracy dropped
3	SGD	35.80 M	93 x 146s	95.19	Accuracy dropped
4	Adamax	35.80 M	91 x 144s	96.81	Accuracy dropped
5	RMSprop	35.80 M	96 x 144s	94.33	Accuracy dropped
Case Study 10: altering learning rate					
Configuration No.	Learning rate	Time complexity	Epoch x training time	Test accuracy (%)	Finding
1	0.01	35.80 M	83 x 140s	96.73	Accuracy dropped
2	0.007	35.80 M	87 x 142s	97.23	Accuracy dropped
3	**0.001**	**35.80 M**	**80 x 142s**	**98.65**	**Highest accuracy**
4	0.0007	35.80 M	89 x 144s	96.82	Accuracy dropped
5	0.0001	35.80 M	80 x 142s	97.47	Previous accuracy

best test accuracy of 97.47% (Table 6). We, therefore, decided to continue our study of model optimization with the Adam optimizer.

x) CASE STUDY 10: CHANGING THE LEARNING RATE

A study was performed using learning rates of 0.01, 0.007, 0.001, 0.0007, and 0.0001. We initiate our base model with a 0.0001 learning rate. It can be seen (Table 6) that the accuracy dropped when we trained our model with a learning rate of 0.01, 0.007, and 0.0007. However, after tuning the model with a 0.001 learning rate, the best test accuracy of 98.65% was achieved. We use this learning rate for our proposed model.

Fig 14 depicts the optimal configuration of our proposed model architecture after conducting ten case studies. It also illustrates the importance of performing model optimization, enhancing the model accuracy in some stages without degrading the model's performance in other stages. We can see the increase in test accuracy and decrease in the time complexity in Fig 14. These characteristics indicate that the shallow architecture of our proposed model is effective.

4) PROPOSED MODEL

After performing ten case studies described in the previous section, we propose a shallow architecture (RetNet-10). Our proposed architecture consists of several modules and layers.

The proposed model, RetNet-10, comprises of ten layers, five convolutional layers, three max-pooling layers, and two dense layers. Three blocks are present in the model, where the last two blocks contain two convolutional layers followed by one max-pooling layer, while the first block includes only one convolution layer followed by a max-pooling layer. All the 3×3 kernel-sized convolutional layers have the parametric rectified linear unit (PReLU), a non-linear activation function and a stride size of 1×1. PReLU performed better in our model than ReLU because PReLU avoids the direct death of the neurons [80]. Equation (16) is the mathematical equation to represent PReLU, where x is the value of the neuron and a is a coefficient that determines the negative slope, which can be stated as:

$$f(x) = \begin{cases} x & if\ x > 0 \\ ax & if\ x < 0 \end{cases} \tag{16}$$

In our proposed model, RetNet-10, there are 39,954,821 trainable parameters. During model training, the initial weights extract features from the input data, and the loss function calculates the network's error rate. The kernels' weights are then estimated based on the error rate after each training epoch. This allows for the adjustment of the kernels after each epoch and the extraction of the best features. The input is sent to Block-1, where the first convolution layer contains 16 filters and a total of 788992 training parameters. For the first convolution layer a low number of filters (16) preserves the structural details and distinct textural features of the input image. Each input generates a total of 16 feature maps which are then corrected with PReLU, only maintaining the feature maps' non-negative values. The first convolutional

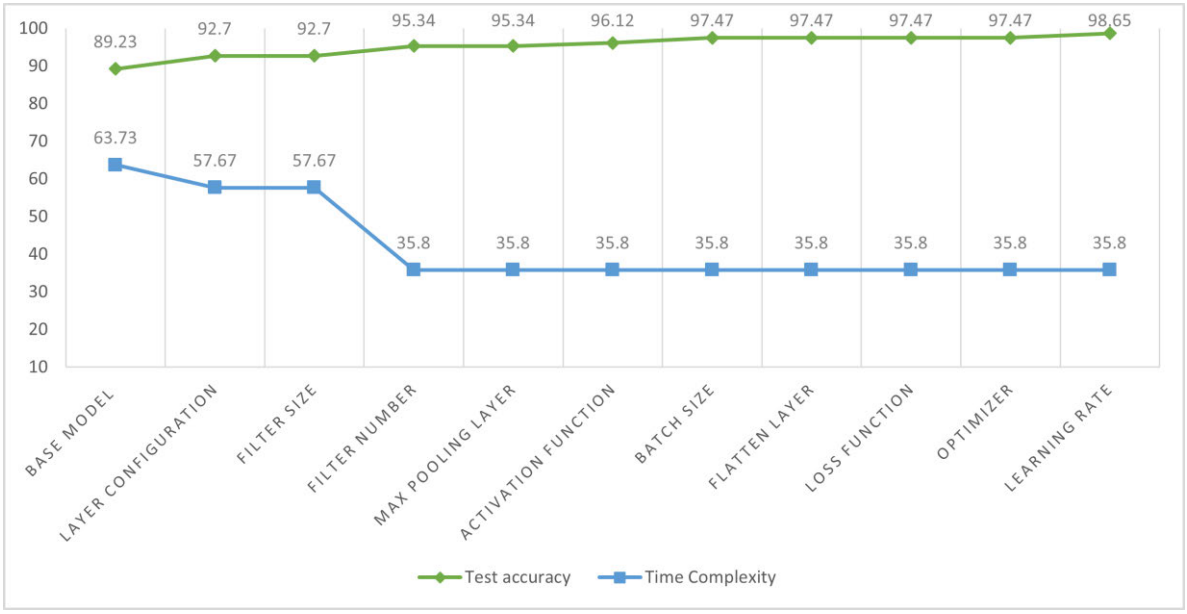

FIGURE 14. Visualization of each case study's time complexity (computed in millions and scaled from 0 to 100) and test accuracy.

layer's output feature maps are then shrunk by half using a 2 × 2 max-pool layer. Block-2 and Block-3 consist of 32 and 64 filters in their two convolution layers. Block-2 has 384832 and 751232 parameters for 32 and 64 filters, respectively, and Block-3 has 101696 and 172160 parameters for 32 and 64 filters, respectively. The resulting feature map of the last convolution layer of Block-3, composed of 64 filters, is 49 by 49 pixels in size. The max-pool layer then reduces the feature maps to 24 pixels by 24 pixels, lowering computational complexity while preserving important input image information. After block 3, a total of 64 feature maps have been generated, including deeper features of the input data with more intricate forms and objects than in the preceding blocks.

Block-3's resultant multidimensional feature maps are flattened into a 1D vector comprising 36864 values for each input. The fully connected (FC) layer, which has 1,024 neurons with the PReLU activation function, follows the flatten layer. Each element of the resulting 1D array functions as an input for the first FC layer, connecting each of the layer's neurons with that input neuron. This connection of the input neurons to the FC neurons, the weight, can be updated by backpropagation after each epoch. A dropout layer with a value of 0.5 follows the first FC layer. A second FC layer is then added: a classification layer with five neurons and a softmax activation function. The softmax activation function provides prediction results for all five classes of our dataset as this layer further generalizes the features. The weights of the fully connected layer and convolutional layers are adjusted after each epoch, based on the error rate determined by the categorical loss function. Fig 15 represents the visualization of our proposed architecture RetNet10.

V. RESULTS AND ANALYSIS

A. EVALUATION METRICS

To assess all the classification models, including existing deep learning models as well as our proposed model, we have applied several metrics named precision, recall, specificity, accuracy (ACC), false positive rate (FPR), false negative rate (FNR), false discovery rate (FDR), and the negative predicted value (NPV) in this study. The confusion matrix of the best-proposed model is shown in Fig 16. In general, the performance metrics values are computed by using the false positive (FP), false negative (FN), true positive (TP), and true negative (TN) values according to equations (17) to (25) below. For equation (26) and (27), the total number of observations is denoted as m, and x^p denotes the predicted value of x [76], [81].

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{17}$$

$$recall = \frac{TP}{TP + FN} \tag{18}$$

$$specificity = \frac{TN}{TN + FP} \tag{19}$$

$$precision = \frac{TP}{TP + FP} \tag{20}$$

$$F_1 - score = 2\frac{precision * recall}{precision + recall} \tag{21}$$

$$FPR = \frac{FP}{FP + TN} \tag{22}$$

$$FNR = \frac{FN}{FN + TP} \tag{23}$$

$$FDR = \frac{FP}{FP + TP} \tag{24}$$

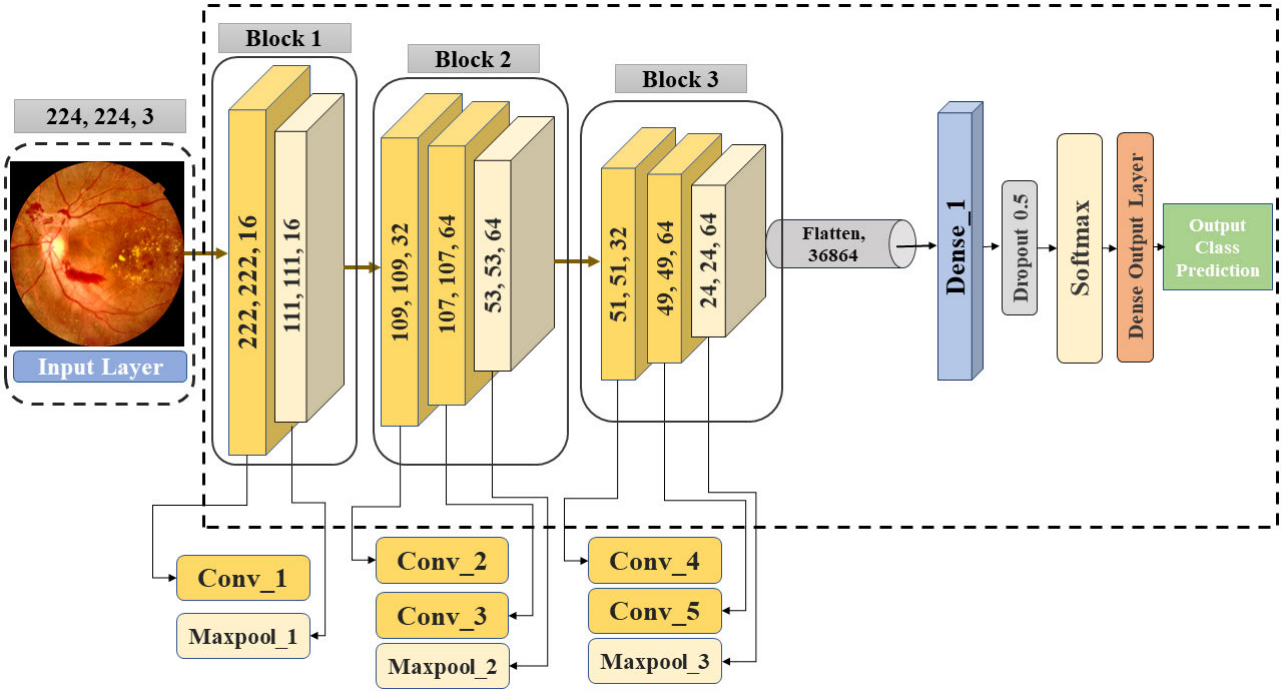

FIGURE 15. RetNet-10 framework after model optimization.

$$NPV = \frac{TN}{TN + FN} \tag{25}$$

$$MAE = \frac{1}{m}\sum_{J=1}^{m}\left|x_j - x_j^p\right| \tag{26}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{J=1}^{m}\left(x_j - x_j^p\right)^2} \tag{27}$$

B. RESULT ANALYSIS OF THE OPTIMAL MODEL

1) PERFORMANCE METRICS OF THE BEST MODEL

After performing ten case studies on our base model, the classification accuracy improved from 89.23% to 98.65% using the optimal optimizer, learning rate, batch size, flatten layer, loss function, activation function, and filter size, changing max pool and convolutional layer, and several filters. In this section, performance metrics are applied to evaluate the models' performance, including specificity, precision, recall, ACC, F1- score, FPR, FNR, FDR, and NPV. From Table 7, it can be seen that our proposed model with optimal configuration achieved a specificity of 99.665%, a precision of 98.650%, a recall of 98.656%, an Acc of 99.463%, an F1-score of 98.653%, an FPR of 0.334, an FNR value of 1.350%, an FDR value of 1.351 and an NPV value of 99.665%. The specificity, precision, recall, ACC, F1-score, and NPV are close to 100%, and the FPR, FNR, and FDR values are also satisfactory. Since all performance metrics are good, it can be stated that our proposed model performs well in classification. Table 7 presents the values of the performance metrics for the optimal proposed model in this study.

Fig 16 represents the confusion matrix, Area under the ROC Curve (AUC), training vs validation accuracy, and training vs validation loss for the optimal model. Fig 16A shows that the training curve converges smoothly from the first to the last epoch, with almost no interruptions. The gap between the validation and training accuracy curves does not show any evidence of overfitting while training. Similarly, the loss curve converges steadily for the training curve, as shown in Fig 16B. Based on the training and loss curves, it can be concluded that there was no indication of overfitting.

Fig 16C represents the confusion matrix of the optimized model.

The row and column values represent the test dataset's actual and the predicted data, respectively, while the diagonal value denotes the TP value. It can be seen that our proposed model is not biased toward any class; instead, it predicts all five disease classes almost equally. In addition, the ROC curve is plotted, and the AUC is found from the ROC curve. The AUC value summarizes the ROC curve representing the model's ability to differentiate between various classes. The model can detect most classes if the AUC value is close to 1. As shown in Fig 16D, the ROC curve is very close to touching the y axis, which indicates the true positive value is close to 1, and the false positive value is close to 0. The AUC value of this study is 98.55%, demonstrating the proposed model's effectiveness.

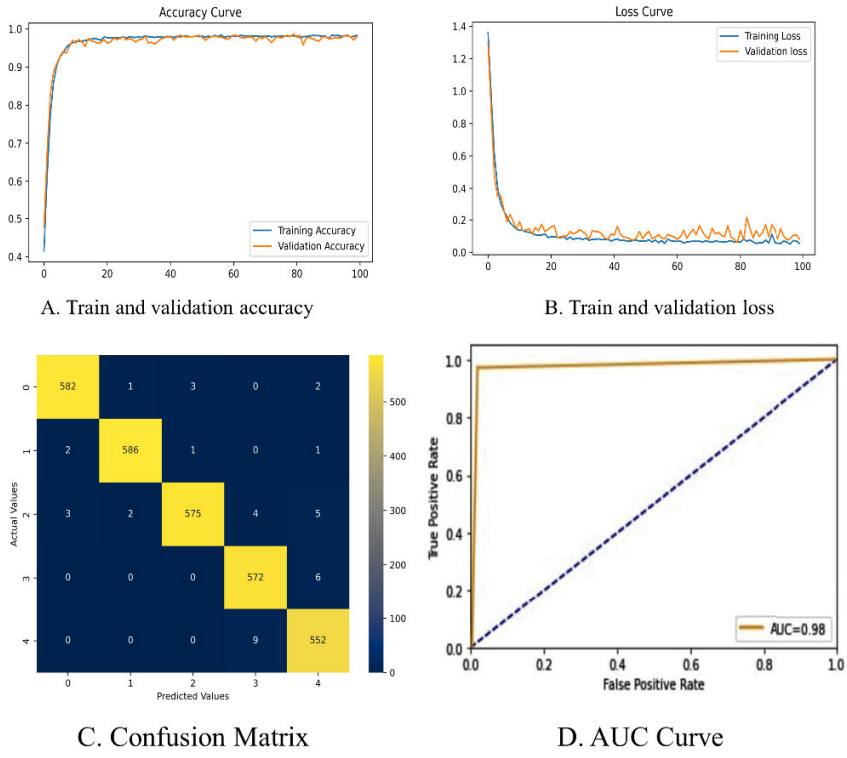

FIGURE 16. Visualization of (A) Accuracy curve (B) Loss curve C) Confusion matrix (D) ROC curve for the optimal result of the proposed RetNet-10 model after model optimization.

TABLE 7. Performance evaluation matrix of the optimal configuration of proposed model.

Performance analysis for the best configuration								
specificity	precision	recall	ACC	F1-score	FPR	FNR	FDR	NPV
99.67%	98.65%	98.66%	99.46%	98.65%	0.33%	1.35%	1.35%	99.67%

C. COMPARISON OF PROPOSED MODEL WITH TRANSFER LEARNING MODEL

In this approach, six pre-trained models named VGG16, VGG19, MobileNetV2, ResNet50, InceptionV3, and Xception are trained and evaluated to observe their performance in terms of accuracy [76], [82], [83]. Table 8 shows a performance comparison of our proposed model and six other CNN-based transfer learning (TL) architectures using the same datasets.

The hyperparameters for all the models are identical, as can be seen in Table 8. The image size is kept at 224 pixels, and we select Adam as the optimizer for all the models. To facilitate the analysis, each model has the same number of epochs: 100. The remaining parameters, learning rate, and batch size for all corresponding models are fixed at the same values, 0.001 and 32, respectively. A performance analysis was carried out to evaluate the robustness of all the models. Each model's input parameters are the same. Table 8 shows the differences in results. MobileNetv2 obtained the best accuracy of 91.42% among all the transfer learning models. The overall results ranged from 87% to 92%, while the proposed RetNet-10 outperformed all the models by achieving a 98.65% test accuracy. It also requires less computational time than all other models. The transfer learning models' execution times range from 151 to 153 seconds on average, which is higher than the proposed model. It can therefore be concluded that the developed RetNet-10 model has the best performance based on accuracy and computation times. Moreover, it can be seen that RetNet-10 has a lower number of layers than the other models, indicating a simple but effective model.

D. EXPERIMENTS WITH DIFFERENT DIMENSIONS AND SPLIT RATIOS

In this study, we have utilized a split ratio of 80:10:10, where we have used 80 percent of the data for the training, 10 percent for the validation and 10 percent for the testing.

TABLE 8. Performance analysis comparison between six traditional transfer learning models and our proposed model.

Model	No. of layers	Epochs	No. of Parameters	Per Epoch time	Optimizer	Batch size	Image size	Learning rate	Test accuracy
VGG16	16	100	138.4 M	151-153s	Adam	32	224	0.001	90.16%
VGG19	19	100	143.7 M	151-153s	Adam	32	224	0.001	88.21%
ResNet50	50	100	25.6 M	151-153s	Adam	32	224	0.001	87.23%
Xception	71	100	22.9 M	151-153s	Adam	32	224	0.001	89.57%
Inception V3	48	100	23.9 M	151-153s	Adam	32	224	0.001	87.68%
MobileNetV2	53	100	3.5 M	151-153s	Adam	32	224	0.001	91.42%
RetNet-10	10	100	39.95 M	140-145s	Adam	32	224	0.001	98.65%

TABLE 9. Comparison of different dimensions with a different split ratio.

Split Ratio (Train: Validation: Test)	Dimensions	Training Accuracy	Validation Accuracy	Test Accuracy	F1 Score	Specificity	Recall	Precision
80:10:10	**512 x 512**	**98.33**	**98.65**	**98.65**	**98.65**	**99.66**	**98.65**	**98.65**
	256 x 256	98.18	98.41	98.63	98.63	99.64	98.63	98.63
	128 x 128	98.1	98.48	98.45	98.45	98.61	98.45	98.45
	64 x 64	98.22	98.48	98.62	98.61	99.65	98.61	98.61
70:10:20	512 x 512	98.03	97.86	97.59	97.6	99.39	97.59	97.6
	256 x 256	98.31	98.24	97.74	97.74	99.43	97.75	97.74
	128 x 128	98.33	97.96	97.66	97.66	99.41	97.65	97.66
	64 x 64	97.96	98.13	97.84	97.85	99.46	97.85	97.85
70:20:10	512 x 512	98.33	97.95	98.1	98.12	99.52	98.1	98.13
	256 x 256	98.14	98.08	98.03	98.03	99.51	98.04	98.03
	128 x 128	97.56	98.07	98.04	98.04	99.51	98.04	98.03
	64 x 64	98.23	98.14	98.03	98.04	99.5	98.03	98.04
60:20:20	512 x 512	98.2	97.41	97.69	97.41	99.34	97.4	97.42
	256 x 256	98.17	97.27	97.33	97.33	99.33	97.34	97.32
	128 x 128	98.09	98.15	97.43	97.44	99.35	97.44	97.44
	64 x 64	98.18	97.41	97.5	97.5	99.37	97.51	97.5

Utilizing a large proportion of images for training enables our model to be trained with a large amount of data, resulting in high accuracy. To maintain uniformity, we have utilized 512 × 512 image dimensions and fed the images to our model. We have experimented with other split ratios and image dimensions to investigate their effect. Table 9 lists the

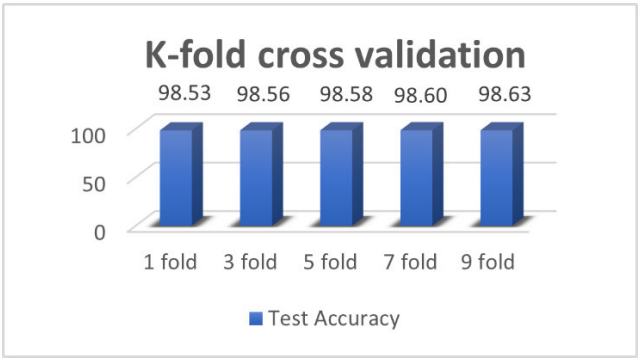

FIGURE 17. K-fold cross validation result.

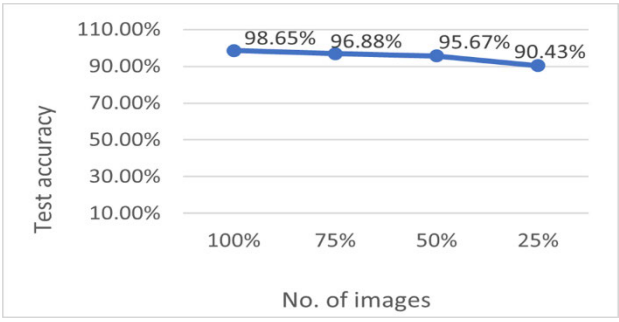

FIGURE 18. Accuracy after reduction of images.

results of these experiments. Table 9 shows that our proposed split ratio and dimensions result in the highest test accuracy of 98.65%. Other performance metrics are also better compared to different configurations. However, it can be seen that for all dimensions and split ratios, the accuracy is above 97%, demonstrating that different dimensions and split ratios only have a minor impact on the accuracy, and it indicates the robustness of our model that our model can attain sublime accuracy regarding the dimensions and split ratio.

E. K-FOLD CROSS VALIDATION AND IMAGE REDUCTION

To assess the strength of the proposed model, two experiments, K-fold cross-validation and image reduction, are conducted. These tests are briefly described below.

1) K-FOLD CROSS VALIDATION

K-Fold cross-validation is a validation test that is carried out using the training and test data set [84]. Initially, the data set is split into multiple k-folds. Subsequently, k iterations of training and validation, each with a separate series of data for training and validation, are carried out [85]. This technique allows observation of the impact of variability, bias and randomness. Bias is indicated by a difference between the real and the predicted accuracy [86]. It is employed for evaluating our models' robustness, reliability, and stability. K- Fold cross-validation is conducted with 1-fold, 3-fold, 5-fold, 7-fold, and 9-fold values, acquiring 98.53%, 98.56%, 98.58%, 98.60%, and 98.63% testing accuracy, respectively. Our best-proposed model attained 98.65%, the highest testing accuracy in classification. After comparing all accuracies, it could be determined that all accuracies were close to the highest accuracy of the optimal proposed model, and the testing accuracy did not drop significantly for any fold. We can therefore expect that our proposed model will acquire high accuracy even in a distinct training scenario with this same dataset. Fig 17 illustrates the test accuracy of several independent K -folds in cross-validation.

2) REDUCTION OF NUMBER OF THE IMAGES

In this section, the number of input images is gradually reduced to evaluate the performance consistency of the proposed RetNet-10 model. For each step, the image number of the data set is reduced to about a fourth of the previous image number. Fig (18) shows the results of reducing the number of images. Since after employing augmentation, our merged dataset consists of 29,279 images, 75%, 50%, and 25%, the dataset consists of respectively, 21,960, 14639, and 7,320 images.

From Fig 18, it can be seen that after training with 100% images (29,279 images), our model achieved 98.65% accuracy. After training with 75% of images (21,960 images), the accuracy dropped by only 2%. With 50% of the images (14639 images), the accuracy dropped by a further 1%, and the RetNet-10 model still achieved a test accuracy of 95.67%. After training with only 25% of the images (7320 images), the model acquired 90.43% test accuracy.

Even using a small number of images (7320 images), the proposed RetNet-10 model can provide a good result demonstrating consistency in the model's performance. Additionally, we can state that the model can efficiently work with 50% of the images without a major loss of test accuracy.

F. COMPARISON WITH EXISTING WORKS

Table 10 provides an overview of the comparison between our proposed model and the existing related work. The proposed RetNet-10 model was compared to some recent studies in diabetic retinopathy classification. Table 10 compares these previous studies and our proposed methodology based on accuracy.

As previously stated, the suggested RetNet-10 model was trained on the merged dataset and reached a test accuracy of 98.65%. We performed a multi-class classification of fundus images, which contains five classes with 5,819 images, before data augmentation. Table 10 shows a comparison of our method with other studies and an overview of their limitations. A number of researchers [16], [18], [19], [20] performed a binary classification utilizing a single public fundus image dataset, obtaining classification accuracies ranging from 96% to 99.89%. Other researchers [16], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], employed single datasets for multi-class classification, resulting in classification accuracies in the range of 80% to 99.59%. In both cases, the main limitations were utilizing a limited number of

TABLE 10. Accuracy comparison with existing literature.

Paper	Name of the dataset	Model	Classification Types	Accuracy	Limitations
Gayathri et al. [16]	1.IDRiD 2.Messidor 3.DR's Kaggle dataset	J48 Classifier (ML model)	Binary no DR, DR Multiclass no DR, mild, moderate, severe, and PDR	99.89% (Binary classification), 99.59% (multi-class classification)	1. No experimentation with a combined dataset
Gharaibeh et al. [18]	1. Image-Ret	SVM + Naïve Bayes	Binary	98.60%	1. Dataset has a lack of fundus images 2. Data augmentation is missing 3. Model fine-tuning is missing.
Kaushik et al. [19]	1.EyePACS	Stacked CNN	Binary (No DR and Having DR) Multiclass no DR, mild, moderate, severe, and PDR	97.92% (Binary classification), 87.45% (Multi-class classification)	1. Lack of image enhancement techniques 2. Noise removal is not conducted 3. Model fine-tuning is missing. 4. No experimentation with a combined dataset
Yaqoob et al. [20]	1.Messidor-2 (Two grades) 2.EyePACS (Five grades)	CNN	Binary No Referable Diabetic Macular Edema Grade (DME) and Referable DME Multiclass no DR, mild, moderate, severe, and PDR	96% (Messidor2) 75.09% (EyePACS)	1. Lack of image enhancement techniques 2. Noise removal is not conducted 3. Model fine-tuning is missing. 4. No Experimentation with a combined dataset 5. No data augmentation
Gen et al. [21]	Kaggle diabetic Retinopathy 1.Original fundus image dataset 2.Entropy image dataset	Proposed CNN	1)Multiclass no DR, mild NPDR, moderate NPDR, severe NPDR, and PDR	81.80% (Original fundus image dataset) 86.10% (Entropy image dataset)	1. Lack of image enhancement techniques 2. Noise removal is not conducted 3. Model fine-tuning is missing. 4. No experimentation with various TL models 5. No experimentation with a combined dataset
Mohamed et al. [22]	1.APTOS 2019	Proposed CNN	1. Multiclass no DR, moderate DR (mild + moderate NPDR), and severe DR severe NPDR+ PDR)	88%-89%	1. Lack of image enhancement techniques 2. Noise removal is not conducted 3. Data augmentation is missing 5. Model fine-tuning is missing. 6. No Experimentation with a combined dataset 7. Experiment with various TL models is absent
Shankar et al. [23]	1.MESSIDOR	Hyperparameter Tuning Inception-v4 (HPTI-v4)	Multiclass Normal, Stage1 (mild NPDR), Stage2 (Moderate +Severe NPDR), Stage3 (PDR)	99.49%	1. No experimentation with a combined dataset 2. Data augmentation is missing
M. Al hazaimeh et al. [24]	DR's Kaggle dataset	DCNN + SVMGA	Multiclass	98.80%	1. Dataset has a lack of fundus images 2. Data augmentation is missing 3. Model fine-tuning is missing.
Wejdan et al. [25]	1.DDR 2.APTOS Kaggle 2019 dataset	First scenario: CNN512 (Proposed CNN) Second scenario: An adopted YOLOv3 model	1)Multiclass no-DR, mild, moderate, severe and PDR	First scenario: 88.6% (DDR dataset) 84.1% (APTOS dataset) Second scenario: 89%	1.Proper image processing method is absent (need to explore more) 2.Experiment with various TL models is absent 3. Model fine-tuning is missing. 4. No experimentation with a combined dataset
P. Saranya et al. [26]	1.MESSIDOR 2. IDRiD	Proposed CNN	1)Multiclass No DR, Mild DR, Moderate DR, and Severe DR	90.89% (MESSIDOR dataset)	1.Lack of image enhancement techniques 2. Model fine-tuning is missing. 3. No experimentation with a combined dataset 4. Experimentation with various TL models is absent
Gharaibeh et al. [27]	1.DARETDB1	SVMGA (SVM and Genetic Algorithm)	Multiclass	98.4%	1. Dataset has a lack of fundus images 2. Data augmentation is missing 3 Model fine-tuning is missing. 4. Experimentation with various TL models is absent

TABLE 10. *(Continued.)* **Accuracy comparison with existing literature.**

K. Shankar et al. [28]	1.MESSIDOR	Synergic Deep Learning (SDL)	Multiclass Normal, Stage1 (mild NPDR), Stage2 (Moderate +Severe NPDR), Stage3 (PDR)	99.28%	1. Experiment with combine dataset is absent
Sehrish et al. [29]	DR's Kaggle dataset	Ensemble Classifier	Multiclass Normal, mild, moderate, severe and PDR	80.80%	1. Lack of image enhancement techniques 2. Noise removal is not conducted 3. Experiment with combine dataset is absent 4. Experimentation with various TL models is absent
Wu et al. [30]	1.IDRiD 2. DR's Kaggle dataset	CF-DRNet	Multiclass No DR, Mild, Moderate, Severe, and PDR	83.10% (Kaggle dataset) 56.19% (IDRiD dataset)	1. Experiment with combine dataset is absent 2. Image processing is lacking. 3. Model fine-tuning is missing.
Rubina et al. [31]	1.Messidor 2.Messidor-2 3.DRISHTI-GS 4.Retina dataset	Fine-tuned VGG16, and Inception V3 model	1.Multiclass (Normal, DR, DME, Glaucoma, and Cataract) 2. Mild multi-class (Normal, mild DR, mild DME, and mild Glaucoma)	88.3 % (multi-class) 85.95% (Mild multi-class)	1. Lack of image-enhancement techniques 2. Data augmentation is missing 4. Experimentation with various TL models is absent 5. Model fine-tuning is missing.
RetNet-10 (Our proposed work)	**Merge Dataset**	**Shallow CNN**	**Multiclass (No DR, mild NPDR, moderate NPDR, severe NPDR and PDR)**	**98.65%**	1. **The number of raw images in grade 3 and 4 are limited.** 2. Pixelwise image preprocessing techniques and markers segmentation are missing. 3. Progression of the disease not included.

images, a lack of image enhancement, model fine-tuning, data augmentation or comparison with other deep learning models. While some researchers [29], [30], [31] experimented with an extensive dataset for multi-class classification, their proposed model could not achieve accuracies as high as the previous studies. In this case, the core limitations were the lack of model fine-tuning and data augmentation. Achieving a good accuracy for a dataset with different quality images for multi-class classification is always challenging, so complex methods are needed to achieve optimal results. Our study aims to address these research gaps.

G. CONTRIBUTION OF THE STUDY

Research on early diagnosis of DR utilizing a computer-assisted system is a highly demanding research subject due to the extreme endangerment of DR. However, to the best of our knowledge, all the state-of-the-art work only focuses on a single dataset, and none merge multiple public datasets class wise. Working with a single dataset is comparatively less challenging. Although, if a model is tested only on a single dataset and is not validated on the merged dataset, that model has a significant chance of giving the wrong prediction on real-time data due to not seeing enough raw images while testing on a single dataset. In our study, we have addressed this challenge by merging three publicly available benchmark DR datasets class-wise. We worked with a wide range of DR images where images from Grade 3 and Grade 4 increased that contain images with severe DR. By working with a wide range of DR images, our model trained with different types of images with other resolutions and complexity, that makes our model robust for working with real-time data as well. In this study, we have proposed an optimal image preprocessing strategy to eradicate noises from fundus images and enhance the image. We have offered an optimal way to perform augmentation for the data balance process. A brief model optimization strategy is adopted to find a robust, lightweight, yet intuitive model that performs better than six traditional transfer learning models. Therefore, the significant contributions discussed in this study can extensively impact and benefit researchers and aid clinicians in diagnosing DR early.

VI. CONCLUSION

This study resolves a multiple-class classification issue for DR with good accuracy. A shallow CNN model framework is proposed, and the time complexity and training time are considered. Three different datasets are used with different resolutions and quality of images, as these images were collected from different sources. Since image qualities were diverse and different artifacts and noises were present in these images, further processing was challenging. A new, high-quality image processing technology has been introduced. Since these three-fundus datasets have insufficient images in some classes, working with this imbalanced dataset was another challenge for this study. Finding an effective method

to balance the dataset was crucial. We, therefore, applied different data augmentation methods and built a base model to evaluate these augmentation methods. After identifying the optimal augmentation technique, ten optimization studies are performed to determine the best configuration and improve the model's classification accuracy. The RetNet-10 model achieved the highest accuracy at the 80th epoch which helps to reduce the time complexity. Our study shows that our model, created after fine-tuning, has lightweight characteristics and achieves excellent results with a test accuracy of 98.65% while working with a wide range of images. The K-Fold cross-validation and the Loss and Accuracy curves of validation and training demonstrate that our proposed model (RetNet-10) has no overfitting and underfitting issues. A comparison with different TL models illustrates the performance stability of our proposed model with a lower convergence time than traditional TL models and a better accuracy. Experiments with different split ratios, dimensions, and image reduction show that our model can perform in a different scenario and with various case studies with optimal accuracies, which indicates the robustness of RetNet-10. Experiments with different split ratios, image dimensions, and a reduced number of images are further indications of the robustness of RetNet-10. Our methodology has been compared with 15 state-of-the-art studies and could handle a wide range of images, utilizing a novel preprocessing strategy. Finding optimal augmentation techniques and fine-tuning a base CNN model resulted in a lightweight yet intuitive model, contributing to the DR classification challenge.

On the basis of this study, for DR multiclass classification, applying optimal image preprocessing, augmentation, and model-building methods, including model optimization, are essential. As the fundus image dataset contains complex characteristics, the RetNet-10 model architecture can be an appropriate approach since it appears to extract significant hidden attributes from the images, resulting in high classification accuracy.

VII. LIMITATIONS AND FUTURE RESEARCH

The proposed CNN (RetNet-10) model performed significantly better than other traditional classifiers for multiclass classification. However, there are limitations which can be addressed in future work. For instance, after combining the three datasets, our final dataset consists of 5,819 images with different characteristics. However, the number of raw images for grades three and four is still limited. In the future, we can add more images to overcome this issue. In addition, although our image preprocessing techniques perform well for this dataset, different image processing techniques to manage noisy input images, including segmentation of the different features of the retinal fundus images, can further be explored. How our proposed model would perform on real-time data could also be evaluated. Additionally, to understand the DR's progression, we may explore geometrical deep learning and graph neural networks.

REFERENCES

[1] N. H. Cho, J. E. Shaw, S. Karuranga, Y. Huang, J. D. da Rocha Fernandes, A. W. Ohlrogge, and B. I. D. F. Malanda, "IDF diabetes atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045," *Diabetes Res. Clin. Pract.*, vol. 138, pp. 271–281, Apr. 2018.

[2] Diabetes. *Eye Complications*. Accessed: Jan. 15, 2023. [Online]. Available: https://diabetes.org/diabetes/complications/eye-complications

[3] A. M. Ashir, S. Ibrahim, M. Abdulghani, A. A. Ibrahim, and M. S. Anwar, "Diabetic retinopathy detection using local extrema quantized Haralick features with long short-term memory network," *Int. J. Biomed. Imag.*, vol. 2021, pp. 1–12, Apr. 2021.

[4] K. Boyd, "Diabetic retinopathy: Causes, symptoms, treatment," Amer. Acad. Ophthalmol., Oct. 2022. Accessed: Apr. 30, 2023. [Online]. Available: https://www.aao.org/eye-health/diseases/what-is-diabetic-retinopathy

[5] S. H. M. Alipour, H. Rabbani, and M. R. Akhlaghi, "Diabetic retinopathy grading by digital curvelet transform," *Comput. Math. Methods Med.*, vol. 2012, pp. 1–11, Sep. 2012.

[6] U. R. Acharya, C. M. Lim, E. Y. K. Ng, C. Chee, and T. Tamura, "Computer-based detection of diabetes retinopathy stages using digital fundus images," *Proc. Inst. Mech. Eng., H, J. Eng. Med.*, vol. 223, no. 5, pp. 545–553, Jul. 2009.

[7] S. Onal and H. Dabil-Karacal, "Improved automated vessel segmentation for diagnosing eye diseases using fundus images," *J. Biomed. Graph. Comput.*, vol. 6, no. 1, pp. 23–33, Nov. 2015.

[8] I. Kaur and L. M. Singh, "A method of disease detection and segmentation of retinal blood vessels using fuzzy C-means and neutrosophic approach," *Imperial J. Interdiscipl. Res.*, vol. 2, no. 6, pp. 551–557, 2016.

[9] W. L. Alyoubi, W. M. Shalash, and M. F. Abulkhair, "Diabetic retinopathy detection through deep learning techniques: A review," *Informat. Med. Unlocked*, vol. 20, Jan. 2020, Art. no. 100377.

[10] W. R. Memon, B. Lal, and A. A. Sahto, "Diabetic retinopathy;: Frequency at level of HbA1C greater than 6.5%," *Prof. Med. J.*, vol. 24, no. 2, pp. 234–238, 2017.

[11] S. Haneda and H. Yamashita, "International clinical diabetic retinopathy disease severity scale," *Nihon Rinsho Jpn. J. Clin. Med.*, vol. 68, pp. 228–235, Nov. 2010.

[12] "Diabetic macular edema-personalizing treatment," Amer. Acad. Ophthalmol., May 2016. Accessed: Apr. 30, 2023. [Online]. Available: https://www.aao.org/eyenet/article/diabetic-macular-edema-personalizing-treatment

[13] R. R. A. Bourne, G. A. Stevens, R. A. White, J. L. Smith, S. R. Flaxman, H. Price, J. B. Jonas, J. Keeffe, J. Leasher, K. Naidoo, K. Pesudovs, S. Resnikoff, and H. R. Taylor, "Causes of vision loss worldwide, 1990–2010: A systematic analysis," *Lancet Global Health*, vol. 1, no. 6, pp. e339–e349, 2013.

[14] D. Atlas, *International Diabetes Federation*, vol. 33, 7th ed. Brussels, Belgium: International Diabetes Federation, IDF Diabetes Atlas, 2015.

[15] S. Jan, I. Ahmad, S. Karim, Z. Hussain, M. Rehman, and M. A. Shah, "Status of diabetic retinopathy and its presentation patterns in diabetics at ophthalmogy clinics," *J. Postgraduate Med. Inst.*, vol. 32, no. 1, pp. 1–4, 2018.

[16] G. S., V. P. Gopi, and P. Palanisamy, "A lightweight CNN for diabetic retinopathy classification from fundus images," *Biomed. Signal Process. Control*, vol. 62, Sep. 2020, Art. no. 102115.

[17] G. S. Scotland, P. McNamee, A. D. Fleming, K. A. Goatman, S. Philip, G. J. Prescott, P. F. Sharp, G. J. Williams, W. Wykes, G. P. Leese, and J. A. Olson, "Costs and consequences of automated algorithms versus manual grading for the detection of referable diabetic retinopathy," *Brit. J. Ophthalmol.*, vol. 94, no. 6, pp. 712–719, Jun. 2010.

[18] N. Gharaibeh, O. M. Al-Hazaimeh, A. Abu-Ein, and K. M. O. Nahar, "A hybrid SVM naïve-Bayes classifier for bright lesions recognition in eye fundus images," *Int. J. Electr. Eng. Informat.*, vol. 13, no. 3, pp. 530–545, Sep. 2021.

[19] H. Kaushik, D. Singh, M. Kaur, H. Alshazly, A. Zaguia, and H. Hamam, "Diabetic retinopathy diagnosis from fundus images using stacked generalization of deep models," *IEEE Access*, vol. 9, pp. 108276–108292, 2021.

[20] M. K. Yaqoob, S. F. Ali, M. Bilal, M. S. Hanif, and U. M. Al-Saggaf, "ResNet based deep features and random forest classifier for diabetic retinopathy detection," *Sensors*, vol. 21, no. 11, p. 3883, Jun. 2021.

[21] G.-M. Lin, M.-J. Chen, C.-H. Yeh, Y.-Y. Lin, H.-Y. Kuo, M.-H. Lin, M.-C. Chen, S. D. Lin, Y. Gao, A. Ran, and C. Y. Cheung, "Transforming retinal photographs to entropy images in deep learning to improve automated detection for diabetic retinopathy," *J. Ophthalmol.*, vol. 2018, pp. 1–6, Sep. 2018.

[22] M. Shaban, Z. Ogur, A. Mahmoud, A. Switala, A. Shalaby, H. A. Khalifeh, M. Ghazal, L. Fraiwan, G. Giridharan, H. Sandhu, and A. S. El-Baz, "A convolutional neural network for the screening and staging of diabetic retinopathy," *PLoS ONE*, vol. 15, no. 6, Jun. 2020, Art. no. e0233514.

[23] K. Shankar, Y. Zhang, Y. Liu, L. Wu, and C.-H. Chen, "Hyperparameter tuning deep learning for diabetic retinopathy fundus image classification," *IEEE Access*, vol. 8, pp. 118164–118173, 2020.

[24] O. M. Al-Hazaimeh, A. Abu-Ein, N. Tahat, M. Al-Smadi, and M. Al-Nawashi, "Combining artificial intelligence and image processing for diagnosing diabetic retinopathy in retinal fundus images," *Int. J. Online Biomed. Eng.*, vol. 18, no. 13, pp. 131–151, Oct. 2022.

[25] W. L. Alyoubi, M. F. Abulkhair, and W. M. Shalash, "Diabetic retinopathy fundus image classification and lesions localization system using deep learning," *Sensors*, vol. 21, no. 11, p. 3704, May 2021.

[26] P. Saranya and S. Prabakaran, "Automatic detection of non-proliferative diabetic retinopathy in retinal fundus images using convolution neural network," *J. Ambient Intell. Humanized Comput.*, no. 0123456789, 2020, doi: 10.1007/s12652-020-02518-6.

[27] N. Gharaibeh, O. M. Al-Hazaimeh, B. Al-Naami, and K. M. Nahar, "An effective image processing method for detection of diabetic retinopathy diseases from retinal fundus images," *Int. J. Signal Imag. Syst. Eng.*, vol. 11, pp. 206–216, Jan. 2018.

[28] K. Shankar, A. R. W. Sait, D. Gupta, S. K. Lakshmanaprabu, A. Khanna, and H. M. Pandey, "Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model," *Pattern Recognit. Lett.*, vol. 133, pp. 210–216, May 2020.

[29] S. Qummar, F. G. Khan, S. Shah, A. Khan, S. Shamshirband, Z. U. Rehman, I. A. Khan, and W. Jadoon, "A deep learning ensemble approach for diabetic retinopathy detection," *IEEE Access*, vol. 7, pp. 150530–150539, 2019.

[30] Z. Wu, G. Shi, Y. Chen, F. Shi, X. Chen, G. Coatrieux, J. Yang, L. Luo, and S. Li, "Coarse-to-fine classification for diabetic retinopathy grading using convolutional neural network," *Artif. Intell. Med.*, vol. 108, Aug. 2020, Art. no. 101936.

[31] R. Sarki, K. Ahmed, H. Wang, and Y. Zhang, "Automated detection of mild and multi-class diabetic eye diseases using deep learning," *Health Inf. Sci. Syst.*, vol. 8, no. 1, p. 32, Dec. 2020.

[32] Kaggle. *APTOS 2019 Blindness Detection*. Accessed: Feb. 20, 2023. [Online]. Available: https://www.kaggle.com/competitions/aptos2019-blindness-detection/data

[33] E. Decencière et al., "Feedback on a publicly distributed image database: The Messidor database," *Image Anal. Stereol.*, vol. 33, no. 3, p. 231, 2014.

[34] P. Porwal, S. Pachade, R. Kamble, M. Kokare, G. Deshmukh, V. Sahasrabuddhe, and F. Meriaudeau, "Indian diabetic retinopathy image dataset (IDRiD): A database for diabetic retinopathy screening research," *Data*, vol. 3, no. 3, p. 25, 2018.

[35] S. P. Praveen, P. N. Srinivasu, J. Shafi, M. Wozniak, and M. F. Ijaz, "ResNet-32 and FastAI for diagnoses of ductal carcinoma from 2D tissue slides," *Sci. Rep.*, vol. 12, no. 1, pp. 1–16, Dec. 2022.

[36] R. Taylor and D. Batey, *Handbook of Retinal Screening in Diabetes: Diagnosis and Management*. Hoboken, NJ, USA: Wiley, 2012.

[37] M. García, C. I. Sánchez, J. Poza, M. I. López, and R. Hornero, "Detection of hard exudates in retinal images using a radial basis function classifier," *Ann. Biomed. Eng.*, vol. 37, no. 7, pp. 1448–1463, Jul. 2009.

[38] Early Treatment Diabetic Retinopathy Study Research Group, "Grading diabetic retinopathy from stereoscopic color fundus photographs—An extension of the modified airlie house classification: ETDRS report number 10," *Ophthalmology*, vol. 98, no. 5, pp. 786–806, 1991.

[39] P. H. Scanlon, A. Sallam, and P. Van Wijngaarden, *A Practical Manual of Diabetic Retinopathy Management*. Hoboken, NJ, USA: Wiley, 2017.

[40] F. Bandello, M. A. Zarbin, R. Lattanzio, and I. Zucchiatti, Eds., *Clinical Strategies in the Management of Diabetic Retinopathy: A Step-by-Step Guide for Ophthalmologists*, 2014th ed. New York, NY, USA: Springer, 2014.

[41] L. Janice, "How to treat diabetic retinopathy," wikiHow, Jan. 2017. Accessed: Apr. 30, 2023. [Online]. Available: https://www.wikihow.com/Treat-Diabetic-Retinopathy

[42] C. P. Wilkinson, F. L. Ferris, R. E. Klein, P. P. Lee, C.-D. Agardh, M. Davis, D. Dills, A. Kampik, R. Pararajasegaram, and J. T. Verdaguer, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales," *Ophthalmology*, vol. 110, no. 9, pp. 1677–1682, Sep. 2003.

[43] Q. Abbas, I. Fondon, A. Sarmiento, S. Jiménez, and P. Alemany, "Automatic recognition of severity level for diagnosis of diabetic retinopathy using deep visual features," *Med. Biol. Eng. Comput.*, vol. 55, no. 11, pp. 1959–1974, Nov. 2017.

[44] M. D. Abrámoff, J. C. Folk, D. P. Han, J. D. Walker, D. F. Williams, S. R. Russell, P. Massin, B. Cochener, P. Gain, L. Tang, M. Lamard, D. C. Moga, G. Quellec, and M. Niemeijer, "Automated analysis of retinal images for detection of referable diabetic retinopathy," *JAMA Ophthalmol.*, vol. 131, no. 3, pp. 351–357, Mar. 2013.

[45] I. Qureshi, J. Ma, and K. Shaheed, "A hybrid proposed fundus image enhancement framework for diabetic retinopathy," *Algorithms*, vol. 12, no. 1, p. 14, Jan. 2019.

[46] L. Xiong, H. Li, and L. Xu, "An enhancement method for color retinal images based on image formation model," *Comput. Methods Programs Biomed.*, vol. 143, pp. 137–150, May 2017.

[47] S. Rasta, M. Partovi, H. Seyedarabi, and A. Javadzadeh, "A comparative study on preprocessing techniques in diabetic retinopathy retinal images: Illumination correction and contrast enhancement," *J. Med. Signals Sensors*, vol. 5, no. 1, p. 40, 2015.

[48] G. Pinedo-Diaz, S. Ortega-Cisneros, E. U. Moya-Sanchez, J. Rivera, P. Mejia-Alvarez, F. J. Rodriguez-Navarrete, and A. Sanchez, "Suitability classification of retinal fundus images for diabetic retinopathy using deep learning," *Electronics*, vol. 11, no. 16, p. 2564, Aug. 2022.

[49] T. Y. Goh, S. N. Basah, H. Yazid, M. J. A. Safar, and F. S. A. Saad, "Performance analysis of image thresholding: Otsu technique," *Measurement*, vol. 114, pp. 298–307, Jan. 2018, doi: 10.1016/j.measurement.2017.09.052.

[50] Y. Ming, H. Li, and X. He, "Contour completion without region segmentation," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3597–3611, Aug. 2016, doi: 10.1109/TIP.2016.2564646.

[51] Sonali, S. Sahu, A. K. Singh, S. P. Ghrera, and M. Elhoseny, "An approach for de-noising and contrast enhancement of retinal fundus image using CLAHE," *Opt. Laser Tech.*, vol. 110, pp. 87–98, Feb. 2019.

[52] J. Lachure, A. V. Deorankar, S. Lachure, S. Gupta, and R. Jadhav, "Diabetic retinopathy using morphological operations and machine learning," in *Proc. IEEE Int. Advance Comput. Conf. (IACC)*, Jun. 2015, pp. 617–622.

[53] G. Sathiya and P. Gayathri, "Automated detection of diabetic retinopathy using GLCM," *Int. J. Appl. Eng. Res.*, vol. 9, no. 22, pp. 7019–7027, 2014.

[54] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2005, pp. 60–65.

[55] F. Shao, Y. Yang, Q. Jiang, G. Jiang, and Y.-S. Ho, "Automated quality assessment of fundus images via analysis of illumination, naturalness and structure," *IEEE Access*, vol. 6, pp. 806–817, 2018.

[56] A. S. Shirokanev, N. Y. Ilyasova, and N. S. Demin, "Analysis of convolutional neural network for fundus image segmentation," *J. Phys., Conf. Ser.*, vol. 1438, no. 1, Jan. 2020, Art. no. 012016.

[57] S. C. Wong, A. Gatt, V. Stamatescu, and M. D. McDonnell, "Understanding data augmentation for classification: When to warp?" in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2016, pp. 1–6.

[58] Y. Nie, A. S. Zamzam, and A. Brandt, "Resampling and data augmentation for short-term PV output prediction based on an imbalanced sky images dataset using convolutional neural networks," *Sol. Energy*, vol. 224, pp. 341–354, Aug. 2021.

[59] Z. Hussain, F. Gimenez, D. Yi, and D. Rubin, "Differential data augmentation techniques for medical imaging classification tasks," in *Proc. AMIA Annu. Symp.* Bethesda, MD, USA: American Medical Informatics Association, 2017, p. 979.

[60] P. Chlap, H. Min, N. Vandenberg, J. Dowling, L. Holloway, and A. Haworth, "A review of medical image data augmentation techniques for deep learning applications," *J. Med. Imag. Radiat. Oncol.*, vol. 65, no. 5, pp. 545–563, Aug. 2021.

[61] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, Jul. 2019.

[62] N. E. Khalifa, M. Loey, and S. Mirjalili, "A comprehensive survey of recent trends in deep learning for digital images augmentation," *Artif. Intell. Rev.*, vol. 55, pp. 1–27, Mar. 2021.

[63] C. Khosla and B. S. Saini, "Enhancing performance of deep learning models with different data augmentation techniques: A survey," in *Proc. Int. Conf. Intell. Eng. Manage. (ICIEM)*, Jun. 2020, pp. 79–85.
[64] E. Castro, J. S. Cardoso, and J. C. Pereira, "Elastic deformations for data augmentation in breast cancer mass detection," in *Proc. IEEE EMBS Int. Conf. Biomed. Health Informat. (BHI)*, Mar. 2018, pp. 230–234.
[65] L. Taylor and G. Nitschke, "Improving deep learning with generic data augmentation," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Nov. 2018, pp. 1542–1547.
[66] X. Wang, K. Wang, and S. Lian, "A survey on face data augmentation for the training of deep neural networks," *Neural Comput. Appl.*, vol. 32, no. 19, pp. 15503–15531, Oct. 2020.
[67] S. Das, K. Kharbanda, M. Suchetha, R. Raman, and D. D. Edwin, "Deep learning architecture based on segmented fundus image features for classification of diabetic retinopathy," *Biomed. Signal Process. Control*, vol. 68, Jul. 2021, Art. no. 102600.
[68] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, pp. 1–74, Mar. 2021.
[69] J. Koushik, "Understanding convolutional neural networks," 2016, *arXiv:1605.09081*.
[70] H. J. Jie and P. Wanda, "RunPool: A dynamic pooling layer for convolution neural network," *Int. J. Comput. Intell. Syst.*, vol. 13, no. 1, pp. 66–76, 2020.
[71] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1207–1216, Feb. 2016.
[72] C. de Vente, L. H. Boulogne, K. V. Venkadesh, C. Sital, N. Lessmann, C. Jacobs, C. I. Sánchez, and B. van Ginneken, "Improving automated COVID-19 grading with convolutional neural networks in computed tomography scans: An ablation study," 2020, *arXiv:2009.09725*.
[73] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5353–5360.
[74] F. Lei, X. Liu, Q. Dai, and B. W.-K. Ling, "Shallow convolutional neural network for image classification," *Social Netw. Appl. Sci.*, vol. 2, no. 1, pp. 1–8, Jan. 2020.
[75] S. Montaha, S. Azam, A. K. M. R. H. Rafid, M. Z. Hasan, A. Karim, K. M. Hasib, S. K. Patel, M. Jonkman, and Z. I. Mannan, "MNet-10: A robust shallow convolutional neural network model performing ablation study on medical images assessing the effectiveness of applying optimal data augmentation technique," *Frontiers Med.*, vol. 9, pp. 1–27, Aug. 2022.
[76] K. Fatema, S. Montaha, M. A. H. Rony, S. Azam, M. Z. Hasan, and M. Jonkman, "A robust framework combining image processing and deep learning hybrid model to classify cardiovascular diseases using a limited number of paper-based complex ECG images," *Biomedicines*, vol. 10, no. 11, p. 2835, Nov. 2022.
[77] I. Kandel and M. Castelli, "The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset," *ICT Exp.*, vol. 6, no. 4, pp. 312–315, Jan. 2020.
[78] Y. You, I. Gitman, and B. Ginsburg, "Large batch training of convolutional networks," 2017, *arXiv:1708.03888*.
[79] D. Masters and C. Luschi, "Revisiting small batch training for deep neural networks," 2018, *arXiv:1804.07612*.
[80] W. QingJie and W. WenBin, "Research on image retrieval using deep convolutional neural network combining L1 regularization and PRelu activation function," *IOP Conf. Ser., Earth Environ. Sci.*, vol. 69, no. 1, Jun. 2017, Art. no. 012156.
[81] A. Tiwari, S. Srivastava, and M. Pant, "Brain tumor segmentation and classification from magnetic resonance images: Review of selected methods from 2014 to 2019," *Pattern Recognit. Lett.*, vol. 131, pp. 244–260, Mar. 2020.
[82] I. U. Khan, S. Azam, S. Montaha, A. A. Mahmud, A. K. M. R. H. Rafid, M. Z. Hasan, and M. Jonkman, "An effective approach to address processing time and computational complexity employing modified CCT for lung disease classification," *Intell. Syst. With Appl.*, vol. 16, Nov. 2022, Art. no. 200147.
[83] S. Montaha, S. Azam, A. K. M. R. H. Rafid, P. Ghosh, M. Hasan, M. Jonkman, and F. D. Boer, "BreastNet18: A high accuracy fine-tuned VGG16 model evaluated using ablation study for diagnosing breast cancer from enhanced mammography images," *Biology*, vol. 10, no. 12, p. 1347, 2021.
[84] F. Y. Ahmed, Y. H. Ali, and S. M. Shamsuddin, "Using K-fold cross validation proposed models for spikeprop learning enhancements," *Int. J. Eng. Technol.*, vol. 7, nos. 4–11, pp. 145–151, 2018.
[85] P. Refaeilzadeh, L. Tang, H. Liu, L. Angeles, and C. D. Scientist, "Cross-validation," *Encyclopedia Database Syst.*, vol. 5, pp. 532–538, Jan. 2020.
[86] H. Tabrizchi, M. M. Javidi, and V. Amirzadeh, "Estimates of residential building energy consumption using a multi-verse optimizer-based support vector machine with k-fold cross-validation," *Evolving Syst.*, vol. 12, no. 3, pp. 755–767, Sep. 2021.

MOHAIMENUL AZAM KHAN RAIAAN is currently pursuing the degree with the Computer Science and Engineering Department, United International University (UIU). He is also a Research Assistant with the Computer Science and Engineering Department, UIU. He is also actively engaged in research activities in computer vision, health informatics, artificial intelligence, graph theory, and mental health modeling.

KANIZ FATEMA received the bachelor's degree in computer science and engineering from Daffodil International University, Dhaka, Bangladesh. She is actively involved in research activities, especially in health informatics, computer vision, machine learning, deep learning, and artificial intelligence-based systems. She has published several research papers in journals (Scopus) and international conferences.

INAM ULLAH KHAN received the bachelor's degree from the Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. He has been recently involved in research activities, especially in the fields of machine learning, deep learning, image processing, computer vision, and artificial intelligence.

SAMI AZAM is currently a leading Researcher and a Senior Lecturer with the Faculty of Science and Technology, Charles Darwin University, Australia. He is also actively involved in the research fields relating to computer vision, signal processing, artificial intelligence, and biomedical engineering. He has number of publications in peer-reviewed journals and international conference proceedings.

MD. RAFI UR RASHID received the Bachelor of Science degree in computer science and engineering from the Bangladesh University of Engineering and Technology (BUET), in 2021. He is currently pursuing the Ph.D. degree in computer science and engineering with The Pennsylvania State University, USA. After graduation, he was a Junior Software Engineer with Reve Systems and a Research Worker with Samsung Research and Development Bangladesh. Before beginning the Ph.D. degree, he was also a Lecturer with the Department of Computer Science and Engineering, United International University, Bangladesh. Currently, he is a Graduate Assistant with the Department of Computer Science and Engineering, The Pennsylvania State University. His research interests include security and privacy of machine learning, with a particular interest in federated learning, natural language processing, adversarial machine learning, and applied machine learning.

MD. SADDAM HOSSAIN MUKTA received the Ph.D. degree from the Data Science and Engineering Research Laboratory (Data Laboratory), BUET, in 2018. He is currently an Associate Professor and an Undergraduate Program Coordinator with the Department of Computer Science and Engineering. He has a number of quality publications in both national and international conferences and journals. His research interests include deep learning, machine learning, data mining, and social computing.

MIRJAM JONKMAN (Member, IEEE) is currently a Lecturer and a Researcher with the Faculty of Science and Technology, Charles Darwin University, Australia. Her research interests include biomedical engineering, signal processing, and the application of computer science to real life problems.

FRISO DE BOER is currently a Professor with the Faculty of Science and Technology, Charles Darwin University, Australia. His research interests include signal processing, biomedical engineering, and mechatronics.