ENGINEER - Vol. LVII, No. 03, pp. [45-59], 2024

DOI: https://doi.org/10.4038/engineer.v57i3.7655

Enhancing Diabetic Retinopathy Detection Accuracy with Convolutional Neural Networks

H.M.L.S. Kumari and C.K. Walgampaya

Abstract: Diabetic Retinopathy (DR) has emerged as a major illness that makes diabetic patients blind worldwide. If this disease is not treated promptly, it can progress to permanent blindness. Having the ability to identify people with DR rapidly will be beneficial for both medical professionals and researchers. In the current situation, manually analysing each minute nerve cell from fundus pictures is extremely time-consuming for clinicians. This paper not only concentrates on diabetic retinopathy detection but also on the analysis of DR progression in the given fundus image, which is performed with the help of data augmentation, Convolutional Neural network and transfer learning algorithms. Deep learning models DenseNet121, DenseNet169, DenseNet201, Resnet50, and InceptionV3 models are used on the Diabetic Retinopathy dataset in Kaggle and APTOS. Five DR stages, which are no DR, mild DR, moderate DR, severe DR and proliferate DR, are handled in this work. The fundus images of the eye are used as input to the model. The most robust model in the evaluations, DenseNet201, achieved an accuracy of approximately 98.56% for the APTOS dataset and 97.14% for the Diabetic Retinopathy dataset in Kaggle. These results were obtained using 5-fold stratified cross-validation and hyperparameter tuning, for detecting retinopathy and assessing its stage.

Keywords: Convolutional Neural Network (CNN), Diabetic Retinopathy (DR), Data Augmentation, Transfer Learning Model

1. Introduction

Diabetes is one of the most prevalent chronic diseases, categorized by level of blood glucose (or blood sugar) [1]. This leads over time to serious damage to the eye, blood vessels, heart, kidneys, and nerves. Diabetic Retinopathy (DR) is a diabetic-related condition that impacts the eyes and, when not addressed, can result in blindness [2]. It occurs due to damage to the blood vessels in the retina, which is the light-sensitive tissue at the back of the eye that is responsible for sending visual signals to the brain. In contrast, DR is related to prolonged high blood sugar levels associated with diabetes. The best way to detect DR early and prevent eye blindness is through regular eye examinations, especially for people with diabetes. These regular eye examinations are often done using fundus images [3]. Fundus images, also known as retinal images, are a common and non-invasive diagnostic tool used to capture detailed images of the retina.

Fig. 1(a) and Fig. 1(b) show the features of the fundus image of a normal eye and the features of the fundus image of a diabetic retinopathy eye, respectively. Fig. 2 shows the fundus images of the eyes in mild state and severe state of diabetic retinopathy.

Deep learning-based CNN has recently been proven to be a prospective methodology for different medical image analysis [4]. CNNs are being used in the classification of DR when provided with a large data sample and significant computational power [5].

Transfer learning has revolutionized computer vision by making it more accessible, efficient, and effective. Transfer learning models in CNN involve leveraging pre-trained neural network architecture and transferring their knowledge to solve different but related tasks.

The pre-trained models, such as DenseNet121, DenseNet201, DenseNet169, InceptionV3, and ResNet50, are being widely used for DR detection [6]. Those pre-trained models are already trained on a large dataset for a specific task, such as image classification. These pre-trained models have learned to recognize various features and patterns within images and can train, modify, or fine-tune to adapt to DR detection.

Miss.H.M.L.S. Kumari, *B. Sc. (Hons) Computer Science, Instructor, Faculty of Engineering, University of Peradeniya.*
Email:lihinisangeetha99@gmail.com
https://orcid.org/0009-0001-1410-7840
Eng. (Prof.) C.K. Walgampaya, *AMIE(SL), B.Sc. Eng. (Hons)(Peradeniya), MSc. (Louisville, USA), PhD (Louisville, USA), Professor, Faculty of Engineering, University of Peradeniya.*
Email:ckw@pdn.ac.lk
https://orcid.org/0000-0003-2192-474X

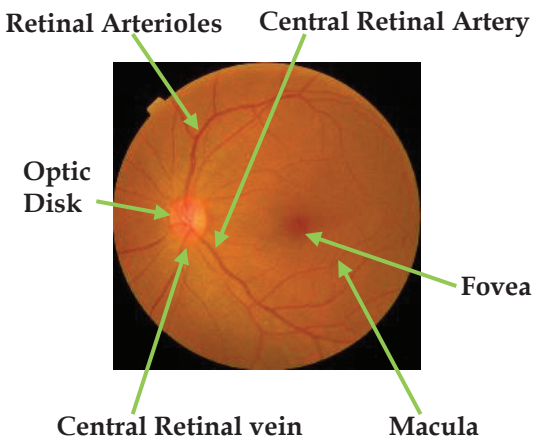

Figure 1 (a). Features of Fundus Image of Normal Eye

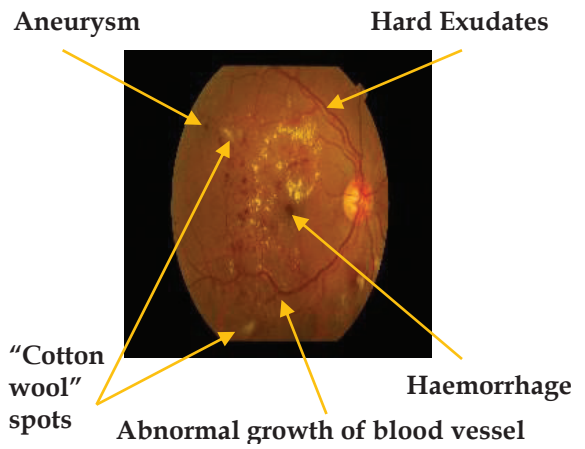

Figure 1 (b). Features of Diabetic Retinopathy in Fundus Image of Eye

CNN with pre-trained models show more accuracy for the massive dataset. However, one cannot easily find a dataset with a large number of images. To improve the sample size, we use data augmentation. In this study, we use the Asia Pacific Tele-Ophthalmology Society (APTOS), which is a publicly available dataset, and the Diabetic Retinopathy dataset that is available on the Kaggle website [5] [7].

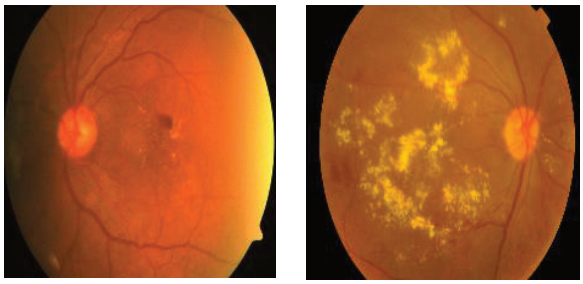

Figure 2 - Fundus Images of the Eyes in Mild and Severe States of Diabetic Retinopathy

2. Related Works

CNN with transfer learning and data augmentation has been extensively studied to detect DR. In most of the studies, small datasets are used with two or four classes. Table 1 illustrates how recent studies have utilized datasets, pre-trained models, and drawn conclusions in the realm of diabetic retinopathy detection.

3. Methodology

3.1 APTOS Dataset

The dataset that is used in this proposed method is the Asia Pacific Tele-Ophthalmology Society 2019 (APTOS-2019) and is taken from the DR Detection Competition on Kaggle (https://www.kaggle.com/c/aptos2019-blindness-detection/data, accessed on 18 June 2024) [5],[7],[20]. This dataset contains 3600 funduscopic images. These samples were collected from many participants in rural India. The Aptos dataset contains fundus images, which give back part of the eye that includes the retina, optic disk, macula, blood vessels, and other important structures [5],[7]. These fundus images were collected in varying conditions and environments over a long period of time. Later, trained doctors labelled those images. As per the scaling system, the dataset is categorized as below:

1. No DR (class 0)
2. Mild DR (class 1)
3. Moderate DR (class 2).
4. Severe DR (class 3)
5. Proliferative DR (class 4)

The number of fundus images in each class APTOS dataset is shown in Table 2. As shown in Table 2, there is a low amount of fundus images in mild DR, severe DR, and proliferate DR. Transfer learning enables the use of models pre-trained on large datasets to effectively train on smaller, task-specific, datasets. Despite the robustness of transfer learning models with limited data, data augmentation remains a crucial technique.

Table 1 - Recent Studies which Utilized Datasets, Pre-Trained Models, and Drawn Conclusions in the Realm of Diabetic Retinopathy Detection

Reference	*Datasets Used*	*Models Considered*	*Conclusions*
Srinivas et al. [9]	APTOS-2019	Densenet121, Densenet169, ResNet50, InceptionV3	Achieved high accuracies (96.64%, 95.95%, 95.71%, 94.73%) using data augmentation (rotation, shearing, flipping, scaling) on grayscale images.
Zhentao Gao et al. [10]	Various	Deep CNN	Achieved 88.72% accuracy for a four-degree classification task with data preprocessing and augmentation.
Rishab Gargeya et al. [11]	MESSIDOR 2, E-Optha, Public datasets	Deep learning model	Achieved 0.93 AUC, 94% sensitivity, and 98% specificity for diabetic retinopathy detection.
Ibrahem Kandel et al. [12]	Messidor, Kaggle, DR1, E-Optha, STARE	Transfer learning with CNNs	Review of transfer learning models for diabetic retinopathy detection.
Masood et al. [13]	Kaggle	InceptionV3	Achieved 48.2% accuracy using cropped images from Kaggle dataset.
Mohammadian et al. [14]	Kaggle	InceptionV3, Xception	Reported 87.12% and 74.49% accuracy for InceptionV3 and Xception, respectively, on a dataset with 35,126 images.
Wang et al. [15]	Kaggle	AlexNet, VGG16, InceptionV3	Achieved accuracies of 37.43%, 50.03%, and 63.23% for AlexNet, VGG16, and InceptionV3, respectively, on Kaggle dataset.
Huang et al. [16]	CIFAR-10, CIFAR-100, SVHN, ImageNet	Dense CNN (DenseNet)	Proposed model achieved high accuracy on object detection benchmarks.
Zahra Mungloo-Dilmohamudet et al. [5]	APTOS, Mauritian	VGG16, ResNet50, DenseNet169	Achieved accuracies of 97.6%, 95.9%, and 91.1% for VGG16, ResNet50, and DenseNet169, respectively, using data augmentation.
Vani Ashok et al. [17]	APTOS	ResNet50	Developed an automatic model for detecting and staging DR with 82% accuracy using ResNet50.
Jeevika Rai et al. [18]	Kaggle	EfficientNet, DenseNet, VGG	Achieved highest accuracy of 97% using DenseNet121 after applying data augmentation.

Table 2 - Number of Fundus Images in Each Class and the Ratio Concerning the Whole Dataset

	Image Count	*Ratio*
No DR	1805	49
Mild DR	370	10
Moderate DR	999	27
Severe DR	193	5
Proliferative DR	295	8
Total	3662	100

Data Augmentation enhances generalization, mitigates overfitting, simulates real-world variability, balances class distributions, and overall improves the performance of the proposed transfer learning model.

Hence, data augmentation is used in this proposed method. The sample of the dataset with five classes is shown in Fig. 4 [20].

3.2 Diabetic Retinopathy Dataset from the Kaggle Website

The dataset of diabetic retinopathy was sourced from the Kaggle website and comprises images classified into five distinct classes [8] [24].

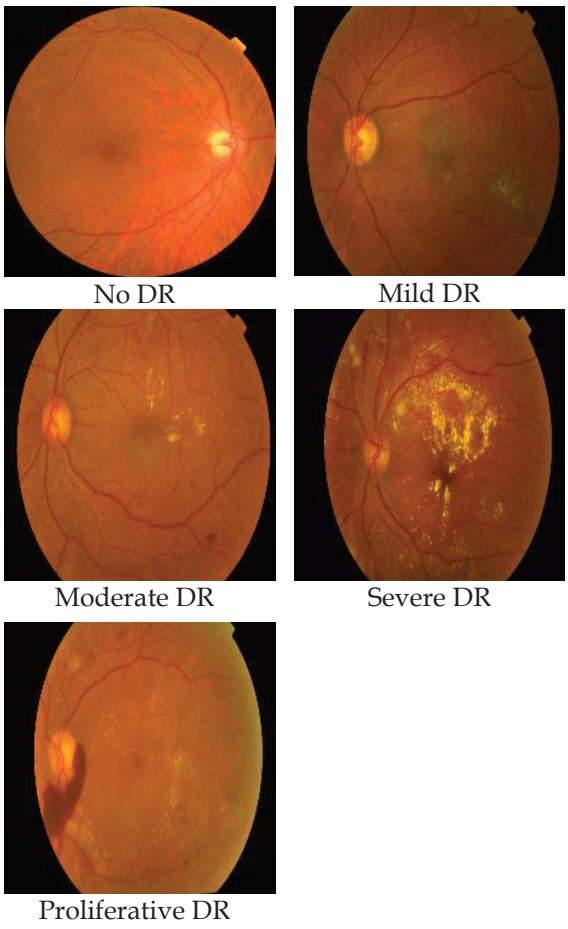

Figure 3 - Fundus Images in Five Classes of the APTOS Dataset

Table 3 illustrates that, even though the data is categorized into five distinct classes, there is significant variability in the number of samples across each class, creating an imbalanced dataset.

As elaborated in previous studies [21, 22], an imbalanced dataset contributes to elevated misclassification rates and suboptimal performance. This occurs because the model tends to be biased towards the majority class, learning its features more effectively while underrepresenting the minority class. Consequently, the model becomes less sensitive to the minority class, leading to higher misclassification rates for those examples.

Table 3 - Number of Fundus Images in Each Class

Train	*Test*	*Validation*
0 -141 images	0 -286 images	0 -110 images
1 -141 images	1 -286 images	1 -110 images
2 -141 images	2 -297 images	2 -121 images
3 -81 images	3 -143 images	3 -66 images
4 -56 images	4 -99 images	4 -33 images
Total=560 Images	Total=1111 images	Total=440 images

3.3 Data Augmentation

The technique of broadening the volume and diversity of data is referred to as data augmentation. Rather than collecting fresh data, we have modified existing data [9], [19]. The image augmentation techniques used were rotation, shear, zoom, flipping the image left and right, flipping the image up and down, gaussian blur, additive gaussian noise, contrast normalization, and so on [9], [18]. By employing these techniques, we get an enhanced number of images within the dataset. After data augmentation, the training dataset consists of three augmented images with their original image.

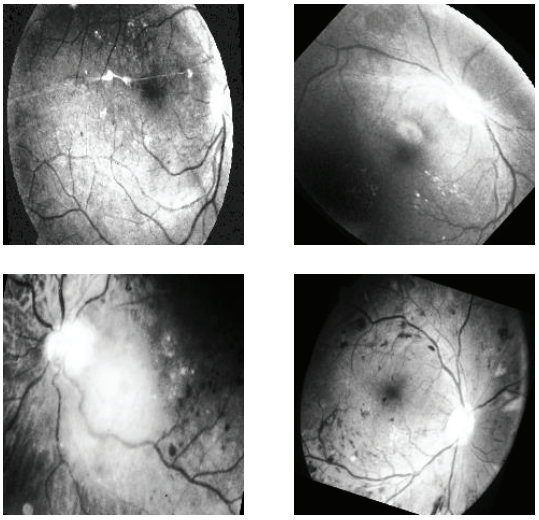

Figure 4 - Sample Augmented Fundus Images from the APTOS Dataset

3.4 Ben Graham Preprocessing and Auto Cropping

The quality of fundus images is very high, but detecting the features of diabetic retinopathy in the eye is not easy. The symptoms that the model used as a feature to classify the images into different classes in DR are:

1. Haemorrhage
2. Hard Exudates
3. Intra-Retinal Microvascular Abnormality (IRMA)
4. Neovascularization
5. Preretinal Haemorrhage

The first problem in the detection of diabetic retinopathy is detecting the actual stage between mild and moderate because there are overlapping symptoms that can be seen in the mild stage and the moderate stage [15], [30]. Haemorrhages can be seen in both mild and moderate stages as shown in Fig. 5. Then we need more accurate images to detect those

stages. In this work, we used different preprocessing techniques for fundus images before feeding them to a model for training [23]. Those features add difficulty to processing and training models. To solve the previously stated problems, we investigated a preprocessing technique inspired by Ben Graham, as employed in the Kaggle competition [7]. Thus, we can remove the dark background of the fundus image. Fig. 6 shows fundus images after Ben Graham preprocessing in 5 classes of the APTOS dataset.

Fig. 7 shows fundus images with Ben Graham preprocessing after auto-cropping in five classes of the APTOS dataset.

Fig.5 depicts the following stages: (a) Stage I: no diabetic retinopathy; (b) Stage II: mild non-proliferative diabetic retinopathy; (c) Stage III: moderate non-proliferative diabetic retinopathy; (d) Stage IV: severe non-proliferative diabetic retinopathy; and (e) Stage V: proliferative diabetic retinopathy [15].

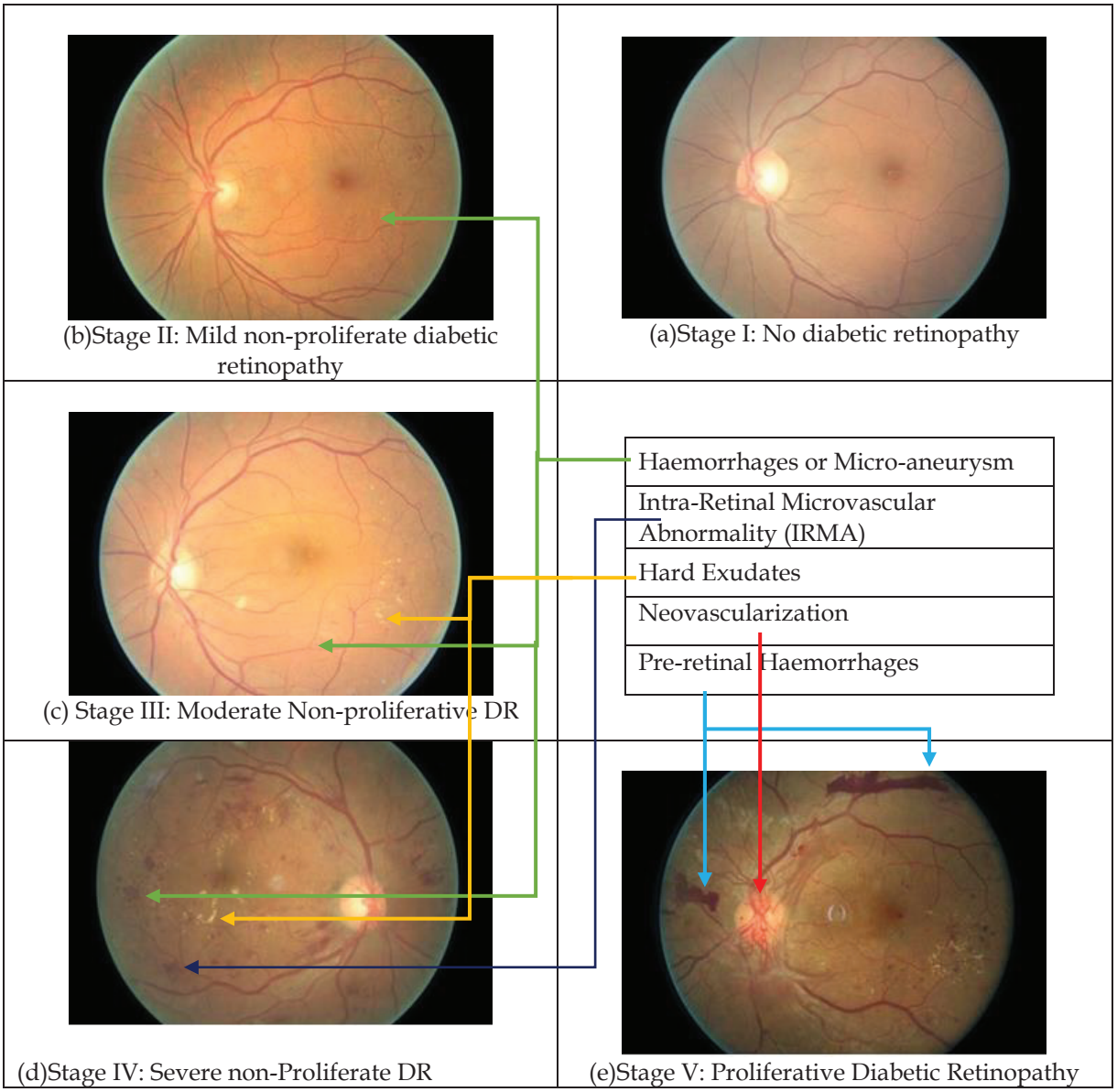

Figure 5 - Fundoscopic Images of Different Stages of Diabetic Retinopathy

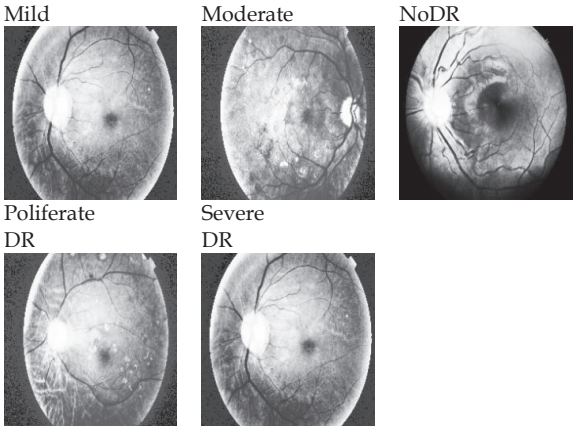

Figure 6 - Fundus Images After Ben Graham Preprocessing in 5 Classes of the APTOS Dataset

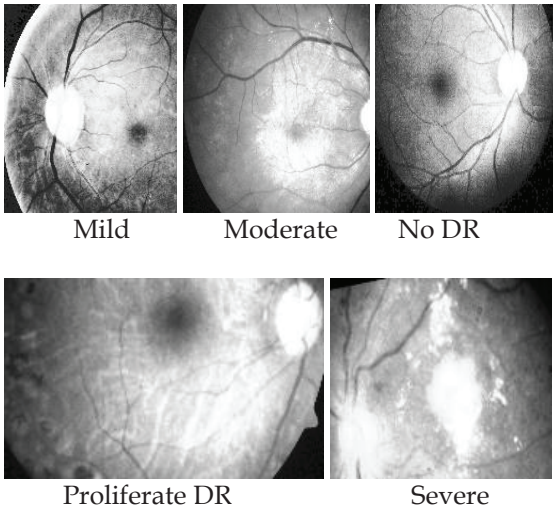

Figure 7 - Fundus Images with Ben Graham Preprocessing after Auto-Cropping in 5 classes of the APTOS dataset

3.5 CNN and Transfer Learning Models

Deep learning is a subset of machine learning that involves neural networks with many layers (often referred to as deep neural networks) to model and understand complex patterns in data. It encompasses a variety of architectures, including but not limited to CNNs, recurrent neural networks (RNNs), long short-term memory networks (LSTMs), and transformer networks. Deep learning architectures with CNN find applications in diverse domains such as pattern recognition, medical image classification, natural language processing, and so on [19]. A neural network consists of an input layer, one or more hidden layers, and an output layer. CNN consists of four main layers, such as the convolutional layer, pooling layer, RELU-connection layer, and fully connected layer. Therefore, CNN is an appropriate technique for the detection of diabetic retinopathy in fundus images and has been used in recent DR studies [11].

1. The most effective method for diabetic retinopathy detection with CNN is using transfer learning models with CNN. The Keras applications are transfer learning models with pre-trained weights. These deep-learning models are employed for such things as prediction, fine-tuning, and feature extraction. There are a large number of Keras models used as transfer learning models. Densenet121, Densenet169, Densenet201, and Inceptionv3 are some of the models that have been used in recent studies in DR detection with CNN [5], [19]. Fine-tuning is a popular transfer learning technique. Because it is the best technique to perform diabetic retinopathy detection on any dataset from a pre-trained CNN. Fine-tuning is done in three ways. Truncate the last layer (i.e., the softmax layer) of the transfer learning model. This increased depth of the transfer learning model makes it more powerful and suitable for tasks that may require more complex feature extraction. Consider the pre-trained network that has been used, and replace its softmax layer with a layer suitable for the number of classes in the dataset. We need to classify 5 classes of diabetic retinopathy detection, but a pre-trained network is used for 1000 or more categories. Therefore, we keep 5 with the softmax layer. Finally, we used the early stopping technique to avoid overfitting the model.
2. Use a small learning rate when training the pre-trained model with CNN on a dataset.
3. Freeze the few layers of the pre-trained network and write new layers that are appropriate for our dataset.

In this study, we use all the methods stated above and different transfer learning models such as Densenet121, Densenet120, Densenet169, Inception V3, and ResNet50.

DenseNet:

As the name implies, DenseNet represents a progressive advancement aimed at enhancing the depth of CNNs [23]. DenseNet requires a few parameters compared to traditional CNNs. Each layer produces a feature map derived from the input it receives from its preceding layers. A transition layer is situated between

different dense blocks, unifying all the feature maps. Dense blocks yield multiple feature maps of the same size, which the transition layer concatenates. Despite the potential increases in input channels due to concatenation, a bottleneck property mitigates the parameter count. Furthermore, an additional compression step within the transition layer further reduces the number of output feature maps [19]. In the proposed model, we have used DenseNet121, DenseNet169, and DenseNet201.

Table 4 - Architecture of DenseNet121, DenseNet169 and DenseNet201

Layers	*Output size*	*Densnet121 VS Densenet169 VS Densenet201*		
Convolution	112 X 112	7X 7 Conv Stride 2		
Pooling	56 X 56	3 X 3 Max Pool, Stride 2		
Dense Block 1	56 X 56	[(1X1 Conv) (3X3 Conv)]X6		
Transition Layer 1	56 X 56	1 X 1 Conv		
	28 X 28	2 X 2 Average Pool, Stride 2		
Dense Block 2	28 X 28	[(1X1 Conv) (3X3 Conv)]X12		
Transition Layer 2	28 X 28	1 X 1 Conv		
	14 X 14	2 X 2 Average Pool, Stride 2		
Dense Block 3	14 X 14	[(1X1 Conv) (3X3 Conv)] X24	[(1X1 Conv) (3X3 Conv)]X 32	[(1X1 Conv) (3X3 Conv)] X48
Transition Layer 3	14 X 14	1 X 1 Conv		
	7 X 7	2 X 2 Average Pool, Stride 2		
Dense Block 4	7 X 7	[(1X1 Conv) (3X3 Conv)] X16	[(1X1 Conv) (3X3 Conv)]X 32	[(1X1 Conv) (3X3 Conv)] X32
Classification Layer	1 X 1	7 X 7 Global Average Pool		
		1000D Fully Connected - softmax		

The pre-processed fundus image moves through different layers of the DenseNet model and finally moves to the softmax layer. DenseNet is already pre-trained for a large number of images and has 1000 classes. Therefore, we need to change the number of classes to 5 in the softmax layer according to the number of classes in the diabetic retinopathy dataset. This step was done for all transfer learning models we used in our proposed study.

DenseNet-121:
DenseNet-121 is a deep architecture with a total of 121 layers, including both convolutional and batch normalization layers. It is known for its effectiveness in image classification and related computer vision tasks and is often used in transfer learning scenarios, where pre-trained versions of the model on large image datasets like ImageNet are fine-tuned for specific applications. Architecture of DenseNet-121 has several layers, such as convolutional layers, pooling layers, dense layers, transition layers, global average layers, softmax layers, etc.

DenseNet-169:
DenseNet-169 is a deeper architecture compared to the original DenseNet, with 169 layers in total, including convolutional and batch normalization layers. This increased depth makes it more powerful and suitable for tasks that may require more complex feature extraction.

DenseNet-201:
DenseNet-201 is a very deep architecture, with a total of 201 layers, including convolutional and batch normalization layers. Its increased depth and complexity make it well-suited for tasks that require sophisticated feature extraction and representation learning.
Similar to other DenseNet variants, DenseNet121, DenseNet169, and DenseNet201 are commonly used in computer vision applications, as are pre-trained versions on large datasets such as ImageNet. Those models can be fine-tuned for specific tasks or used as feature extractors in transfer learning.
Table 4 provides an overview of the architectural structures of DenseNet121, DenseNet169, and DenseNet201. While all layers are uniform across DenseNet121, DenseNet169, and DenseNet201, variations exist in the number of layers within each model.

InceptionV3:
Inception is a CNN designed for object and image processing tasks. InceptionV3 originated as a module for Googlenet. The architecture was trained on a dataset comprising 1000 classes from the original ImageNet dataset, involving over a million training images.
Notably, the TensorFlow version incorporates an extra "background" class, resulting in a total of 1001 classes.
While the computational cost is slightly higher than that of GoogLeNet, InceptionV3 boasts a

depth of 42 layers, rendering it more efficient than efficientNet.
Inception's computational expense is notably more economical than that of VGGNet and its more proficient successors. This affordability has facilitated the utilization of Inception networks in extensive data environments where substantial amounts of data need to be processed cost-effectively. It is also advantageous in scenarios with constrained memory or computational capabilities, such as in mobile vision.

ResNet50:
ResNet50, an influential CNN architecture, is composed of 50 layers structured into residual blocks. Renowned for its utilization of skip connections, ResNet50 effectively solves the vanishing gradient problem. Within its architecture, ResNet50 incorporates 48 convolution layers, along with 1 MaxPool layer and 1 AveragePool layer. In our diabetic retinopathy application, this design choice proves beneficial, as it allows later layers to focus on learning less semantic information, with the earlier layers capturing more intricate details. Spatial convolution is executed using a 3 × 3 filter, which is subsequently downsized through the application of the max-pooling method.

3.6 Proposed DR Detection Model

Fig. 9. shows the proposed DR detection model with CNN and a pre-trained model. The dataset is divided into 70% for training set, 20% for the testing set and 10% for the validation set. The data augmentation is done by synthesizing one image into three images from its original image in the training dataset. The training dataset contains augmented and original images. The data augmentation methods used in this study are explained under the Methodology section.
Ben Graham's preprocessing methods and auto-cropping are used for augmented training images.
The dashed lines of Fig. 9 show the proposed CNN model with a transfer learning approach. The model freezes all layers of the transfer learning model using "layer. trainable=False" but keeps the last 8 layers to detect fine information in the images, such as edges of the images. This is done to avoid the problem of overfitting and to avoid training the entire network. Fig. 8 shows the activated layers of the CNN model in the transfer learning approach.

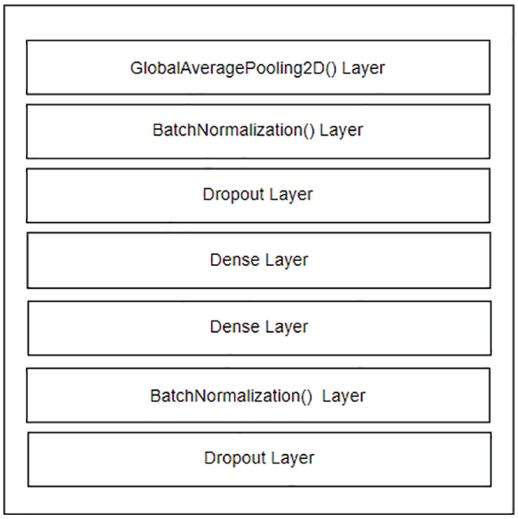

Figure 8 - Activated 8 Layers of Proposed CNN Model

The next step is to feed the resulting training images, validation images and testing images labelled as a list, into the proposed CNN model. The model is trained using 5-fold stratified cross-validation, where the dataset is split into five stratified folds to ensure each fold has a representative distribution of classes. For each fold, the model is trained on the training subset and validated on the validation subset. Early stopping is employed during the training of each fold to halt training when the performance on the validation subset ceases to improve [30]. Early stopping avoids overfitting by stopping training when the model's performance on the validation set starts to degrade and prevents the model from memorizing the training data.

4. Experimental Results and Discussion

The proposed model is trained using hyperparameter tuning by varying batch sizes (32 and 64) for 75 epochs with a 224 x 224 image size. We utilized Google Colab with Python and Keras libraries running on the TensorFlow backend for this study. The Google Colab environment provides access to the NVIDIA Tesla K80 GPU.

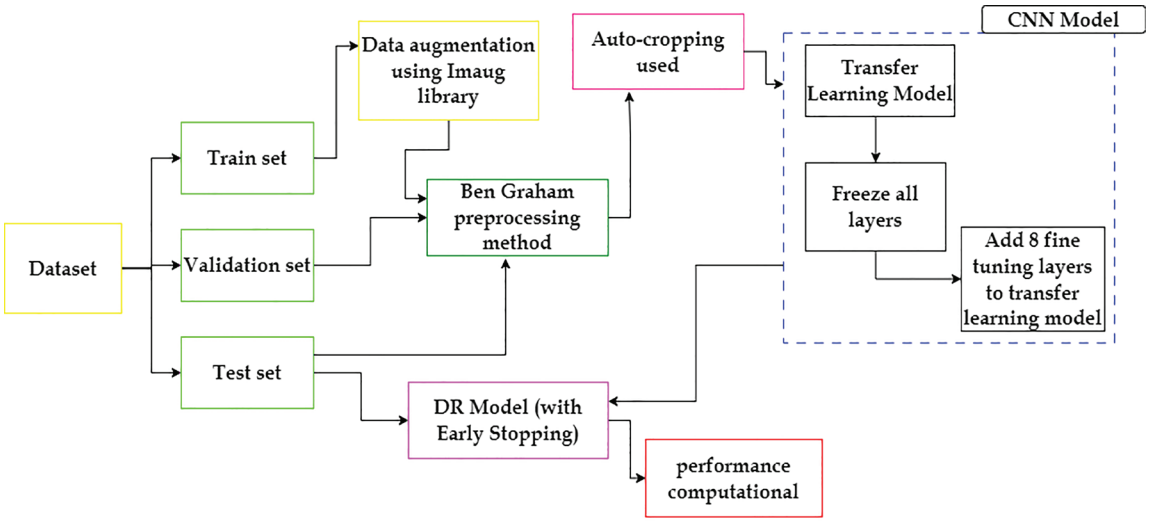

Figure 9 - Proposed DR Model

4.1 Evaluation Metrics

Accuracy, F1 score, and precision are the evaluation metrics used in this study, obtained through cross-validation with hyperparameter tuning.

Accuracy = (TP+TN) /(TP+TF+FP+FN)
Precision= TP /(TP+FP)
Recall= TP /(TP+FP)

F1 Score =2((Precision Recall)/(Precision+ Recall))*

TP=true positive, TN=true negative, FP=false positive, FN=false negative

4.2 Testing Results

The proposed study evaluates accuracy using the APTOS dataset, Diabetic Retinopathy in Kaggle, by using augmentation and preprocessing. A large dataset is advantageous for training CNNs with transfer learning as it can enhance the accuracy on an unseen test dataset. A comparison of validation accuracy is conducted across different scenarios for each transfer learning model employed in the study. When combined with 5-fold stratified cross-validation and hyperparameter tuning, this approach ensures robust model evaluation and optimization, leading to improved generalization and performance across diverse datasets.

1. using split original datasets
2. using the split original datasets with Ben Graham preprocessing
3. using the split original datasets with Ben Graham preprocessing along with the auto-cropping preprocess method.

Recent research has not integrated data augmentation, transfer learning, and fine tuning in the combination to achieve optimal accuracy. All the above models are trained using the early stopping technique.

Table 5 highlights the peak evaluation metrics of the proposed model when employing various architectures (Densenet121, Densenet169, Densenet201, Inception V3, and ResNet50) on the APTOS dataset. The evaluation was conducted using 5-fold stratified cross-validation. The training set comprised 2562 images, while the test set included 736 images. The highest validation accuracies were attained by optimizing hyperparameters, specifically with a dropout rate of 0.5, dense layer sizes of 1024 and 512, batch size of 64, a learning rate of 0.001, and training for 75 epochs across all experiments.

Table 6 illustrates the highest validation evaluation metrics of the proposed model using different architectures (Densenet121, Densenet169, Densenet201, Inception V3, and ResNet50) on the split APTOS dataset after Ben Graham preprocessing with 5-fold stratified cross-validation. The experiments are conducted with hyperparameters including a batch size of 64, dropout layers of 0.5, dense layers of size 1024 and 512, and 75 epochs. The training set consists of 8,068 images, and the test set consists of 736 images. Additionally, the learning rate was set to 0.0001 for all trials.

Table 7 displays the highest validation evaluation metrics of the proposed model using different architectures (Densenet121, Densenet169, Densenet201, Inception V3, and ResNet50) on the split APTOS dataset with Ben Graham preprocessing and auto-cropping, using 5-fold cross-validation. Hyperparameter tuning was used to achieve high accuracy with a batch size of 64, dropout layer of 0.5, dense layers of 1024 and 512 units, and 75 epochs, Utilizing 8068 images in the training set and 736

images in the test set. The learning rate was set to 0.001 for all trials.

DenseNet201 and DenseNet169 show the best validation accuracy among used pre-trained models. The loss and accuracy curves of DenseNet201, which show maximum accuracy of 98.56% are shown in Figs. 10(a), 10(b), respectively. On the loss curve, if the training loss continues to decrease while the validation loss increases, it indicates that the model is likely overfitting [7]. Based on the above figures, it is evident that the models do not exhibit overfitting. It occurs when models perform well in training data by memorizing data patterns, noise, and random fluctuations of the training images but cannot perform well in unseen images in test images. The implementation of data augmentation, early stopping, hyperparameter tuning, and 5-fold stratified cross-validation during the training of the CNN with a transfer learning approach are the techniques used to avoid overfitting in the proposed model.

Fig. 11 shows the highest mean validation accuracies of the split APTOS dataset, using Ban Graham preprocessing and auto-cropping, for Densenet121, Densenet169, Densenet201, InceptionV3, and Resnet50 separately, with 5-fold stratified cross-validation.

The proposed model in Table 8 refers to our newly developed model, which we designed and implemented to improve upon existing methods. While the recent works referenced in the table also used the APTOS dataset and pre-trained models, there are differences in other aspects of the experimental setup. To ensure a meaningful comparison, we adopted the same dataset and leveraged pre-trained models similar to those used in previous studies.

Table 5 - Cross-Validation Results of the Proposed Model for the Split APTOS Dataset

Transfer learning model	*Accuracy (%)*	*F1 Score*	*Precision*
DenseNet121	84.28	85.09	83.42
DenseNet169	84.26	85.21	83.69
DenseNet201	82.32	82.11	82.34
InceptionV3	79.89	79.21	79.61
ResNt50	78.42	78.11	78.21

Table 6 - Cross-Validation Results of the Proposed Model for the Augmented APTOS Dataset with Ben Graham preprocessing

Transfer learning model	*Accuracy (%)*	*F1 Score*	*Precision*
DenseNet121	93.79	93.83	93.73
DenseNet169	94.42	94.42	94.20
DenseNet201	98.56	98.52	98.53
InceptionV3	84.57	85.77	83.462
ResNet50	80.29	80.23	80.63

Table 7 - Cross-Validation Results of the Proposed Model for the Augmented APTOS Dataset with Ben Graham Preprocessing and Auto-Cropping

Transfer learning model	*Accuracy (%)*	*F1 Score*	*Precision*
DenseNet121	91.63	91.91	91.56
DenseNet169	89.89	90.368	89.90
DenseNet201	96.56	96.66	96.47
InceptionV3	83.17	83.30	82.73
ResNet50	81.29	81.63	81.53

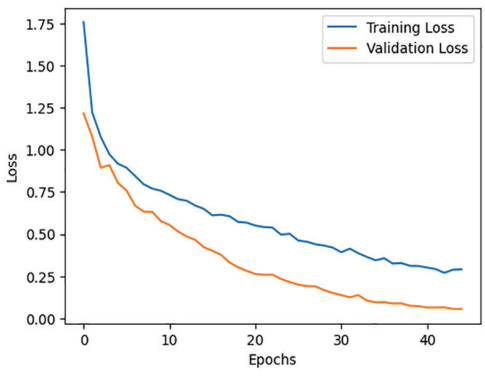

Figure 10 (a) - Loss Curve of the DenseNet201 Model

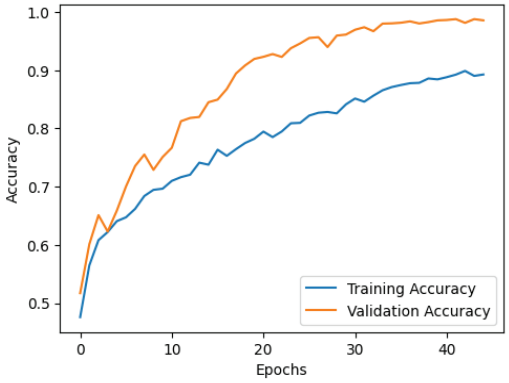

Figure 10 (b) - Accuracy Curve of the DenseNet201 Model

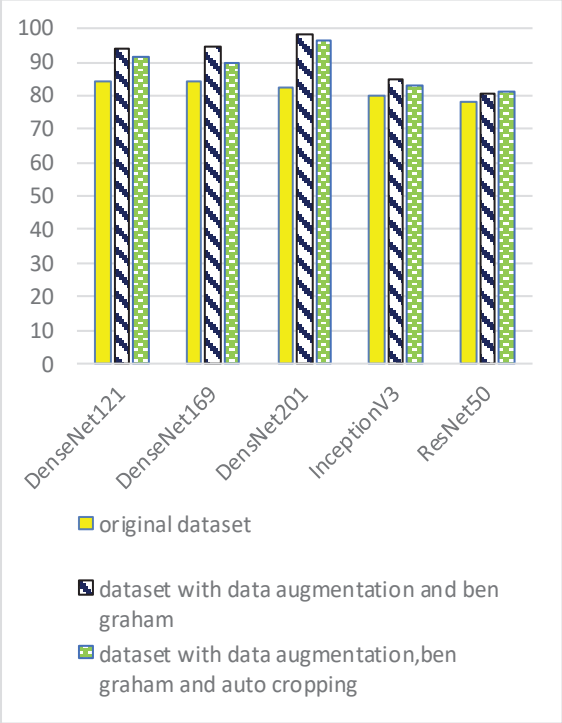

Figure 11 - Comparison of the Highest Accuracy of Aptos Dataset

However, our approach differs in terms of data preprocessing steps, cross-validation strategies, and hyperparameter tuning. By highlighting these differences, we aim to provide a clear and valid comparison of our model's performance relative to the referenced works. This approach allows us to accurately evaluate the improvements achieved by our proposed model while acknowledging the variations in experimental methodologies.

Table 8 - How Proposed Study has Higher Accuracy than Previous Studies

Study/ Paper	*Dataset*	*Transfer learning models used*	*Accuracy*
[19]	APTOS dataset	Densenet121 Densenet169 ResNet50 InceptionV3	96.64% 95.95%, 95.71% 94.73%
[5]	APTOS dataset	Vgg16 ResNet50 DensNet169	97.6% 95.9% 91.1%
[17]	APTOS dataset	ResNet50	82%
Proposed DR detection Model	**Aptos Dataset**	**Densenet121 Densenet169 Densenet201 InceptionV3 ResNet50**	**93.79% 94.42% 98.56% 84.57% 81.29%**

Table 9 presents the highest performance metrics of the proposed model, assessed for DenseNet121, DenseNet169, DenseNet201, InceptionV3, and ResNet50 architectures, using the split Diabetic Retinopathy Kaggle dataset with 5-fold stratified cross-validation. These experiments were also conducted with hyperparameters including a batch size of 32, dense layers of 1024,512, dropout 0.5 and trained over 75 epochs. The training dataset consisted of 1512 images, while the test dataset contained 332 images. Additionally, a learning rate of 0.001 was utilized for all experiments.

Table 10 illustrates the highest performance metrics achieved by the proposed model when tested individually on the augmented Diabetic Retinopathy Kaggle dataset, using Ben Graham preprocessing. The evaluation utilized split data, 5-fold stratified cross-validation. The experiments were conducted with a batch size of 64 over 75 epochs. The training dataset comprised 6048 images, while the test dataset consisted of 332 images. Additionally, a learning rate of 0.001 was employed for all trials.

Table 11 displays the highest performance metrics of the proposed model, evaluated separately on the split augmented Diabetic Retinopathy Kaggle dataset using Ben Graham preprocessing and auto-cropping, along with previously mentioned transfer learning models. This evaluation was conducted using 5-fold stratified cross-validation.

The experiments were conducted with a batch size of 64, dense layers of 1024, 512, dropout of

0.4 over 74 epochs. The training dataset comprised 6048 images, while the test dataset consisted of 332 images. Additionally, a learning rate of 0.0001 was used for all models.

Furthermore, we can see how accuracy increases with transfer learning with CNN and data augmentation, along with fine tuning. DenseNet201 shows the best accuracy among pre-trained model used in proposed method. Fig. 12 shows the highest accuracies of the split Diabetic retinopathy Kaggle dataset with Ben Graham preprocessing and auto-cropping for DenseNet121, DenseNet169, DenseNet201, InceptionV3 and ResNet50 separately.

The model presented in Table 12 represents our new developed approach, designed and implemented to enhance existing methodologies. Although the recent studies cited in the table also utilized the Diabetic retinopathy Kaggle dataset and pre-trained models, there are notable differences in their experimental setups. To facilitate a meaningful comparison, we used the same dataset and pre-trained models as the prior studies. However, our method varies in data preprocessing steps, cross-validation strategies, and hyperparameter tuning. By highlighting these distinctions, we aim to offer a clear and valid comparison of our model's performance against the referenced works. This allows us to accurately assess the improvements made by our proposed model while acknowledging the differences in experimental methodologies.

Table 9 - Cross-Validation Results of the Proposed Model with the Split Diabetic Retinopathy Kaggle Dataset

Transfer learning model	*Accuracy (%)*	*F1 Score*	*Precision*
DenseNet121	85.68	85.89	85.44
DenseNet169	85.50	85.56	85.01
DenseNet201	87.504	87.17	86.93
InceptionV3	82.47	81.81	81.04
ResNet50	64.65	64.44	64.62

Table 10 - Cross-Validation Results of the Proposed Model with the Split Diabetic Retinopathy Kaggle Dataset with Ben Graham Pre-Processing.

Transfer learning model	*Accuracy (%)*	*F1 Score*	*Precision*
DenseNet121	93.95	94.85	94.06
DenseNet169	95.13	95.56	95.22
DenseNet201	97.14	97.41	97.96
InceptionV3	89.93	89.85	89.51
ResNet50	65.65	65.62	64.32

Table 11 - Cross-Validation Results of the Proposed Model with the Split Diabetic Retinopathy Kaggle Dataset Ben Graham Preprocessing and Auto-Cropping

Transfer learning model	*Accuracy (%)*	*F1 Score*	*Precision*
DenseNet121	91.66	92.31	91.22
DenseNet169	92.57	93.70	92.65
DenseNet201	92.98	93.26	92.78
InceptionV3	88.18	88.81	88.04
ResNet50	64.65	64.44	64.62

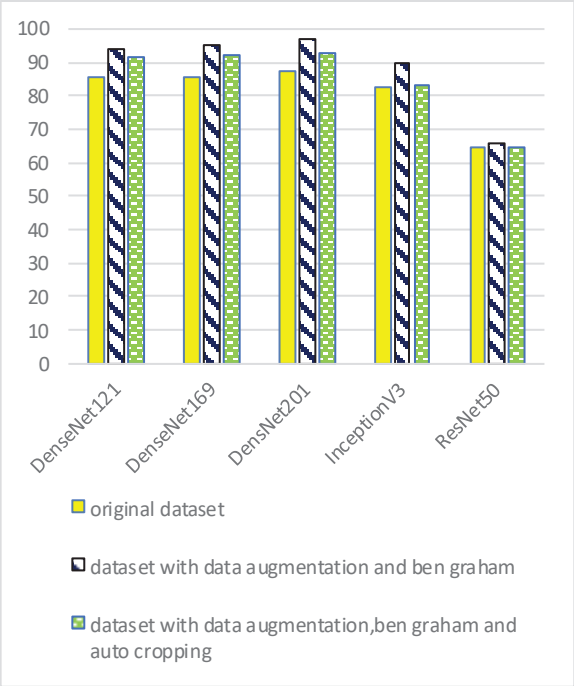

Figure 12 - Comparison of the Highest Accuracy of Diabetic Retinopathy Kaggle Dataset

Table 12 - How Proposed Study has Higher Accuracy than Previous Studies

Study/Paper	*Dataset*	*Transfer learning models used*	*Accuracy*
[13]	Kaggle dataset	InceptionV3	48.8%
[14]	Kaggle dataset	InceptionV3 Xception	87.12% 74.49%
[15]	Kaggle dataset	InceptionV3, Vgg16	63.23% 50.03%
Proposed DR detection Model	**Kaggle dataset**	**DenseNet121 DenseNet169 DenseNet201 InceptionV3 Resnet50**	**93.95% 95.13% 97.14% 89.93% 65.65%**

In the first scenario, data augmentation increases the diversity of the training data by applying random transformations, such as rotations, flips, and zooms, which help the model generalize better by exposing it to a variety of perspectives of the same image.

Ben Graham preprocessing likely includes image resizing, normalization, and enhancement, improving image quality and making relevant features more pronounced. In the second scenario, auto cropping is added to focus on the region of interest, typically the retinal area in medical images. While this can remove irrelevant parts of the image, it also risks cropping out important contextual information, potentially hindering the model's accuracy. Auto cropping might lead to an overemphasis on certain features within the cropped area, making the model less robust when encountering variations in the test set. Additionally, the diversity of augmented images might be reduced when applied to cropped images, affecting the model's generalization ability.

The effectiveness of auto cropping depends on its consistency and accuracy; if the cropping algorithm is not reliable, it could introduce noise and variability in the training data. As a result, the loss of contextual information, reduced data diversity, and potential inconsistencies in cropping can lead to lower accuracies in models trained with auto cropping compared to those trained with only data augmentation and Ben Graham preprocessing.

5. Conclusions

In this proposed work, transfer learning models such as DenseNet121, DenseNet169, DenseNet201, InceptionV3, and ResNet50 are trained with different fine-tuning approaches to detect diabetic retinopathy (DR) from fundus images. Any fundus image representing a stage of DR can be easily detected by this model. The APTOS dataset and the Diabetic Retinopathy Kaggle dataset were utilized in this work. To enhance performance, it is important to have a large amount of data during the training of these models. Therefore, data augmentation was employed. To improve the efficiency of the models, we applied advanced pre-processing methods—the Ben Graham preprocessing and auto-cropping techniques designed specifically for diabetic retinopathy. As a result, there has been an improvement in accuracy. Additionally, introducing new layers to the transfer learning model elevated it to a higher level, achieving higher accuracy than that found in recent research. After multiple fine-tunings, the model was developed to detect fundus images with high precision. The proposed study demonstrates that data augmentation, Ben-Graham preprocessing methods, auto-cropping, along with a transfer learning model and new layers, influence the accuracy of the model and can provide better accuracy for any dataset, such as APTOS and the Diabetic Retinopathy Kaggle dataset.

Following the implementation of all methods, DenseNet201 achieved accuracies of over 98% on the APTOS dataset, indicating a major improvement in its ability to identify diabetic retinopathy in fundus images. On the Diabetic Retinopathy Kaggle dataset, a similar accuracy increment was observed. DenseNet201 surpassed 97% accuracy, showcasing our model's performance in handling different datasets and the significant contribution of advanced pre-processing methods alongside the innovative approach of transfer learning models in enhancing its performance. The proposed research offers valuable insights into the field of diabetic retinopathy detection, promising better diagnostic abilities for this critical health issue.

References

1. "Diabetes," World Health Organization, https://www.emro.who.int/noncommunicable-diseases/diabetes/index.html# (accessed Nov. 20, 2023).

2. Klein, Ronald, Barbara EK Klein, and Scot E. Moss. "Visual Impairment in Diabetes." *Ophthalmology* 91.1 (1984): 1-9.

3. Ravishankar, Saiprasad, Arpit Jain, and Anurag Mittal. "Automated Feature Extraction for Early Detection of Diabetic Retinopathy in Fundus Images." *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009.

4. Litjens, Geert, et al. "A Survey on Deep Learning in Medical Image Analysis." *Medical Image Analysis* 42 (2017): 60-88.

5. Litjens, Geert, et al. "A Survey on Deep Learning in Medical Image Analysis." *Medical Image Analysis* 42 (2017): 60-88.

6. Asiri, Norah, et al. "Deep Learning Based Computer-Aided Diagnosis Systems for Diabetic Retinopathy: A Survey." *Artificial Intelligence in Medicine* 99 (2019): 101701.

7. Samanta, Abhishek, et al. "Automated Detection of Diabetic Retinopathy using Convolutional Neural Networks on a Small Dataset." *Pattern Recognition Letters* 135 (2020): 293-298.

8. Kandel, Ibrahem, and Mauro Castelli. "Transfer Learning with Convolutional Neural Networks for Diabetic Retinopathy Image Classification. A Review." *Applied Sciences* 10.6 (2020): 2021.

9. Naik, Srinivas, et al. "Efficient Diabetic Retinopathy Detection using Convolutional Neural Network and Data Augmentation." *Soft Computing* (2023): 1-12.

10. Gao, Zhentao, et al. "Diagnosis of Diabetic Retinopathy using Deep Neural Networks." *IEEE Access* 7 (2018): 3360-3370.

11. Gargeya, Rishab, and Theodore Leng. "Automated Identification of Diabetic Retinopathy using Deep Learning." *Ophthalmology* 124.7 (2017): 962-969.

12. Kandel, Ibrahem, and Mauro Castelli. "Transfer Learning with Convolutional Neural Networks for Diabetic Retinopathy Image Classification. A Review." *Applied Sciences* 10.6 (2020): 2021.

13. Masood, Sarfaraz, et al. "Identification of Diabetic Retinopathy in Eye Images using Transfer Learning." *2017 International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2017.

14. Mohammadian, Saboora, Ali Karsaz, and Yaser M. Roshan. "Comparative Study of Fine-Tuning of Pre-Trained Convolutional Neural Networks for Diabetic Retinopathy Screening." *2017 24th National and 2nd International Iranian Conference on Biomedical Engineering (ICBME)*. IEEE, 2017.

15. Wang, Xiaoliang, et al. "Diabetic Retinopathy Stage Classification using Convolutional Neural Networks." *2018 IEEE International Conference on Information Reuse and Integration (IRI)*. IEEE, 2018.

16. Masood, Sarfaraz, et al. "Identification of Diabetic Retinopathy in Eye Images using Transfer Learning." *2017 International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2017.

17. Huang, Gao, et al. "Densely Connected Convolutional Networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

18. Ashok, v., Hosmane, N., Mahagaonkar, G.,. Gudigar, A. and A. P, "Diabetic Retinopathy Detection using Retinal Images and Deep Learning Model," *International Journal of Innovative Technology and Exploring Engineering*, Vol. 10, No. 9, pp. 35–39, 2021.

19. Thiagarajan, Aswin Shriram, et al. "Diabetic Retinopathy Detection using Deep Learning Techniques." *J. Comput. Sci* 16.3 (2020): 305-313.

20. Dane, S., Karthik, Maggie. (2019). APTOS 2019 Blindness Detection. Kaggle. https://kaggle.com/competitions/aptos2019-blindness-detection.

21. Stoyanov, Danail, et al., eds. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, dlmia 2018, and 8th International Workshop, ml-cds 2018, Held in Conjunction with miccai 2018, Granada, Spain, September 20, 2018, proceedings*. Vol. 11045. Springer, 2018.

22. Zhou, Yi, et al. "DR-GAN: Conditional Generative Adversarial Network for Fine-Grained Lesion Synthesis on Diabetic Retinopathy Images." *IEEE Journal of Biomedical and Health Informatics* 26.1 (2020): 56-66.

23. Huang, Gao, et al. "Densely Connected Convolutional Networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

24.https://www.kaggle.com/datasets/kaggledrdataset/diabetic-retinopathy-detection-2015-competition.

25. Chollet, François. "Xception: Deep Learning with Depthwise Separable Convolutions." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

26. He, Kaiming, et al. "Deep Residual Learning for Image Recognition." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016.

27. Chollet, François. "Xception: Deep learning with Depthwise Separable Convolutions." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

28. What is overfitting in deep learning [+10 ways to avoid it]. V7. (n.d.). https://www.v7labs.com/blog/overfitting.

29. Huang, Gao, et al. "Densely Connected Convolutional Networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017.

30. Xception architectural design. Xception Architectural Design - coding. (n.d.). https://stephanosterburg.gitbook.io/coding/coding/mldl/tensorfow/ch3xception/xception-architectural-design