Diabetic Retinopathy Detection using Transfer Learning from Pre-trained Convolutional Neural Network Models

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

18-01-2022 / 24-01-2022

CITATION

Chilukoti, Sai venkatesh; Maida, Anthony S; Hei, Xiali (2022): Diabetic Retinopathy Detection using Transfer Learning from Pre-trained Convolutional Neural Network Models. TechRxiv. Preprint.
<https://doi.org/10.36227/techrxiv.18515357.v1>

DOI

[10.36227/techrxiv.18515357.v1](https://doi.org/10.36227/techrxiv.18515357.v1)

Diabetic Retinopathy Detection using Transfer Learning from Pre-trained Convolutional Neural Network Models

Sai Venkatesh Chilukoti, Dr. Anthony S Maida, and Dr. Xiali Hei

Abstract—Diabetic Retinopathy is the most common eye disease that diabetes can cause. It can even result in vision loss and complete blindness. Early detection prevents complete vision loss. However, it is challenging to detect it early because it may not show symptoms in the early stages. The existing models for diabetic retinopathy cannot detect all the stages of diabetic retinopathy. The most widely used metrics like accuracy, f1-score, precision, and recall; do not consider the level of disagreement between labels, which helps in detecting all the stages of diabetic retinopathy. This paper used ResNet, VGG, and EfficientNet pre-trained models. We performed results evaluation using quadratic weighted kappa, which is appropriate for classifying different stages of diabetic retinopathy based on the severity. Furthermore, it considers the level of disagreement between actual and predicted labels. We have achieved the quadratic weighted kappa score of 0.85 using the EfficientNet b3 network, surpassing the existing models like support vector machines, decision trees, convolutional neural networks, and DenseNet pre-trained model.

Index Terms—Convolutional Neural Network, Diabetic Retinopathy, EfficientNet, Inverted Residual Block, Quadratic Weighted Kappa, ResNet, VGGNet.

I. INTRODUCTION

DIABETIC RETINOPATHY is the most common diabetic eye disease and a leading cause of blindness in American adults [1]. Because of the excessive blood sugar, the tiny blood vessels in the retina will be broken and lead to hemorrhages in the retina, and this will cause diabetic retinopathy. Any diabetes can result in diabetic retinopathy. The longer one has diabetes, the higher the risk of diabetic retinopathy. Depending on the severity of the disease, the effect can be in the range of near-normal vision to complete loss of sight. [2], [3]. Diabetic Retinopathy is more likely to attack adults over 40 who have diabetes. In the United States of America, approximately 4.1 million US adults suffer from diabetic retinopathy [4]. Early detection of diabetic retinopathy can prevent 95% vision loss

[5]. Diabetic Retinopathy may not give any symptoms in the earlier stages as it happens inside the eye, even when the blood sugar level is maintained and vision is very normal. Because of this, the doctor can only detect diabetic retinopathy after proper examination [6].

Diabetic Retinopathy can be divided into two stages: Non-Proliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). The NPDR can further be divided into three phases: Mild, Moderate and Severe [3], [7]. NPDR is due to excessive sugar levels that start affecting the tiny blood vessels in the retina, which leads blood vessels to become more swollen and leak fluid; as a result, the retina lacks oxygen and nutrients [3], [8]. As a result, the body produces a vascular endothelial growth factor (VEGF) to provide nutrients and oxygen to the eye's retina. But these new cells are fragile and can be easily damaged, resulting in more swelling and leaking. This advanced stage is called Proliferative Diabetic Retinopathy (PDR) which is hazardous as it often causes permanent vision loss [3], [9].

Using the PyTorch framework, we developed VGG, ResNet, and EfficientNet pre-trained models and fine-tuned them to suit diabetic retinopathy detection. PyTorch supports automatic differentiation and efficiently uses GPUs for parallel processing [10]. Furthermore, we used Google Colab Pro, which gives GPU run time for a limited time and RAM of 25.46 GB to train the developed models.

As diabetic retinopathy, early detection can prevent the patients from vision loss. We have focused on the early detection of diabetic retinopathy using pre-trained models like ResNet, VGG, EfficientNet. We have used the publicly available dataset from Kaggle [11] to develop the state-of-the-art model for diabetic retinopathy detection. We first used VGG, followed by ResNet, and finally EfficientNet. We also have demonstrated the results of all the models using the most relevant metric known as Quadratic Weighted Kappa [12]. The EfficientNet.b3_60 model has achieved a state-of-the-art QWK score of 0.85. The proposed model can be used in healthcare to aid the doctors in the decision-making on the patient's model as the model can distinguish each class of diabetic retinopathy.

II. BACKGROUND AND LITERATURE REVIEW

Many researchers have used different classes of algorithms to detect diabetic retinopathy. We have classified the literature into Computer-Aided Detection (CAD) algorithms, Machine

This work was supported in part by NSF under Grant CNS-2117785.

Ch. S. Venkatesh is with the School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, LA 70503, USA (e-mail: sai-venkatesh.chilukoti1@louisiana.edu).

A. S. Maida, is with the School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, LA 70503, USA (maida.@louisiana.edu).

X. Hei is with the School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, LA 70503, USA (e-mail: sai-venkatesh.chilukoti1@louisiana.edu, maida.@louisiana.edu, xiali.hei.@louisiana.edu)

Learning (ML) algorithms, and Deep Learning (DL) algorithms. Most researchers used metrics like accuracy, recall, etc. Some researchers used Quadratic Weighted Kappa [12], most relevant to diabetic retinopathy detection. Still, the reported Quadratic Weighted Kappa was less than 0.40, which is very low and might not detect all the stages of diabetic retinopathy with high probability.

A. Computer-aided detection and Machine Learning algorithms

In the early 2000s, Computer-Aided Detection (CAD) systems were developed and introduced in the clinical workflow to aid doctors in decision-making and disease diagnosis. But the CAD systems have shown limitations. One of them is that the adversaries report more false positives than the human reader, resulting in further analysis, additional cost and time for the doctor to decide on patient's health condition [13].

Machine Learning requires structured data. Decision trees, support vector machines, linear regression, logistic regression, etc., are some examples of Machine Learning examples. The advantages of machine learning models are less complex and can be implemented in conventional computers with CPUs. In addition, machine learning systems can be set up and operated quickly. The main drawback is that it will not work very well for unstructured data like images, audio data, etc. However, machine learning algorithms can work very well even when the size of the dataset is significantly small [14].

Ramya [15] have used SVMs for diabetic retinopathy detection of two-class classification and got the detection rate of 86% for the diabetic affected eye. However, before feeding the data to the SVM, she has to preprocess and extract the features using the Middle Filtering Method and histogram equalization.

Enrique *et al.* [16] used SVM and Decision Trees (DT) for classifying the different grades: 0 - Normal, 1- Mild NPDR, 2- Moderate NPDR, and 3 - Severe NPDR. The overall accuracy, sensibility, specificity, and AUC score (Area under the receiver operating characteristic (ROC) curve) for SVM are 85.1%, 49.4%, 88.4%, 77%, respectively, whereas for the Decision Trees, they are 85.1%, 49.2%, 88.4%, 77.3%, respectively.

Based on this review, it is evident that ML algorithms require an image processing technique to extract features and a machine learning classifier for classification. On the other hand, CAD algorithms require more time and cost to extract retinal images' features for diabetic retinopathy detection. Therefore, in section 2.2, we discuss some deep learning techniques for end-to-end detection of diabetic retinopathy. So, there is no need for image processing techniques for feature extraction and ML classifier for classification as both can be done by deep learning.

B. Deep Learning algorithms

Deep Learning is a sub-field of machine learning inspired by both the functionality and structure of the brain [17]. Hardware accelerators such as the Graphical Processing Unit (GPU) and Tensor Processing Unit (TPU) have been the backbone of many modern deep learning architectures to provide faster training and inference. Deep Learning algorithms can be

enhanced with more data, unlike other algorithms that get saturated after a specific limit is reached. Furthermore, deep learning algorithms can automatically extract features from the data without human intervention, unlike machine learning algorithms requiring manually extracted features as in [15]. Deep Learning algorithms can be used when there is enough computing power, high storage capacity, and extensive data.

Pratt *et al.* [18] have proposed a Convolutional Neural Network (CNN) model to diagnose patients suffering from diabetic retinopathy from digital fundus images. They took a publicly available dataset of 80,000 images from Kaggle. They got an overall accuracy of 75% on 5000 validation images. They have used a 13 - layered network composed of 10 layers of CNN for feature extraction and three layers of fully connected (FC) layers to classify the given input image. The acquired dataset comprises five classes: 0 - No DR, 1 - Mild DR, 2 - Moderate DR, 3 - Severe DR, 4 - Proliferate DR.

Doshi *et al.* [19] used Deep Convolutional Neural networks to classify high-resolution digital fundus retinal images into five stages based on the severity of diabetic retinopathy. They used quadratic weighted kappa as the metric. They obtained scores of 0.386 and 0.3996 for a single convolutional network and a combination of the three similar models. The main drawback of this technique is the Quadratic Weighted Kappa scores are less than 0.40. So, a low QWK score might indicate that all the stages of diabetic retinopathy might not be detected with a high probability.

Gao *et al.* [20] has proposed an ensemble framework of Deep CNNs for diabetic retinopathy detection. They have also used the dataset from Kaggle. Furthermore, they have used two deep CNN models with an ensemble technique. The accuracy for the imbalanced model 1, imbalanced model 2, balanced model 1, flat model 2 are 78.13, 80.36, 60.8, 60.89, respectively. The demerit of this paper is that even though the accuracy is high, it is not enough as it can give a high score even when a few stages of diabetic retinopathy are detected.

Through the survey on diabetic retinopathy detection using deep learning models, we have found that proposed models must be improved to a much higher Quadratic Weighted Kappa, and they must detect all five stages of diabetic retinopathy in Fig. 1.

C. Transfer Learning

Transfer Learning is a prevalent method in Deep Learning that transfers knowledge from one learning task to another. In Transfer Learning, one will first download the pre-trained models, usually trained on large datasets such as ImageNet [24]. Then, fine-tune it by replacing it with either fully connected layers or the machine learning-based classifier such as Random Forest, Support Vector Machines, and Decision Trees. It is suited to the task at hand. Several pre-trained models are available, which were trained on ImageNet [24]. We have taken ResNet, VGG, and EfficientNet as they have been used extensively in academic research and the industry because of their features. The main features are skip-connections introduced in ResNet, enhancing the trainability of

deeper networks. Lower-dimensional filters used in the VGG network resulted in fewer trainable parameters in the filter simultaneously having a larger receptive field. Furthermore, an efficient compound scaling method used in the EfficientNet effectively scales the network's height, width, and depth to learn the most complex features.

Simonian *et al.* [21] have proposed a CNN architecture known as Visual Geometry Group (VGG). In this paper, they have used only 3*3 convolutional filters. This architecture has secured first and second places in the localization and classification tasks in the ImageNet challenge 2014.

He *et al.* [22] have proposed Deep Residual Learning for Image Recognition. They have developed a residual learning framework to ease the training for the deeper models. Because of the depth of the models, they have achieved better accuracy. For the ImageNet [24] dataset, they have developed layers of depth up to 152 layers.

Tan *et al.* [23] have developed EfficientNet. Their paper has proposed a new scaling method that uniformly scales all depth/width/resolution dimensions using a simple and very effective compound coefficient. They have also proposed a baseline network using Neural Architecture Search (NAS) and then scaled it up to obtain a family of models known as EfficientNets, which have achieved better accuracy and efficiency than previous convolutional networks. The authors of the EfficientNet model developed the compound scaling method by considering the depth, width, and height are dependent on each other.

D. Our goal

Most of the work on diabetic retinopathy detection focuses on the presence or absence of diabetic retinopathy. Furthermore, most papers did not use the quadratic weighted kappa, which must be used to indicate all the five stages of diabetic retinopathy are detected with high probability. However, if used, the reported score is deficient, like 0.39 [19], and some of the models like [18] cannot detect the Mild DR stage. Therefore, the main goal of this paper is to develop a model that can efficiently detect all stages of diabetic retinopathy with a high quadratic weighted kappa. High quadratic weighted kappa indicates that all the stages are detected, unlike metrics like accuracy, such that it can help doctors to make a better decision on the patient.

We have proposed a model using the EfficientNet b3 pre-trained model in this paper. It is scaled uniformly in width, depth, and resolution-wise appropriately to detect complex features from the given retinal images. Hence, it achieves the state-of-the-art QWK score of 0.85 and can efficiently detect all the stages of diabetic retinopathy.

To build the final model, we have analyzed the pre-trained models like ResNet, VGG, and EfficientNet. We have observed that the EfficientNet has performed well. The reason for this could be an effective scaling method to increase the depth, width, and resolution of the images that were employed in building the EfficientNet, unlike other scaling methods that scale depth, height, and width arbitrarily.

III. METHODOLOGY

A. Dataset description

Output labels	Label counts	Target Class	Left Eye Image	Right Eye Image
0	25180	No DR
1	2443	Mild DR
2	5292	Moderate DR
3	873	Severe DR
4	708	Proliferate DR

Fig. 1. Dataset Description

This project uses the resized version of the diabetic retinopathy Kaggle competition dataset [11]. This dataset contains 35,126 digital fundus retinal images. The given images are high-resolution digital fundus images of the retina taken under different imaging conditions. We had split the data into three datasets for training, evaluation, and testing. The training set contains 24,590 retinal images, whereas the validation and test dataset has 5,268 retinal images. We have resized the images to speed up the training to have (150, 150) resolution.

Fig. 1. illustrates the output labels indicated as 0-5, label counts describe the number of images for each class, and the target class denotes the grade of diabetic retinopathy. The grade can help doctors to diagnose the patient with appropriate treatment since treatment depends on the severity of diabetic retinopathy. The last two columns correspond to the left and right eye of an image respectively for each stage of diabetic retinopathy.

B. Quadratic Weighted Kappa

Most of the researchers have considered the metrics like Accuracy, Sensitivity, Roc-Score, which are not suitable for this problem because they do not consider the level of disagreement between labels. Thus, we have considered the quadratic weighted kappa [12] to evaluate the model. The quadratic weighted kappa considered the level of disagreement between actual and predicted labels. So, for example, if the predicted label is 0 and the actual label is 2, then the level of disagreement is double that of when the true label is 2, and the predicted label is 1. Therefore, we think this metric

is very appropriate for this project as target labels must be penalized based on the level of disagreement. The quadratic weighted kappa can be computed using the three matrices, namely expected matrix (E), output matrix (O), and weighted matrix (W).

The quadratic weighted kappa can be computed as follows:

Step 1: calculate the expected matrix (E) by taking the outer product between the actual labels vector and target labels vector.

Step 2: construct the output matrix by building a confusion matrix of actual labels and predicted labels.

Step 3: calculate weight matrix as follows:

$$W_{ij} = \frac{(i - j)^2}{(k - 1)^2} \quad (1)$$

Where i is an actual label, j is the predicted label, and k is the number of classes.

Step 4: normalize the expectation matrix (E) and output matrix (O) by dividing them by their sum.

Step 5: calculate the weighted kappa using the following formula:

$$\text{Quadratic Weighted Kappa} = 1 - \frac{\text{num}}{\text{den}} \quad (2)$$

where num is the sum of elements obtained using element-wise multiplication between weight matrix (W) and output matrix (O), and den is the sum of elements obtained using element-wise multiplication between weight matrix (W) and expectation matrix (E).

The quadratic weighted kappa can range from -1 to 1. The larger the value, the lower the disagreement between an actual label and a true label. The metric score of 1 is a perfect level of agreement, whereas the metric score of -1 is extreme disagreement. Since weights are quadratic, the level of disagreement between actual labels is penalized quadratically. It assumed the given target labels and predicted labels were ordinal. The target labels of this project are ordinal as they are indicated based on the severity of diabetic retinopathy, which follows the order; we have used this metric.

C. Confusion matrices for models

This section demonstrates the confusion matrix for multi-class classification scenarios using the finalized model results as illustrated in Table I.

The following steps illustrate how to calculate the True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) for the No Dr class.

TP: The actual value and predicted value should be the same. So, for the No Dr class, a cell, its value is 3772.

FN: The sum of values of all the cells in the corresponding row except the first cell, which is $8 + 65 + 0 + 2 = 75$.

FP: The sum of values of all the cells in the corresponding column except the first cell, which is $244 + 114 + 3 + 9 = 380$.

TN: The sum of all the remaining cells, which is $31 + 128 + 0 + 0 + 9 + 624 + 5 + 0 + 0 + 77 + 46 + 13 + 0 + 14 + 6 + 98 = 1051$.

TABLE I
CONFUSION MATRIX FOR EFFICIENTNET B3 MODEL

		Predicted Classes				
Actual Classes	NO DR	3772	8	65	0	2
	Mild DR	244	31	128	0	0
	Moderate DR	114	9	624	5	0
	Severe DR	3	0	77	46	13
	Proliferate DR	4	9	0	14	6

The TP, TN, FP, FN can also be calculated similarly for the remaining classes. We have considered the weighted average for the accuracy, precision, recall, and f1-score as well.

D. Model Description

In this project, we have used VGG, ResNet, and EfficientNet based on the architecture's simplicity, efficiency, and performance in the classification and localization tasks on the ImageNet dataset [24]. Furthermore, we have fine-tuned these models to suit the diabetic retinopathy classification task. All the pre-trained models in this work are frozen in the training and validation phases. Only the final classifier is trained to detect the five stages of diabetic retinopathy. Initially, all the models are trained for 30 epochs and selected the best-performed, efficient net b3 model. After finding the best model, we again trained efficient_b3 model for another 30 epochs, and the best weights were saved, resulting in the model's improvement.

TABLE II
EFFICIENTNET BASELINE MODEL ARCHITECTURE

stage _i	F _i	H _i XW _i	C _i	L _i
1	Conv 3 X 3	224X224	32	1
2	MBCConv1, k3x3	112X112	16	1
3	MBCConv6, k3x3	112X112	24	2
4	MBCConv6, k5x5	56X56	40	2
5	MBCConv6, k3x3	28X28	80	3
6	MBCConv6, k5x5	14X14	112	3
7	MBCConv6, k5x5	14X14	192	4
8	MBCConv6, k3x3	7X7	320	1
9	Conv 1 X 1 & Pooling & FC	7X7	1280	1

1) EfficientNet architecture: In this section, we describe the details of EfficientNet architecture. Table II illustrates the architecture details of the baseline model of the EfficientNet family, which is EfficientNetb.0. The main building block of EfficientNet models is inverted residual block [26] as illustrated in Fig.6. which is indicated as MBCONV in Table II . First, the squeeze and excitation [25] technique is applied to the MBCONV. Next, the input feature maps are

increased depth-wise using 1x1 convolutions. Then, 3x3 depth-wise convolution and point-wise convolution are performed to reduce the channels in the output feature map. Finally, the short connection connects the input and output feature map. The primary motivation for authors of the EfficientNet is to improve efficiency and accuracy. They found that it is essential to balance the network depth, width, and resolution while scaling the network. Therefore, they have proposed an effective compound scaling method to obtain the higher versions of the EfficientNet from the baseline architecture. The compound scaling method can be formulated as (3).

$$\begin{aligned}
 \text{depth} : d &= \alpha^\phi \\
 \text{width} : w &= \beta^\phi \\
 \text{resolution} : r &= \gamma^\phi \\
 \text{s.t. } \alpha\beta^2\gamma^2 &\approx 2 \\
 \alpha \geq 1, \beta \geq 1, \gamma \geq 1
 \end{aligned} \quad (3)$$

By setting ϕ equal to 1 and using the grid search method, the parameters α , β , γ can be found by choosing the parameters that give the best accuracy. Then, by increasing the ϕ value, the authors have developed different versions of Efficient Net models.

2) Workflow: The workflow of the model will be explained in step-by-step as follows.

Step 1: All the input images were resized to a size of 150 x 150.

Step 2: The data is fed into the pre-trained model for feature extraction.

Step 3: The extracted features were trained using the classifier to predict the classes of diabetic retinopathy.

Step 4: Finally, we evaluated the model using test dataset using the metrics like accuracy, precision, recall, f1-score, and quadratic weighted kappa.

The general structure of our models is shown in Fig. 2. In the model structure, the pre-trained model is a convolutional neural network. Then, a feed-forward neural network is used to develop the final classifier.

Fig. 2. Structure of Model

3) Model Improvement Techniques: To improve the performance of the models, we used the Adam optimizer, which combines the capabilities of both RmsProp, which adjusts the effective learning rate based on the gradients of the weights and momentum, which helps the model to overcome the local minima. Furthermore, to overcome the exploding gradient problem, we used gradient clipping with a clip value equal to 0.1. Finally, we have also used a weight decay value of 10^{-4} . As a result, the learning rate of all our models is 0.001.

Fig. 3. The Architecture of the Final Model

4) Final Modelling: We have used the EfficientNet models from b0-b6. Out of all the models, the EfficientNet b3 achieved a state-of-the-art result. First, the EfficientNet b3 produces a feature of size: 7 x 7 x 1,536. Then it uses adaptive average pooling 2d of output size 1 x 1 to generate the feature of size 1 x 1 x 1,536; then, it is flattened to have a size of 1,536, which is fed to the classifier. We have replaced the classifier used in the EfficientNet b3 with the developed final classifier to detect five stages of diabetic retinopathy. First, the feature vector having dimension 1,536 has been fed to the fully connected layer, giving a feature vector of size 512. Then, it is passed to the dropout layer of rate equals 0.5, followed by the ReLU activation layer. Next, the output of ReLU is passed to the fully connected layer, giving the output size of 512, which is fed to the dropout layer having a rate equal to 0.25 and then to the ReLU activation layer. Finally, the output of the ReLU activation layer is given to the fully connected layer, which has five units equal to the number of output classes. It is then fed to the Softmax activation layer to generate the probabilities for each category. The class with the highest probability will be the grade of diabetic retinopathy. The detailed architecture of the final model is depicted Fig. 3.

IV. RESULTS

This section demonstrates the results evaluated using accuracy, precision, recall, f1_score, and quadratic weighted kappa.

A. 4.1 Metric scores for models

Fig. 4. shows accuracy, f1_score, precision, recall, and quadratic weighted kappa of the VGG, ResNet, and EfficientNet models. The Efficient_b3 model trained for 30 and 60 epochs are denoted as Efficient_Net_b3_30 and Efficient_Net_b3_60, respectively. The metric quadratic weighted kappa is denoted as QWK. The QWK score of VGG 16, VGG 19, ResNet 18, ResNet 34, and ResNet 50 are absolute zero

because of strong disagreement between the actual labels and predicted labels generated by the label. In the following subsection, we show the confusion matrices for all the models. The confusion matrices help illustrate the QWK score of 0. The QWK score of 0 is because the given images are taken under various imaging conditions, and models are not robust to varying conditions. The different versions of the ResNet and VGG models are only scaled by the depth, which is not efficient enough to extract the complex features in the input images.

However, ResNet models used skip connections to overcome the vanishing gradient problems such that model architecture can be increased to enormous depth. The higher versions of ResNet, which are ResNet101 and ResNet152, give the little positive QWK score, indicating that increasing the depth will increase the level of agreement between the actual labels and true labels, which is desirable. By observing the QWK scores for VGG and ResNet models, we found that the models scaled only by the depth are ineffective in tackling images taken in various imaging conditions. Thus, we have selected the EfficientNet models that use the compound scaling method. The QWK score for all the versions of the EfficientNet is comparatively good. Out of all the versions, EfficientNet b3 trained for 60 epochs has performed very well, which is indicated by the QWK score of 0.85. The reason is that different versions of the EfficientNet have different resolutions, depths, and widths. The EfficientNet b3 model's scaling parameters of resolution, depth, and width are suited to given input resized images. Therefore, even the resolution, depth, and width are scaled more than needed will under perform, which is evident from the QWK scores of EfficientNet b4, b5, b6 models. The remaining metrics, accuracy, precision, recall, and f1_score, are similar for all the remaining models. The reason for picking the QWK is to show that even when the model has high accuracy, recall, etc. is not the best model unless it has a high QWK metric score. The possible reason is that even if the model cannot detect all the stages, the accuracy, precision, recall, and f1_score will be much higher than the QWK score. A detailed view of the results is depicted in Fig. 4. It is visible that all the metric scores are higher for the EfficientNet b3.60 model. The QWK scores for the ResNet 18, 34, 50, and VGG models are precisely zero. The remaining metrics are similar for most of the models.

B. Confusion Matrices for Models

In this section, we demonstrate the confusion matrices for all the models. From Fig.5. we found the QWK score of the VGG and ResNet models is either 0 or very low because the models only identify class 0 and misclassify the remaining classes completely. Different from them, the EfficientNet models could recognize the other classes as well. Moreover, the EfficientNet b3_30 and Efficient Net b3_60 have detected all the classes with high probability; hence the QWK score is very high for EfficientNet models compared to the VGG and ResNet models.

TABLE III
METRIC SCORES FOR ALL THE MODELS

Model	Accu.	F1_score	Prec.	QWK	Recall
VGG16	0.73	0.62	0.53	0	0.73
VGG19	0.73	0.62	0.54	0	0.73
ResNet18	0.72	0.61	0.52	0	0.72
ResNet34	0.72	0.61	0.52	0	0.72
ResNet50	0.73	0.62	0.53	0	0.73
ResNet101	0.73	0.61	0.56	0.01	0.73
ResNet152	0.72	0.61	0.58	0.07	0.72
EffNet_b0	0.77	0.72	0.69	0.50	0.77
EffNet_b1	0.78	0.72	0.69	0.48	0.78
EffNet_b2	0.77	0.71	0.67	0.50	0.77
EffNet_b3_30	0.85	0.84	0.83	0.82	0.85
EffNet_b3_60	0.87	0.84	0.85	0.85	0.87
EffNet_b4	0.77	0.70	0.69	0.40	0.77
EffNet_b5	0.72	0.61	0.52	0	0.72
EffNet_b6	0.77	0.72	0.70	0.53	0.77

C. Results comparison with existing literature

Fig.7. illustrates the comparison of our proposed model results with existing literature. In this section, we will explain details about Fig.7.

Ramya [15] used SVM to classify the normal eye and diabetic affected eye. She used the Middle Filtering method for feature extraction. As a result, she can classify the normal eye with the recognition rate of 86%, whereas for the diabetic affected eye, the recognition rate is 82%.

Enrique *et al.* [16] used Messidor database, Extracted three features, namely Blood Vessels, Hard exudates, and Microaneurysms, for extracting the features and also support vector machines and decision tree classifiers for classification of diabetic retinopathy into four stages, namely Normal, Mild NPDR, Moderate NPDR, and Severe NPDR. They have got an accuracy of 85.1% for both the SVM and DT. For the remaining metrics, both the SVM and DT have similar results.

Pratt *et al.* [18] used a publicly available dataset from Kaggle. They developed a convolutional neural network to detect all the stages of diabetic retinopathy. They obtained an accuracy of 75% and specificity of 95%. The main drawback of this method is their proposed model completely misclassifies the Mild DR class, which can be observed from the confusion matrix that for class 1 (Mild Dr), the total number of correct predictions is zero.

Doshi *et al.* [19] used EyePACs dataset. They developed an ensemble model using three similar convolutional neural networks to classify high-resolution retinal images into five stages of diabetic retinopathy. They achieved 0.3996 quadratic weighted kappa, which is very low can be improved much further.

Gao *et al.* [20] used the Kaggle dataset and obtained an Imbalanced accuracy of 80.36% and a recall of 47.70% by using the transfer learning technique for the classification of diabetic retinopathy into five stages. The five stages are Normal, Mild NPDR, Moderate NPDR, Severe NPDR, and PDR.

We used a Kaggle dataset and developed the EfficientNet_b3.60 model, which is able to detect all the five stages of diabetic retinopathy with a QWK score of 0.85, which is

Fig. 4. Metrics scores for all the models.

Model Name	Vgg16					Vgg 19					Resnet 18					Resnet 34					Resnet 50				
Real Result	Predicted Result					Predicted Result					Predicted Result					Predicted Result					Predicted Result				
	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
0	363	0	0	0	0	388	0	0	0	0	382	0	0	0	0	383	0	0	0	0	386	0	0	0	0
1	363	0	0	0	0	376	0	0	0	0	371	0	0	0	0	391	0	0	0	0	365	0	0	0	0
2	763	0	0	0	0	793	0	0	0	0	806	0	0	0	0	804	0	0	0	0	806	0	0	0	0
3	131	0	0	0	0	131	0	0	0	0	152	0	0	0	0	138	0	0	0	0	116	0	0	0	0
4	113	0	0	0	0	113	0	0	0	0	114	0	0	0	0	105	0	0	0	0	117	0	0	0	0
Model Name	Resnet 101					Resnet 152					Efficient net b0					Efficient net b1					Efficient net b2				
Real Result	Predicted Result					Predicted Result					Predicted Result					Predicted Result					Predicted Result				
	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
0	384	0	0	0	0	377	0	0	0	0	378	0	105	2	5	383	0	68	0	0	376	0	120	0	0
1	383	0	0	0	0	360	0	0	0	0	321	0	20	0	0	357	0	17	0	0	346	0	29	0	0
2	783	0	0	0	0	814	0	0	0	0	532	0	228	25	1	501	0	273	0	2	487	0	307	0	0
3	140	0	0	0	0	128	0	0	0	0	27	0	67	39	2	30	0	89	0	3	24	0	86	0	0
4	103	0	0	0	0	90	0	0	0	0	28	0	47	17	15	34	0	41	0	18	17	0	91	0	0
Model Name	Efficient net b3_30					Efficient net b3_60					Efficient net b4					Efficient net b5					Efficient net b2				
Real Result	Predicted Result					Predicted Result					Predicted Result					Predicted Result					Predicted Result				
	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
0	386	3	20	0	0	377	8	65	0	0	385	0	38	0	2	372	0	83	1	0	380	0	59	0	2
1	326	2	34	0	0	244	31	128	0	0	348	0	3	0	0	363	0	11	0	0	335	0	13	0	0
2	247	3	501	0	0	114	9	624	5	0	605	0	194	1	8	587	0	234	21	0	513	0	269	1	10
3	10	0	69	38	0	3	0	77	46	13	42	0	71	4	2	36	0	70	33	0	26	0	87	5	18
4	16	0	29	13	37	9	0	14	6	98	44	0	38	1	16	19	0	63	21	0	33	0	66	1	21

Fig. 5. Confusion matrices for all the models

Fig. 6. Inverted Residual block.

Fig. 7. Results comparison with existing literature

Researchers	classifier	Dataset	Feature extraction	Performance																																																			
V.Ramya [15]	SVM	<table><tr><th>Training Set</th><th>Testing Set</th></tr><tr><td>18</td><td>12</td></tr><tr><td>10</td><td>5</td></tr></table>	Training Set	Testing Set	18	12	10	5	Middle filtering method	<table><tr><th>Type of eye</th><th>Recognition Rate</th></tr><tr><td>Normal Eye</td><td>86%</td></tr><tr><td>Diabetic Affected Eye</td><td>82%</td></tr></table>	Type of eye	Recognition Rate	Normal Eye	86%	Diabetic Affected Eye	82%																																							
Training Set	Testing Set																																																						
18	12																																																						
10	5																																																						
Type of eye	Recognition Rate																																																						
Normal Eye	86%																																																						
Diabetic Affected Eye	82%																																																						
Enrique et al. [16]	SVM, DT	Messidor database	Extracted Blood vessels, Hard exudates, Microaneurysms	<table><tr><th>Metric</th><th>SVM</th><th>DT</th></tr><tr><td>Accuracy</td><td>85.1%</td><td>85.1%</td></tr><tr><td>Sensibility</td><td>49.4%</td><td>49.2%</td></tr><tr><td>Specificity</td><td>88.4%</td><td>88.4%</td></tr><tr><td>AUC</td><td>77%</td><td>77.3%</td></tr></table>	Metric	SVM	DT	Accuracy	85.1%	85.1%	Sensibility	49.4%	49.2%	Specificity	88.4%	88.4%	AUC	77%	77.3%																																				
Metric	SVM	DT																																																					
Accuracy	85.1%	85.1%																																																					
Sensibility	49.4%	49.2%																																																					
Specificity	88.4%	88.4%																																																					
AUC	77%	77.3%																																																					
Harry Pratt et al. [18]	CNN	Kaggle dataset		<table><tr><th>Metric</th><th>Metric value</th></tr><tr><td>Specificity</td><td>95%</td></tr><tr><td>Accuracy</td><td>75%</td></tr><tr><td>Confusion Matrix</td><td><table><tr><th>Real Result</th><th colspan="6">Predicted Result</th></tr><tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><th>0</th><td>3456</td><td>0</td><td>145</td><td>1</td><td>34</td></tr><tr><th>1</th><td>344</td><td>0</td><td>27</td><td>0</td><td>1</td></tr><tr><th>2</th><td>543</td><td>0</td><td>179</td><td>5</td><td>40</td></tr><tr><th>3</th><td>40</td><td>0</td><td>63</td><td>10</td><td>15</td></tr><tr><th>4</th><td>28</td><td>0</td><td>23</td><td>3</td><td>43</td></tr></table></td></tr></table>	Metric	Metric value	Specificity	95%	Accuracy	75%	Confusion Matrix	<table><tr><th>Real Result</th><th colspan="6">Predicted Result</th></tr><tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><th>0</th><td>3456</td><td>0</td><td>145</td><td>1</td><td>34</td></tr><tr><th>1</th><td>344</td><td>0</td><td>27</td><td>0</td><td>1</td></tr><tr><th>2</th><td>543</td><td>0</td><td>179</td><td>5</td><td>40</td></tr><tr><th>3</th><td>40</td><td>0</td><td>63</td><td>10</td><td>15</td></tr><tr><th>4</th><td>28</td><td>0</td><td>23</td><td>3</td><td>43</td></tr></table>	Real Result	Predicted Result							0	1	2	3	4	0	3456	0	145	1	34	1	344	0	27	0	1	2	543	0	179	5	40	3	40	0	63	10	15	4	28	0	23	3	43
Metric	Metric value																																																						
Specificity	95%																																																						
Accuracy	75%																																																						
Confusion Matrix	<table><tr><th>Real Result</th><th colspan="6">Predicted Result</th></tr><tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><th>0</th><td>3456</td><td>0</td><td>145</td><td>1</td><td>34</td></tr><tr><th>1</th><td>344</td><td>0</td><td>27</td><td>0</td><td>1</td></tr><tr><th>2</th><td>543</td><td>0</td><td>179</td><td>5</td><td>40</td></tr><tr><th>3</th><td>40</td><td>0</td><td>63</td><td>10</td><td>15</td></tr><tr><th>4</th><td>28</td><td>0</td><td>23</td><td>3</td><td>43</td></tr></table>	Real Result	Predicted Result							0	1	2	3	4	0	3456	0	145	1	34	1	344	0	27	0	1	2	543	0	179	5	40	3	40	0	63	10	15	4	28	0	23	3	43											
Real Result	Predicted Result																																																						
	0	1	2	3	4																																																		
0	3456	0	145	1	34																																																		
1	344	0	27	0	1																																																		
2	543	0	179	5	40																																																		
3	40	0	63	10	15																																																		
4	28	0	23	3	43																																																		
Darshit Doshi et al. [19]	Ensemble of 3 CNNs	EyePACs dataset		<table><tr><th>Model</th><th>Quadratic kappa score</th></tr><tr><td>CNN Model 1</td><td>0.3066</td></tr><tr><td>CNN Model 2</td><td>0.35</td></tr><tr><td>CNN Model 3</td><td>0.386</td></tr><tr><td>Ensemble of 3 CNN models</td><td>0.3996</td></tr></table>	Model	Quadratic kappa score	CNN Model 1	0.3066	CNN Model 2	0.35	CNN Model 3	0.386	Ensemble of 3 CNN models	0.3996																																									
Model	Quadratic kappa score																																																						
CNN Model 1	0.3066																																																						
CNN Model 2	0.35																																																						
CNN Model 3	0.386																																																						
Ensemble of 3 CNN models	0.3996																																																						
Gao Jinfeng et al. [20]	Two deep CNN models with ensemble technique	Kaggle dataset		<table><tr><th>Metric</th><th>Metric score</th></tr><tr><td>Imbalanced acc.</td><td>80.36%</td></tr><tr><td>Balanced acc</td><td>60.89%</td></tr><tr><td>Recall</td><td>47.70%</td></tr><tr><td>Specificity</td><td>85.94%</td></tr></table>	Metric	Metric score	Imbalanced acc.	80.36%	Balanced acc	60.89%	Recall	47.70%	Specificity	85.94%																																									
Metric	Metric score																																																						
Imbalanced acc.	80.36%																																																						
Balanced acc	60.89%																																																						
Recall	47.70%																																																						
Specificity	85.94%																																																						

much higher than existing literature.

V. CONCLUSION AND FUTURE WORK

Until now, little research has been done on detecting all the stages of diabetic retinopathy with higher Quadratic Weighted Kappa. In this paper, we have used the three pre-trained models ResNet, VGG, and EfficientNet. Among them, VGG and ResNet can only detect class 0 with a higher probability. Our results suggest that using pre-trained models like ResNet, VGG, which are scaled only by their depth, cannot learn the features from the images that correspond to all the stages of diabetic retinopathy. Then, we found the EfficientNet models that use the compound scaling method uniformly scale the depth, width, and resolution. So, an efficient net model can detect more than 1 stage of diabetic retinopathy.

The EfficientNet b3 model can detect all the stages of diabetic retinopathy with a higher quadratic weighted kappa score of 0.85. As, we have used the Google Colab Pro for model training, the resources are not sufficient to train the efficient net b7 model.

However, we achieved the higher quadratic weighted kappa score of 0.85. There is still an improvement that can be done by using higher resolution images, using the ensemble of different CNN models, or the same model that achieved the best result. In our case, it is efficient net b3 and more advanced architectures like CoAtNet, which uses depth-wise convolution and self-attention can be used and check how it performs.

REFERENCES

- [1] "Diabetic Retinopathy Data and Statistics." <https://www.nei.nih.gov/learn-about-eye-health/outreach-campaigns-and-resources/eye-health-data-and-statistics/diabetic-retinopathy-data-and-statistics>.
- [2] "Diabetic Retinopathy." <https://brailleinstitute.org/diabetic-retinopathy/>
- [3] "DIABETES AND YOUR EYES." <https://www.noweeyesee.com/diabetes-and-your-eyes>.
- [4] T. E. D. P. R. Group*, "The Prevalence of Diabetic Retinopathy Among Adults in the United States," *Arch. Ophthalmol.*, vol. 122, no. 4, pp. 552–563, Apr. 2004, doi: 10.1001/archophth.122.4.552.
- [5] "Saving Vision for Patients Living with Diabetes Starts with You." <https://www.hillrom.com/en/solutions/enable-earlier-diagnosis-and-treatment/>.
- [6] "Diabetic retinopathy." <https://www.mayoclinic.org/diseases-conditions/diabetic-retinopathy/symptoms-causes/syc-20371611>.
- [7] "Help Detect Progression of Disease Severity1." <https://www.diabeticretinaldisease.com/disease-education/stages-severity> (accessed Nov. 29, 2021).
- [8] "Nonproliferative Diabetic Retinopathy (NPDR) and Macular Edema." <https://louisvillediabeticeyedoctor.com/truck-drivers/nonproliferative-diabetic-retinopathy-npdr-and-macular-edema/>.
- [9] "Diabetic retinopathy." <https://www.aoa.org/healthy-eyes/eye-and-vision-conditions/diabetic-retinopathy?sso=y>.
- [10] "Pytorch." <https://pytorch.org/>.
- [11] "Dataset." <https://www.kaggle.com/tanlikesmath/diabetic-retinopathy-resized>.
- [12] "Quadratic weighted kappa." <https://www.kaggle.com/aroraaman/quadratic-kappa-metric-explained-in-5-simple-steps>.
- [13] "PMC6901900."
- [14] "Machine Learning VS Deep Learning." <https://flatironschool.com/blog/deep-learning-vs-machine-learning>.
- [15] V. Ramya, "SVM Based Detection for Diabetic Retinopathy," *Iccad*, vol. V, no. I, pp. 11–13, 2018, [Online]. Available: www.ijltemas.in.
- [16] E. V. Carrera, A. Gonzalez, and R. Carrera, "Automated detection of diabetic retinopathy using SVM," *Proc. 2017 IEEE 24th Int. Congr. Electron. Electr. Eng. Comput. INTERCON 2017*, no. April 2018, 2017, doi: 10.1109/INTERCON.2017.8079692.
- [17] "What is Deep Learning?" <https://machinelearningmastery.com/what-is-deep-learning/>.
- [18] "H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and Y. Zheng, "Convolutional Neural Networks for Diabetic Retinopathy," *Procedia Comput. Sci.*, vol. 90, no. July, pp. 200–205, 2016, doi: 10.1016/j.procs.2016.07.014.
- [19] D. Doshi, A. Shenoy, D. Sidhpura, and P. Gharpure, "Diabetic retinopathy detection using deep convolutional neural networks," *Int. Conf. Comput. Anal. Secur. Trends, CAST 2016*, pp. 261–266, 2017, doi: 10.1109/CAST.2016.7914977.
- [20] G. Jinfeng, S. Qummar, Z. Junming, Y. Ruxian, and F. G. Khan, "Ensemble framework of deep CNNs for diabetic retinopathy detection," *Comput. Intell. Neurosci.*, vol. 2020, 2020, doi: 10.1155/2020/8864698.
- [21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1–14, 2015.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-December, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.
- [23] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 10691–10700, 2019.
- [24] "ImageNet." <https://www.image-net.org/>.
- [25] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, 2020, doi: 10.1109/TPAMI.2019.2913372.
- [26] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "Mobilenetv2," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018.

Ch. S. Venkatesh Ch. S. Venkatesh received the B.Tech in Electronics and Communication Engineering from Velagapudi RamaKrishna Siddhartha Engineering College. During his Undergraduate studies he has done Machine Learning courses to gain practical knowledge. He is Doctoral student at the School of Computing and Informatics, University of Louisiana at Lafayette. He worked as Graduate Teaching Assistant for the Introduction to Computer Engineering Course. Currently, he is working as Graduate Research Assistant. His research interests are Machine Learning, Privacy preserving Federated Learning, Quantum Computing.

Anthony S. Maida Anthony S. Maida received the BA in mathematics, MS in computer science, and PhD in psychology at the State University of New York at Buffalo. He received postdoctoral training at Brown University and the University of California at Berkeley. He is Associate Professor at the School of Computing and Informatics, University of Louisiana at Lafayette. His research is in biologically inspired artificial intelligence, brain simulation, and deep learning.

X. Hei Xiali (Sharon) Hei is an assistant professor in the School of Computing and Informatics at the University of Louisiana at Lafayette. Prior to joining the University of Louisiana at Lafayette, she was an assistant professor at Delaware State University from 2015-2017. Sharon received her Ph.D. in computer science from Temple University in 2014, focusing on computer security.

She was awarded an NSF MRI award, a Facebook award, an LA BoRSF CEMC Talent Initiative Fund, an LA BoRSF Seed fund, and a Delaware DEDO grant, etc. Also, she earned an M.S. in Software Engineering from Tsinghua University and a B.S. in Electrical Engineering from Xi'an Jiaotong University. She got several awards such as ACM 2014 MobiHoc Best Poster Runner-up Award, Dissertation Completion Fellowship, and IEEE INFOCOM, etc. Her papers were published at USENIX Security Symp., ACM CCS, and IEEE INFOCOM, etc. She is a TPC member of the USENIX Security Symp., IEEE GLOBECOM, SafeThings, etc. She is an IEEE senior member since 2019.