Diabetic Retinopathy Detection and Classification

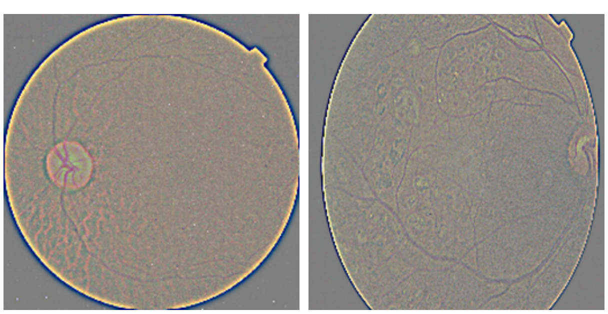
A P Vaideeswaran[1]

***Abstract*— Diabetic retinopathy (DR) is the leading cause of blindness in the working-age population. The aim is to detect and classify DR into four stages if the disease exists thereby estimating the severity even before visiting a doctor. This is crucial as the disease barely presents any symptoms in the early stages. Early medical attention can prevent loss of eyesight or slow down the disease. The patient's condition can be monitored remotely using such models. DR detection is time-consuming and requires sizable expertise and infrastructure. Therefore, such models can help doctors and patients alike, giving the health sector time to improve the infrastructure required to cater to the needs of an increasing diabetic population, especially in rural areas. In this paper, we look at the performance of an augmented version of EfficientNetB4 where we add an extra ANN layer before the output layer and train all the layers.**

I. Introduction

The World Health Organization estimates that 347 million people have the disease worldwide. Diabetic Retinopathy (DR) is an eye disease associated with long-standing diabetes. Present DR detection is costly regarding time and manual effort, where a clinician examines and evaluates digital color fundus photographs of the retina. When reviews are submitted a few days later, the delayed results lead to lost follow-up, miscommunication, and delayed treatment. Clinicians identify DR by observing vascular abnormalities that manifest in the form of lesions. While this approach is effective, its resource demands are high.

Therefore, there is a need for an effective automated method of DR screening based on ML image classification algorithms. Colored fundus have sufficient resolution, which, when trained upon, might result in the models even having clinical potential [4].



(a) (b)

Fig. 1: Fundus photographs of the retina: (a) Healthy Eye; and (b) Proliferative Diabetic Retinopathy

[1]Department of Electrical Engineering, Indian Institute of Technology Hyderabad, India.

EfficientNetB4 is a Convolution Neural Network (CNN) based Deep Learning Model. It has been used previously for classification purposes. This sets the expectation that it can handle this problem efficiently as well after training. This model was identified as the Golden Bullet at the end of the ideation phase, which involved Transfer Learning, where we compared the performances of VGG16, ResNet50, EfficientNetB3, and B4. To make EfficientNetB4 handle the nuances of this task, we train all the model's layers. The pre-trained model was obtained from Keras-Applications [2]. In this paper, we present the performance of the EfficientNetB4.

II. Literature Review

Raj Sunil Salvi et al. [6] compared the various deep-learning models like VGG16, ResNet50, and EfficientNetB0 in the transfer-learning framework. They found that VGG16 is better than the others, given the transfer-learning framework.

R. N. Lazuardi et al. [5] used EfficientNetB4 and B5 and preprocessing methods like contrast limited adaptive histogram equalization (CLAHE) and image central cropping. They also employed a cyclical learning rate scheduler and obtained high QWK, F1, and accuracy scores of 0.79, 0.82, and 83.88%. These values were nearly the same across both models.

III. Materials and Methods

A Kaggle annotated public dataset comprising retinal fundus images was used to train all the EfficientNetB4 layers [3].

A. Data Description

The dataset includes nearly 2750 retinal fundus images, each 224x224x3. For each image, the severity has been provided along with the index. For instance, given the name Severe DR_10.png, Severe DR indicates that the disease is in the third stage, and 10 is used only to index the dataset. A clinician has rated the presence of the disease in each image on a scale of 0 to 4, according to the following scale: 0 - No DR 1 - Mild 2 - Moderate 3 - Severe 4 - Proliferative DR.

B. Preprocessing Methods

ZCA Whitening was performed to decorrelate the pixels. The rotation range was set to 30 degrees, implying that the input images might be rotated by angles of at most 30 degrees either clockwise or counterclockwise. The newly created pixel values after the above transformations are interpolated based on the nearest existing pixel. Resizing was not required as all the images were of the same size.

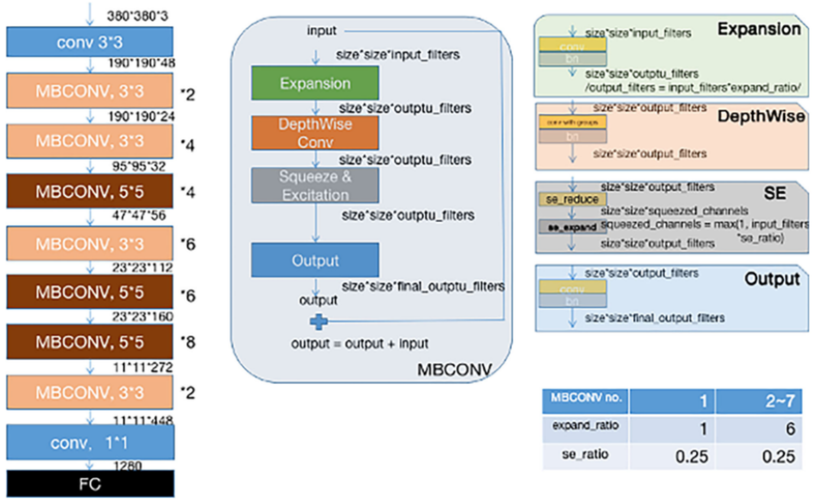


Fig. 2: ML model: EfficientNetB4
[1]

C. Training Phase

An additional penultimate layer comprising 64 nodes was added over and above the EfficientNetB4 base layers. The ReLU Activation function was used. The output layer had 5 nodes corresponding to the 5 classes with the Soft Max Activation Function. The training was done for 20 epochs employing the Adam Optimizer. The loss function employed was the Categorical Cross Entropy, setting the learning rate at 0.001, and the chosen batch size was 20.

Softmax Function for a given class s_i: f(s_i) = $\left(\frac{e^{s_p}}{\sum_j^C e^{s_j}}\right)$ Where s_j are the scores inferred for each class in C.

ReLU Function: f(x) = max(0,x)

Accuracy: $\frac{TP+TN}{TP+TN+FP+FN}$ where TP = True-Positive, TN = True-Negative, FP = False-Positive, and FN = False-Negative.

Precision: $\frac{TP}{TP+FP}$

Recall: $\frac{TP}{TP+FN}$

F1-Score: $\frac{NetTP}{NetTP+0.5(FP+FN)}$

Quadratic Weighted Kappa Metric: 1- $\frac{\Sigma w_{i,j} O_{i,j}}{\Sigma w_{i,j} E_{i,j}}$ where $O_{i,j}$ corresponds to the number of i (actual) that received a predicted value j, $w_{i,j}$ = $\frac{(i-j)^2}{(N-1)^2}$ An NxN matrix, E, is calculated assuming that there is no correlation between values. This is calculated as the outer product between the actual histogram vector of outcomes and the predicted histogram vector, normalized such that E and O have the same sum.

Categorical-Cross Entropy = $-\log\left(\frac{e^{s_p}}{\sum_j^C e^{s_j}}\right)$ where s_p is the CNN score for the positive class.

D. Outcome Prediction

The model predicts a class label with a certain probability. We choose the class that corresponds to the maximum probability. The performance was assessed using the metrics: Accuracy, Precision, Recall, F1_Score, and the Quadratic Weighted Kappa Metric. In this case, the Precision, Recall, and F1_Score will take the same value as we are using micro-averaging to make up for the fact that there is a class imbalance.

IV. Experimental Results

The performance of the augmented model was evaluated after training all the layers at the end of 20 epochs. The following results were obtained:

Accuracy = Precision = Recall = F1_Score = 0.7163636363636363.

Quadratic Weighted Kappa Metric = 0.7591537406352221

Fig. 3: Accuracy and Loss Plots

V. Discussion

Despite the training accuracy plot being non-monotone, the general trend with the increase in the number of epochs

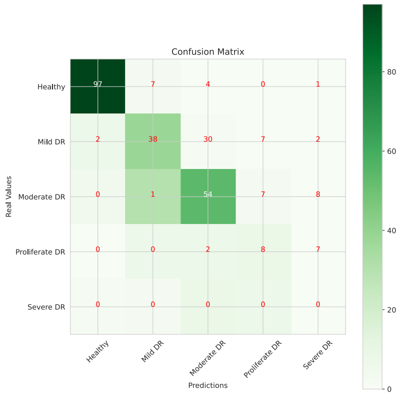


Fig. 4: Confusion Matrix

is greater training accuracy. The number of epochs was not increased beyond 20 as the validation accuracy already hits its maxima at the 13th epoch, and increasing the number of epochs might result in overfitting. EfficientNetB4 is already a very deep model. Therefore, it is unlikely that increasing the width and depth of the ANN introduced above the base model will enhance performance substantially. The model performs well in most of the classes except the class: Severe DR, which comprises the least number of samples.

To improve the performance of EfficientNetB4, oversampling of the non-dominant classes can be done to make up for the class imbalance. However, this should be done keeping in mind the learnability of the classes. Some of the classes are inherently easier to learn than others and hence may require a lesser number of samples to prevent overfitting. Other datasets can also be explored, comprising a similar distribution of images but with a higher resolution (above 224x224x3). Alternatively, this model can be used in conjunction with segmentation models as the segmentation models can provide additional insight regarding the microaneurysms, thereby enhancing performance.

Overall, we have been able to meet the initial goal at an intermediate level. Given more resources, better datasets, and more models to experiment with, this project can be turned into a product with a good practical value capable of helping the society at large.

REFERENCES

[1] https://www.researchgate.net/publication/350928633/figure/fig1/AS:1143128118756283 2@1694280214213/Original-efficientnet-b4-architecture.tif.

[2] https://keras.io/api/applications/#usage-examples-for-image-classification-models.

[3] https://www.kaggle.com/datasets/sachinkumar413/diabetic-retinopathy-dataset, 2021.

[4] https://www.kaggle.com/c/diabetic-retinopathy-detection/, 2025.

[5] Rachmadio Noval Lazuardi, Nyoman Abiwinanda, Tafwida Hesaputra Suryawan, Muhammad Hanif, and Astri Handayani. Automatic diabetic retinopathy classification with efficientnet. In *2020 IEEE REGION 10 CONFERENCE (TENCON)*, pages 756–760, 2020.

[6] Raj Sunil Salvi, Shreyas Rajesh Labhsetwar, Piyush Arvind Kolte, Veerasai Subramaniam Venkatesh, and Alistair Michael Baretto. Predictive analysis of diabetic retinopathy with transfer learning. In *2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*, pages 1–6, 2021.

VI. CODE SUBMISSION

Code Files