Prediction of Diabetic Retinopathy using Convolutional Neural Networks

Manal Alsuwat, Hana Alalawi, Shema Alhazmi, Sarah Al-Shareef
Computer Science Department, College of Computer Science and Information System
Umm AlQura University, Makkah, 21955, Saudi Arabia

***Abstract*—Diabetic retinopathy (DR) is among the most dangerous diabetic complications that can lead to lifelong blindness if left untreated. One of the essential difficulties in DR is early discovery, which is crucial for therapy progress. The accurate diagnosis of the DR stage is famously complicated and demands a skilled analysis by the expert being of fundus images. This paper detects DR and classifies its stage using retina images by applying conventional neural networks and transfer learning models. Three deep learning models were investigated: trained from scratch CNN and pre-trained InceptionV3 and EfficientNetsB5. Experiment results show that the proposed CNN model outperformed the pre-trained models with a 9 to 25% relative improvement in F1-score compared to pre-trained InceptionV3 and EfficientNetsB5, respectively.**

***Keywords*—CNN; convolutional neural networks; deep learning; transfer learning; medical imaging; diabetic retinopathy; retina fundus images**

I. Introduction

Diabetic retinopathy (DR) is one of the diseases associated with diabetes and causes blindness to 4.4 million Americans over age 40 [1]. DR is an eye condition developed quickly in diabetes mellitus patients in type 1 or 2 [2]. DR often has no obvious symptoms in the early stages, but it becomes more pronounced as the disease progresses to more severe stages. An experienced ophthalmologist schedules a plan, which may run from weeks to months, to examine diabetic patients to determine their stage based on the retina's lesions and the severity level. Essentially, DR affects light-sensitive tissue blood vessels (i.e., retina) [3]. DR can be either nonproliferative or proliferative. In nonproliferative DR (NPDR), no abnormal blood vessel growth is found in the retina. Still, small outpouchings exist as the wall of retinal capillaries is weakened due to high blood glucose. These outpouchings are known as microaneurysms. NPDR can be mild, moderate, or severe based on the number of found microaneurysms and distortion of the blood vessels in the retinal exam. As the disease progresses, blood vessels may grow abnormally covering the retina; hence, DR becomes proliferative (PDR), leading to severe visual consequences.

In preventing blindness caused by the DR, detection, diagnosis, and treatment in earlier stages will control the disease and reduce vision loss. Diagnosis of the DR is complicated and requires high potential abilities [4]. One well-known obstacle for DR is that even for diabetic macular oedema, there are no early warning signs. Therefore, it is highly desirable to detect DR on time. Currently, DR diagnosis needs a well-trained doctor to diagnose the disease and manually evaluate digital images of the fundus of the retina. DR is recognised through identifying lesions connected with vascular malformations resulting from diabetes. This process may require longer time and effort depending on the experience and efficiency of the examiner doctor.

With the recent advancements in intelligent solutions, deep learning and transfer learning techniques showed significant success in object recognition and detection tasks. This research aims to automate DR diagnosis through exploiting convolutional neural networks (CNN) and transfer learning to identify DR from retina images. The Asia Pacific Tele-Ophthalmology Society (APTOS) dataset was used for blindness diagnosis and detection in this research. In addition, a comparison of the evaluation of different models to detect the disease effectively. This intelligent solution would help the health community diagnose the disease more efficiently, using less time and resources.

The remainder of this paper is organised as follows. In Section II, the related studies on the topic have been reviewed. Section III presents the research models used in this study followed by a description of the dataset used in the diagnosis DR in Section IV. Section V lays evaluation metrics and experimental design. Section VI presents the results of the experiments, and finally, the study is concluded in Section VII.

II. Related Work

Early DR detection is critical and time-consuming, and ophthalmologists are burdened. This attracted many researchers to develop early DR detectors and classifiers. Here, an overview of the deep learning techniques used in the previous literature is presented. Also, the used DR datasets in those studies are summarised. All of the reviewed literature detected DR from retinal fundus images. If detected, DR was classified into one of four severity levels: mild, moderate, severe NPDRs and PDR.

In the deep learning approach, CNN extracts features from input images and feeds them to the deeper layer in the model. Shan et al. [19] distinguished microaneurysms from fundus images using stacked sparse autoencoder (SSAE). Their model reached 91.3% for F1-score and an AUC of 96.2%. Singh et al. [20] employed a densely connected neural network architecture to detect the DR severity efficiently. Experimental findings showed that the DR severity could be successfully identified through the model with an accuracy of 83.6%.

Some researchers fine-tuned pre-trained models, known as transfer learning (TL), instead of training their models from scratch. These pre-trained models were initially trained using a large amount of out-of-domain data for object recognition and

TABLE I. PREVIOUS WORK IN EARLY DR DETECTION USING TRANSFER LEARNING. PREPROCESSING COLUMN INDICATES WHETHER ANY IMAGE PROCESSING WAS APPLIED BEFORE MODEL FINE-TUNING.

Work	Preprocessing	Pre-trained models	Dataset	Performance
Nquyen et al. [5]	Highlight spot, Crop, Drop outliers, Convert B/W, Rotate tree, Special Filtering	VGG-16, VGG-19	EyePACS	(Ensemble) Accuracy 82% Sensitivity 80% Specificity 82%
Masood et al. [6]	Downsize to a common radius, Normalize, Crop a borders	InceptionV3	EyePACS	Accuracy 48.2%
Maswood et al. [7]	Ben's preprocessing	EfficientNet-B5	APTOS2019	Accuracy 93%
AbdelMaksoud et al. [8]	Filtering using median filter, Resize into 256 × 256, Transformation Processes, Normalize	EffecientNet-B0	IDRiD	Accuracy 86%
Qummar et al. [9]	Resize into 786 × 512, Mean normalized	Resnet50, Inceptionv3, Xception,Dense121, Dense169	EyePACS	(Ensemble) Accuracy 80.8% Sensitivity 51.5% Specificity 86.7%
Gao et al. [10]	Remove black borders, Resize into 672 × 672	(modified) MobileNet-Dense, MobileNetV2	MESSIDOR, EyePACS	(Ensemble) Accuracy 96.2%
Taufiqurrahman et al. [11]	Resize 224 × 224	MobileNetV2-SVM, MobileNetV2	APTOS2019	(MobileNetV2-SVM) Accuracy 85%
Khojasteh et al. [12]	Patch Exraction	SVM/KNN/OPF, DRBM,ResNet-50	DIARETDB1, e-Ophtha	(ResNet-50-SVM) Accuracy 98.2% Sensitivity 99% Specificity 96%
Hemanth et al. [13]	Histogram equalisation	—	MESSIDOR	Accuracy 97% Sensitivity 94% Specificity 98%
Kose et al. [14]	Kirsch's template	—	MESSIDOR	Accuracy 89.8% Sensitivity 79.6% Specificity 93.2%
Pham et al. [15]	Subtracting the average local colour using a Gaussian mask	EfficientNet-B5	APTOS2019	—
Shankar et al. [16]	Histogram based segmentation	ResNet50	MESSIDOR	Accuracy 99.28% Sensitivity 98% Specificity 99%
Jiang et al. [17]	—	Inception V3, Resnet152, Inception-Resnet-V2	Beijing Tongren Hospital	(Ensemble) Accuracy 88.21% Sensitivity 85.57% Specificity 90.85%
Tymchenko et al. [18]	—	EfficientNet-B4, EfficientNet-B5, SE- ResNeXt50	MESSIDOR, APTOS2019, EyePACS, IDRiD	(Ensemble) Accuracy 99% Sensitivity 99% Specificity 99%

detection. Then, only the output layer is replaced according to the given task and number of classes. Table I lists some of these studies along with the used pre-trained models and their performance. Whenever a study investigated more than one pre-trained model, an ensemble was applied to combine all these models and produce an optimal model.

Traditionally, the output layer of pre-trained models is replaced by a multi-layer neural network classifier and a softmax layer with a size equivalent to the number of classes to be recognised. Nevertheless, Taufiqurrahman et al.[11] suggested restructuring the MobileNetV2 model by replacing the fully connected layer with a Support Vector Machine (SVM) classifier. This modified version, MobileNetV2-SVM, obtained better performance than its original model. MobileNetV2-SVM achieved an accuracy of 85% and AUC of 92.8%. In a similar fashion, Khojasteh et al.[12] replaced the softmax layer with several classifiers: OPF, SVM, and KNN. Combing Resnet-50 and SVM outperformed other models with an accuracy of 98.2%, a sensitivity of 99%, and a specificity of 96%.

Several models can be combined using ensemble learning to improve prediction performance or reduce the bias in the learning process. Jiang et al. [17] introduced an image-based method to detect the DR early using an interpretable ensemble deep learning model. The proposed model is working on three main steps. Firstly, the fundus images preprocessing. Secondly, three different deep learning models have been used independently and trained sufficiently: Inception V3, Resnet152, and Inception-Resnet-V2. Finally, the Adaboost optimiser algorithm combined all the models' results to generate the final score. The integrated model proved a high performance in all evaluation metrics used: sensitivity, specificity, accuracy, AUC, 85.57%, 90.85%, 88.21%, and 0.946, respectively. Also,

Tymchenko et al. [18] developed a DR detector using three-head CNN, which trained classification, regression and ordinal model. They used the output of these three heads for DR detection and achieved the sensitivity and specificity of 0.99.

As in the research, [17], [8], [6], [11], [7], this research aims to use InceptionV3, CNN, EfficientNetsB5 to detect DR due to their efficiency previous studies. However, these models will be validated using the same dataset to compare their results. Besides handling the issue of imbalanced distribution of classes on APTOS 2019 dataset, that not highlighted in previous researches.

III. Methodology

The purpose of this research is to classify a retinal fundus image whether it has a DR and at which severity. According to previous literature, deep learning and transfer learning models can solve this task. Transfer learning is a method that allows using the knowledge gained from other tasks to tackle new similar problems quickly and effectively. Hence, CNN models that are pre-trained will be fine-tuned utilizing domain dataset. Two pre-trained models will be selected for this study, InceptionV3 and EfficientNetsB5, for their effectiveness in diagnosing DR in the work of [17], [8], [6], [11], [7]. The performance of the fine-tuned pre-trained models will be compared with the performance of a CNN without pre-training.

A common issue in medical imaging datasets is the disparity in the number of samples within classes due to the difficulty of obtaining such samples. This problem is known as the class imbalance, and it pushes classifiers to prefer classes with higher training samples, reducing classification performance.

This section describes the techniques mentioned above.

A. Convolutional Neural Network (CNN)

Deep neural networks are artificial neural networks with more hidden layers to perform more complicated tasks and deal with massive amounts of data. The convolutional neural network (CNN) is one of the deep learning model networks with multiple layers such as convolution, pooling, fully connected, and non-linearity layer. CNN has been used with many applications, especially those that deal with spatial information, such as document analysis, image and video recognition, and computer vision [21]. The main aim of CNN is to increase or decrease the image dimensions into a more manageable form and extract the significant features, then process it to provide better predictions.

In this study, three convolution layers were employed with the same kernel size of (3,3). ReLU is used as an activation function with all layers, followed by a max-pooling layer with (2,2) pooling size to reduce the size of the large images. The results were flattened before the fully connected layer with a dropout of 0.2 to avoid overfitting. A softmax activation layer was used as the output layer. The architecture of the model is shown in Fig. 1. Some of the model's configurations were based on the work of [22], [23], [24].

B. Pre-Trained CNN Models

It has become customary to utilize a pre-trained CNN model and fine-tune it with in-domain dataset for the majority of computer vision applications. A pre-trained CNN model is a CNN model that has been trained on a large volume of data, such as ImageNet, for image classification [25]. Two pre-trained CNN models will be investigated in this paper: InceptionV3 and EfficientNetsB5.

Inception-v3 is a CNN architecture from the Inception family that contains 48 deep layers. Inception is characterised by implementing multiple kernels of different sizes in each layer (means become wider) instead of increasing the number of layers and going deeper in the network [26]. Each unit consists of four parallel operations: 1×1, 3×3, 5×5 conv layers and 3×3 max-pooling. All feature maps that come from different paths are concatenated together as the input of the next layer. Because in the image classification, the feature size of the image can diversify and deciding a fixed kernel size is difficult. Lager kernels are effective when the features are distributed over a wide area in the image. On the contrary, smaller kernels are useful and give excellent results in detecting small areas distributed across the image frame. To effectively recognise this variable size feature, kernels of different sizes are needed, which are provided in Inception models [27], [28].

EfficientNets family has a highly significant performance that achieves state-of-art on ImageNet, CIFAR-100, Flowers, and three other transfer learning datasets [29]. The architecture of the EfficientNets model involves convnet designs to reduce the space of the model with each layer to be scaled uniformly with a constant ratio to optimise the accuracy performance. It focuses on three aspects of scaling width, depth, and resolution. According to that, the EfficientNets family produces seven models with different image dimensions, and there is no change of layers operator of baseline network. This research proposes to apply EfficientNets B5 version.

C. Data Augmentation

Many approaches have been proposed to overcome the imbalanced dataset problem that can be classified into two categories: creating algorithms to resample the data and data preprocessing to generate new samples [30]. Resampling a dataset is a method used to balance the class distribution of the dataset. This is achieved by either adding samples to the minority class (oversampling) or removing samples from the majority class to balance the data (undersampling) [31]. However, data augmentation is a common technique used to generate new samples of the data to provide the image in a different representation.

Data augmentation techniques help improve the deep learning model's ability by generating artificial new images to achieve high variation in the training dataset and avoid overfitting problems. Many transform operations could be applied for data augmentation, such as random rotation, brightness, zoom, and image preprocessing techniques, such as Gaussian blur or CLAHE [32]. The data augmentation techniques included in this research are horizontal and vertical flip, zoom, and rotation. Fig. 2 shows a sample image from class 0 augmented after preprocessing image phase.

IV. Diabetic Retinopathy Dataset

Several datasets are available for the retina to detect DR and the vessels. Often these datasets are utilised for training,

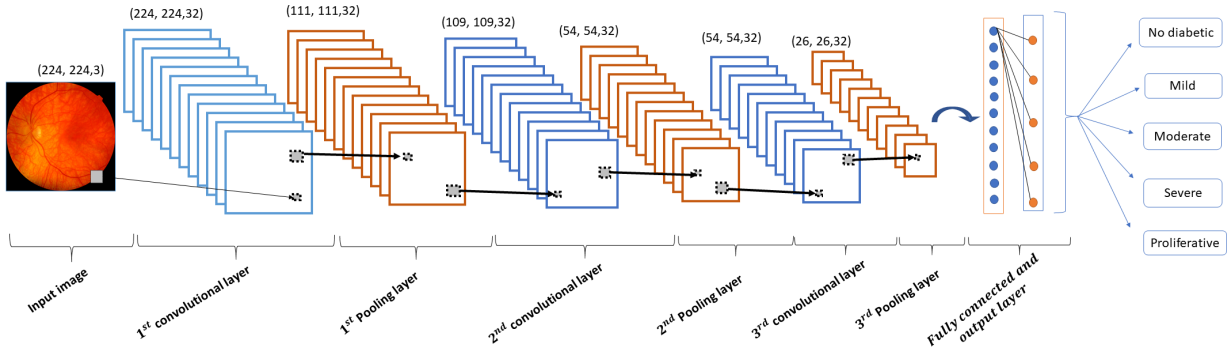


Fig. 1. CNN Module Architecture used in this Study.

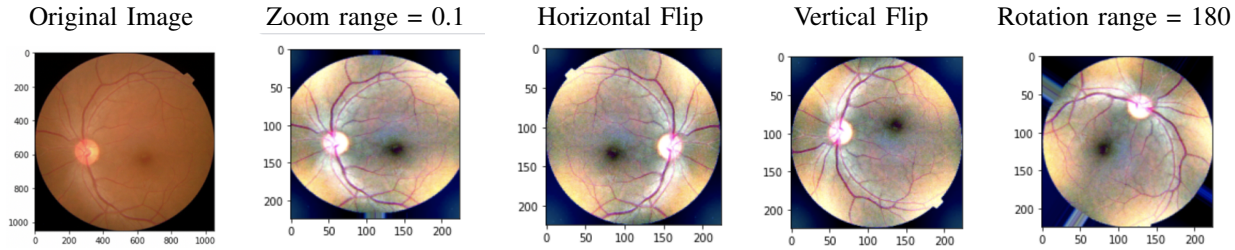


Fig. 2. Applying Data Augmentation Techniques in a Sample of Retinal Fundus Image.

validation, and testing deep learning models. The Asia Pacific Tele-Ophthalmology Society (APTOS) published this dataset in the second quarter of 2019. As shown in Table I, several studies used the APTOS2019 dataset [33] for blindness detection, containing a large set of retina images taken using fundus photography. Initially, two sets were published: labelled images, known as the train set, and unlabelled images, known as the test set. Only the labelled images were included in this study, consisting of 3662 fundus images. Each image was labelled into one of five classes, representing the severity of DR. Table II shows samples of each class and its characteristics that differentiate it from the others. As many of the medical dataset, APTOS2019 suffers from class imbalance as shown in Fig. 3 with majority of the cases towards healthy images without DR. However, there is a balance between the sum of all DR images regardless of their severity and healthy images.

The image size is a more critical factor that will impact the classification tasks. As shown in Fig. 4, there is a different distribution of image height and width, which suggests that not all images are in a perfect square shape.

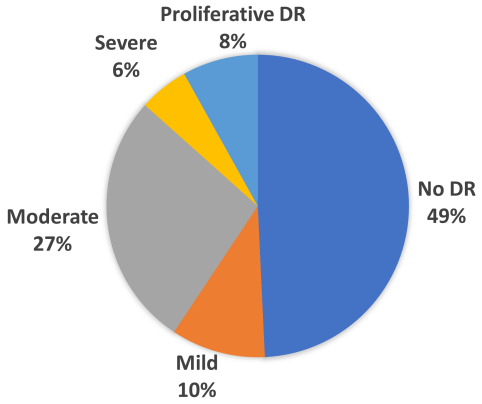


Fig. 3. Class Distribution among APTOS2019 Training Set.

V. Experiments

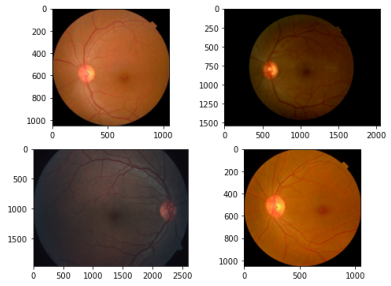
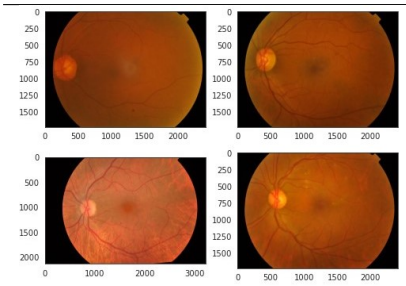
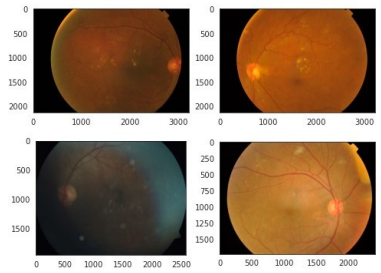
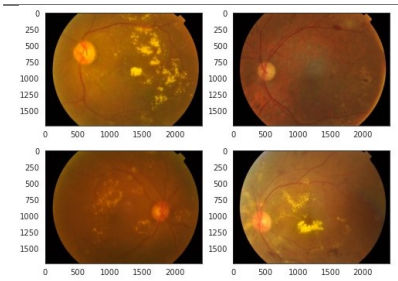
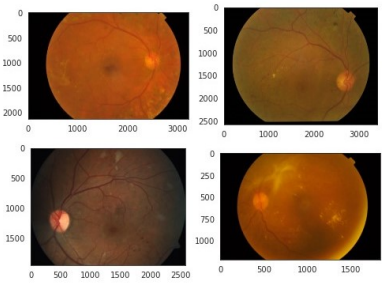
A. Experimental Design

All experiments were implemented and evaluated using Python [34] and leverage TensorFlow and Keras library [35] using Kaggle GPUs, Kaggle presents free access to NVidia K80 GPUs in kernels. In particular, these GPUs can be used to train deep learning models [33]. For this study, the labelled set was split into three homogeneous sets: training, validation and testing sets with a ratio of 68%, 20%, and 12%, respectively. The distribution of classes within each split is shown in Table III. Two sets of experiments were performed: fine-tuning and training using an imbalanced training set, 2489 samples, and a balanced training set after augmentation, 6158 samples.

B. Data Preprocessing

As most of the pre-trained models in this study were trained using images of size 224×224, images of APTOS2019 were rescaled accordingly. Moreover, images were converted into grayscale, which increases the visibility of some abnormalities. Following [18], [7], further image processing processes were applied: uninformative black areas were removed using circular crop, blending using Gaussian blur with alpha=4, beta=-4,

TABLE II. LABEL DESCRIPTION AND CHARACTERISTICS OF DR SEVERITY LEVELS WITH ITS SAMPLES FROM APTOS2019

Sample	Characteristics
	**Label 0: No diabetic retinopathy (NoDR)**
	**Label 1: Mild nonproliferative retinopath**: In this early stage of the disease, small patches of balloon-like swelling in the small blood cells in the retina, known as microaneurysms. The fluid will leak into the retinas through these microaneurysms as shown in the left images.
	**Label 2: Moderate nonproliferative retinopathy**: As the disease progresses, Blood vessels feeding the retina may swell and distort and also lose blood transportation capacity. These conditions cause significant changes to the appearance of the retina and can contribute to diabetic macular edema (DME) as shown in the left images.
	**Label 3: Severe nonproliferative retinopathy**: Many further blood vessels are blocked, which deprive the retinal region of blood supply. These regions secrete growth factors that suggest that the retina is forming new blood vessels as shown in the left images.
	**Label 4: Proliferative diabetic retinopathy (PDR):** This more serious form called proliferative diabetic retinopathy. Damaged blood vessels are blocked in the retina in this case, causing the development of irregular new blood vessels, and can flow into the clear, jelly-like substance that fills the center of the eye (vitreous). Scar tissue stimulated through new blood vessel growth can gradually separate the retina from the posterior of the eye. Therefore, retinal detachment could lead to permanent eyesight loss as shown in the left images.

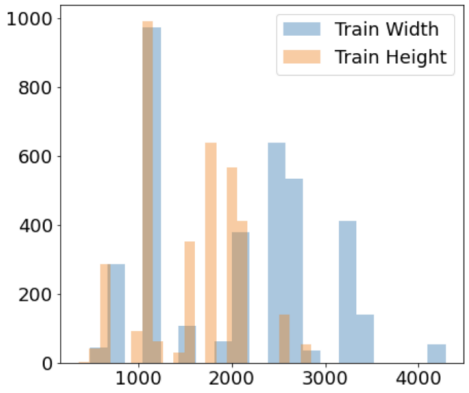


Fig. 4. Distribution of Image Width and Height in APTOS2019.

TABLE III. THE SPLIT OF APTOS2019 TRAIN SET USED IN THIS STUDY AND ITS CLASS DISTRIBUTION.

Class	Training set	Testing set	Validation set
No DR	1229	217	359
Mild	265	43	62
Moderate	670	124	205
Proliferative DR	132	25	36
Severe	193	31	71
Total	2489	440	733

and gamma=128. Consequently, the resulting images are not entirely greyscaled as modifications were applied separately on every pixel's colour channel. This helps improve the blood vessel's visibility and its growth in the eye, as shown in Fig. 5. All image preprocessing techniques was applied using Python (cv2) OpenCV library [36], [37].

C. Data Augmentation

Data augmentation was implemented using 'ImageDataGenerator' class from Keras library [35]. As shown in Fig. 3, the number of cases in each category varies significantly, with no_DR as the majority class (49.3% of total images). The number of the augmented images is different based on the number of the original images, as shown in Fig. 6. The augmentation phase enriched the diversities of the classes to provide high-quality images to the learning models. This operation was performed only on the training dataset. Image augmentation for the minority classes was applied via zooming, flipping, and rotation, which acquired a dataset three times larger than the original set.

D. Fine-Tuning the Pre-Trained CNN Models

For every pre-trained model included in this study, the input layer was set to be 224×224 and three channels. However, the output layer was modified to match the number of classes in this task, i.e. five classes. Then, all layers were frozen during the fine-tuning process except for the modified last layers. The last layers were trained using Adagrad optimiser with a learning rate of 0.01 and for 30 epochs. Similar training configurations were employed when training CNN model.

E. Evaluation Metrics

A multi-class classification task necessitates factors such as class balance and expected outcomes when picking the optimal

TABLE IV. EVALUATION METRICS AND THEIR FORMULAS.

Metric	Description	Formula
Accuracy	The average number of correct predictions.	$Acc = \frac{\sum_{i=1}^{C} \frac{TP_i+TN_i}{TP_i+FN_i+FP_i+TN_i}}{C}$
Precision	Capability of identifying the correct instances for each class.	$Pre = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} TP_i+FP_i}$ / C
Recall	Capability to recognise the true positive out of the total true positive cases.	$Rec = \frac{\sum_{i=1}^{C} TP_i}{\sum_{i=1}^{C} TP_i+FN_i}$ / C
F1-score	The harmonic average of precision and recall.	$F1 = 2 * \frac{Pre_M * Rec_M}{(Pre_M + Rec_M)}$

TABLE V. RESULTS WHEN TRAINING MODELS USING THE IMBALANCED DATASET.

	Precision	Recall	F1-score	Accuracy	kappa
	Training set				
CNN	61%	69%	69%	67%	61%
Inceptionv3	84%	71%	74 %	88%	84 %
EfficientNet B5	35%	34%	32%	62%	29%
	Validation set				
CNN	58%	65%	61%	65%	58 %
Inceptionv3	51%	51%	51%	76%	70%
EfficientNet B5	40%	36%	34%	64%	30%
	Testing set				
CNN	64%	71%	67%	73%	65%
Inceptionv3	62%	53%	54%	78%	72%
EfficientNet B5	30%	34%	31%	63%	30%

metrics to evaluate the performance of a particular classifier against a given dataset. One performance metric may assess a classifier from a specific perspective while others can not, and vice versa. Hence, there is no standardised (unified) metric for defining the generalised performance measurement of the classifier. In this paper, several metrics are chosen to measure the models' performance: Accuracy, Precision, Recall and F1-score. Table IV. summarises how each of the first four metrics is calculated for a multi-class classifier with C classes, where TP_i and TN_i are the number of cases correctly diagnosed for class C_i or not, respectively. And FP_i and FN_i are the number of cases that were incorrectly diagnosed to the class C_i or not, respectively.

As one of the experiments uses an imbalanced dataset, Cohen's kappa was used as an additional metric. It can be computed as follows:

$$K = \frac{P_0 - P_e}{1 - P_e},$$

where P_0 denotes the overall accuracy and P_e denotes a measure of the probability of the agreement between the prediction class values and the actual class values as it occurs by chance [38]. $K = 1$ if classes are in complete agreement while $K = 0$ proves the opposite.

VI. RESULTS AND DISCUSSION

A. Training with Imbalanced Dataset

Each pre-trained model was fine-tuned using the imbalanced training set, with 2489 samples and no_DR as the majority class. When training the CNN model from scratch, the same imbalanced set was used for training. Table V. lists the results obtained during models training. Since accuracy is

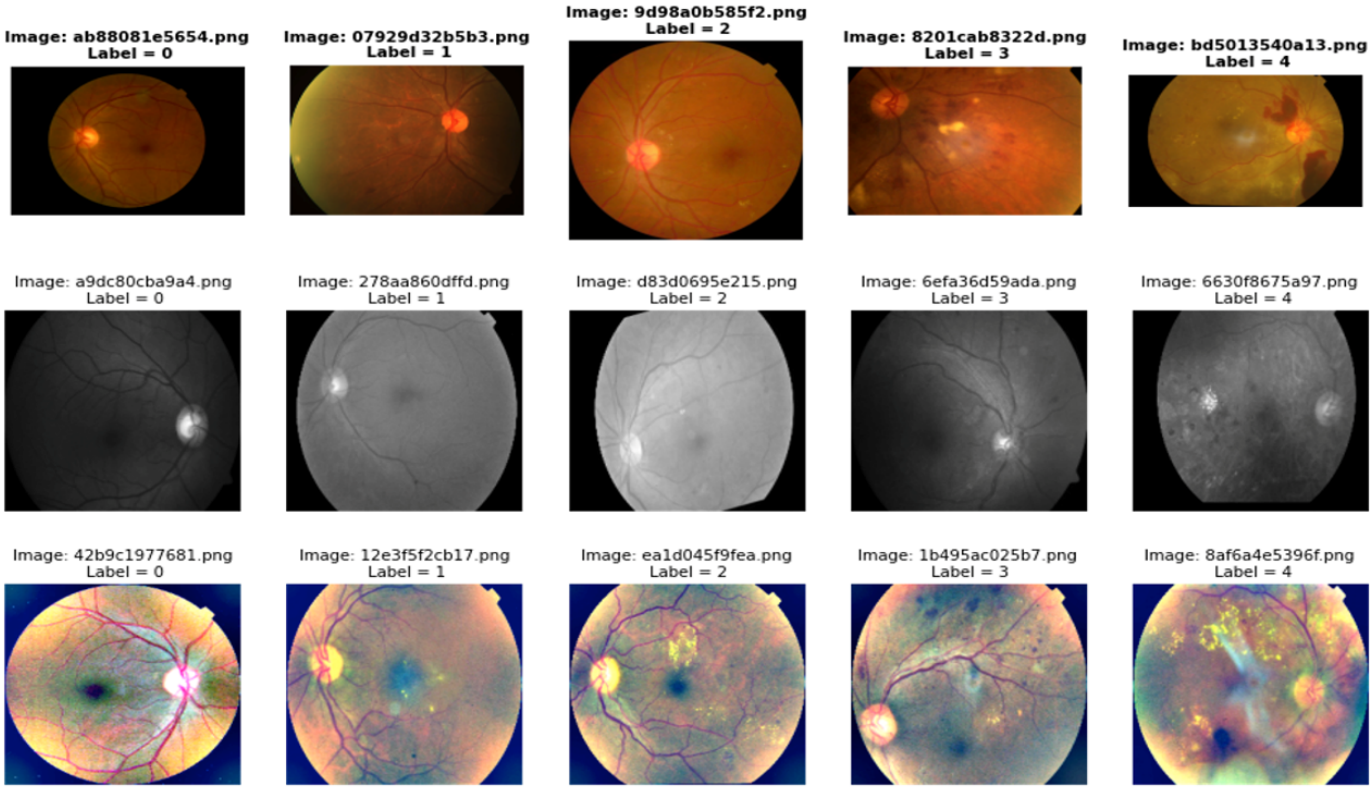


Fig. 5. Applying Image Preprocessing Techniques on Some Samples from APTOS2019. The First row Shows the Original Sample from each Class. The Second Row Shows the Same Samples after Converting them into Grayscale using the cvtColor Function and COLOR_BGR2GRAY as a Parameter. The Third Row Shows the Same Samples after Applying Gaussian Blur and Circular Cropping.

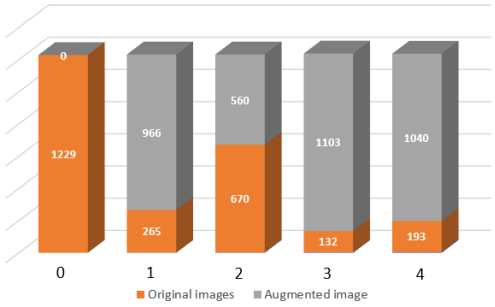


Fig. 6. Samples Per Class in APTOS2019 Training Set. Class 0 (no_DR) is the Majority Class so Other Classes were Augmented with Different Numbers of Samples to Obtain a Balanced Training Set.

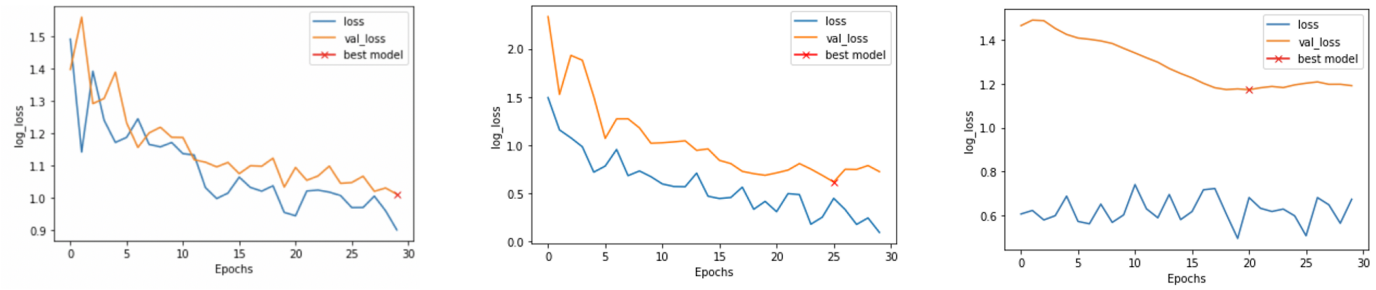
unreliable when evaluating models trained on an imbalanced dataset, F1-score and kappa are the primary evaluating metric. CNN model achieves the highest F1-score with 67%, while the InceptionV3 model obtained 54%. On the other hand, the EfficientNetB5 model has the lowest performance.

To investigate the reasons for EfficientNetB5 performance, the learning curve for each of these models are depicted in Fig. 7. As shown in the figures, the learning curves of the CNN and InceptionV3 model in training and validation phases was improving smoothly, while the EfficientNetB5 model suffered from a high overfitting problem, which caused its low results.

Fig. 8 visualises the confusion matrix of these models, which indicates the number of predictions produced by the model where it classified the classes correctly or incorrectly. The diagonal expresses the correctly diagnosed states for each class, where the off-diagonal elements represent the misclassified samples. In general, all models have their best recognition for Class 0 (no_DR) and 2 (mild NPDR) aligned with the class majority shown in Fig. 3. with Class 0 and 2 with the largest samples, respectively. However, most confusion was between different classes of DR, not no_DR and any DR. This observation was accurate for all models. In other words, these models have good DR detection but poor severity level classification. The detection rate can be calculated by mapping all DR severity levels 1-4 to 1. Hence, the obtained detection rates are 90%, 96% and 83% for CNN, InceptionV3 and EfficientNetB5, respectively.

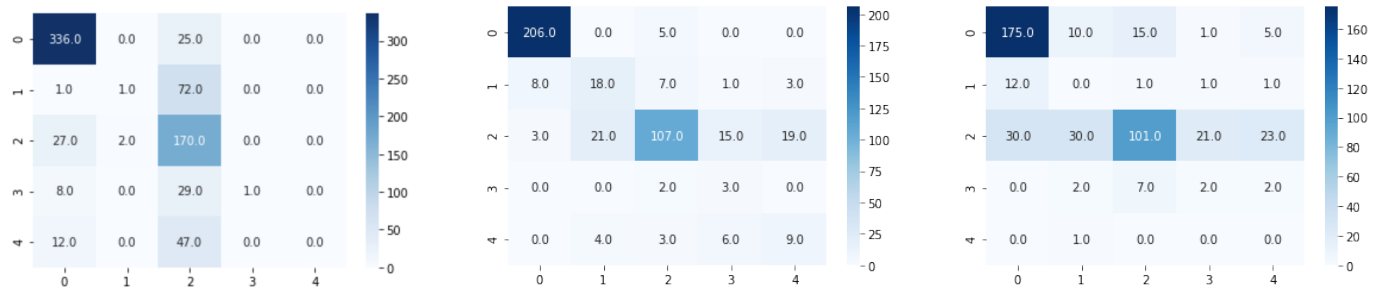
B. Training with Balanced Dataset

In this experiment, pre-trained models were fine-tuned using the balanced training set via augmentation, with 6158 samples. The same set was used for training when training the CNN model from scratch. Table VI lists the results obtained during models training. CNN model achieves the highest F1-score with 64%, while the InceptionV3 model obtained 58%. On the other hand, the EfficientNetB5 model has the lowest performance with a 48% F1-score. Looking at the learning curves for these models in Fig. 9. the performance of the validation set improved better than the training set for the CNN



(a) CNN (b) InceptionV3 (c) EfficientNetB5

Fig. 7. Learning Curves for the Models Trained using the Imbalanced Dataset using the Train Set (2489 Samples) and the Validation Set (733 Samples).



(a) CNN (b) InceptionV3 (c) EfficientNetB5

Fig. 8. The Confusion Matrix when Evaluating on the Test Set (440 Samples) for the Models Trained using the Imbalanced Dataset. The x-Axis Represents the Actual Labels, while the y-axis Represents Predicted Labels. Label 0: no_DR, Label 1: mild NPDR, Label 2: Moderate NPDR, Label 3: Severe NPDR, and Label 4: PDR.

TABLE VI. RESULTS WHEN TRAINING MODELS USING THE BALANCED DATASET

	Precision	Recall	F1-score	Accuracy	kappa
	Training set				
CNN	50%	50%	49%	50%	28%
Inceptionv3	76%	75%	75%	75%	64%
EfficientNet B5	59%	59%	59%	59%	33%
	Validation set				
CNN	57%	65%	61%	65%	53%
Inceptionv3	64 %	60%	60%	79%	69%
EfficientNet B5	59%	47%	47%	74%	62%
	Testing set				
CNN	61%	68%	64%	68%	57%
Inceptionv3	59%	59%	58%	78%	70%
EfficientNet B5	57%	46%	48%	73%	60%

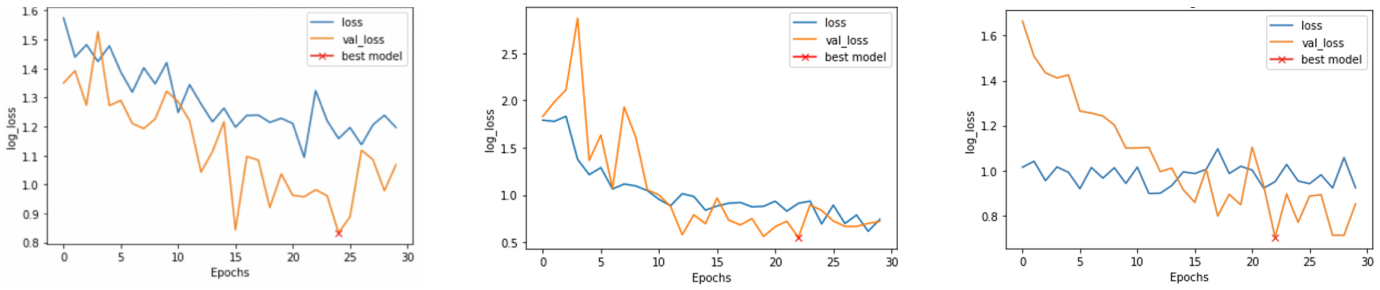
model, which indicates that some samples were difficult for the models to learn from the features. However, this was not observed for InceptionV3 and EfficientNetB5 models, which means the performance of training and validation sets were approximate are similar.

Fig. 10 visualises the confusion matrix of these models. In general, all models could not recognise Class 4 (PDR) successfully compared to other classes. As in the previous experiment, most confusion was between different classes of DR, not no_DR and any DR. The obtained detection rates here are 84%, 95% and 90% for CNN, InceptionV3 and EfficientNetB5, respectively.

This study performed two experiments; the first was on a dataset imbalanced between classes and only processed by scaling and resizing the image. The second experiment was on a balanced dataset by utilising augmentation data and applying image preprocessing techniques. F1-score was used to measure and compare the performance in both experiments because it is a standard measure of imbalanced data classification, in addition to the rest metrics mentioned in Section V-E. The performance was improved when using balanced since the InceptionV3 and EfficientNetB5 models obtained higher results. InceptionV3 model's performance improved in Recall and F1-score when using a balanced training set while the results of the CNN model decreased in all measures. On the other hand, the results of the EfficientNetB5 model improved in all metrics when using a balanced training set. Hence, fine-tuning pre-trained models could benefit from the augmented samples and enhanced features, which was not the case for the CNN model.

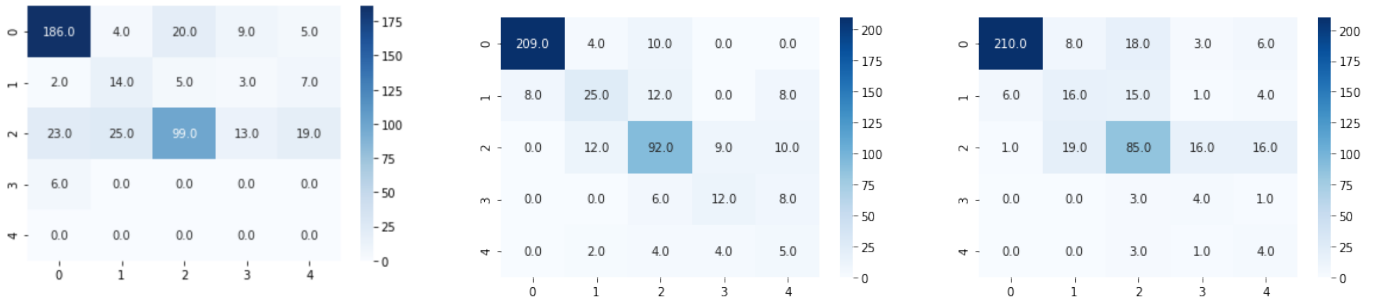
Furthermore, the CNN model achieves the highest results in the two experiments which are 67% and 64% of F1-score, in the first and second experiments, respectively. When looking at their learning curves, overfitting was an issue in pre-trained models, indicating the need for more powerful regularisation for these advanced architectures. In other words, the more complex the architecture, the more prone to overfit.

In general, the detection ability of these models was better than its classification between DR severity levels. For Efficient-NetB5, the DR detection was improved by 7% absolute when



(a) CNN (b) InceptionV3 (c) EfficientNetB5

Fig. 9. Learning Curves for the Models Trained using the Balanced Dataset using the Augmented Train Set (6158 Samples) and the Validation Set (733 Samples).



(a) CNN (b) InceptionV3 (c) EfficientNetB5

Fig. 10. The Confusion Matrix when Evaluating on the Test Set (440 Samples) for the Models Trained using the Augmented Balanced Dataset. The x-Axis Represents the Actual Labels, while the y-Axis Represents Predicted Labels. Label 0: no_DR, Label 1: mild NPDR, Label 2: Moderate NPDR, Label 3: Severe NPDR, and Label 4: PDR.

using a balanced training set, while it was the opposite case for CNN as its detection accuracy dropped by 6% absolute.

VII. Conclusion

DR is currently one of the dominant diseases that significantly affect people with diabetes. The paper covers the details of the implementation and evaluation of several deep learning models: CNN, InceptionV3, and EfficientNetsB5 for classifying DR using the APTOS2019 dataset. Two different experiments were conducted, the first with the original images and the second after processing the images and balancing the classes. The InceptionV3 model performed the best accuracy on the dataset in both experiments, while the CNN model got the highest F1-score in both experiments. Using these prediction results, effective DR detection systems can be implemented using deep learning models so that the patient can be treated and dealt with in the early stages. The results of this research may not be the same as previous research due to the difference in the dataset used and the data processing method. This research's main challenges and limitations are that the image dataset was imbalanced, and there was a shortage of efficiency of the devices utilised in processing even when using online GPU, such as Kaggle, the allotted time was limited. For further work, this research can expand to address these deficiencies by using other methods to balance data and apply other pre-trained models to diagnose DR.

Acknowledgment

The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: (22UQU4260137DSR02)

References

[1] Mayo Clinics. Diabetic retinopathy. [Online]. Available: https://www.mayoclinic.org/diseases-conditions/diabetic-retinopathy/symptoms-causes/syc-20371611, [Accessed: May 25, 2021]

[2] L. Chen, D. J. Magliano, and P. Z. Zimmet, "The worldwide epidemiology of type 2 diabetes mellitus—present and future perspectives," *Nature reviews endocrinology*, vol. 8, no. 4, p. 228, 2012.

[3] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros *et al.*, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *Jama*, vol. 316, no. 22, pp. 2402–2410, 2016.

[4] NHS. Overview-Diabetic retinopathy. [Online]. Available: https://www.nhs.uk/conditions/diabetic-retinopathy/, [Accessed: 2022-Jan-01]

[5] Q. H. Nguyen, R. Muthuraman, L. Singh, G. Sen, A. C. Tran, B. P. Nguyen, and M. Chua, "Diabetic retinopathy detection using deep learning," in *Proceedings of the 4th International Conference on Machine Learning and Soft Computing*, 2020, pp. 103–107.

[6] S. Masood, T. Luthra, H. Sundriyal, and M. Ahmed, "Identification of diabetic retinopathy in eye images using transfer learning," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2017, pp. 1183–1187.

[7] M. M. S. Maswood, T. Hussain, M. B. Khan, M. T. Islam, and A. G. Alharbi, "Cnn based detection of the severity of diabetic retinopathy from the fundus photography using efficientnet-b5," in *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 2020, pp. 0147–0150.

[8] E. AbdelMaksoud, S. Barakat, and M. Elmogy, "Diabetic retinopathy grading system based on transfer learning," *arXiv preprint arXiv:2012.12515*, 2020.

[9] S. Qummar, F. G. Khan, S. Shah, A. Khan, S. Shamshirband, Z. U. Rehman, I. A. Khan, and W. Jadoon, "A deep learning ensemble approach for diabetic retinopathy detection," *IEEE Access*, vol. 7, pp. 150 530–150 539, 2019.

[10] J. Gao, C. Leung, and C. Miao, "Diabetic Retinopathy Classification Using an Efficient Convolutional Neural Network," *Proceedings - 2019 IEEE International Conference on Agents, ICA 2019*, pp. 80–85, 2019.

[11] S. Taufiqurrahman, A. Handayani, B. R. Hermanto, and T. L. E. R. Mengko, "Diabetic retinopathy classification using a hybrid and efficient mobilenetv2-svm model," in *2020 IEEE REGION 10 CONFERENCE (TENCON)*, 2020, pp. 235–240.

[12] P. Khojasteh, L. A. Passos Júnior, T. Carvalho, E. Rezende, B. Aliahmad, J. P. Papa, and D. K. Kumar, "Exudate detection in fundus images using deeply-learnable features," *Computers in Biology and Medicine*, vol. 104, no. July 2018, pp. 62–69, 2019. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2018.10.031

[13] D. J. Hemanth, O. Deperlioglu, and U. Kose, "An enhanced diabetic retinopathy detection and classification approach using deep convolutional neural network," *Neural Computing and Applications*, vol. 32, no. 3, pp. 707–721, 2020. [Online]. Available: https://doi.org/10.1007/s00521-018-03974-0

[14] U. Kose, O. Deperlioglu, J. Alzubi, and B. Patrut, "Diagnosing diabetic retinopathy by using a blood vessel extraction technique and a convolutional neural network," *Studies in Computational Intelligence*, vol. 909, pp. 53–72, 2021.

[15] H. N. Pham, R. J. Tan, Y. T. Cai, S. Mustafa, N. C. Yeo, H. J. Lim, T. T. Do, B. P. Nguyen, and M. C. H. Chua, "Automated grading in diabetic retinopathy using image processing and modified efficientnet," in *International Conference on Computational Collective Intelligence*. Springer, 2020, pp. 505–515.

[16] K. Shankar, A. R. W. Sait, D. Gupta, S. K. Lakshmanaprabu, A. Khanna, and H. M. Pandey, "Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model," *Pattern Recognition Letters*, vol. 133, pp. 210–216, 2020. [Online]. Available: https://doi.org/10.1016/j.patrec.2020.02.026

[17] H. Jiang, K. Yang, M. Gao, D. Zhang, H. Ma, and W. Qian, "An interpretable ensemble deep learning model for diabetic retinopathy disease classification," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 2045–2048.

[18] B. Tymchenko, P. Marchenko, and D. Spodarets, "Deep learning approach to diabetic retinopathy detection," *arXiv preprint arXiv:2003.02261*, 2020.

[19] J. Shan and L. Li, "A deep learning method for microaneurysm detection in fundus images," in *2016 IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*. IEEE, 2016, pp. 357–358.

[20] A. Singh and W. Kim, "Detection of diabetic blindness with deep-learning," in *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2020, pp. 2440–2447.

[21] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*. Ieee, 2017, pp. 1–6.

[22] hman. How to add a reshape layer to the start of a pre-trained cnn. [Online]. Available: https://stackoverflow.com/questions/61742075/how-to-add-a-reshape-layer-to-the-start-of-a-pre-trained-cnn, [Accessed Feb 2021]

[23] P. Huilgol. Top 4 pre-trained models for image classification with python code. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/08/top-4-pre-trained-models-for-image-classification-with-python-code/, [Accessed Feb 2021]

[24] A. H. Inception model with custom input tensor. [Online]. Available: https://groups.google.com/g/keras-users/c/G9C55N7e8S4?pli=1/, [Accessed March 5, 2021]

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[26] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[27] A. Anwar. Difference between AlexNet, VGGNet, ResNet, and Inception. [Online]. Available: https://towardsdatascience.com/the-w3h-of-alexnet-vggnet-resnet-and-inception-7baaaecccc96, [Accessed: 2022-Jan-01]

[28] S.-H. Tsang. Review: GoogLeNet (Inception v1) Winner of ILSVRC 2014 Image Classification. [Online]. Available: https://medium.com/coinmonks/paper-review-of-googlenet-inception-v1-winnr-of-ilsvlc-2014-image-classification-c2b3565a64e7, [Accessed: 2022-Jan-01]

[29] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 10 691–10 700, 2019.

[30] D. Ramyachitra and P. Manikandan, "Imbalanced dataset classification and solutions: a review," *International Journal of Computing and Business Research (IJCBR)*, vol. 5, no. 4, pp. 1–29, 2014.

[31] V. Ganganwar, "An overview of classification algorithms for imbalanced datasets," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 4, pp. 42–47, 2012.

[32] J. Brownlee, "How to Configure Image Data Augmentation in Keras." [Online]. Available: https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/, [Accessed: 2022-Jan-01]

[33] A. P. T.-O. Society. APTOS 2019 Blindness Detection dataset. [Online]. Available: https://www.kaggle.com/c/aptos2019-blindness-detection, [Accessed: 2022 Jan, 01]

[34] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[35] F. Chollet *et al.* Keras. [Online]. Available: https://github.com/fchollet/keras, [Accessed: 2022 Jan, 01]

[36] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[37] Divakar. Crop black border of image using numpy. [Online]. Available: https://codereview.stackexchange.com/a/132934, [Accessed: Feb 25, 2021]

[38] M. Widmann. Cohen's Kappa. [Online]. Available: https://www.knime.com/blog/cohens-kappa-an-overview, [Accessed: 2022-Jan-01]