Diabetic Retinopathy Classification using Feature Extraction and Deep Neural Networks

Sneha Singh
snehasingh300197@gmail.com
PhD, NT

Vinay Ummadi
ummadi.vinay2000@gmail.com
MTech, SMST

April 16, 2022

Abstract

The early diagnosis of diabetic retinopathy(DR) has will help in saving patients vision. It to identify it, atleast yearly eye checkup is necessary and many patients are not able to do it due to lack of diagnosis centers and ophthalmologists. From a technical perspective, researchers attempt to solve this using automated algorithms. Given an retinal image as input the algorithm will identify the the severity of DR on scale of 0-4. Several algorithms involving different methodologies are developed in an effort to automate DR identification and relive stress on ophthalmologists. These are algorithms first extract the significant features from the available set of images and then feed those features to a classifier. The most recent algorithms use end-to-end differential algorithms, resulting in significant accuracy improvements. We will be attempting to solve DR classification using three different methods and compare the methods using quantitative results. The deep neural net based algorithms are the best out of all three with an accuracy of 99.5%.

1 Introduction

All people with diabetes, including those with Type I (juvenile onset) and Type II (adult onset), are at risk of developing diabetic retinopathy. The longer you have diabetes, the more likely you are to develop it. There is a genetic component to this disease, as well. Diabetic retinopathy can begin without any warning symptoms, which makes a yearly comprehensive eye exam even more critical for diabetics. In order to prevent vision loss, early diagnosis and treatment of diabetic retinopathy are essential. The risk of blindness can be reduced by 90% with timely treatment and follow- up care. During the early stages of the disease, no treatment is needed. As the disease advances, the following treatment options are available like laser surgery and intraocular injection. Hence it becomes important to detect it at very early stage. Diabetic retinopathy is major cause of blindness in India and accounts for 30% of DR cases in the world.

In this term project we are reviewing implementing the existing methods for DR recognition and classification. The task of classifying retinal images is not new and existing since 2000s. In the early years the resolution of the digital cameras is not good enough and also they are too expensive. Since the rise of the high resolution cameras and fast computer, this led to use more digital imaging in medical applications. Before the rise of deep learning, researchers developed retinal image analysis to classification and segmentation tasks. In retinal image classification task, it can be either simple binary classification saying that it is DR or Non DR or can be multi class classification task that result on of the following five classes No DR, Mild DR, Moderate DR, Severe DR and Proliferate DR. lets us first understand how clinicians classify retinal images based on their properties. In No DR case, none of the hemorrhages, clots, exudates are present. In mild DR, the blood vessels are slightly swelled and microanursyms are also present. In Moderate DR, to small bleads, exudates and cotton wool spots are present. In Severe DR case, in addition to moderate DR properties, significant damage of blood vessels is seen. In the most final Proliferate DR stage, blood vessels are completely damaged and tractional retinal detachment is seen.

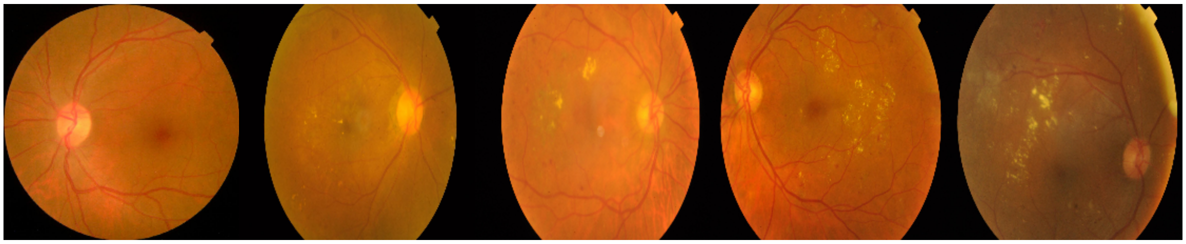
2 Objective

The brief goal of this project is to analyse the retinal images to recognise presence of diabetic retinopathy(DR) and its severity. We attempt to solve binary DR classification and multi class DR classification task.

3 Dataset

The dataset used in this sourced from kaggle [1] which is modifed version of APTOS 2019 Blindness Detection. The shape of the image is 224 x 224 x 3 (RGB) and number of images in each category are tabulated below. From the class wise count it is evident that dataset is classed imbalanced. No DR class is dominating followed by Moderate class. Sample images from each class are show in Figure 1.

Class	**Count**
No DR	1806
Mild	371
Moderate	1000
Severe	194
Proliferate	236
Total	3667



(a) No DR (b) Mild DR (c) Moderate DR (d) Severe DR (e) Proliferate DR

Figure 1: One randomly drawn sample image for each class.

It is clearly observable that hemorrhages, blood clots and exudates are progressively increasing from No DR to Proliferate DR.

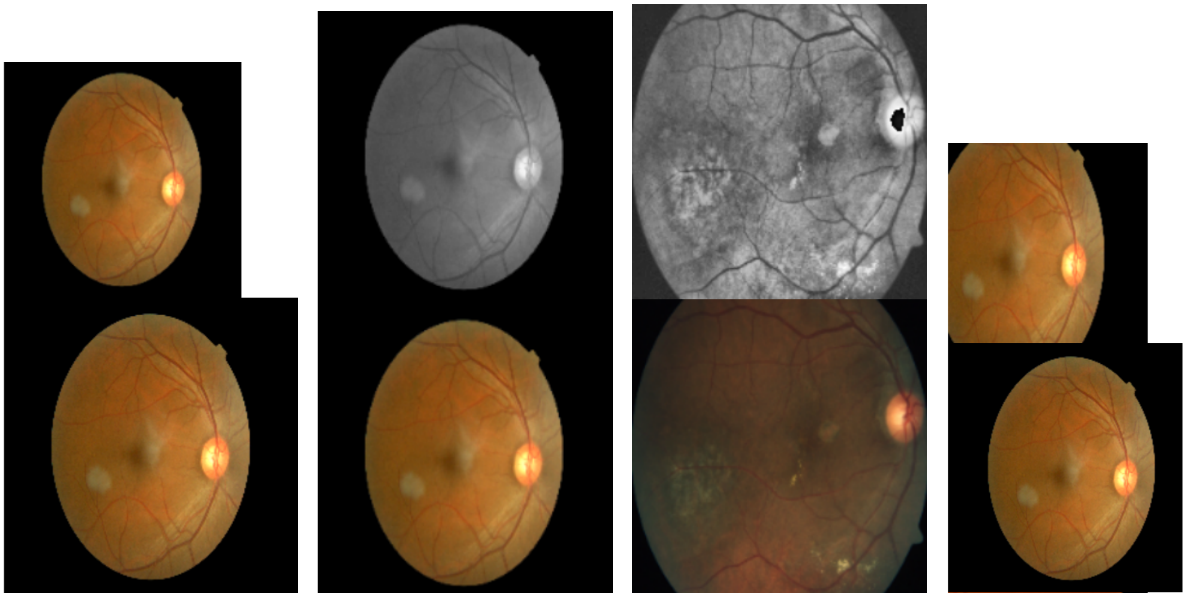
3.1 Data augmentation

For data augmentation, we are applying gray scaling, CLAHE histogram, gaussian noise, normalization, rotation of 90,rotation of 180, resizing and cropping to all the images.

4 Methodologies

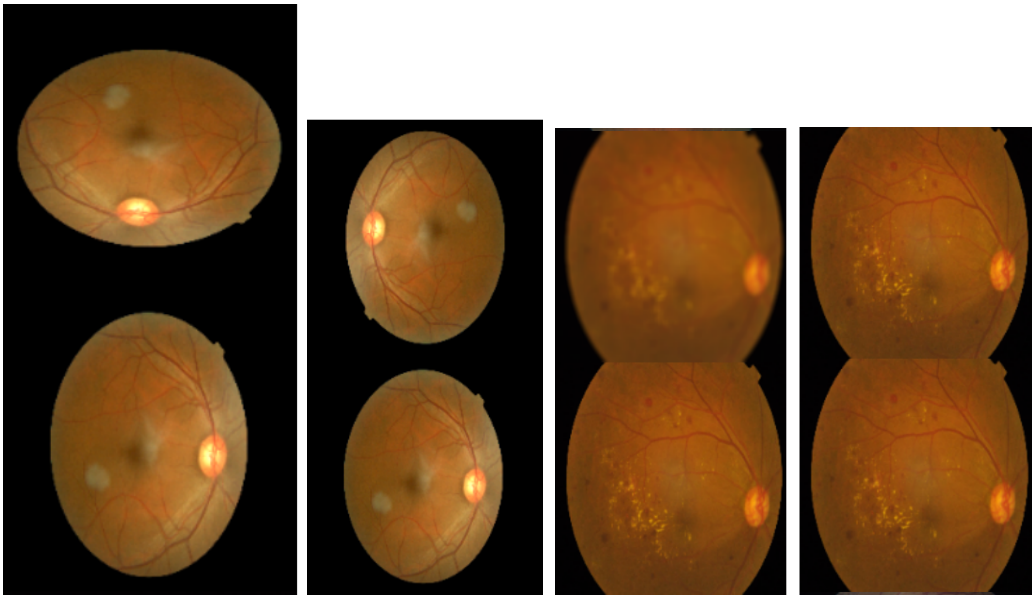
We approaching this problem in three different methods. First method starts with traditional image processing to segment the exudates followed by feature extraction and finally external classification. In the second methods hand crafted feature extraction is eliminated and features are extracted by a deep neural net and these are fed into external classifier. In the third method end-to-end classification network is used which eliminates external classifier. The described methods are detailed in the following sub sections.

To get moving with the problem, in the first phase the problem is considered as binary classification problem instead of multi-class classification problem. The four DR classes(Mild, Moderate, Severe, Proliferate) are merged into one meta class called DR. Now we are left with only classes No DR and DR with 1806 and 1861 images respectively. Later the harder multi-class classification problem is attacked.



(a) Resized Image (b) Gray Scale (c) Clahe Histogram (d) Crop

Figure 2: Data augmentation; Augmented image is shown top



(a) Rotate by 90 (b) Rotate by 180 (c) Guassian Noise (d) Normalize

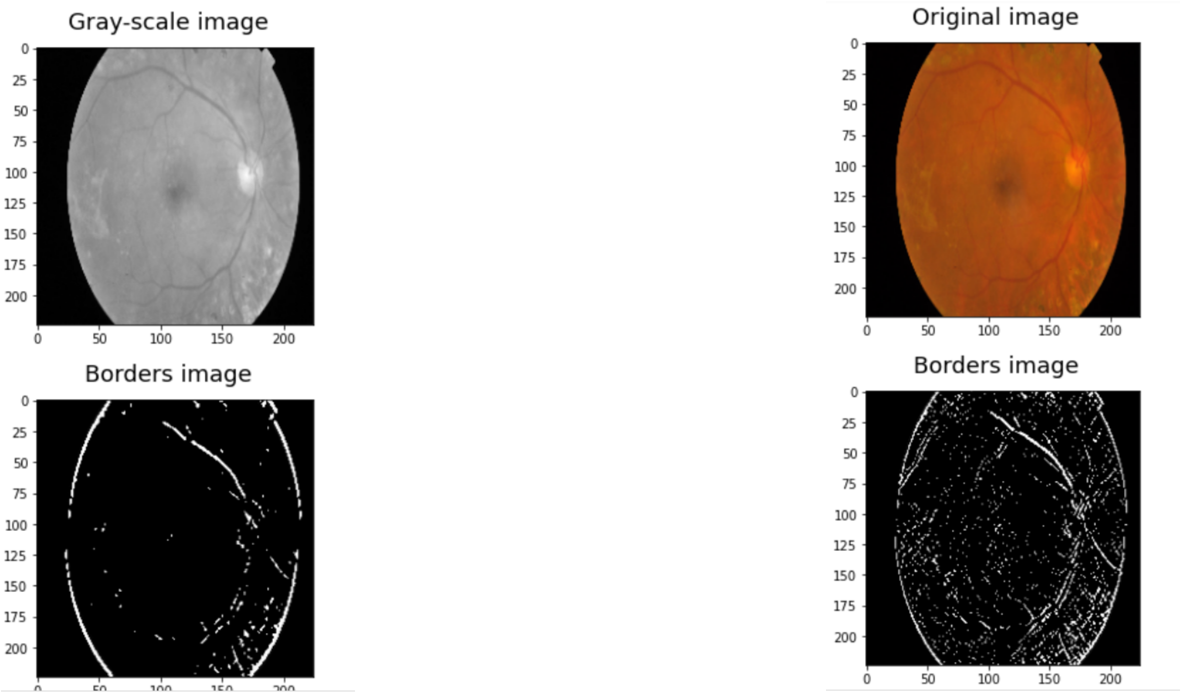
Figure 3: Data augmentation; Augmented image is shown top

4.1 Image processing based feature extraction + classification

In order to get the borders information from the fundus image, we convolve the gray image with the guass kernel with different sigma and n values. The image of which is shown below: We have also applied canny edges and then contour to the image whose result is shown below:

4.2 Deep neural network for feature extraction + classification

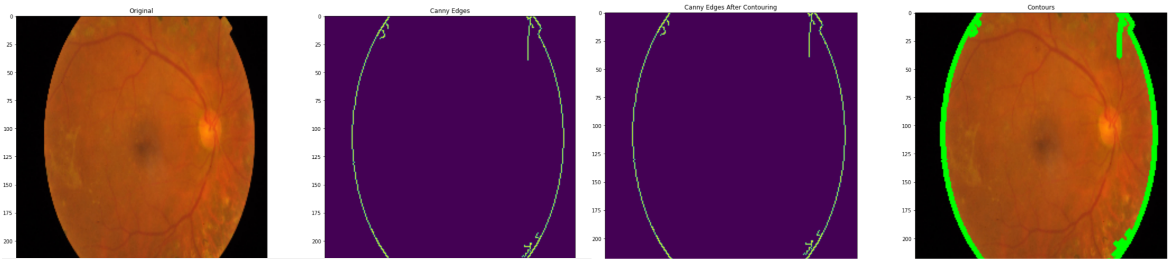
Firstly we started without data augmentation and we have used the InceptionV3 [2] for feature extraction and then gave it to various classifiers such as k-NN, SVM, Random forest classifier, adaboost



(a) With sigma = 1.08 and n =35

(b) With sigma = 0.5 and n=10

Figure 4: Border Images and Original Image



(a) canny edge without contour

(b) canny edge with contour

Figure 5: Canny Edge Detection

and MLP classifier. The Gaussian kernel is defined in 1-D, 2D and N-D respectively as

$$f(x,y) = A \exp\left(-\left(\frac{(x-x_0)^2}{2\sigma_X^2} + \frac{(y-y_0)^2}{2\sigma_Y^2}\right)\right).$$

Mobilenet is a model which does the same convolution as done by CNN to filter images but in a different way than those done by the previous CNN. It uses the idea of Depth convolution and point convolution which is different from the normal convolution as done by normal CNNs. This increases the efficiency of CNN to predict images and hence they can be able to compete in the mobile systems as well. Since these ways of convolution reduce the comparison and recognition time a lot, so it provides a better response in a very short time and hence we are using them as our image recognition model.

Later with the data augmentation, from 3667 samples we have increased it to 26675 samples. The same model is then again trained using various classifiers and the accuracy is improved. The accuracy results are listed below:

4.3 End-to-end deep neural network based classification

Starting 2012, Convolutional neural networks(CNNs) become the defacto method to solve computer vision problems. The CNN is a class of deep neural networks(DNNs), where convolution operation is performed in each layer. The term end-to-end means, input image is supplied to one end of the network and output is availed at other end of the network with need of any external modules. To train a any DNN end-to-end there are two primary requirements :

1. Sufficient training samples

2. Computation requirements considering upon model parameters and dataset size

The dataset size(3667) we have is not sufficient if we train the network from scratch. In such scenario the network will not converge well. So transfer learning is a technique, where pre-trained network is used and fine tuned to the targeted task. We trained our models from scratch and transfer learning, where the later one resulting in much better performance. For this task, five pre-trained networks are being used and appropriate comparison analysis is performed. The used network architectures briefly described below. All the five network architecture figures are not shown here to keep the report short and precise.

Network	**Short details**
VGG	VGG19 [3] is a standard 19 layer CNN for image classification and feature extraction. Batch normalization is added in every conv block.
ResNet	Residual network(ResNet) [4] is popular for stacking more layers by introducing skip connections from block to block. It usually performs better than VGG network. ResNet18 and ResNet50 are used in this project.
EfficientNet	EffiecintNet [5] achieves better performance by scaling the depth, width, resolution appropriately.
ConvNext	ConvNext [6] uses the best network design practices that outperform vision transformer to achieve SOTA results on vision tasks. We are using convnext-tiny with 3 convnext blocks.

4.3.1 Implementation details

The training recipe is as follows: Networks are trained using gradient decent based optimization algorithm called Adaptive momentum (Adam) with a learning rate of 0.0003 and trained for 10 epochs only. The lower number of iterations is due to computational constraints. The cross-entropy loss is used as error function to penalize the misclassifications. Cross entropy loss is defined as $H(p, q) = -\sum_{x\in\mathcal{X}} p(x)\,\log q(x)$. Images are sent in batches of 64 in each iteration. Horizontal flipping is added to transforms pipeline to allow model to learn rotation in-variance. All the networks are trained on Tesla T4 GPU (16GB memory) using training recipe mentioned.

5 Results

In this section we will present the quantitative and qualitative results for different methodologies described. The Table 1 shows the accuracy scores for the various classifiers without the data augmentation. After the data augmentation, the dataset is increased from 3667 to 29301 which is resulting into better accuracy and more dataset to learn on for feature extraction. The reason for the higher accuracy in case of Ada boost and XGB boost could be

Model	**Top 1 Accuracy**	**Precision**	**Recall**	**F1 Score**
KNN	0.7822	0.78	0.78	0.78
SVM	0.822	0.82	0.82	0.82
Random Forest	0.846	0.84	0.84	**0.84**
XGB	0.842	0.84	0.84	**0.84**
MLP	0.7714	0.77	0.77	0.77
AdaBoost	**0.861**	0.861	0.861	0.813

Table 1: Binary classification results using various classifiers without data augmentation

Model	**Top 1 Accuracy**	**Precision**	**Recall**	**F1 Score**
KNN	0.9064	0.90	0.89	0.89
SVM	0.9102	0.91	0.90	0.90
Random Forest	0.9707	0.97	0.96	0.96
XGB	0.9776	0.97	0.96	0.96
MLP	**0.9757**	0.97	0.97	**0.97**
AdaBoost	0.975	0.97	0.97	**0.97**

Table 2: Binary classification results for various classifiers with data augmentation

The table 3 contains quantitative results for binary classification task. In binary classification the data split ratio 73% for training, 5% for validation and 22% for testing. So 800 images are in the testset which are never seen by the model. The statistical measures namely accuracy, precision, recall and F1-score are tabulated in table 3. In addition it has false classifications made by the model out of 800 images in the testset. It is observed that all the four models are performing significantly better when fine-tuned using transfer learning instead of training from scratch. To quantitatively compare VGG19, ResNet18, EfficientNet-B0 are almost giving similar measures, but ConvNext Tiny outperforming other three by atleast 1%. Thirteen misclassifications out of 800 is quite acceptable. Since there are ambiguities in class labels itself, the model may learn wrong class information which leads to increase in error rates.

Model	**Top 1 Accuracy**	**Precision**	**Recall**	**F1 Score**	**False-classifications(out of 800)**
VGG19	0.977	0.98	0.98	0.98	18
ResNet18	0.974	0.97	0.97	0.97	21
EfficeintNet B0	0.974	0.97	0.97	0.97	21
ConvNext Tiny	**0.984**	0.98	0.98	0.98	**13**

Table 3: Binary classification results for DR recognition

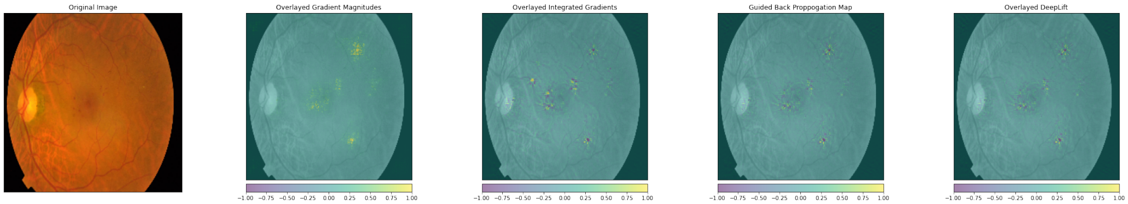
In multi-class DR classification, by just snapping at the results table it is quite surprising that the classification performance is better than binary DR task. One of the reasons to understand this phenomenon is to understand the cross-entropy loss used. Cross entropy loss heavily penalizes for a wrong classification, which keeps the model to learn the deep features for each class. Since in the multi-class classification validation set is not created so that 5% is added to the training set. This might also contributed to learning and performance improvement.

Model	**Top 1 Accuracy**	**Precision**	**Recall**	**F1 Score**	**False-classifications (out of 704)**
VGG19	0.9644	0.96	0.92	0.94	25
ResNet18	0.99	0.98	0.98	0.98	7
EfficeintNet B0	**0.997**	0.99	0.99	0.99	**2**
ConvNext Tiny	0.995	0.99	0.99	0.99	3

Table 4: Classification results for DR recognition and classification

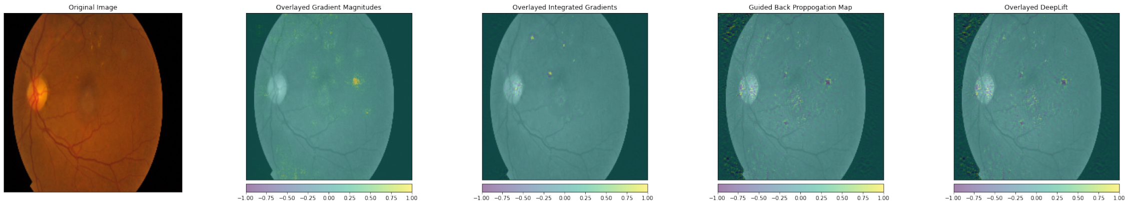
We have seen the results from the tabular presentation above, the models are pretty good, but it is important to understand what is happening inside the model. Explainability of the models is utmost important before taking it any further, since it has direct implications on human lives if deployed. To place an effort on the model interpretation, now we look into most commonly used model interpretation techniques. Attribution is a technique that gives information about which parts of the input image are giving rise predictions. There are different gradient based methods like saliency feature map, Integrated gradients(IG) [7], Guided back propagation(GBP) [8] and DeepLift(DL) [9]. The saliency map gives information about prime regions of interest as compared to our human eyes. Integrated gradients is a simple, yet powerful axiomatic attribution method that requires almost no modification of the original network. The attribution results are shown in Figures 6 7 8 9. One random image is sampled from each DR class and their attribution results are using convnext-tiny and are shown below. In the attribution maps negative weight corresponds to negative effect on the prediction and positive weight corresponds to significant features for prediction.

From the attribution maps it is quite observable that the model is not only focusing on salient features and but features that are invisible to naked human eye. Some of the regions that are particular to class and helps unique identification of class are also captured by the model. Still these are attributions are incomplete and there is a need for more rigours approach to understand the models.


Original Image
Overlayed Gradient Magnitudes
Overlayed Integrated Gradients
Guided Back Proppogation Map
Overlayed DeepLift
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00

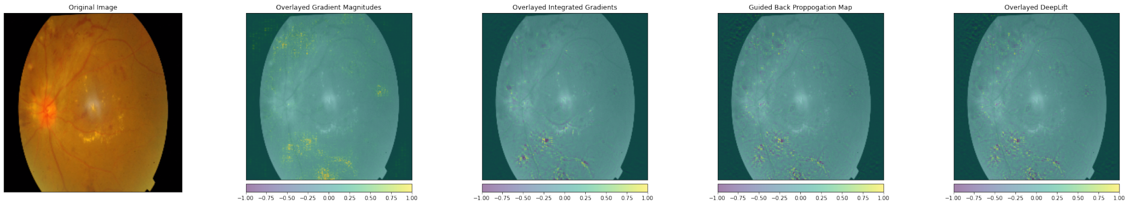
(a) Mild (b) Saliency Map (c) IG (d) GBP (e) DeepLift

Figure 6: Gradient based attributions for Mild class.


Original Image
Overlayed Gradient Magnitudes
Overlayed Integrated Gradients
Guided Back Proppogation Map
Overlayed DeepLift
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00

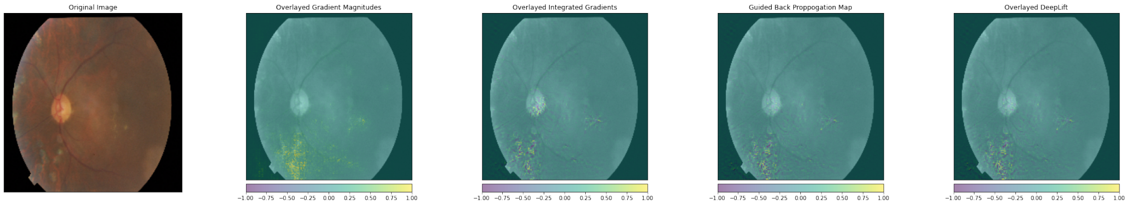
(a) Moderate (b) Saliency Map (c) IG (d) GBP (e) DeepLift

Figure 7: Gradient based attributions for Moderate class.


Original Image
Overlayed Gradient Magnitudes
Overlayed Integrated Gradients
Guided Back Proppogation Map
Overlayed DeepLift
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00

(a) Severe (b) Saliency Map (c) IG (d) GBP (e) DeepLift

Figure 8: Gradient based attributions for Severe class.


Original Image
Overlayed Gradient Magnitudes
Overlayed Integrated Gradients
Guided Back Proppogation Map
Overlayed DeepLift
−1.00 −0.75 −0.50 −0.25 0.00 0.25 0.50 0.75 1.00

(a) Proliferate (b) Saliency Map (c) IG (d) GBP (e) DeepLift

Figure 9: Gradient based attributions for Proliferate class.

6 Conclusions

The shown methods worked on significantly on a both binary DR as well as multi class DR classification tasks. Since the test is a potion of main data set, in which features and images are similar, the models will do well. If the same model used to test on a different it, performance drops heavily. So now the current days task to train more robust algorithms that will withstand the invariances in size, modalities, dataset population, etc. If we want to see the computer algorithms running real-time in hospitals the dataset invariance should be learnt robustly. A more recent approach called federated learning is being under active research, in which model is training is distributed in a centralized process, and trained on multiple datasets that are stored securely at different hospitals.

7 About Team member contributions

Sneha's Cotributions	**Vinay's Contributions**
Sneha is responsible for implementing data augmentation, deep feature extraction using InceptionV3 and ML classifiers. She also worked on image processing based feature extraction	Vinay is responsible for implementing algorithms in Section End-to-end deep neural network based classification and their corresponding results in tables Table 3 and 4. In addition to that a model interpretation studies is done through attribution methods and results are interpreted

References

[1] Asia Pacific Tele-Ophthalmology Society (APTOS). Aptos 2019 blindness detection, 2019.

[2] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[3] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[5] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[6] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *arXiv preprint arXiv:2201.03545*, 2022.

[7] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[8] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[9] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMLR, 2017.