Acta Scientific MEDICAL SCIENCES (ISSN: 2582-0931)

Volume 4 Issue 12 December 2020

Research Article

Diabetic Retinopathy Detection-MobileNet Binary Classifier

Aruna Pavate*, Jay Mistry, Rahul Palve and Nirav Gami

Information Technology Department, Atharva College of Engineering, Mumbai University, India

***Corresponding Author:** Aruna Pavate, Information Technology Department, Atharva College of Engineering, Mumbai University, India.

Received: September 28, 2020

Published: November 27, 2020

Abstract

Background and Purpose: According to the International Diabetes Federation (IDF) the total number of people in India who are suffering from diabetes were around 50.8 million in the year 2010, and it will rise to 87.0 million by 2030. Diabetic Retinopathy is a one of the major complications that exhibits because of Type II diabetes. Diabetic Retinopathy causes blindness in the population of age mostly in between 20 to 64 years. In long term diabetic retinopathy blood vessels disturb the normal flow of fluid out the eye and that comes pressure on the eyeball and this may cause damage to nerves that emerge in glaucoma. Early detection of Diabetic retinopathy and treatment can significantly reduce the risk of vision loss.

Analysis Method: The manual diagnosis of Diabetic retinopathy by ophthalmologists takes time, effort and also includes more costs and can be misdiagnosed if computer aided diagnosis systems are not used. Recently, Deep Learning has become one of the most common methods to achieve high performance results in many areas, especially in medical image analysis and classification. This work addresses the problem of prediction of diabetic retinopathy in advance to avoid further complications in the near future. The proposed classifier was built using MobileNet architecture-a lightweight, mobile friendly architecture, which is trained using retinal fundus images from Aptos 2019 challenge dataset.

Findings: The proposed enhanced model gives an accuracy of 95% and precision, recall, f-1 scores are 0.95, 0.98 and 0.97 respectively. Presented results demonstrate that this model achieves promising results and can be deployed as an application for clinical testing. This work attempts to suggest the diabetic retinopathy complications in advance. The intention of the work is to help the practitioners not to replace the ophthalmologist.

Keywords: Diabetic Retinopathy; MobileNet; CNN

Introduction

Diabetes is the most suitable disease for applying deep learning concepts [1]. There are many researchers working on prediction of diabetes disease and complications arising from diabetes. There are many applications available which helps the practitioners to study the disease and complications but many applications have their own advantages and flaws. According to [2], Indian peoples are more prone to diabetes because of lots of reasons including lifestyle, consumption of type of food and inadequate physical activities. Diabetic Retinopathy is one of the major complications that affects an eye human eye of diabetic person. Damage to the blood vessels of light-sensitive tissue of the retina causes this disease. Diabetic Retinopathy (DR) is a compilation of diabetes that causes the blood vessel of the retina to swell and leak fluids and blood. It is the leading cause of blindness for people aged 20 to 64 years. Diabetic Retinopathy (DR) is a leading cause of vision loss globally. According to the article presented in [3] claims that approximately one-third of the 285 million population having diabetes mellitus worldwide intimates signs of diabetic retinopathy. As per [4] survey, the global population suffering from diabetes in 2025 will be 438 million and it will surpass by 25 million will be 578 million in adults by 2030 and 700 millions by 2045 as shown in figure 1a and 1b shows the number of death cases during the year 2019 because of the diabetes disease.

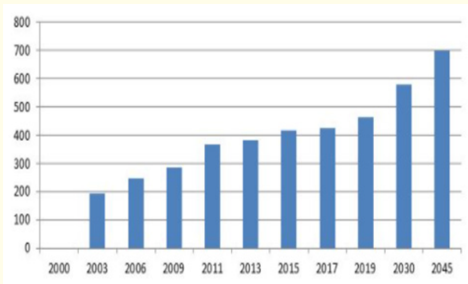

Figure 1a: Estimation of diabetes patients by 2045 [4].

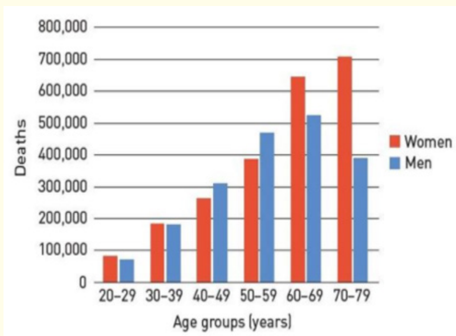

Figure 1b: Number of deaths in adults because of diabetes patients in 2019 [4].

In India, currently around 127,000 ophthalmologists but around 45% of the patients suffer from blindness before being diagnosed with the disease. Currently, diagnosing diabetic retinopathy is a slow process that requires trained doctors to analyze color photographs of retinas. Then experts classify the level of deterioration of the patient's eye has experienced into one of the four categories of diabetic retinopathy. This process is effective but it is a very slow process that requires around 2 days to get results and as it takes this much time to get the results it may be harder to reach patients. In India, in rural areas to access the trained clinicians the proper screening of equipment is in limited amounts and hence many of those peoples are left without any support for medication. As the number of people with diabetes is increasing this system, the proposed system, is going to be more efficient. To improve this situation, the Google research team worked closely with EyePACS in the USA and three eye hospitals in India [5], Aravind Eye Hospital, Sankara Nethralaya, and Narayana Nethralaya. Therefore, building better deep learning-enabled software for automatic diabetic retinopathy detection becomes necessary. In recent times there is an increasing progress in collaboration between Diabetes care physicians and Ophthalmologists. There is a shortage of ophthalmologists who can treat and diagnose diabetic retinopathy properly. In economically developed countries like the USA, about one third of diabetes have never undergone an ophthalmologic examination, and according to other data, only half have been examined by an eye doctor over the past year. The quality of the ophthalmologist examination is not up to the mark.

A common way to detect the diabetic eye is to examine fundus images and study the severity of the disease. There are four main stages of Diabetic Retinopathy; in its most advanced stage, abnormal blood vessels propagate on the surface of the retina, which can lead to scarring and cell loss in the retina.

Mobile-Based AI helps to detect diabetic Retinopathy, has been applied in some studies [6]. As mobile devices contain less memory capacity and less computation limit, so most of the major research work focuses on using architectures like dense which are heavy and computationally expensive. This work uses MobileNet, which is efficient and lightweight architecture, used to classify diabetic retinopathy into binary classes. The remaining content of this work is organized as follows. Section II introduces the related work, Section III describes the implementation details like dataset used, methodology Section IV result and analysis of the work and summarizes main findings.

Related work

In recent times there has been rising interest in building small and efficient networks in recent literature [7,8]. There are many approaches researchers have tried including compressing pretrained models to training small network models. Some authors have tried the Fuzzy inference system to find out the risk of the major complications arising because of diabetes [9].

Andrew G. Howard., *et al*. [10] proposed an efficient model to build light weight neural networks to be applicable to various computer vision applications like fine grain classification, object detection etc. in order to provide benefit of small, low latency models.

Yuchen Wu., *et al*. [11] proposed classification of diabetic retinopathy using the concept of transfer learning and tested the samples on VGG19, InceptionV3 and Resnet models. Transfer learning concept applied on Kaggle dataset and experimental accuracy shows 60%, claimed better than the model learning from scratch.

Yunlei Sun., *et al*. [12] proposed a model to detect diabetes retinopathy. Electronic Health Record from 201 hospital applied over five different algorithms such as Decision tree, Random Forest, Support vector machine, Logistic Regression, Naïve Bayesian and performed the comparative analysis of the models to get a better diagnosis. Among all the five models, the Random Forest model shows 92% of accuracy.

Y. sun [13] developed a hybrid model which is a combination of CNN and the BN layer for diagnosing diabetic retinopathy disease based on convolution Neural Network Method, applied to one dimensional datasets. This model helps to improve the training speed and improve the accuracy. The results showed training accuracy 99.85% and test accuracy of 97.56%.

Lam C., *et al*. [14] proposed an early detection of diabetic retinopathy using convolutional neural networks (CNNs) on color fundus images. Author claimed of a new multinomial classification model to discover the preprocessing which improves recognition of precise features. The concept of transfer learning applied on different models like GoogLeNet with test accuracy 74.5%, AlexNet with 68.8% test accuracy and ImageNet with 52.2%.

Carson Lam., *et al*. [14] proposed detection of diabetic retinopathy using deep learning concepts and proved the validation sensitivity of 95%. The main contribution in this research is the preprocessing with limited adaptive histogram equalization.

At the same time, many researchers have got attention on machine learning model for detection of diabetes of retinopathy [15-19].

Methodology

Dataset

MobileNet model trained and tested on the APTOS dataset which is freely available on kaggle [20]. The dataset rated each image according to the levels of severity of the diabetes on a scale of 0 to 4. Here 0 - No DR, 1 - Mild, 2 - Moderate, 3 - Severe, 4 - Proliferative DR. These images are processed and targeted into binary labels as shown in figure 2. The dataset contains 3662 images out of that 650 images were used for validation of the system. Here image size is 224 X 224 X 3. First dataset divides the dataset of 5 classes of grade [0, 1, 2, 3, 4] into a binary dataset. This is done to make the dataset unbiased, i.e. to equal the number of images having and not having the disease. As for preprocessing the dataset images are shrunk to (224, 224, 3) pixels and passed to the preprocessing method specially designed for MobileNet architecture. The normalization of images is performed to decrease the variance in data points.

	id_code	diagnosis	file_name	binary_target
0	000c1434d8d7	2	000c1434d8d7.png	1
1	001639a390f0	4	001639a390f0.png	1
2	0024cdab0c1e	1	0024cdab0c1e.png	1
3	002c21358ce6	0	002c21358ce6.png	0
4	005b95c28852	0	005b95c28852.png	0

Figure 2: Levels of severity of disease targeted into binary labels.

Architecture

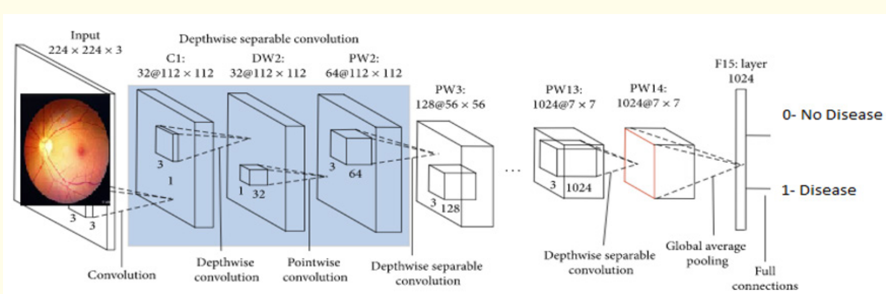

Figure 3: Mobilenet Architecture to train APTOS dataset.

Mobilenet is a lightweight architecture. It uses depth wise separable convolutions which basically means it performs a single convolution on each color channel rather than combining all three and flattening it. This has the effect of filtering the input channels. It is very low maintenance thus performs very good with high speed. Mobilenet Architecture used to train APTOS dataset. For model building the top 6 layers of MobileNet are removed and a Dropout layer and a Dense layer is attached to it, using" SoftMax" activation. The first 23 layers are then made untrainable while the remaining layers that are few remaining layers of MobileNet and 2 added layers are allowed to be trained. The optimizer used Adam and the loss function used is categorical_crossentropy while the metrics used is categorical_accuracy. The callbacks used are Early Stopping, ReduceLrOnPlateau, checkpoint and csv_logger. Basic steps for classification of diabetes retinopathy include:

- Import APTOS dataset
- Build the network model
- Select the optimizer
- Use the trained model
- Test the model.

Summary of the model is shown below:

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 224, 224, 3)]	0
conv1_pad (ZeroPadding2D)	(None, 225, 225, 3)	0
conv1 (Conv2D)	(None, 112, 112, 32)	864
conv1_bn (BatchNormalization	(None, 112, 112, 32)	128
conv1_relu (ReLU)	(None, 112, 112, 32)	0
conv_dw_1 (DepthwiseConv2D)	(None, 112, 112, 32)	288
conv_dw_1_bn (BatchNormaliza	(None, 112, 112, 32)	128
conv_dw_1_relu (ReLU)	(None, 112, 112, 32)	0
conv_pw_1 (Conv2D)	(None, 112, 112, 64)	2048
conv_pw_1_bn (BatchNormaliza	(None, 112, 112, 64)	256
conv_pw_1_relu (ReLU)	(None, 112, 112, 64)	0
conv_pad_2 (ZeroPadding2D)	(None, 113, 113, 64)	0
conv_dw_2 (DepthwiseConv2D)	(None, 56, 56, 64)	576
conv_dw_2_bn (BatchNormaliza	(None, 56, 56, 64)	256
conv_dw_2_relu (ReLU)	(None, 56, 56, 64)	0
conv_pw_2 (Conv2D)	(None, 56, 56, 128)	8192
conv_pw_2_bn (BatchNormaliza	(None, 56, 56, 128)	512
conv_pw_2_relu (ReLU)	(None, 56, 56, 128)	0
conv_dw_3 (DepthwiseConv2D)	(None, 56, 56, 128)	1152
conv_dw_3_bn (BatchNormaliza	(None, 56, 56, 128)	512
conv_dw_3_relu (ReLU)	(None, 56, 56, 128)	0
conv_pw_3 (Conv2D)	(None, 56, 56, 128)	16384
conv_pw_3_bn (BatchNormaliza	(None, 56, 56, 128)	512
conv_pw_3_relu (ReLU)	(None, 56, 56, 128)	0
conv_pad_4 (ZeroPadding2D)	(None, 57, 57, 128)	0
conv_dw_4 (DepthwiseConv2D)	(None, 28, 28, 128)	1152
conv_dw_4_bn (BatchNormaliza	(None, 28, 28, 128)	512
conv_dw_4_relu (ReLU)	(None, 28, 28, 128)	0
conv_pw_4 (Conv2D)	(None, 28, 28, 256)	32768
conv_pw_4_bn (BatchNormaliza	(None, 28, 28, 256)	1024
conv_pw_4_relu (ReLU)	(None, 28, 28, 256)	0
conv_dw_5 (DepthwiseConv2D)	(None, 28, 28, 256)	2304
conv_dw_5_bn (BatchNormaliza	(None, 28, 28, 256)	1024
conv_dw_5_relu (ReLU)	(None, 28, 28, 256)	0
conv_pw_5 (Conv2D)	(None, 28, 28, 256)	65536
conv_pw_5_bn (BatchNormaliza	(None, 28, 28, 256)	1024
conv_pw_5_relu (ReLU)	(None, 28, 28, 256)	0
conv_pad_6 (ZeroPadding2D)	(None, 29, 29, 256)	0
conv_dw_6 (DepthwiseConv2D)	(None, 14, 14, 256)	2304
conv_dw_6_bn (BatchNormaliza	(None, 14, 14, 256)	1024
conv_dw_6_relu (ReLU)	(None, 14, 14, 256)	0
conv_pw_6 (Conv2D)	(None, 14, 14, 512)	131072
conv_pw_6_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_pw_6_relu (ReLU)	(None, 14, 14, 512)	0
conv_dw_7 (DepthwiseConv2D)	(None, 14, 14, 512)	4608
conv_dw_7_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_dw_7_relu (ReLU)	(None, 14, 14, 512)	0
conv_pw_7 (Conv2D)	(None, 14, 14, 512)	262144
conv_pw_7_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_pw_7_relu (ReLU)	(None, 14, 14, 512)	0
conv_dw_8 (DepthwiseConv2D)	(None, 14, 14, 512)	4608
conv_dw_8_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_dw_8_relu (ReLU)	(None, 14, 14, 512)	0
conv_pw_8 (Conv2D)	(None, 14, 14, 512)	262144
conv_pw_8_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_pw_8_relu (ReLU)	(None, 14, 14, 512)	0
conv_dw_9 (DepthwiseConv2D)	(None, 14, 14, 512)	4608
conv_dw_9_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_dw_9_relu (ReLU)	(None, 14, 14, 512)	0
conv_pw_9 (Conv2D)	(None, 14, 14, 512)	262144
conv_pw_9_bn (BatchNormaliza	(None, 14, 14, 512)	2048
conv_pw_9_relu (ReLU)	(None, 14, 14, 512)	0
conv_dw_10 (DepthwiseConv2D)	(None, 14, 14, 512)	4608
conv_dw_10_bn (BatchNormaliz	(None, 14, 14, 512)	2048
conv_dw_10_relu (ReLU)	(None, 14, 14, 512)	0
conv_pw_10 (Conv2D)	(None, 14, 14, 512)	262144
conv_pw_10_bn (BatchNormaliz	(None, 14, 14, 512)	2048
conv_pw_10_relu (ReLU)	(None, 14, 14, 512)	0
conv_dw_11 (DepthwiseConv2D)	(None, 14, 14, 512)	4608
conv_dw_11_bn (BatchNormaliz	(None, 14, 14, 512)	2048
conv_dw_11_relu (ReLU)	(None, 14, 14, 512)	0
conv_pw_11 (Conv2D)	(None, 14, 14, 512)	262144
conv_pw_11_bn (BatchNormaliz	(None, 14, 14, 512)	2048
conv_pw_11_relu (ReLU)	(None, 14, 14, 512)	0

conv_pad_12 (ZeroPadding2D)	(None, 15, 15, 512)	0
conv_dw_12 (DepthwiseConv2D)	(None, 7, 7, 512)	4608
conv_dw_12_bn (BatchNormaliz	(None, 7, 7, 512)	2048
conv_dw_12_relu (ReLU)	(None, 7, 7, 512)	0
conv_pw_12 (Conv2D)	(None, 7, 7, 1024)	524288
conv_pw_12_bn (BatchNormaliz	(None, 7, 7, 1024)	4096
conv_pw_12_relu (ReLU)	(None, 7, 7, 1024)	0
conv_dw_13 (DepthwiseConv2D)	(None, 7, 7, 1024)	9216
conv_dw_13_bn (BatchNormaliz	(None, 7, 7, 1024)	4096
conv_dw_13_relu (ReLU)	(None, 7, 7, 1024)	0
conv_pw_13 (Conv2D)	(None, 7, 7, 1024)	1048576
conv_pw_13_bn (BatchNormaliz	(None, 7, 7, 1024)	4096
conv_pw_13_relu (ReLU)	(None, 7, 7, 1024)	0
global_average_pooling2d (Gl	(None, 1024)	0
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 2)	2050

Total params: 3,230,914
Trainable params: 3,209,026
Non-trainable params: 21,888

Result and Discussion

The developed architecture is MobileNets based on depth wise separable convolutions**.** The model is trained for 100 Epoch, while the best model is found in 24th Epoch with f1-score of 0.96. The overall accuracy of the model is.096 with a good result in confusion metrics and with a kappa score of 0.93 as shown in figure 4.

	epoch	categorical_accuracy	loss	lr	val_categorical_accuracy	val_loss
9	9	0.990151	0.027905	0.00250	0.963989	0.178187
12	12	0.997538	0.006863	0.00125	0.963989	0.221748
13	13	0.998769	0.005212	0.00125	0.963989	0.234287

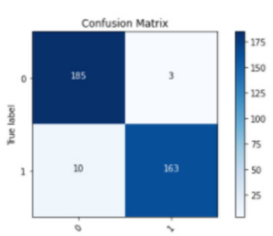

Figure 4: Confusion Matrix.

In order to compare the performance of the model different parameters like precision, recall, f1-score and support are calculated as shown in figure 5. The MobileNet Network models maintain the rise of the depth of the convolutional neural network, as the use of small convolutional kernels gives great impact on the final classification result, but at the same time it also drops the performance of the model and therefore it is difficult to apply practically. Figure 6 shows the training, validation loss and training and validation accuracy of the model.

	precision	recall	f1-score	support
0	0.95	0.98	0.97	188
1	0.98	0.94	0.96	173
accuracy			0.96	361
macro avg	0.97	0.96	0.96	361
weighted avg	0.96	0.96	0.96	361

Figure 5: Performance analysis of system using various parameters.

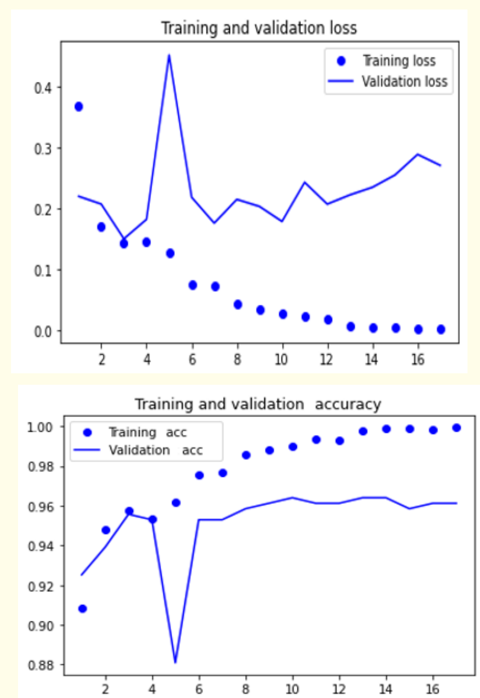

Figure 6: Graphical Analysis of Training and validation samples for accuracy and loss of the model.

Conclusion

Diabetic Retinopathy is one of the major complications that take place because of Type II diabetes mellitus where blood vessels swell as well as can even break. Early detection of disease helps to prevent further complications and helps the expert to treat the patient in early stages. The proposed system is based on MobileNet architecture with dense blocks for image classification. Though compression and acceleration of network model reduces the classification accuracy including dense blocks allows to improve the performance of the MobileNet. In future, the same architecture can be applied to detect the further other complications taking place because of diabetes mellitus.

Bibliography

1. Zainuddin Z., *et al*. "A neural network approach in predicting the blood glucose level for diabetic patients". *World Academy of Science, Engineering and Technology* 6.50 (2009): 981-988.
2. A Pavate and N Ansari. "Risk Prediction of Disease Complications in Type 2 Diabetes Patients Using Soft Computing Techniques". 2015 Fifth International Conference on Advances in Computing and Communications (ICACC), Kochi (2015): 371-375.
3. https://www.auctoresonline.org/journals/endocrinology-and-disorders/archive/173
4. https://diabetesexpo.com/
5. http://www.eyepacs.com/blog/eyepacs-helps-google-rethink-the-future-of-eye-health
6. R Rajalakshmi., *et al*. "Automated diabetic retinopathy detection in smartphone-based fundus photography using artificial intelligence". *Eye* 32.6 (2018): 1138-1144.
7. J Jin., *et al*. "Flattened convolutional neural networks for feedforward acceleration". arXiv preprint arXiv (2014): 1412.5474.
8. M Wang., *et al*. "Factorized convolutional neural networks". arXiv preprint arXiv (2016): 1608.04337.
9. Pavate A., *et al*. "Early prediction of five major complications ascends in diabetes mellitus using fuzzy logic". In: Nayak, J., Abraham, A., Krishna, B.M., Chandra Sekhar, G.T., Das, A.K. (eds.) Soft Computing in Data Analytics. AISC 758 (2019): 759-768.
10. Andrew G Howard., *et al*. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications". arXiv:1704.04861v1 [cs.CV] (2017).
11. Y Wu and Z Hu. "Recognition of Diabetic Retinopathy Basedon Transfer Learning". 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China (2019): 398-401.
12. Y Sun and D Zhang. "Diagnosis and Analysis of Diabetic Retinopathy Based on Electronic Health Records". in IEEE Access 7 (2019): 86115-86120.
13. Y Sun. "The Neural Network of One-Dimensional Convolution-An Example of the Diagnosis of Diabetic Retinopathy". in IEEE Access 7 (2019): 69657-69666.
14. Lam C., *et al*. "Automated Detection of Diabetic Retinopathy using Deep Learning. AMIA Joint Summits on Translational Science proceedings". *AMIA Joint Summits on Translational Science* 2017 (2018): 147-155.
15. Xu K., *et al*. "Deep Convolutional Neural Network-Based Early Automated Detection of Diabetic Retinopathy Using Fundus Image". *Molecules* 22.12 (2017): 2054.
16. Narayanan BN., *et al*. "Hybrid machine learning architecture for automated detection and grading of retinal images for diabetic retinopathy". *Journal of Medical Imaging (Bellingham)* 7.3 (2020): 034501.
17. Hacisalihoglu RE., *et al*. "Deep Learning Frameworks for Diabetic Retinopathy Detection with Smartphone-based Retinal Imaging Systems". *Pattern Recognition Letters* 135 (2020): 409-417.
18. Riaz H., *et al*. "Deep and Densely Connected Networks for Classification of Diabetic Retinopathy". *Diagnostics (Basel)* 10.1 (2020): 24.
19. Hatanaka Y. "Retinopathy Analysis Based on Deep Convolution Neural Network". *Advances in Experimental Medicine and Biology* 1213 (2020): 107-120.
20. https://www.kaggle.com/c/aptos2019-blindness-detection