International Journal of Computer Science Trends and Technology (IJCST) – Volume 10 Issue 3, May-Jun 2022

RESEARCH ARTICLE OPEN ACCESS

Diabetic Retinopathy Detection using Deep Learning

Kalyani Bang [1], Rutika Kamble [2], Shraddha Pandey [3],
Vijayraj Mukwane [4], A.W.Bhade [5]

[1],[2],[3],[4] Student, Department of Information Technology, GCOE Amravati, Maharashtra - India
[5] Professor, Department of Information Technology, GCOE Amravati, Maharashtra - India

ABSTRACT

This paper provides a solution for the diagnosis of diabetes retinopathy. Diabetes retinopathy is a disease carried by people with diabetes and many people lose their sight due to this disease. Various handmade procedures are available to diagnose Diabetic Retinopathy, but it requires a skilled doctor and needs a lot of time. So this project can be used to reduce such kind of hand procedures and an eye doctor can spend more time in caring for the patient properly or at least minimizing the severity of the disease. This can be achieved by creating a Convolutional neural network model that can automatically view a patient's eye image and measure the severity of blindness in a patient. The goal here is to measure their efforts professionally, so that they may have the ability to automatically diagnose disease images and provide information on how serious the condition can be. This can help to shorten the time and hence leads to evaluating the process of treating diabetic retinopathy on a large scale.

Keywords: Diabetic Retinopathy, ophthalmologist, blindness, convolution neural network, severity.

I. INTRODUCTION

Diabetic retinopathy is a major cause of blindness these days. Over 65.1 million people in India are affected with diabetes and this number is expected to achieve rise upto 109 million by 2035. Manual diagnosing process of Diabetic Retinopathy by eye doctors is time-consuming, time-consuming, and costly. Diabetic Retinopathy(DR) can be categorized from primary to most advanced when examining the condition of the retina fundus. The disease has two main stages: Non-Proliferative Diabetic Retinopathy (NPDR) and Proliferative Diabetic Retinopathy (PDR). However, the NPDR introduced three sub-categories, medium, and solid. The damage of this first stage is limited and does not extend beyond the inner lining of the retina. Treatment at this stage will work well. DR increases when the ischemia damage results in the growth of blood vessels beyond the retina. PDR contains low levels of premature, high risk, and high levels. Diabetic retinopathy is found to be caused when any patient has diabetes for at least 10-12 years without being even diagnosed and patient is unaware of it. DR can therefore be prevented if it is diagnosed early with a medical examination and systematic treatment for diabetes.

II. RELATED RORK

Huge research has been carried out for binary classification of Diabetic Retinopathy. Farrikh Alzami [1] described a grade classification of DR based on fractal analysis and random forest with MESSIDOR dataset. The system which they have built was able to segment images and compute fractal dimensions as a features but they failed to distinguish mild mild diabetic retinopathy to severe diabetic retinopathy. Neural Network has also been used to classify the three categories of DR. Nayak et al [2] used features such as the area of exudates and the location of blood vessels and tissue boundaries. Elements embedded in the neural network to separate images into normal, abnormal and progressive retinopathy. The neural network uses these features as input. The findings were confirmed in comparison with an optician specialist program. They demonstrated 93% class accuracy, 90% sensitivity and 100% clarity. This was done on 140 image websites and the output feature was required for all images in both time training and testing. Most five-phase division studies have done using vector support equipment (SVMs). Acharya et al [3] created an automated method of identifying five classes. Features, extracted from raw data using a high-quality spectra method, are included in the SVM filter and capture form variations and concert images. This SVM method has reported 82% of average accuracy, 82% sensitivity and 88% clarity.

III. METHODOLOGY

A. DATASET

Every image processing requires a dataset to train on. In our project, we have used the dataset provided by the Kaggle, [5]it's community of data scientist and machine learning practitioners. For our project we needed two distinguish datasets, healthy eyes and affected eyes. Dataset size is one of the parameter on which accuracy of the model depends. So, we tried to make sure the size of dataset should be as big as possible.

A. METHODS

1. Convolutional Neural Network

Convolutional Neural Networks (CNNs), a branch of in-depth learning, has an excellent history of applications in image analysis and interpretation, including medical imaging.

Network structures designed to work with image data were typically already built in the 1970s 10 with useful applications and surpassed other challenging tasks such as handwriting recognition recognition11. However, until several breakthroughs have emerged in neural networks such as 12-stop initiation, 13 fixed-unit units and a corresponding increase in computer power per image processing units (GPUs) where they work on complex image recognition problems. Currently, major CNNs are used to successfully address the most complex image recognition tasks in the multi-item categories; at an impressive level. CNN is used in many of the current high-quality image separation functions such as ImageNet and COCO challenges.

The Convolution neural network works as follows:

Step 1: Convolution

The convolution layer is the basis of construction of CNN. Contains a large part of the network accounting burden. This layer in actual generates the dot production between the two matrics, among them one matrix is a set of readable parameters which is also known as a kernel, and the other matrix is a limited part of the reception field. The kernel is smaller in size than the image but much deeper. This means that, if the image is composed of three channels (RGB), the kernel height and width will be less geographically, but the depth extends across all three channels.

Step 2: Pooling

This layer actually replaces the output of the network at certain locations followed by deriving a summary statistic of the nearby outputs which further helps in reducing spatial size of the representation, which ultimately decreases the required amount of computation and weights. The pooling operation is processed on every slice of the representation individually.

Step3: Flattening

It is a process of converting the data into a 1- dimensional array for inputting it to the next layer. Here we usually flatten the output of the convolutional layers for creating a single long feature vector.

And it is connected to the final classification model, which is called a ***fully-connected*** layer. In other words, we put all the pixel data in one line and make connections with the final layer.

DENSE NET ARCHITECTURE

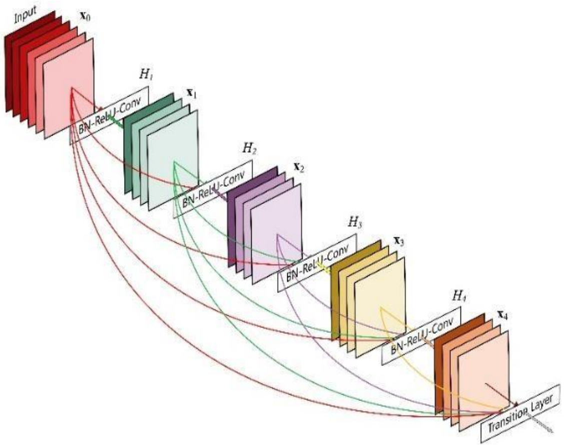
DenseNet (Dense Convolutional Network) is a project that focuses on making deep learning networks even deeper, but at the same time making them more effective for training, through short links between layers. DenseNet is a convolutional neural network where each layer is connected to all the other deeper layers in the network, i.e., the first layer is connected to the 2nd, 3rd, 4th and so on, the second layer is connected to the 3, 4, 5 and so on. This is done to enable the flow of large amounts of information between layers of the network.

Dense Convolutional Network (DenseNet), which connects each layer to all other layers in a forward-looking way. Although traditional convolutional networks have L layers have L-connections - one between each layer and its subsequent layer.

Somewhat different from one of most efficient model of DenseNet-BC (DenseNet-Bottleneck-Compressed) networks are being trained. By using the DenseNet-BC-121-32 model as a basic model.

Advantages of DenseNet-BC are:

- Reduced number of parameters
- Similar or Better performance
- Better accuracy



This webapp has been created with flask. Flask is a web application framework which has been written in Python. It was developed by Armin Ronacher, who led a team of international Python enthusiasts called Poocco. Flask is completely based on the Werkzeg WSGI toolkit and the Jinja2 template engine.Both are said to be Pocco projects.

The dataset used for training the model is available at kaggle, where we have 2 csv files, one for training and the other for testing the trained model. Both the files have the same features. There are 13000 images of different severity are available in this dataset.

By using the Machine Learning libraries – scikit-learn [6], numpy [7], tensorflow[8], keras [9], opencv [10] and pandas [11], we have created a Machine Learning model and dumped it into a file. This file is deployed on a Flask Server. After testing model with trained data, we got accuracy of 90 percent.

CONCLUSION

This paper provides a solution for the diabetic patient for early detection of retinopathy. Once we provide images to the application, it provides results in the form of severity and thus helps in early diagnosis of diabetic retinopathy.

ACKNOWLEDGEMENT

Prof. A.W.Bhade, Head of Department of Information Technology, Government College of Engineering, Amravati, has helped us a lot and provided great support and guidance to author throughout this paper.

REFERENCES

[1] International Seminar on Application for Technology of Information and Communication, 2019. https://www.researchgate.net/publication/342120641_Diabetic_Retinopathy_Detection_using_Machine_Learning

[2]Automated identification of diabetic retinopathy stages using digital fundus images. J Med Syst 2008;32(2):107–115.https://www.sciencedirect.com/science/article/pii/S1877050916311929

[3] Application of higher order spectra for the identification of diabetes retinopathy stage. https://www.sciencedirect.com/science/article/pii/S1877050916311929

[4] The dataset is available here: https://www.kaggle.com/c/aptos2019- blindness-detection/data

[5] Scikit-Lear.org | Available at https://scikitlearn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

[6] Numpy | Available at: https://numpy.org/

[7] Tensorflow" | Available at: https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi

[8] Keras" | Available at: https://keras.io/

[9] OpenCV" | Available at: https://opencv.org/author/opencv/

[10] Pandas – Python Data Analysis Library" | Available at: https://pandas.pydata.org/