Detection of Blindness and its stages caused by Diabetes using CNN

Krishna Murthy K.T[1], Aishwarya Anil Kulkarni[2], Shreya Gupta[3], Keerthana M[4]

[1,2,3,4] *Department of Electronics and Instrumentation Engineering, B.M.S College of Engineering, Bengaluru, India*

Abstract **– Diabetic retinopathy is an eye condition that occurs due to diabetes in humans. It can arise because of the high blood glucose level caused by diabetes. Diabetes Retinopathy is the leading cause of new incidents of blindness in those aged 20 to 74. If DR is discovered and treated early enough, vision impairment caused by it can be managed or avoided. Blindness can be avoided if detected early enough. The Convolutional Neural Network (CNN), based on deep learning, is a promising tool in biological image processing. A deep-learning system for the categorization of Diabetes Retinopathy (DR) grades from fundus pictures is used to detect early blindness. In our paper, sample Diabetes Retinopathy (DR) photographs were grouped into five groups based on ophthalmologist competence. For the classification of DR phases, a group of deep Convolutional Neural Network algorithms was used. State-of-the-art accuracy result has been achieved by Resnet50, which demonstrates the effectiveness of utilizing deep Convolutional Neural Networks for DR image recognition.**

Keywords **– AI.ML, Diabetes Retinopathy, Deep-learning based CNN**

I. INTRODUCTION

The retina is a light-sensitive layer in the eye's rear portion. It transforms incoming light into electrical signals and transmits them to the brain. These signals are converted into visuals by the brain. Blood must be supplied to the retina on a regular and continuous basis. It gets the blood from tiny blood veins that reach it. High levels of sugar, the damage these tiny vessels in the blood leading to the development of Diabetes Retinopathy. Vitreous haemorrhage occurs when vessels in blood bleed into the primary jelly that fills the eyes, known as the vitreous. Floaters are common in mild cases, but vision loss is possible in more severe cases because the blood in the vitreous inhibits light from entering the eye. Bleeding in the vitreous can resolve spontaneously if the retina is not injured. Diabetic retinopathy is responsible for almost 86 percent of blindness in the younger-onset group. In the older-onset group, in which other eye diseases were common, one-third of the cases of legal blindness was developed due to diabetic retinopathy. The lack of trained eye doctors and level of operator's expertise also determines the fate of the patient. Lack of facilities in rural and semi-rural and lack of awareness about the disease contribute largely to the number of rising cases. This grave problem motivated us to come up with a state-of-the-art solution based on Deep Learning and with high accuracy to help detect the stage of blindness to provide the required treatment.

II. Diabetic Retinopathy

The backs of your eyes are supplied with oxygen and nutrients by some of your body's smallest and most sensitive blood vessels. When blood arteries are injured, the cells that surround them may begin to die. The retina is in the back of your eye and is responsible for sending "sight messages" to your brain. When light enters your eye, it bounces off the retina and is relayed to the optic nerve by light-sensitive cells. If those cells start to die, your retina will no longer be able to convey a clear image to your brain, resulting in vision loss.

Diabetic Retinopathy can be categorized into 5 stages:

1.NPDR (non-proliferative diabetic retinopathy)

2.Mild Non-proliferative Retinopathy

3. Moderate Non-proliferative Retinopathy

4.Severe Non-proliferative Retinopathy

5.PDR (proliferative diabetic retinopathy)

III. System Design

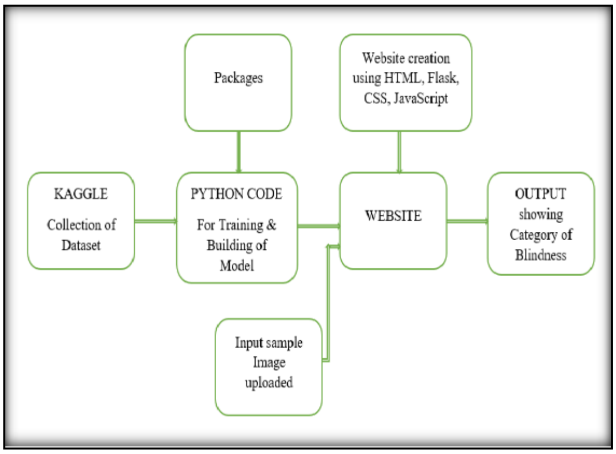

Fig.1. System Architecture

1. Kaggle

It contains 88,702 high-resolution photos acquired from various cameras with resolutions ranging from 433 289 pixels to 5184 3456 pixels. Each image is assigned to one of five DR phases. Only the ground facts for training photos are provided to the public. Many of the photographs on Kaggle are of poor quality and have inaccurate labeling.

2. Python 3 Spyder

We used the Scripting language- Python 3 programming language to train and build the model (on Spyder IDE).

3. Anaconda command prompt

Anaconda command prompt is like the command prompt, but it ensures that we may use Anaconda and conda commands without changing directories or paths from the prompt. We'll note that when we launch the Anaconda command prompt, it adds/("prepends") a variety of locations to your PATH.

4. Tensor Flow

For our model, we made use of TensorFlow. It's a free artificial intelligence package that builds models using data flow graphs. It gives programmers the ability to build large-scale neural networks with numerous layers. Classification, perception, understanding, discovering, prediction, and creation are some of the most common applications for TensorFlow. It is used to obtain the online Dataset. It is an end-to-end platform that made it easy for us to build and deploy our model.

5. Keras

Keras is a Python-based deep learning API that runs on top of TensorFlow, a machine learning platform. It was created with the goal of allowing for quick experimentation. It's crucial to be able to go from idea to result as quickly as feasible when conducting research.

5.1. Keras model

In our project, we adopted the sequential model. It enables us to build models in a layer-by-layer manner. However, we are unable to develop models with many inputs or outputs. It's best for simple layer stacks with only one input tensor and one output tensor. Layers and models are Keras' primary data structures.

```
from tensorflow.keras.models import Sequential

model = Sequential()
```

Fig.2. Sequential Model

5.2. Keras layers

In Keras, layers are the fundamental building elements of neural networks. A layer is made up of a tensor-in tensor-out calculation function (the layer's call method) and some state (the layer's weights) stored in TensorFlow variables. Keras Models are made up of functional building components called Keras Layers. Several layer () functions are used to generate each layer. These layers receive input information, process it, perform some computations, and then provide the output. Furthermore, the output of one layer is used as the input of another layer.

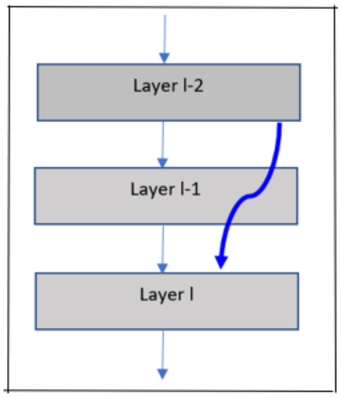

Fig.3. Canonical form of Residual Neural network (ResNet)

6. Web development

6.1. Flask

We used the Python Flask API, which allows us to create web applications. Flask is a Python-based microweb framework. It is referred to as a microframework because it does not necessitate the usage of any specific tools or libraries. It doesn't have a database abstraction layer, form validation, or any other components that rely on third-party libraries to do typical tasks.

6.2. HTML

HTML (Hypertext Markup Language) is the coding that defines how a web page, and its content are structured. Content could be organized using paragraphs, a list of bulleted points, or graphics and data tables. HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) are two of the most important Web page construction technologies. For a range of devices, HTML provides the page structure and CSS offers the (visual and auditory) layout. JavaScript is a text-based programming language that allows you to construct interactive web pages on both the client and server sides. Whereas HTML and CSS provide structure and aesthetic to web pages, JavaScript adds interactive components that keep users engaged.

IV. IMPLEMENTATION

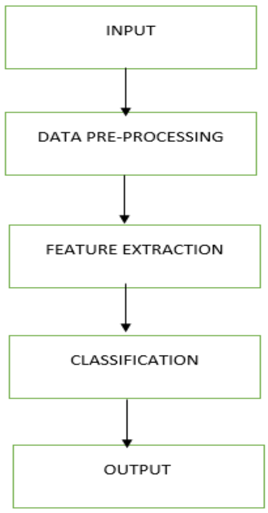

Fig.4. Shows the implementation of the mode

1. Collected database

Collected Database Fundus Images to train the model. Typical fundus images from the database that represent different DR stages: mild, moderate, NoDR, PDR, severe, and proliferative diabetic retinopathy.

2. Data Pre-processing

Data preprocessing is the most important aspect of our paper. Pre-processing is done because images may not befit those criteria, the geometry can be distorted, the lighting can be irregular, there can be noise in the images, and so on... so Preprocessing algorithms aim to prepare data (images), so they can be used efficiently by other types of algorithms.

We analyzed many projects and papers on Diabetic Retinopathy and found that they key to building a robust system was enough data preprocessing steps to improve the quality of the images.

The data preprocessing steps includes

Intensity normalization was done to convert pixel values to 0 to 255 only. It may be 50 to 150. 0 to 255 will be better distributed as RGB. This refers to a homogenization of the signal intensity of the white and grey matter, to better distinguish between the features and make it easier to segment. For this we used the Numpy package so that we can perform operations on the pixels.

Noise Removal by Cropping, zooming, and rescaling to zoom into only the features and surrounding eye area will be cropped which doesn't have any features thus noise is removed.

Rescaling has been performed to make sure all pixel values lie between 0 and 1

Data augmentation by Horizontal flip to increase no of images. Import from PIL is used so that now from Python Imaging Library various image editing capabilities like load image crop image can now be performed.

After all the images have been standardized, data augmentation is performed to boost the training data and hence the training process' quality. The augmentation is accomplished by flipping each input image vertically. Invariance isa characteristic of a convolutional neural network that allows it to reliably categories objects even when positioned in different orientations.

3. Data Balancing

Imbalanced data is a classification problem in which the number of observations per class is not evenly distributed; you'll often have a lot of data/observations for one class (referred to as the majority class) and a lot less for one or more other classes (referred to as the minority classes). As a result, data balancing is the following phase in our model. This step is done to provide the CNN model with an equal split of data for each DR grade, which can help remove bias during the training phase. The training model is provided an equal number of images for normal, moderate, medium, severe, and prophylactic diabetes.

4. Feature Extraction

The mathematical statistical process that extracts the quantitative parameter of resolution is known as feature extraction. It is a term used to describe changes or anomalies that are not visible to the human eye. Identifying anomalies is what Feature Extraction is all about. We decided to take GLCM out of the equation (texture-based features). The probability density function and the frequency of occurrence of similar pixels are used to create the Gray Level Co-occurrence Matrix (GLCM). The suggested methodology uses the proprietary Res-Net50 CNN model, which has 50 weighted layers, to perform transfer learning. Res-Net50 is divided into four stages, each with three convolutional layers and n replications. Res-Net refers to a residual network in which convolutional layer blocks are bypassed utilising shortcut connections. This feature can help the ResNet model learn global features particular to the data. The parameters of the convolutional layers are transferred without alteration, and the fully connected layer (FC1000 layer) is substituted with a shallow classifier with four class labels, reflecting the DR grades, to implement transfer learning.

5. Classification

We used following classifications.

5.1. First-stage classification

To categorize a DR image, we tested two types of shallow classifiers in this stage. The first classifier is a pixel-wise feedforward Neural Network (NN) classifier, which is made up of one layer of fully connected weights between the Resnet50 models feature vector and an output layer with several nodes equal to the DR grades. Based on the selected features, the second one does a binary linear kernel Supported Vector Machine (SVM) classification. Four class labels are utilized to train the first-stage system classifiers: normal, mild, moderate, and severe/PDR. Due to the similarity of their visuals, the DR grades "Severe" and "PDR" have been bundled into one label "Severe/PDR" throughout all experiments. The suggested system was tested using a conventional performance evaluation criterion, namely classification accuracy, to identify the best classifier.

5.2. Second-stage classification

Because the two classes (Severe and PDR) are so similar, the original Resnet is unable to distinguish them among the DR grades. A second stage Resent model is introduced to provide higher accuracy. Exclusively corrected Severe/PRD images from the first stage are fed to this stage, which is taught offline to identify only between the two classes (Severe and PDR). This stage's output is either "Severe" or "PDR."

V. RESULTS

Output – Following are the results:

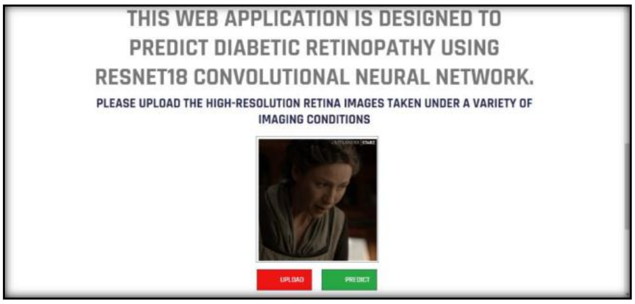

Fig.5. Developed website using HTML

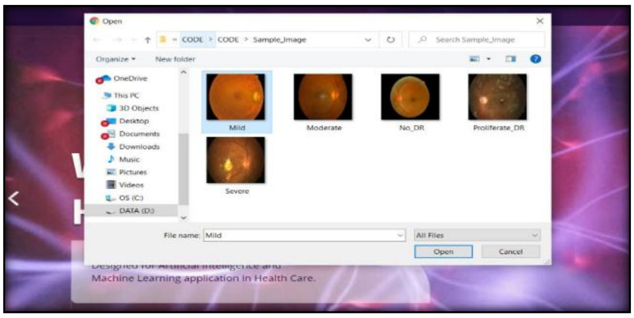

Fig.6. Prompt to upload the input sample image

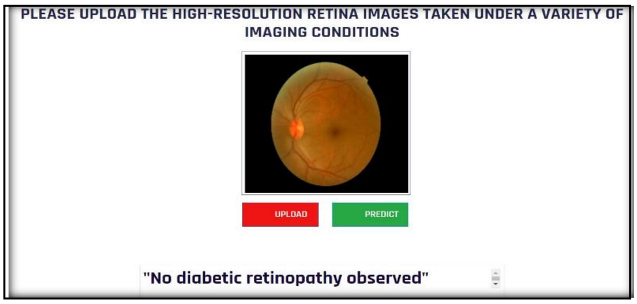

Fig.7. Prediction for No DR. "No diabetic retinopathy observed

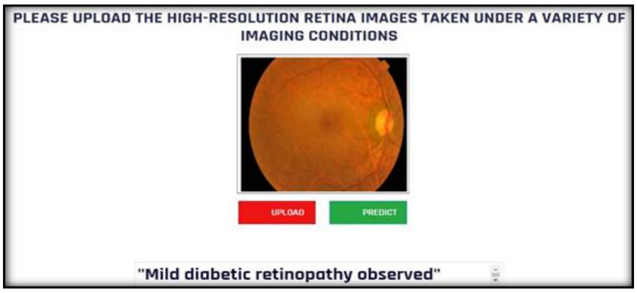

Fig.8. Prediction for Mild DR. "Mild diabetic retinopathy observed

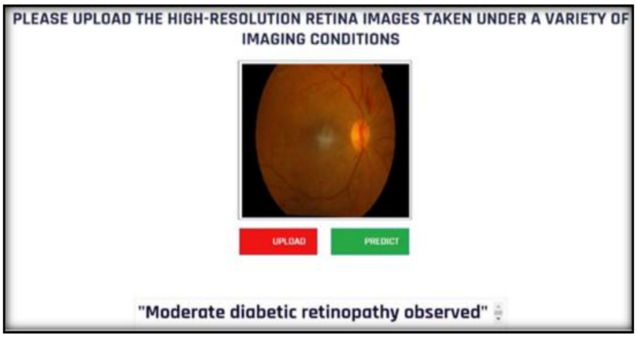

Fig.9. Prediction for Moderate DR. "Moderate diabetic retinopathy observed

Fig.10. Prediction for Severe DR. "Severe diabetic retinopathy observed

Fig.11. Prediction for Proliferative DR. "Proliferate diabetic retinopathy observed

VI. CONCLUSION

It's worth noting that classifying Diabetes Retinopathy based on fundoscopic images is not an easy task, even for a highly trained human specialist. The results of our study show that CNNs are beneficial and effective in staging Diabetes Retinopathy. Although the accuracy of Resnet50 predictive models is adequate given the short training data quantity, there is still much opportunity for improvement in the future. This publication is the first step in laying the groundwork for additional research into machine learning-based classifiers for Diabetes Retinopathy staging. We hope that our efforts will result in a valuable software-based tool for ophthalmologists to assess the severity of diabetes mellitus by recognising distinct stages of Diabetes Retinopathy, which will aid in the right management of Diabetes Retinopathy prognosis. This project takes input as fundus image and processes it to detect Diabetes Retinopathy. Project can run on any Python (Anaconda) installed- platform and can manage well with stack of fundus images, given one at a time. The result stage will diagnose the type of damage caused to retina.

VII. ACKNOWLEDGEMENT

The authors sincerely acknowledge Department of Electronics and Instrumentation, BMSCE and our instructor KRISHNA MURTHY K.T for his guidance.

VIII. REFERENCES

[1] G. Danaei, M. M. Finucane, Y. Lu, G. M. Singh, M. J. CowanC. J. Paciorek, J. K. Lin, F. Farzadfar, Y.-H. Khang, G. A. Stevens, M. Rao, M. K. Ali, L. M. Riley, C. A. Robinson, and M. Ezzati, "National, regional, and global trends in fasting plasma glucose and diabetes prevalence since 1980: systematic analysis of health examination surveys and epidemiological studies with 370 country-years and 2·7 million participants," The Lancet, vol. 378, issue 9785, 2011, pp. 31- 40.

[2] L. Wu, P. Fernandez-Loaiza, J. Sauma, E. Hernandez-Bogantes, and M. Masis, "Classification of diabetic retinopathy and diabetic macular edema," World Journal of Diabetes, vol. 4, issue 6, Dec. 2013, pp. 290- 294.

[3] C. P. Wilkinson, F. L. Ferris, R. E. Klein, P. P. Lee, C. D. Agardh, M. Davis, D. Dills, A. Kampik, R. Pararajasegaram, J. T. Verdaguer, and G. D. R. P. Group, "Proposed international clinical diabetic retinopathy and diabetic macular edema disease severity scales," Ophthalmology, vol. 110, issue 9, Sep. 2003, pp. 1677-1682.

[4] T. Y. Wong, C. M. G. Cheung, M. Larsen, S. Sharma, and R. Simó, "Diabetic retinopathy," Nature Reviews Disease Primers, vol. 2, Mar. 2016, pp. 1-16.

[5] H. Zhong, W.-B. Chen, and C. Zhang, "Classifying fruit fly early embryonic developmental stage based on embryo in situ hybridization images," in Proceedings of IEEE International Conference on Semantic Computing (ICSC 2009), IEEE, Sep. 2009, pp. 145-152.

[6] J. D. Osborne, S. Gao, W.-B. Chen, A. Andea, and C. Zhang, "Machine classification of melanoma and nevi from skin lesions," in Proceedings of the 2011 ACM Symposium on Applied Computing (SAC 2011), ACM, Mar. 2011, pp. 100-105.

[7] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," Medical image analysis, vol. 42, Dec. 2017, pp. 60-88.

[8] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," IEEE transactions on medical imaging, vol. 35, issue 5, May 2016, pp. 1207-1216.

[9] Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," Nature, vol.542, issue 7639, Feb. 2017, pp.115-118.

[10] P. Wilkinson et al, "Proposed International Clinical Diabetic Retinopathy and Diabetic Macular Edema Disease Severity Scales ", Ophthalmology 2003;110: 1677–1682 ,2003 by the American Academy of Ophthalmology.

AUTHOR PROFILE

KRISHNA MURTHY K.T

Assistant Professor
Department of Electronics and Instrumentation, BMSCE