International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 06 | June 2020 www.irjet.net p-ISSN: 2395-0072

Detecting Diabetic Retinopathy using Convolutional Neural Network

Raghavendra Sanem[1], SriGanga Savalagi[2], Manisha Allenki[3], Dr. S Krishna Anand[4]

[123]*Junior Research Scholar, Sreenidhi Institute of Science and Technology, Hyderabad – 501301, Telangana, India.*
[4]*Professor, Dept. of Electronics and Computer Engineering, Sreenidhi Institute of Science and Technology, Hyderabad – 501301, Telangana, India.*

Abstract - *The condition that leads to loss of vision because of increased blood's glucose level is Diabetic retinopathy. This happens because of the blood leakage around the retina as the blood vessels become too weak. Around 40 million people suffer from this disease and in India alone there are 7 million patients. The leading cause of blindness is Diabetic retinopathy in the world as there are very less ophthalmologists. In the United states there are around 75 ophthalmologists for every million people but in India there are only 11. Out of 12 thousand ophthalmologists in India only 3 thousand are specialized in retina. In the early stages of this condition its symptomless but only then its treatable. Our project is to accelerate the detection of this disease by building an AI model using a concept called image recognition, by implementing deep learning techniques. In deep learning, artificial neural Networks which mimic the functionality of the cognition by simulating a neural network similar to the interconnected human brain cells are used. The model will then be able to recognize the blood leakages when the images of the eyes are fed into it and precautionary measures must be taken at as early as possible thereby reducing the chances of vision loss among people.*

Key Words: Diabetic retinopathy, Convolutional Neural Networks, deep learning etc.

1. INTRODUCTION

Diabetes is a disease which causes abnormal fluctuation in the sugar level of human blood. People suffering from diabetes have either less production of insulin or body being not able to use the total amount of insulin produced. Hence, Diabetes affects the metabolism of the victim. High amounts of sugar in human blood may result in formation of ruptures in the blood vessels present in the human retina which eventually leads to a condition called Diabetic Retinopathy. Retina is a thin layer of tissue that is at the back of the human eye on the inside which is connected to the optic nerve. The routine of retina is to receive the light captured by the eye then transmit the light in the form of signals to the brain for visual recognition. The ruptures in the blood vessels due to high sugar level will result in leakage of blood vessels which result in cell loss in the retina. High cell loss in the retina will eventually result in eye blindness in the worst case. It is a widely spread and severe disease which results in loss of eye vision. The disease's risk is found to increase as age grows. Hence, older and middle aged are more likely to be affected by Diabetic Retinopathy. The transition from zero retinopathy to Proliferative Diabetic Retinopathy might take twenty years or more and this slow rate helps in identifying Diabetic Retinopathy and can be treated as early as possible. The progression and development of Diabetic Retinopathy are dependent on the duration and how one controls diabetes. DR in its early stage is responsive to treatment as it is usually asymptomatic. Diabetic retinopathy is of four stages:

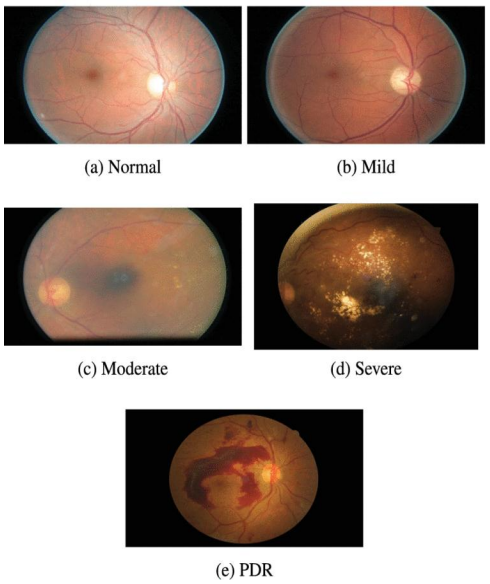

Fig -1 Diabetic Retinopathy Stages

2. RELATED WORK

With the help of fundoscopy diagnosis of any pathological findings can be performed with a technique. It is by getting the retinal view of human eye depending on a decent range of features in the provided fundus image. For the victims in early stage the disease is very hard to be diagnosed because at early the disease appearance is dependent on visibility of the presence of micro-aneurysms which are small swells or pouches formed in the blood vessel of the eye. These micro-aneurysms will

further lead to blood vessel ruptures and thereby leakage of the blood.

With no symptoms shown by the victim during the early stage of the disease, tracing of such pathological formations in the retina is a real big burden to the ophthalmologists. Machine based techniques were used to diagnose diabetic retinopathy in earlier days. As a result of humongous amounts of researches there came the first artificial neural networks that is capable to classify patches of normal retina with many conditions provided. The accuracy in finding the micro- aneurysms from the normal patches of retina by the first deployed neural networks was found to be 74%.

Top hat algorithm was also used to detect the presence of micro-aneurysm. This was achieved by performing various studies on high bias, low variance digital image processing techniques. But then, various features of the eye would show an effective result in detecting the disease.

Apart from the traditional methods, additional methods were also used to detecting micro- aneurysms and grading DR like k-NN algorithm, support vector machines, and ensemble- based method. These techniques managed to get the sensitivities and specificities within the 90% range by extracting various features of the eye.

Prior studies on CNN for DR fundus images seemed to achieve sensitivities and specificities in the range of 90% for binary classification that merges two classes into one that is healthy or sensible stage of the disease versus rigorous or fully diseased stage. This was performed on very bulk amount of datasets ranging from 80,000 to 120,000 images. However, accuracy measures for the detection of four classes of DR became quite a difficult task with sensibilities and specifications. This project will determine the sensitivity and specificity of the 4-ary classification model and evaluate performance by comparing results to currently published research data.

3. PROPOSED MODEL

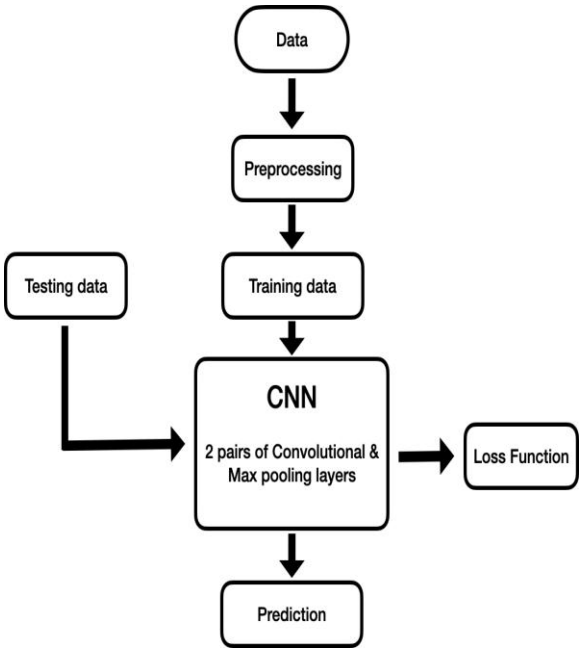

Fig -2 Flowchart

4. DATAFRAME CREATION & PREPROCESSING

4.1 Dataset

A huge data set consisting of retina images with very high resolution has been taken with various imaging conditions from Kaggle. Each subject's left and right detail is provided in the dataset as shown in Fig.3. Labeling of images is done with an id describing whether it is a left or right eye (e.g. 1_left.jpeg is that the left eye of patient id 1).

trainLabels

image	level
10_left	0
10_right	0
13_left	0
13_right	0
15_left	1
15_right	2
16_left	4
16_right	4
17_left	0
17_right	1
19_left	0
19_right	0
20_left	0
20_right	0
21_left	0
21_right	0

Fig -3 Train Labels

Fig -4 Train Images

4.2 Preprocessing

Here, the data set was taken from an online platform named Kaggle. The size of the data set was trimmed to 350 images. Before feeding the input directly into the model, the data which is the set of fundus images must undergo some preprocessing steps which includes i. resizing of images size from 3888 *2951.to 786 * 786 dimension. ii. perform flip-flop operations which are rotating the fundus image by 90 degrees. Flipping of images is done in order to exercise the model in an efficient way. The input data set is classified into three different categories. They are a. Training dataset, the dataset that is used to train or exercise the model. This data is labeled data set. b. Testing data set, which is used to test the model. c. Validation data set, the dataset that is used to validate the model. Validation data set is used to ensure that the model is not over-fitting whereas training data helps to minimize the loss function. Updating of weights happens accordingly when the training data set is exercised in the model but validation data set does not involve any updating process. Training dataset and validation dataset are labeled but not the testing dataset. Also, one hot encoding is performed on the training labels.

4.3 Training and Testing Data

After preprocessing is done, the dataset is divided into two parts as Training and Testing.

The training data is used to train the model whereas, the testing data is used to validate the model.

5. MODEL ARCHITECTURE - CNN Layers

The architecture consists of six layers: 1 Input layer, 2 pairs of Convolutional layers & Max Pooling Layers, 1 Output Layer.

1. Input Layer
This layer consists of 786 x 786 neurons which is equal to the count of pixels of each individual image being passed. Here, the pixels values of the training images are sent to the input layer.

2. Convolutional Layer 1
This layer consists of 32 neurons. There is a connection between each of the neurons present in this layer to all of the neurons in the previous layer. Convolution is performed on the input pixels, which is a process of performing dot product on the pixel values with arbitrary numbers called as filters. So, the layer's output is further passed to the max pooling layer.

3. Max Pooling Layer 1
With the filters provided max pooling operation is performed on the received input which is identification of highest value in each patch of feature map.

4. Convolution Layer 2
The max pooling layer's output is concatenated to a convolution layer (convolution layer 2) with 16 filters, kernel size as 4*4 and activation function as ReLu. This layer is further passed to a max pooling layer.

5. Max Pooling Layer 2
The max pooling layer performs the max pooling operation on the received input.
Then the output of the max pooling layer is flattened. Flattening is a process of converting any matrix into one dimensional array. Flatten function is applied on the convolution layer to create a single long feature vector.

6. Output Layer
The total amount of neurons existing in this layer is equal to the number of levels the disease is classified into. The neuron consisting of the maximum value ranging between 0-1 will be the output i.e., the level in which the disease is. This output will be compared with the actual values and the error is determined. Based on the error the model tunes its underline parameters such that the error is as minimum as possible. This operation is performed on each and every training image.

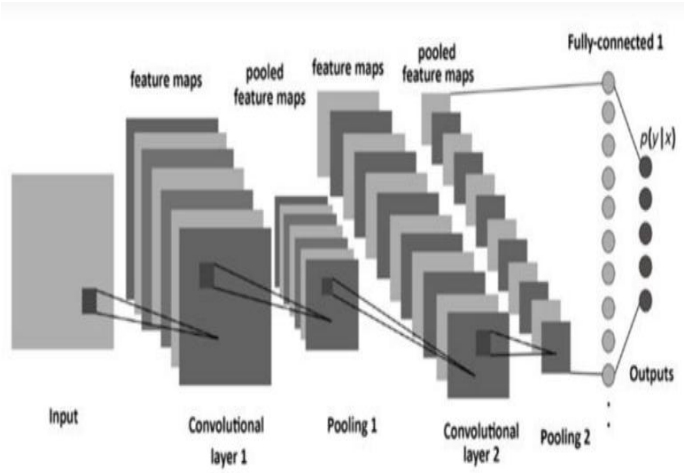

Fig -5 Model

5.1 Optimization Technique

Adam is a flexible learning rate optimization algorithm. It was designed exclusively for the sake of deep neural networks' training. Adam was presented in 2014, the presentation consisted of assuring diagrams, displaying massive performance improvements with respect to speed of training. Nonetheless, after some time people started observing, that Adam finds the worse solution than "stochastic gradient descent" in some cases.

The algorithms leverage the adaptive learning rates methods power to find single learning rates for every parameter. The pros of Ad grad are also present, which performs good in sparse gradients settings, but in neural networks' non-convex optimization it struggles, and RMSprop, that tackles to reduce few of the Ad grad problems and performs good enough coming to on-line settings.

5.2 Prediction

The built model is now evaluated using the testing data and the accuracy is computed which acts as the performance metric of the model.

6. TESTING

Testing is making certain if a tolerable job of precisely segregating data with adequate generalization is done by the model and avoiding any overfitting or underfitting issues. Validation techniques are used for this purpose and withholding a chunk of the training data is used for the validating.

6.1 Unit Testing

Unit Testing is used to test the code at component level. It compares the actual output with the expected output. This has scope to component level as the range of scope.

DaraFrame Testing:

```
df_train = pd.read_csv("~/Desktop/trainLabels.csv")
```

```
df_train.values
```

```
array([['10_left', 0],
       ['10_right', 0],
       ['13_left', 0],
       ...,
       ['44348_right', 0],
       ['44349_left', 0],
       ['44349_right', 1]], dtype=object)
```

```
df_train.tail()
```

	image	level
35121	44347_right	0
35122	44348_left	0
35123	44348_right	0
35124	44349_left	0
35125	44349_right	1

Fig -6 Data Frame Testing

DataFrame creation is the first and important unit of the project. Testing on this unit is performed by checking the tail value of the created data frame.

One Hot Encoding Testing:

Hot encoding the categorical labels given in the DataFrame by checking the first ten records after hot encoding performed.

```
targets_series = pd.Series(df_train['level'])
one_hot = pd.get_dummies(targets_series, sparse = True)
```

```
targets_series[:10]
```

```
0    0
1    0
2    0
3    0
4    1
5    2
6    4
7    4
8    0
9    1
Name: level, dtype: int64
```

```
one_hot[:10]
```

	0	1	2	3	4
0	1	0	0	0	0
1	1	0	0	0	0
2	1	0	0	0	0
3	1	0	0	0	0
4	0	1	0	0	0
5	0	0	1	0	0
6	0	0	0	0	1
7	0	0	0	0	1

Fig -7 One Hot Encoding Testing

6.2 Integration Testing

The main goal of Integration Testing is to detect the presence of any inconsistencies between the units that are integrated together. In simple terms it makes sure that the Individual units work fine alone and also when integrated together called assemblages. The dependent unit in the assemblage must get the correct input from the independent unit.

Model Architecture Testing:

By checking the number trainable and non-trainable parameters, the architecture can be validated.

```
model.summary()
```

```
Model: "model_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_6 (InputLayer)         (None, 786, 786, 3)       0
_________________________________________________________________
conv2d_11 (Conv2D)           (None, 783, 783, 32)      1568
_________________________________________________________________
max_pooling2d_11 (MaxPooling (None, 391, 391, 32)      0
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 388, 388, 16)      8208
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 194, 194, 16)      0
_________________________________________________________________
flatten_6 (Flatten)          (None, 602176)            0
_________________________________________________________________
dense_11 (Dense)             (None, 10)                6021770
_________________________________________________________________
dense_12 (Dense)             (None, 5)                 55
=================================================================
Total params: 6,031,601
Trainable params: 6,031,601
Non-trainable params: 0
_________________________________________________________________
```

Fig -8 Model Architecture Testing

6.3 System Testing

The testing which is conducted on a whole integrated system in order to evaluate the compliance of the system with its specified requirements is called System Testing. The software is tested against inputs, and changes are made to the model to increase its accuracy.
The built model's overall performance is defined by its accuracy.

```
test_acc
```

```
0.8888888955116272
```

Fig -9 System Testing

7. RESULTS AND CONCLUSION

Diabetes is one among the burgeoning diseases in today's world. Based many surveys, a diabetes patient has about 0.3 probability to be affected with Diabetic Retinopathy (DR). It starts from mild symptoms and advances to severe and then finally PDR (Proliferative Diabetic Retinopathy). As the disease advances in stages, it causes blurred vision, floaters and finally can make the patient blind if its diagnosis is ignored in the initial stages. Physical diagnosis of these images needs highly trained experts and is a tedious, delayed and difficult process. For detecting this disease in its different stage's computer vision-based approaches have been suggested in the literature. This project is concentrated on classifying all stages of Diabetic Retinopathy, exclusively the early stages, as it is the major defect of existing techniques. The proposed solution is a CNN based framework to diagnose and classify the

different stages of the back of retina called fundus images. The results present that this model can spot the stages of the disease and also out functions other modern methods. Therefore, the deployed model is able to perform classification with an accuracy of 88.88%. In future, we plan to train specific models for specific stages and then consider those outcomes together in order to increase the accuracy of early stages. Finally, after testing the accuracy of 88.88% is obtained.

```
x_valid_raw = np.array(X_valid)

y_valid_raw = np.array(Y_valid)

test_loss, test_acc = model.evaluate(x_valid_raw, y_valid_raw)

18/18 [==============================] - 2s 122ms/step

test_loss

1.9940619468688965

test_acc

0.8888888955116272
```

Fig -10 Results

8. FUTURE WORK

Since anomalies emerge and disappear on the retina, diabetic retinopathy is considered a continuous disease. Patients have frequent dilated retinal exams, hence their images from earlier visits will be available. By evaluating the continuous deformities in the fundus images over a certain period, exquisite oddities could be spotted. Practically this implies comparing the older data with the images most recently taken and analyzing the differences between them. Temporal change detection is an approach that could be used for automatic screening in the distant future. Retinal registration is a requirement for the advancement of a variance detection algorithm. By developing algorithms for retinal change detection and retinal image registration the Fundus Image Analysis system can be improved, which consequently aids the ophthalmologist to investigate the post medication advances.

REFERENCES

[1] W. R. Memon, B. Lal, A. A. Sahto, "Diabetic retinopathy", The Prof. Med. J., Vol. 24, pp. 2017, 234-238.

[2] J. Amin, M. Sharif, M. Yasmin, H. Ali, S. L. Fernandes, "A method for the detection and classification of diabetic retinopathy using structural predictors of bright lesions", J. Comput. Sci., Vol. 19, pp., Mar. 2017, 153-164.

[3] L. Seoud, T. Hurtut, J. Chelbi, F. Cheriet, J. M. P. Langlois, "Red lesion detection using dynamic shape features for diabetic retinopathy screening", IEEE Trans. Med. Imag., Vol. 35, No. 4, pp., Apr. 2016, 1116-1126.

[4] R. Gargeya, T. Leng, "Automated identification of diabetic retinopathy using deep learning", Ophthalmology, Vol. 124, No. 7, pp., 2017, 962-969.

[5] G. Quellec, K. Charrière, Y. Boudi, B. Cochener, M. Lamard, "Deep image mining for diabetic retinopathy screening", Med. Image Anal., Vol. 39, pp., Jul. 2017, 178-193.

[6] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms", Pattern Recognit., Vol. 30, No. 7, pp., 1997, 1145-1159.

[7] S. Haneda, H. Yamashita, "International clinical diabetic retinopathy disease severity scale", Nihon Rinsho. Jpn. J. Clin. Med., Vol. 68, pp., Nov. 2010, 228-235.

[8] R. A. Welikala, M. M. Fraz, J. Dehmeshki, A. Hoppe, V. Tah, S. Mann, T. H. Williamson, "Genetic algorithm-based feature selection combined with dual classification for the automated detection of proliferative diabetic retinopathy", Computerized Med. Imag. Graph., Vol. 43, pp., Jul. 2015, 64-77.

[9] F. Mercaldo, V. Nardone, and A. Santone, “Diabetes mellitus affected patients classification and diagnosis through machine learning techniques,”Procedia Comput. Sci., Vol. 112, pp. , Sep.2017, 2519_2528.

[10] V. V. Vijayan and C. Anjali, ``Prediction and diagnosis of diabetesmellitus_A machine learning approach,'' in Proc. IEEE Recent Adv. Intell.Comput. Syst. (RAICS), Dec. 2016, pp. 122-127.

[11] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting", Journal of Machine Learning Research, Vol. 15, 2014, 1929–1958.

[12] https://www.kaggle.com/c/diabetic-retinopathy-detection/data.

[13] https://brailleworks.com/diabetic-retinopathy/

[14] https://www.nei.nih.gov/health/diabetic/retinopathy

[15] https://www.analyticalvidhya.com/blog/2018/12/guide-convolutional-neural-network