DOI: 10.17148/IJARCCE.2023.125163

Convolutional Neural Networks for Diabetic Retinopathy Detection

Karthik Raj S L[1], Sahana R[2], Sahana S[3], Simran G[4], Akash Anil Kumar[5]

Assistant Professor, Department of ISE, The Oxford College of Engineering, Bangalore, India[1]

Student, BE, Department of ISE, The Oxford College of Engineering, Bangalore, India [2-5]

Abstract: Diabetes Mellitus, also known as diabetes, causes persistently high blood sugar levels that can lead to diabetic retinopathy, which has been linked to damage to the retina's tiny blood vessels. The retina must first register light before the optic nerve can transmit signals to the brain. Until diabetic retinopathy begins to advance towards Proliferate DR/PDR, treatment is frequently postponed. As Diabetic Retinopathy (DR) worsens, more regular comprehensive dilated eye exams are required. Severe non-proliferative diabetic retinopathy patients are at a high risk of developing PDR and may require a thorough dilated eye exam every two to four months. [1] Therefore, in our work, we constructed a model called "retina.model" that can recognise even the slightest variation between each stage of DR and is 100% reusable with a growing amount of cognition over time as the computer tries to learn new patterns.

Keywords: matrix handling, Diabetes , American Optometric Association (AOA), Deep Learning, CNN architecture, Diabetic Retinopathy (DR), Image Classification, retina of the eye, Optometrist, Gaussian filters, Mild DR and Moderate DR, "retina.model", Severe non-proliferative diabetic retinopathy and Cotton wool spots.

I. INTRODUCTION

Any type of diabetes, including type 1, type 2, and gestational, can cause diabetic retinopathy. Risk increases with diabetes duration. Between 40 and 45 percent of Americans with diabetes have diabetic retinopathy, yet only about half of them are aware of it. In pregnant women who develop or have diabetes, diabetic retinopathy may quickly worsen or develop. There are situations when diabetic retinopathy results in permanent vision loss. However, early detection and intervention can greatly reduce the risk of blindness by 95%. Diabetes patients should have a comprehensive dilated eye exam at least once a year because diabetic retinopathy frequently shows no early signs. More frequent eye exams may be necessary for those with diabetic retinopathy. A comprehensive dilated eye exam should be performed as soon as possible once a woman with diabetes learns she is expecting. Additional tests may be required to diagnose gestational diabetes, which could become permanent throughout pregnancy. The American Optometric Association (AOA),[2–5] who also provided the reference dataset for training that would automate the process of detection and include the feature of intuition and perception based on AI Architecture for prediction, is where the image set was obtained from.[6-8]

II. STRUCTURE OF THE DR EYE

Until it has progressed to an advanced state, diabetic retinopathy typically does not show any symptoms. It might cause abrupt blindness if it is not diagnosed and treated. The main signs of diabetic retinopathy are rapid changes in vision, hazy vision, gradual loss of vision over time, eye pain, double vision, floaters in the field of vision, and difficulty seeing at night.[8][9] Figure 3 illustrates how many persons with early diabetic retinopathy experience no symptoms prior to significant eye bleeding. Early diabetic retinopathy, also known as non-proliferative, is characterised by smaller blood vessels in the eye, occasionally blocked blood vessels, retinal hemorrhages, and fluid leakage into the retina. Small scars form on the retina and in other areas of the eye in more proliferative, more severe retinopathy. We may also see new, fragile blood vessels beginning to emerge in the eye. Microvascular retinal alterations are a result of diabetic retinopathy. [10-15] Vascular walls become ineffective as a result of intramural pericyte mortality brought on by hyperglycemia and thickening of the basement membrane. These cause harm to the blood-retinal barrier and increase the permeability of the retinal blood vessels. Small blood vessels in the eye are particularly sensitive because they are damaged when glucose and/or fructose are accumulated excessively. Most people do not detect any change in their vision during the non-proliferative diabetic retinopathy stage, which is the first stage. Simplex retinopathy is the word occasionally used to describe early, reversible abnormalities. Macular edoema is a condition that some people get. [11][16] wherein the blood arteries get damaged, causing fluid and lipid leaking onto the macula, a portion of the retina. The macula swells as a result of this fluid, causing vision blur. Figure 1 depicts the anomalies of a DR eye. They are mostly [11] [17]

i. Cotton wool spots
ii. Aneurysm
iii. Hemorrhages
iv. Hard exudates
v. Abnormal growth of blood vessels

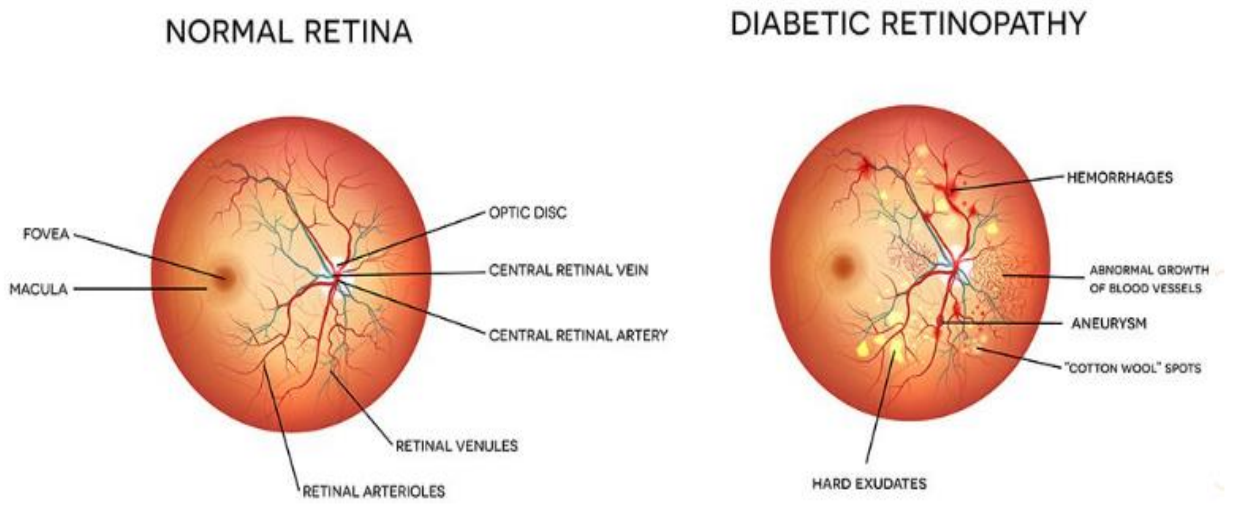


Figure 1 shows the anatomy of a normal eye and Diabetic Retinopathy eye.

III. EXISTING CNN MODELS

i. Inception V3: The Inception architecture was proposed by Szegedy et al. in 2014. The initial design was known as GoogLeNet. Inception Vn (n is the version number) was the name given to all subsequent iterations. Inception V2 enhanced Inception V1 by including Batch Normalization. Inception introduced V3 model factorization techniques as an advancement over V2.[6-9]

ii. ResNet50: The Residual Networks architecture was proposed by He and colleagues in 2015. It features 50 convolutional layers with skip connections, which contribute to the model's improved learning accuracy. Additionally, the model size is decreased by using global average pooling rather than completely connected layers.[10].

iii. MobileNet: Howard et al. suggested MobileNet, a new CNN architecture in 2017. Each colour channel in this separable convolution is treated independently rather than as a whole, and the separable convolution has been grouped depth-wise. In this architecture, computation costs are decreased.[11][12]

iv. Xception: In 2017, François Chollet created Xception. Since depth-wise separable convolutions have been used in place of Inception's modules, this model might be thought of as an improvised version of Inception. Speed and precision are strengths of this most recent and reliable type.[18]

IV. PROBLEM STATEMENT

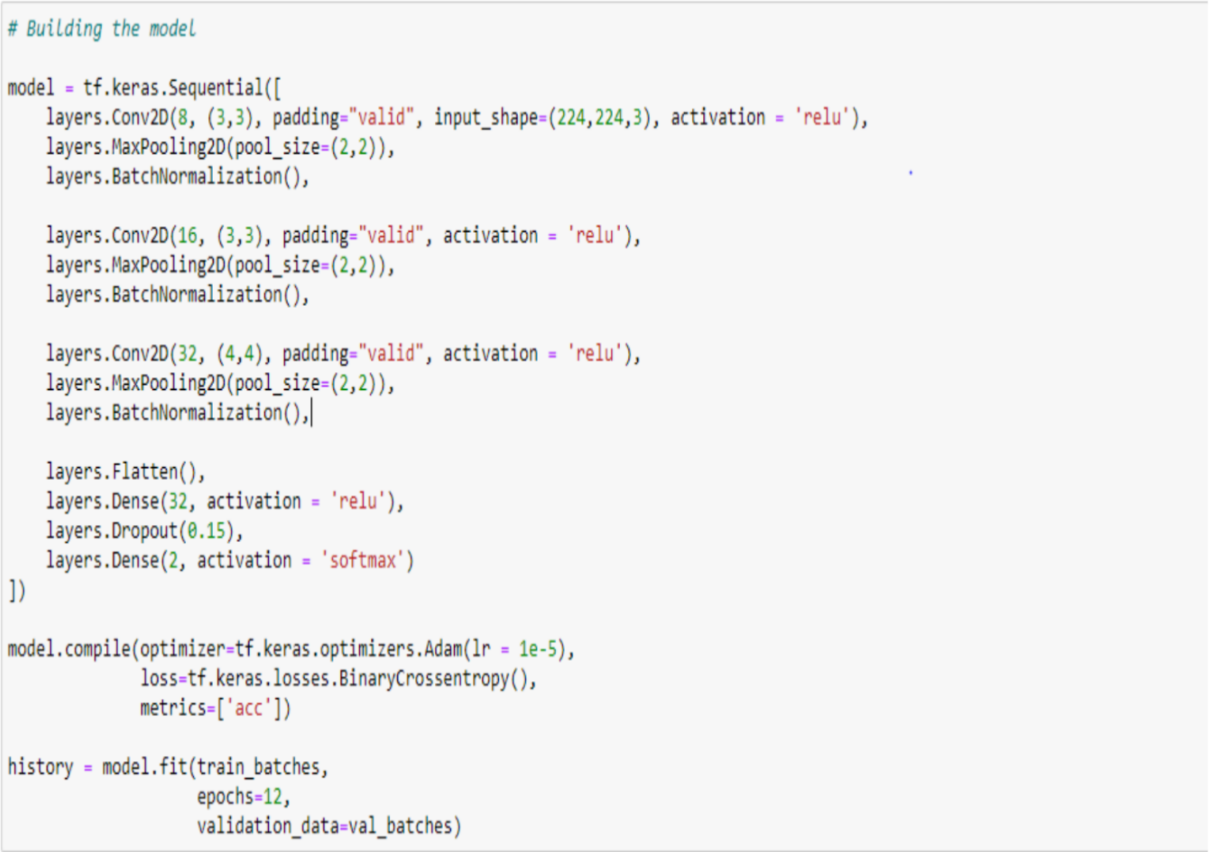
We forecast the stage of diabetic retinopathy using fundus picture images and CNN's Keras_h5 model. The primary objective of the study is to identify diabetic retinopathy as early as possible to avoid blindness. By classifying retinal images of the patient into five labels ranging from 0 to 4, each label named Normal, Mild DR, Moderate DR, Severe DR, and Proliferate DR represents a disease complication, we can identify the disease's complication using deep transfer learning and classification techniques[12]. One of these five stages is recognised as an output label for the supplied input fundus image. For Normal, Mild DR, Moderate DR, Severe DR, and Proliferate DR, predictions for real-time photos or fresh images of newer subjects are necessary. Additionally, the CNN model needs to be revalidated using the feedback method for increased accuracy.

This project work is divided into four parts
First. Image Acquisition, categorization and applying filters [29]
Second. Image/ data Pre-processing, array formation and matrix handling.[1]
Third. Using CNN and validating 'retina.model' for Prediction.
Fourth. Testing on real time images and verification.

DOI: 10.17148/IJARCCE.2023.125163

V. METHODOLOGY

As part of the training process, the images are passed through the network by a number of filters applied at each layer when we train a deep convolutional neural network using a dataset of previously processed images. The values of the filter matrices are multiplied by the activations of the image at each layer. The class to which the image belongs is determined by the activations of the final layer. The optimal values for each of these filter matrices should be chosen while training a deep network in order to successfully identify the class to which an image belongs. In this manner, the network's output activations can accurately determine the class to which an image belongs. The technique utilised to calculate these filter matrix values is gradient descent. We can observe something pretty intriguing based on perception when we train a conv net on the ImageNet dataset and then check what each filter on the conv net has learned to detect or what triggers it. The filters on the first few layers of the convolutional network distinguish colours and certain horizontal and vertical lines. [19-22] Using the lines and colours discovered in the previous layers, the following layers gradually learn to recognise insignificant shapes. Following that, the subsequent layers pick up on textures, and after that, object parts like legs, eyes, noses, etc. The output is then produced when the last layer's filters are activated by complete objects. We can learn how to combine previously learned characteristics to recognise the objects in new datasets by utilising a pre-trained network for transfer learning. This is done by simply adding a few dense layers to the end of the pre-trained network.[23][24][29]



```
# Building the model

model = tf.keras.Sequential([
    layers.Conv2D(8, (3,3), padding="valid", input_shape=(224,224,3), activation = 'relu'),
    layers.MaxPooling2D(pool_size=(2,2)),
    layers.BatchNormalization(),

    layers.Conv2D(16, (3,3), padding="valid", activation = 'relu'),
    layers.MaxPooling2D(pool_size=(2,2)),
    layers.BatchNormalization(),

    layers.Conv2D(32, (4,4), padding="valid", activation = 'relu'),
    layers.MaxPooling2D(pool_size=(2,2)),
    layers.BatchNormalization(),

    layers.Flatten(),
    layers.Dense(32, activation = 'relu'),
    layers.Dropout(0.15),
    layers.Dense(2, activation = 'softmax')
])

model.compile(optimizer=tf.keras.optimizers.Adam(lr = 1e-5),
              loss=tf.keras.losses.BinaryCrossentropy(),
              metrics=['acc'])

history = model.fit(train_batches,
                    epochs=12,
                    validation_data=val_batches)
```

Figure 2 shows the process of building the keras model.

The activation function in a neural network is in charge of converting the node's summed weighted input into the activation of the node or output for that input. If the input is positive, the rectified linear activation function, or ReLU for short, will output the input directly; if it is negative, it will output zero. It has evolved into the standard activation function for our model "retina.model" because it is simpler to train and frequently results in higher performance. Due to the possibility of interpreting the outcome as a probability distribution, Softmax is frequently employed as the activation for the final layer of a classification network.[25-28] The method of creating the model is depicted in Figure 2, along with the configuration of the initial activation units with ReLU [29] and the final layers with softmax. Either 12 or 25 can be used as the epoch. An epoch is one cycle of training the neural network using all the training data. We only use each piece of information once within an epoch.

A pass is considered to be one pass if it is both forward and backward. Each epoch consists of one or more batches in which the neural network is trained using a portion of the dataset. We refer to the process of going through a batch of training examples as iteration. The operations throughout an epoch and an iteration are depicted in Figure 3. Figure 4 shows validation with 81 iterations in each of 12 epochs.

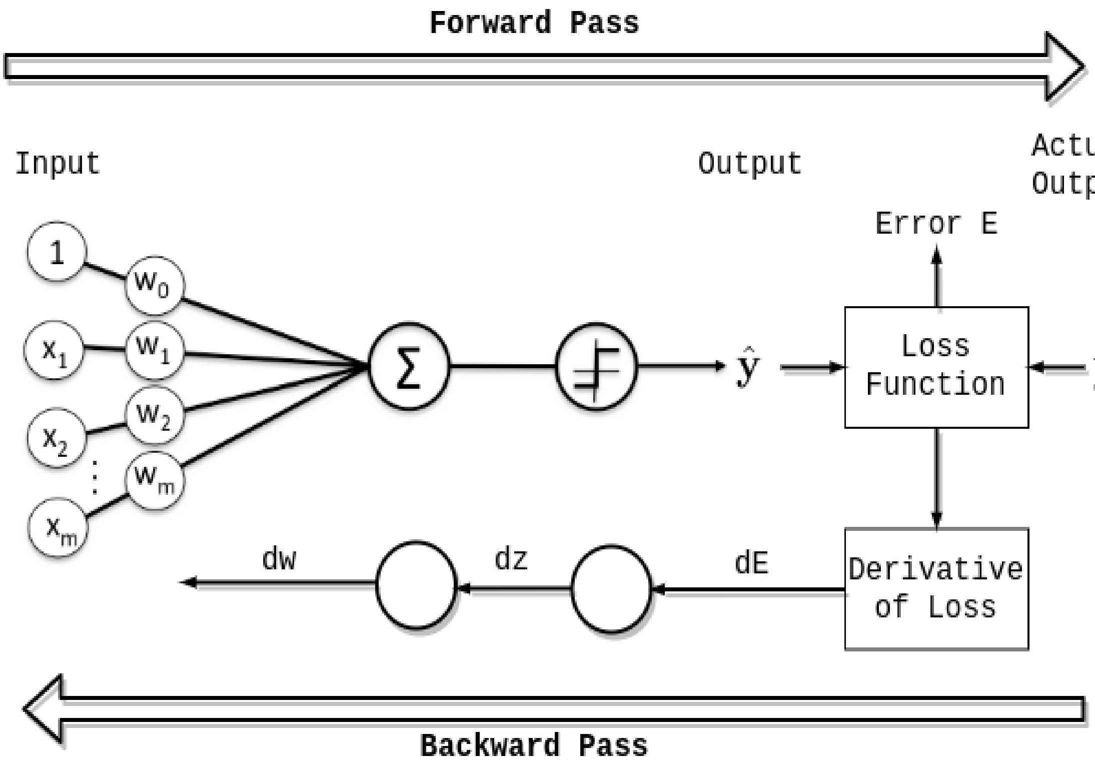
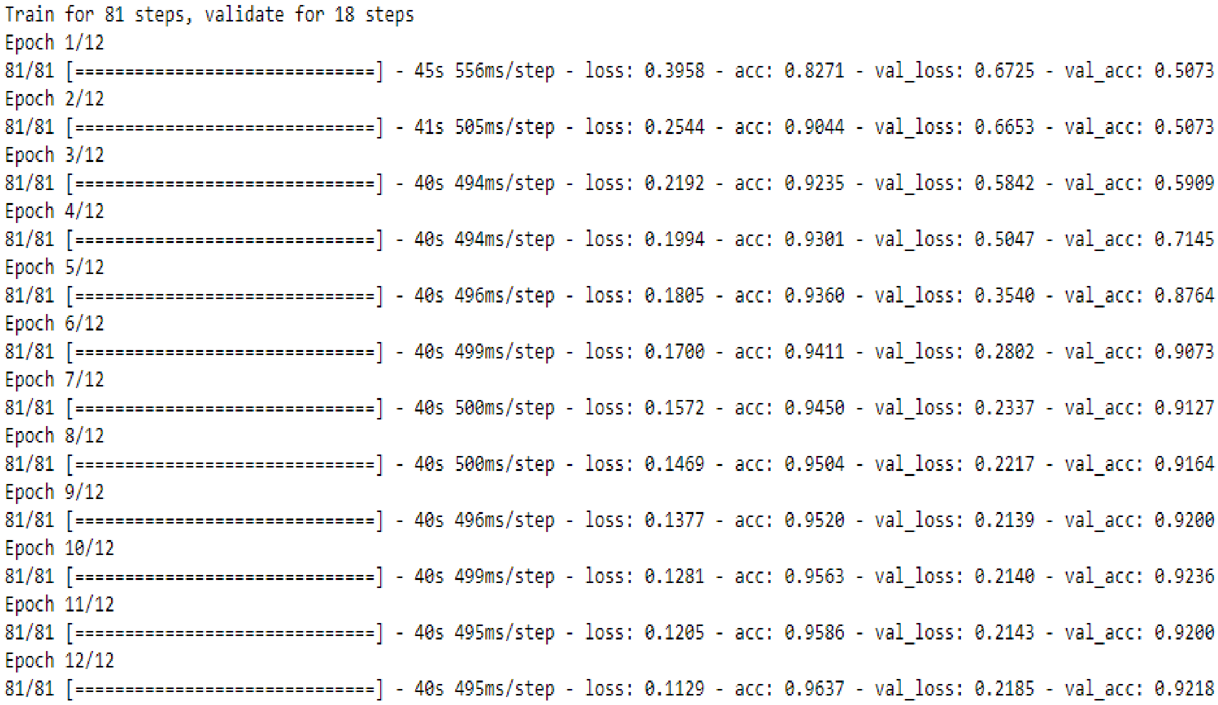


Figure 3 shows the functions during an epoch and an iteration.

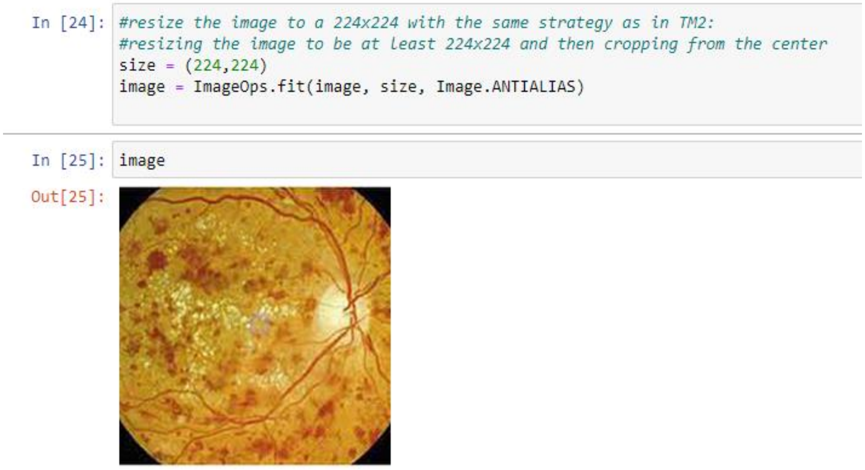


```
Train for 81 steps, validate for 18 steps
Epoch 1/12
81/81 [==============================] - 45s 556ms/step - loss: 0.3958 - acc: 0.8271 - val_loss: 0.6725 - val_acc: 0.5073
Epoch 2/12
81/81 [==============================] - 41s 505ms/step - loss: 0.2544 - acc: 0.9044 - val_loss: 0.6653 - val_acc: 0.5073
Epoch 3/12
81/81 [==============================] - 40s 494ms/step - loss: 0.2192 - acc: 0.9235 - val_loss: 0.5842 - val_acc: 0.5909
Epoch 4/12
81/81 [==============================] - 40s 494ms/step - loss: 0.1994 - acc: 0.9301 - val_loss: 0.5047 - val_acc: 0.7145
Epoch 5/12
81/81 [==============================] - 40s 496ms/step - loss: 0.1805 - acc: 0.9360 - val_loss: 0.3540 - val_acc: 0.8764
Epoch 6/12
81/81 [==============================] - 40s 499ms/step - loss: 0.1700 - acc: 0.9411 - val_loss: 0.2802 - val_acc: 0.9073
Epoch 7/12
81/81 [==============================] - 40s 500ms/step - loss: 0.1572 - acc: 0.9450 - val_loss: 0.2337 - val_acc: 0.9127
Epoch 8/12
81/81 [==============================] - 40s 500ms/step - loss: 0.1469 - acc: 0.9504 - val_loss: 0.2217 - val_acc: 0.9164
Epoch 9/12
81/81 [==============================] - 40s 496ms/step - loss: 0.1377 - acc: 0.9520 - val_loss: 0.2139 - val_acc: 0.9200
Epoch 10/12
81/81 [==============================] - 40s 499ms/step - loss: 0.1281 - acc: 0.9563 - val_loss: 0.2140 - val_acc: 0.9236
Epoch 11/12
81/81 [==============================] - 40s 495ms/step - loss: 0.1205 - acc: 0.9586 - val_loss: 0.2143 - val_acc: 0.9200
Epoch 12/12
81/81 [==============================] - 40s 495ms/step - loss: 0.1129 - acc: 0.9637 - val_loss: 0.2185 - val_acc: 0.9218
```

Figure 4 shows validation with 81 iterations in each of 12 epochs.

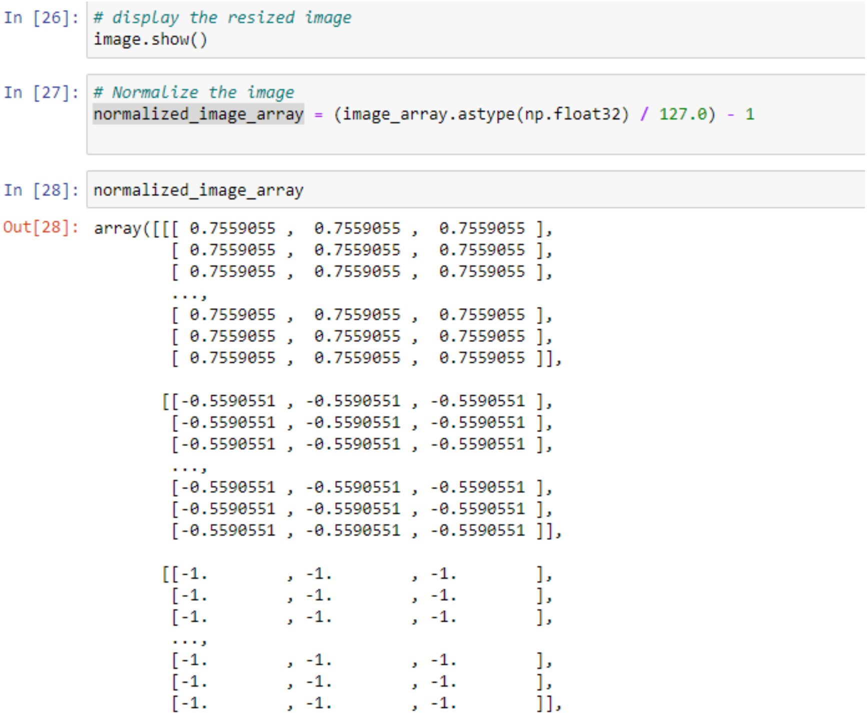
VI. RESULTS AND DISCUSSIONS

After building the model, the model is saved as 'retina.model' and can now be used to make predictions when any realtime image is fed to our project. Figure 5 shows a realtime photograph being fed to the system after resizing (224*224). Due the model the image gets converted into matrix array. Now the information is no more in Image format. Each row in the matrix contains encoded information of the image. Later the array obtained is normalized. Figure 6 shows the conversion of image to normalized matrix array. Figure 7 shows the prediction of DR present in the eye of the patient. Any image in .png, .jpg, .jpeg can be fed to our designed model and prediction is obtained with about 97% accuracy.



```
In [24]: #resize the image to a 224x224 with the same strategy as in TM2:
         #resizing the image to be at least 224x224 and then cropping from the center
         size = (224,224)
         image = ImageOps.fit(image, size, Image.ANTIALIAS)

In [25]: image
Out[25]:
```

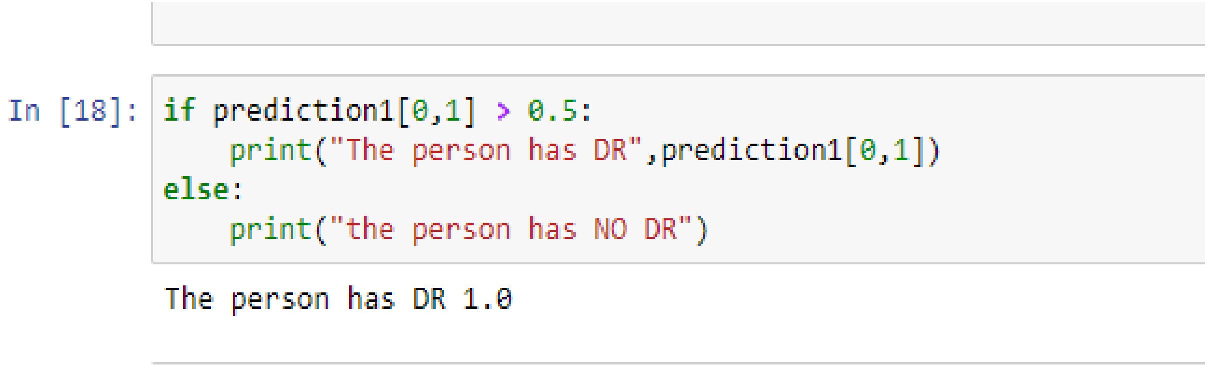
Figure 5 shows a realtime image being fed to the system.



```
In [26]: # display the resized image
         image.show()

In [27]: # Normalize the image
         normalized_image_array = (image_array.astype(np.float32) / 127.0) - 1

In [28]: normalized_image_array
Out[28]: array([[[ 0.7559055 ,  0.7559055 ,  0.7559055 ],
                 [ 0.7559055 ,  0.7559055 ,  0.7559055 ],
                 [ 0.7559055 ,  0.7559055 ,  0.7559055 ],
                 ...,
                 [ 0.7559055 ,  0.7559055 ,  0.7559055 ],
                 [ 0.7559055 ,  0.7559055 ,  0.7559055 ],
                 [ 0.7559055 ,  0.7559055 ,  0.7559055 ]],

                [[-0.5590551 , -0.5590551 , -0.5590551 ],
                 [-0.5590551 , -0.5590551 , -0.5590551 ],
                 [-0.5590551 , -0.5590551 , -0.5590551 ],
                 ...,
                 [-0.5590551 , -0.5590551 , -0.5590551 ],
                 [-0.5590551 , -0.5590551 , -0.5590551 ],
                 [-0.5590551 , -0.5590551 , -0.5590551 ]],

                [[-1.        , -1.        , -1.        ],
                 [-1.        , -1.        , -1.        ],
                 [-1.        , -1.        , -1.        ],
                 ...,
                 [-1.        , -1.        , -1.        ],
                 [-1.        , -1.        , -1.        ],
                 [-1.        , -1.        , -1.        ]],
```

Figure 6 shows the conversion of image to normalized matrix array.

DOI: 10.17148/IJARCCE.2023.125163



```
In [18]: if prediction1[0,1] > 0.5:
             print("The person has DR",prediction1[0,1])
         else:
             print("the person has NO DR")

         The person has DR 1.0
```

Figure 7 shows the prediction of DR present in the eye of the patient.

VII. CONCLUSIONS AND FUTURE WORK

Utilizing our trained CNN in real-time whenever a new image is acquired could be useful because it is capable of classifying thousands of photographs per minute. When a patient walks in for screening, in actuality, images are sent to doctors for grading and are not properly assessed. The trained CNN enables a quick diagnosis and a prompt response to a patient. The network used just one image per eye to get these results as well. The picture of a healthy eye is quickly recognised by the network. This is most likely a result of the dataset's abundance of healthy eyes. Much less learning was required to classify the photographs at the extreme ends of the spectrum during training. The low sensitivity, which primarily came from the mild and moderate classes, suggests that the network may have had trouble learning detailed enough properties to distinguish some of the more complicated components of DR. A related issue was identified and validated by a clinician: more than 10% of the images in our collection are classified as ungradable by national UK guidelines [24]. These images were categorised as a class because they include a minimum level of DR. Given that the images were misclassified for both training and validation, our results may have been substantially hindered.

REFERENCES

[1] Vishesh S, D S Pavan, Rishi Singh, Rakesh Gowda B, 2022. Prediction of Diabetic Retinopathy using Neural Networks.

[2] Saaddine, J.B., Honeycutt, A.A., Narayan, K.V., Zhang, X., Klein, R. and Boyle, J.P., 2008. Projection of diabetic retinopathy and other major eye diseases among people with diabetes mellitus: United States, 2005-2050. Archives of ophthalmology, 126(12), pp.1740-174

[3] Yau, J.W., Rogers, S.L., Kawasaki, R., Lamoureux, E.L., Kowalski, J.W., Bek, T., Chen, S.J., Dekker, J.M., Fletcher, A., Grauslund, J. and Haffner, S., 2012. Global prevalence and major risk factors of diabetic retinopathy. Diabetes care, 35(3), pp.556-564.

[4] Duh, E.J., Sun, J.K. and Stitt, A.W., 2017. Diabetic retinopathy: current understanding, mechanisms, and treatment strategies. JCI insight, 2(14).

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

[6] Salz, D.A. and Witkin, A.J., 2015. Imaging in diabetic retinopathy. Middle East African journal of ophthalmology, 22(2), p.145.

[7] Fong, D.S., Aiello, L., Gardner, T.W., King, G.L., Blankenship, G., Cavallerano, J.D., Ferris, F.L. and Klein, R., 2004. Retinopathy in diabetes. Diabetes care, 27(suppl 1), pp.s84-s87.

[8] Vitale, S., Maguire, M.G., Murphy, R.P., Hiner, C., Rourke, L., Sackett, C. and Patz, A., 1997. Interval between onset of mild nonproliferative and proliferative retinopathy in type I diabetes. Archives of Ophthalmology, 115(2), pp.194-198.

[9] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009, June. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-264)

[10] LeCun, Y., Boser, B.E., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W.E. and Jackel, L.D., 1990. Handwritten digit recognition with a backpropagation network. In Advances in neural information processing systems (pp. 396-404).

DOI: 10.17148/IJARCCE.2023.125163

[11] Sadman Sakib, Tharat Tazrin, Mostafa M. Fouda, Zubair MD. Fadlullah, Mohsen Guizani., DL-CRC: Deep Learning-Based Chest Radiograph Classification for COVID-19 Detection.

[12] Vishesh S, Sujay Singh, Rishi Singh, 2021. CNN Algorithm: H5 model for Accurate Prediction of COVID-19.

[13] Gao, Z., Li, J., Guo, J., Chen, Y., Yi, Z. and Zhong, J., 2018. Diagnosis of Diabetic Retinopathy Using Deep Neural Networks. IEEE Access, 7,pp.3360-3370.

[14] Yuxin Wu,Kaiming He Group Normalisation for Deep learning arXiv preprint arXiv:1803.08494V3. [15] G.Deng,L.W.Cahill,An adaptive Gaussian filter for noise reduction and edge detection.1993 IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference.

[16] Xin He, Kaiyong Zhao, Xiaowen Chu, Department of Computer Science, Hong Kong Baptist University, AutoML:A Survey of the State-of-the-Art arXiv preprint arXiv:1908:00709v2[cs.LG]

[17] Thomas Elsken,Jan Hendrik Metzen,Frank Hutter, Neural Architecture Search:A survey arXiv preprint arXiv:1808.05377v3.

[18] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, MobileNetV2:Residuals and bottlenecks arxiv preprint arXiv:1801.0438[cs.cv]

[19] Cun, Y.L., Boser, B., Denker, J.S., Howard, R.E., Habbard, W., Jackel, L.D., et al. Advances in neural information processing systems 2. Citeseer. ISBN 1-55860-100-7; 1990, p. 396–404.

[20] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.. Dropout: A simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15(1):1929–1958.

[21] Nair, V., Hinton, G.E.. Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10). 2010, p. 807–814.

[22] Ioffe, S., Szegedy, C.. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv 2015;URL: arXiv:1502.03167.

[23] He, K., Zhang, X., Ren, S., Sun, J.. Deep residual learning for image recognition. arXiv 2015;URL: arXiv:1512.03385.

[24] Gardner, G., Keating, D., Williamson, T., Elliott, A.. Automatic detection of diabetic retinopathy using an artificial neural network: a screening tool. Brit J Ophthalmol 1996;80(11):940–944.

[25] Nayak, J., Bhat, P.S., Acharya, R., Lim, C., Kagathi, M.. Automated identification of diabetic retinopathy stages using digital fundus images. J Med Syst 2008;32(2):107–115.

[26] Acharya, R., Chua, C.K., Ng, E., Yu, W., Chee, C.. Application of higher order spectra for the identification of diabetes retinopathy stages. J Med Syst 2008;32(6):481–488.

[27] Acharya, U., Lim, C., Ng, E., Chee, C., Tamura, T.. Computer-based detection of diabetes retinopathy stages using digital fundus images. P I Mech Eng H 2009;223(5):545–553.

[28] Adarsh, P., Jeyakumari, D.. Multiclass svm-based automated diagnosis of diabetic retinopathy. In: Communications and Signal Processing (ICCSP), 2013 International Conference on. IEEE; 2013, p. 206–210.

[29] Vishesh S, Rachana S, Chethan K, Harshitha V Raj, 2022. AI to Predict Diabetic Retinopathy: Image Pre-Processing and Matrix Handling.