Exploring the Parametric Impact on a Deep Learning Model and Proposal of a 2-Branch CNN for Diabetic Retinopathy Classification with Case Study in IoT-Blockchain based Smart Healthcare System

Manaswini Jena[1], Smita Prava Mishra[1], Debahuti Mishra [1], Pradeep Kumar Mallick[2] and Sachin Kumar[3]*
E-mail: manaswini.jena88@gmail.com, smitamishra@soa.ac.in, debahutimishra@soa.ac.in, pradeepmallick84@gmail.com, sachinagnihotri16@gmail.com
[1]Department of Computer Science and Engineering, Siksha 'O' Anusandhana Deemed to be University, Bhuvaneswar Odisha, India
[2]School of Computer Engineering, Kalinga Institute of Industrial Technology (KIIT) Deemed to be University, Bhuvaneswar, Odisha, India
[3]Department of Computer Science, South Ural State University, Chelyabinsk, Russia

Keywords: healthcare system, 2-branch CNN, diabetic retinopathy, fundus images, medical diagnosis, internet of things

Received: January 9, 2022

Smart healthcare has changed the way how the patient interacts with the specialists for treatment. However, security and support for various diseases are still the concern for such smart automated systems. One of the critical diseases namely Diabetic Retinopathy (DR), is a major concern for the person with prolonged diabetes and may lead to complete blindness irrespective of age groups. Moreover, in recent years blockchain has gained popularity in providing secure communication between sender and receiver. Hence, this work focus on designing a blockchain-based smart healthcare system for the early detection of diabetic retinopathy. However, early detection of DR impose complexities and requires expert diagnosis, which is not available everywhere. Hence, the proposed smart healthcare model contains a Computer-Aided Diagnosis (CAD) assistance for early detection of symptoms of the disease. The CAD model may assist the ophthalmologists in the early detection of DR, which requires intensive research in developing an efficient and accurate model that can operate without human interaction. This study provides an empirical analysis of these factors to design the best model for early detection of DR. The best model can be used to develop IoT based smart devices to detect DR in diabetic patients. The study also explains the importance of IoT and blockchain-based technology for the development of smart healthcare systems. The values of the parameters and type of hyperparameters choosen from the study is used in a proposed 2-branch CNN model, and the model is validated using the Kaggle fundus image set. Analysis of various parameters and using their best values gives an outstanding performance in the proposed 2-branch CNN model.

Povzetek: Predstavljena je bločna metoda za varno komunikacijo med zdravnikom in pacienti, pri tem je uporabljena metoda globokih nevronskih mrež za problem diabetične slepote.

1 Introduction

Healthcare systems are an essential part of modern society to save people's lives and provide services remotely. However, the concern of the healthcare system is the need for the protection of information of the patient and correctly recognizing the symptoms for early detection. Diabetic Retinopathy(DR) is one of the fatal diseases where early diagnosis is very crucial. Computer-Aided Diagnosis(CAD) assists experts in the early diagnosis of the disease. In recent years, researchers have developed many CAD models to detect DR effectively. However, a complete healthcare model is not yet developed. This research attempts to propose a CAD model with security to protect patient data and a deep learning model to detect DR accurately.

Blockchain, IoT, and AI have shown their potential in almost every domain of our life. Smart healthcare is one of the major sectors that has been influenced by IoT infrastructure and solutions [1-2]. IoT-based healthcare systems have immensely added value to our lifestyle and health monitoring with the use of portable wearable devices. However, there are few complex healthcare areas where Blockchain and IoT have to do some evolutionary steps to protect the healthcare data to ensure patient confidentiality.

Similarly, Convolutional Neural Network (CNN) is widely used for accurate CAD in the healthcare domain. These automatic diagnostic systems can improve the quality and productivity of healthcare services. Moreover, CAD can help in the early detection of diabetic retinopathy, one of the primary causes of blindness in the world [3]. The limited resources of healthcare facilities lead to difficult ac-

*Corresponding author

cess to expert doctors for diabetic retinopathy analysis [4]. Currently, CAD includes convolutional networks to identify the early signs of diabetic retinopathy disease. The CNN is a type of deep feed-forward artificial neural network. This architecture is mainly designed and adopted to classify images using a multilayer architecture [5-7]. Various types of CNNs are proposed and their efficiencies are proved in the complicated image classification techniques [8-11]. Prolonged diabetics result in diabetic retinopathy leading to the significant cause of vision loss worldwide [10-11]. Several technical implementations have done using CNN for diabetic retinopathy identification [12-17]. The performance of the CNN model depends on the number of convolution layers, the number of nodes in convolution layers with several parameters, and hyper-parameters such as learning rate, activation functions, and pooling [18-19]. To our best knowledge, no study has been carried out that can justify the choices made in designing a CNN for diabetic retinopathy classification.

The fundamental units for this convolution network are shared weight, bias, local perceptive, and pooling. The activation function and pooling are critical for model performance. The activation function is an elementary part of every neural network architecture. Moreover, it is responsible for transforming the summed weighted input into its output [20-21]. The activation function and threshold are the basic terminology for every neural network for execution and flow of network as the activation of neurons depend upon it [22-23]. Non-linear transformations and back-propagation are made using activation functions. Several different activation functions are utilized for NN [24-27]. Pooling is applied as a downsampling approach to reduce the input signal resolution without affecting the representable, essential, and large feature elements [24-25]. In addition to activation and pooling, the learning rate and optimizer used for back-propagation in the network has also a great impact on the performance of the neural network. So, all four features are going to be analyzed for the classification of diabetic retinopathy using the CNN model.

The learning rate is a hyper-parameter used to control the learning speed of the neural network model. The key role of the learning rate is to scale the magnitude of weight to reduce the network's loss function. The training or learning is generally performed through updating weights to map the input with the output. These updated weights are calculated by analytical methods via empirical optimization techniques called stochastic gradient descent (SGD). The stepping amount needed to change the weights during the training process is called the learning rate. This is the value used for adjusting the weights according to loss gradient through back-propagation [26].

The value of the learning rate is a small positive value in the range of 0.0 to 1.0. It may vary differently for different models and we can find a good learning rate value via the trial and error method. Though the optimal learning rate calculation is analytically not possible yet performance improvement can be done by achieving a good learning rate. During training, the learning rate can be adjusted to improve the performance and this is called adaptive learning [27-28]. Between the epochs or iterations, the value can be increased or decreased, instead of having a fixed value. The change in value is made is called a learning rate schedule. Here, an attempt is made to visualize the change in accuracy and performance of the model according to the change in learning rate values.

Another feature considered here is the types of optimizers used for back-propagation in the neural network. Back-propagation is a weight updater in neural networks used to update the weights by optimizing the values to get the target result. For this optimization, SGD is generally [29-30] used. SGD is an iterative optimization method that uses shuffled samples for the evaluation of the gradient values. Sometimes the random probabilistic nature of SGD for choosing samples results in a noisy path to reach the destination from typical gradient descent algorithms as it may take a higher number of iterations due to randomness. But, the computationally inexpensive nature of this algorithm makes it suitable for usage. Another type of optimizer available to be used instead of this SGD is Adam. The memory requirement is very less for this optimizer as well as it can be used for large problems in terms of data/parameters and appropriate for non-stationary objectives along with a problem with noisy gradients [31]. AdaDelta is another optimizer based on a moving window of gradient updates. It does not accumulate the past gradients rather continues learning from updates [32]. Here these three optimizers are chosen for the experimentation.

The objective of this study is to analyze the impacts of these four features i.e. activation function, pooling, learning rate, and optimizer parameters on a CNN model for diabetic retinopathy classification. The study is conducted in phases to analyze the impact of each parameter. Finally, the best specifications are used in a proposed CNN model for diabetic retinopathy classification.

The rest of the article is organized as follows: Section II represents the state-of-the-art literature review. Section III and IV provides the methodolgy with model description and an overview of the dataset respectively. Section V provides the experiment details. The results and discussion are in Section VI . Finally, Section VII concludes the study.

2 Literature review

IoT, Blockchain and Cloud technologies are integrated in the medical environment for offering healthcare and telemedical laboratory services along with AI. For the classification of diabetic retinopathy through fundus image CNN is widely used and proved as a great classifiers, but sometimes the data taken directly from the patient may not give the expected accurate report. More features and parameters need to be tuned very minutely and this is the main objective here. This section represents some studies where CNN is used for the classification of DR. In [33] transfer learning is used on pre-trained CNN models and an accuracy of 74.5% is achieved for binary classification model. Tertiary and quaternary classification models are also tested for the classification the study. A set of pre-processing technique and data augmentation is followed by transfer learning for the experiment. In [34] the CNN used with augmentation for the classification of diabetic retinopathy gives better result compared to the CNN used without augmentation *i.e.* 94.5% and 91.5% respectively. Here augmentation improves the performance. DIARETDB1 data is used in [35] for classification of DR by N. Gharaibeh *et al.* Here, in the first phase feature extraction is done followed by classification of features acquiring 98.4% accuracy using CNN. In [39] retinal images are transformed into entropy images to improve the classification. The complexity of original fundus images are represented by quantifying the information of the image. Then classification is done using deep learning which improves the accuracy to 86.0% from 81.8% that was achieved using normal fundus images. Image enhancement techniques like histogram equalization is used in [57] along with deep leaning to improve the classification of DR and an accuracy of 97% is achieved. In [37] the segmented image classification using CNN for DR achieves an accuracy of 99.18% accuracy results. Another work in [36] uses CNN along with Support vector machine (SVM) as classifier and results with 86.17% accuracy. Here max-pooling is replaced by fractional max-pooling in the CNN for extraction of features before using it in SVM. Following Table 1 highlights the findings of the above discussion:

However, a study [56] states that although high accuracy is achieved for detection of DR using various models of machine learning, but the requirements of real time deployment and clinical validation is not yet fulfilled completely. It also indicates to solve the problem arising due to the diversity of datasets in terms of ethnicities and camera used for capturing the images in real practical clinical field. So, fine-tuning of every parameters and features of the network is required to get a perfect model. However, this study of various features and parameters and their effects will help in development of prediction models. Furthermore, IoT/Blockchain based devices provides portable and secure technology for healthcare monitoring, however it is a challenging task to achieve such facility with high security. In [41], authors proposed an approach to deal with these challenges for Internet of Medical Things [IoMT]. Several research studies focused on the effectiveness of IoT and bloackchain for the healthcare ecosystem [42-43]. In addition, cloud-server can be utilized to provide a global secure network to monitor a large number of petients in several regions with cloud based IoT network in health care domain [44-47].

It is observed that, image features are also a very important factor [48], to obtain a high classification performance. Various techniques have been used for accurate classification of the image by extracting the discriminative features. The amalgamation of both global and local features can be fed to a classifier for classification of the image. For this a 2 branch CNN can be designed having the ability to extract features in an optimized way for the classification in a unified optimized framework [49]. A multi-domain CNN performs better than a single frequency domain CNN and spatial domain CNN for detection of image compression [50]. Hyperspectral image classifications show the dominance of a multi-branch CNN in many studies [51-53]. These kinds of architecture extracts feature from each source of the image at their native pixel of resolution and it is done by exploiting both spatial and spectral information from the image. Keeping this in mind, the workflow is designed to create a better model for diabetic retinopathy detection as compared to the existing models using the analysis.

3 Methodology

At first a two class classification model using several CNN configurations were developed to understand the importance of several parameters for the classification of DR in diabetic patients. These configurations are mentioned in Table 2 as follows:

The result of the empirical analysis is used in a 5 class classification model. A two-branch CNN is proposed for the classification of DR using fundus image. The proposed flow diagram of a two-branch CNN model is given in **Figure 1**. Two separate feature sets collected from two separate branches of CNN are assembled and used in the fully connected neural network for classification of DR.

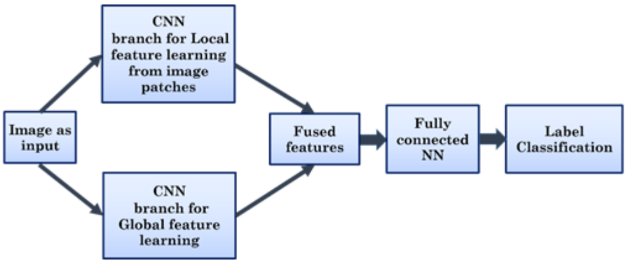

CNN branch for Local feature learning from image patches
Image as input
Fused features
Fully connected NN
Label Classification
CNN branch for Global feature learning

Figure 1: A two branch CNN model.

Some basic pre-processing steps are applied here for the classification as follows:

(i) **Cropping:** Extracting the part of the region of interest (ROI) from the image is done through cropping. Most of the fundus images have a black border which needs to be reduced as possible to concentrate on the region of interest.

(ii) **Resizing of image:** For any machine learning technique or deep learning technique, resizing of the image is required. All the images of data need to be standardized. Here, all the images are converted into one size *i.e.* 256×256×3.

Table 1: Comparison of existing studies for DR classification.

Model used	Dataset used	Accuracy
CNN with pre-processing, feature extraction and augmentation (used for binary classification) [33]	Messidor MildDR	74.50%
CNN with augmentation [34]	Kaggle	94.50%
CNN without augmentation [34]	Kaggle	91.50%
Feature extraction and classification using CNN [35]	DIARETDB1	98.40%
Transformed entropy images and classification using CNN [36]	Kaggle	86.00%
Histogram Equalization and classification using CNN [57]	Messidor	97%
Segmentation and pre- processing with CNN [37]	DiaretDB0,	99.17%
	DiaretDB1	98.53%
	DrimDB	99.18
CNN with SVM as classifier [38]	Kaggle	86.17%

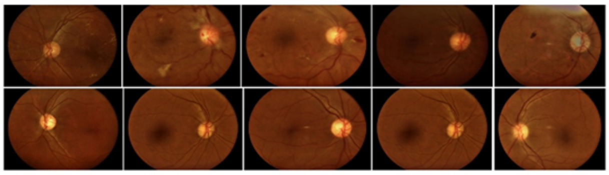


Figure 2: First row represents diseased fundus images and second row represents normal fundus images.

(iii) **Contrast Enhancement:** It is a process of making the image features more prominent by changing the range of pixel values in the image. Contrast Limited Adaptive Histogram Equalization (CLAHE) is the method which improves contrast in digital images. The results of medical image enhancement using this technique are better than other histogram equalization techniques [54]. Here, this CLAHE is used for the contrast enhancement to make the hidden features visible.

(iv) **Normalization:** Max-min normalization is applied here for normalizing the pixel values.

4 Dataset description

The dataset under study have been obtained with the support of the administration of Ruby Eye hospital in Orisha, India. The fundus images have been captured with a high resoulion cameras. The database shows a male and female ratio of 50:26 with diabetic retinopathy patients' in 30-65 age group. For diabetic retinopathy, the separated fundus images of diseased and non-diseased patients are collected. The samples of the infected and non-infected fundus images are shown in Fig 2.

The dataset consists of the fundus images of 102 diabetic patients with the presence of both healthy and diseased images with equal distribution. The two classes of these fundus images are either diabetic retinopathy (DR) *i.e.* the diseased or normal *i.e.* non-diseased. Normalized of the images are performed with max-min normalization in the range [0-1] in order to achieve zero mean and unit variance. This dataset is available online publically for research purpose [55].

Kaggle retina fundus image dataset is also used here for experiment. It contains the images based on five severity levels such as 0=No DR, 1=Mild NPDR, 2=Moderate NPDR, 3=Severe NPDR, 4=PDR. For both the data, the training and testing ratio is taken as 80:20 here.

5 Experimental analysis

The detailed evaluation of hyper-parameters on a large model is a computationally intensive task. Hence, the study uses simple model to evaluate various hyper-parameter settings like pooling, activation, learning rate, and optimization. Finally, the proposed 2-branch CNN model uses the best combination of hyper-parameters learnt from the simple model.

To analyse the significance of pooling, activation, learning rate, and optimization speciation of the CNN model, multiple experiments have been conducted, and these are divided into four phases.

In the first phase (Phase 1), the performance of the parameters or factors are analyzed to be selected for further testing. Following this in the second phase (Phase 2), a selective combination of factors are evaluated to establish the relationship of these to the model performance. In next two phases (Phase 3 and Phase 4), more factors are evaluated by incremental tests to obtain maximum performance for fundus image classification using this dataset.

Here, data augmentation is used for virtually increasing the number of samples in the dataset during training. It is a technique used to manufacture or generate own data

Table 2: Used CNN model and its layer description.

Type	Layer Size	Learnables	Total Learnables
Image_Input	256 × 256 × 3	-	0
Convolution	242 × 242 × 32	Weights 15 × 15 × 3 × 32 Bias 1 × 1 × 32	21632
Batch_Normalization	242 × 242 × 32	Offset 1 × 1 × 32 Scale 1 × 1 × 32	64
ReLU	242 × 242 × 32	-	0
Max_Pooling	121 × 121 × 32	-	0
Convolution	115 × 115 × 64	Weights 7 × 7 × 32 × 64 Bias 1 × 1 × 64	100416
Batch_Normalization	115 × 115 × 64	Offset 1 × 1 × 64 Scale 1 × 1 × 64	128
ReLU	115 × 115 × 64	-	0
Max_Pooling	57 × 57 × 64	-	0
Convolution	53 × 53 × 128	Weights 5 × 5 × 64 × 128 Bias 1 × 1 × 128	204928
Batch_Normalization	53 × 53 × 128	Offset 1 × 1 × 128 Scale 1 × 1 × 128	256
ReLU	53 × 53 × 128	-	0
Max_Pooling	26 × 26 × 128	-	0
Convolution	24 × 24 × 128	Weights 3 × 3 × 128 × 128 Bias 1 × 1 × 128	147584
Batch_Normalization	24 × 24 × 128	Offset 1 × 1 × 128 Scale 1 × 1 × 128	256
ReLU	24 × 24 × 128	-	0
Max_Pooling	12 × 12 × 128	-	0
Keras Flatten	1 × 1 × 18432	-	0
Fully_Connected	1 × 1 × 100	Weights 100 × 18432 Bias 100 × 1	1843300
ReLU	1 × 1 × 100	-	0
Fully_Connected	1 × 1 × 2	Weights 2 × 100 Bias 2 × 1	202
ReLU	1 × 1 × 2	-	0
Classification_Output	-	-	0

with the existing data [56]. Data augmentation have been used to deform the labelled data without semantic information loss[57]. Here five online data augmentations are used such as horizontal and vertical shifts, horizontal and vertical flips and rotation. The horizontal shift and vertical shifts are of 2% and rotation of 90% is applied. The modified input images are introduced during the training process automatically. For experiment, the dataset is divided in the ratio of 80:20 for the training and testing data.

5.1 Evaluation of the parameters

For evaluation various matrices like accuracy, sensitivity, specificity, precision, recall and F_1*-Score* [58-62] are taken into consideration.The analysis of results is done step by step through different phases as described below:

Phase 1: As described before in the experimental setup, different pooling layers, activation functions, optimization parameters and learning rates are evaluated to understand the behaviour of the model. The obtained results for variation in parameter combination are presented pictorially in the following graphs. The 3-dimensional bar graph representation is used to visualize the behaviour of the model with respect to testing accuracies achieved for different combinational approaches made for the CNN through the process of classification of diabetic retinopathy data.

Four pooling types such as max pooling, min pooling, average pooling and maxmin pooling is considered here. For each pooling, the variation in activation function, learning rate are presented in x-axis and y-axis respectively. Furthermore, four widely used activation functions such as, *LReLu, ReLU*, *sigmoid* and *tanh* are chosen to be studied with combination of learning rate of 0.1, 0.01, 0.001 and 0.0001. Finally, the observed testing accuracy is given in the z-axis. The learning ability of the model with respect to various optimizers such as SGD, ADAM, Adadelta is also observed by the colour indicated in graph respectively. Figure 3 shows the changes in accuracy value for max pooling with different activation functions, learning rate and different optimizer for backpropagation in the CNN model.

The Figure 3 (a), (b), (c), and (d) are the 3-dimensional graphical representation of accuracy values measured using max pooling at epoch 25, 50, 75, and 100 respectively.

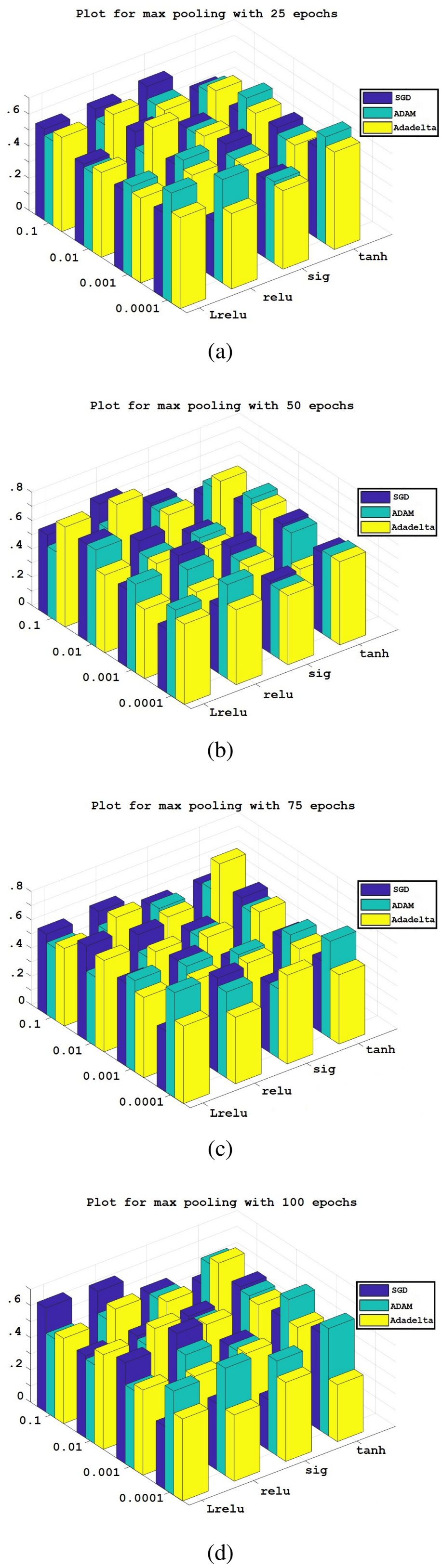
From the Figure 3(a) it is observed that, at epoch 25, a maximum accuracy result of 0.686 is obtained by using LReLu and ReLu activations at 0.01 learning rate with Adadelta optimizer. The same accuracy is observed at 0.0001 learning rate with ADAM optimizer at 100 epochs. The results show that the increase of epoch does not ensure better accuracy here. Moreover, the accuracy can also decrease. This occurs due to overtraining which affects the results or due to improper tuning of the learning rate.

The graphical Figure 3(b) shows increased accuracy values as compared to Figure 3(a). The two activation functions ReLu and LReLu at 0.1 learning rate with Adadelta optimizer present accuracies of 0.725 and0.705, respectively.

Figure 3(c) shows that the Adadelta optimizer presents an accuracy of 0.725 with tanh activation at 0.01 learning rate for epoch 75. Moreover, the ADAM optimizer also presents reliable results with an accuracy value 0.745 with LReLu at 0.0001 learning rate. Figure 3(d) shows the accuracy of the model by using max pooling layer at epoch 100. The best accuracy of 0.686 is achieved with ReLu activation at 0.01 learning rate and SGD optimization and also by the tanh activation function at learning rate of 0.1 and ADAM optimization. The LReLu activation achieves an accuracy of 0.647 with SGD and ADAM optimizers at 0.1 and 0.0001, respectively. The sigmoid function presents the lowest accuracy results.

Figure 4 represents the 3-dimensional graph for the second type of pooling taken min pooling at different epochs and features. The Figure 4(a) is the plot for accuracy values at epoch 25 using min pooling in the model. Here, the ADAM optimizer at 0.0001 learning rate with ReLu activation function achieves an accuracy of 0.764 and the SGD optimizer at 0.01 with LReLu activation gets a 0.66 accuracy. In Figure 4(b) for epoch 50, the ADAM optimizer at 0.0001 learning rate with ReLu activation gets an accuracy of 0.686. In this same condition LReLu activation increases the accuracy to 0.725. In this specific situation, the LReLu performs better than the ReLu activation. Again, the Adadelta optimizer at 0.0001 learning rate with LReLu gives the same accuracy as 0.725. The Figure 4(c) is the accuracy plot taking epoch value as 75, where only one combination gives a marked accuracy result of 0.686 by Adadelta optimizer at 0.0001 learning rate and tanh activation function. The Figure 4(d) gives the bar graph representation for epoch 100. In this point, the LReLu activation at 0.01 learning rate and ADAM optimizer in the model gets an accuracy of 0.705 and tanh function at 0.1 learning rate and Adadelta optimizer gets an accuracy of 0.686. The observed results shows that, the min pooling and max pooling in the model combined with ADAM or Adadelta optimizer performs better than other optimizer combinations. The following **Figure 5** represents the plotting for another type of pooling called average pooling taken in the model.

The Figure 5(a) is for epoch 25 which yields two important accuracy observation as 0.686 and 0.705 using ADAM and Adadelta optimizers. In the Figure 5(b), the ADAM optimizer achieves an accuracy of 0.666 with ReLu activation at 0.0001 learning and 0.745 with LReLu at 0.01 learning rate. From the graph it also can be said that the average pooling method in the network model along with SGD optimizer and learning rate of 0.01works well with tanh function resulting an accuracy value of 0.705 compared to ReLu function that yields 0.666 accuracy. The Figure 5(c) is plotted for 75 epochs and yields accuracies of 0.666 and 0.686 by using SGD optimizer at 0.1 and 0.01 learning rate with ReLu and LReLu activations. The ADAM optimizer also reaches an accuracy of 0.764 with tanh activation function at 0.0001 learning rate. But with increased epoch of 100 at Figure 5(d) the accuracy values does not increase further, rather staying at value of 0.666 by SGD and ADAM opti-



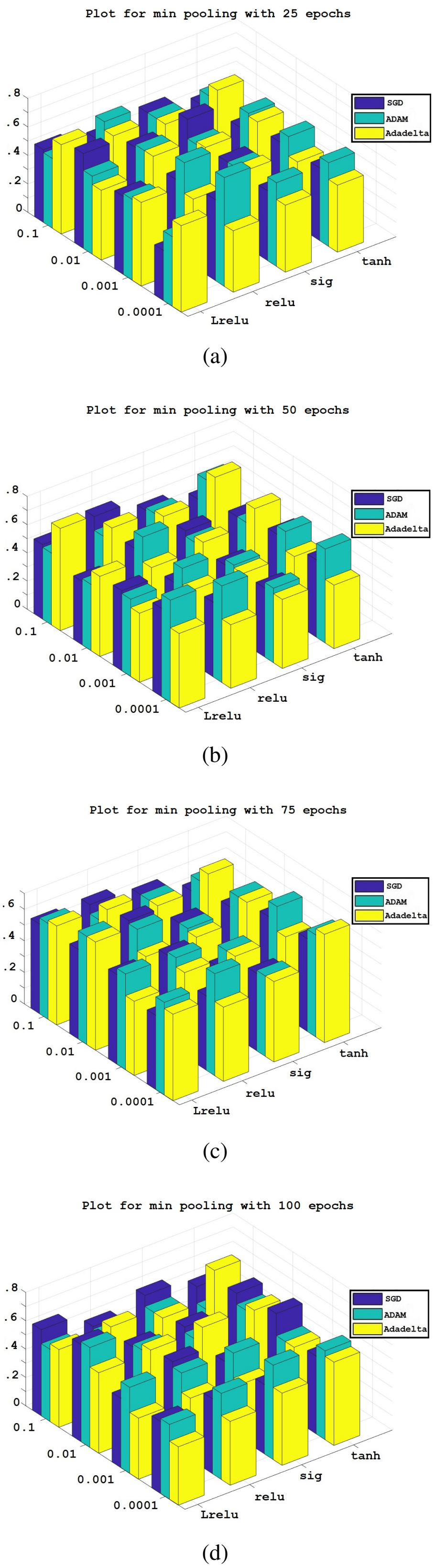
(a)

(b)

(c)

(d)

Figure 3: Plotting of accuracy values by taking max pooling in the model. **(a)** is for epoch =25, **(b)** is for epoch =50 **(c)** is for epoch =75 **(d)** is for epoch =100.



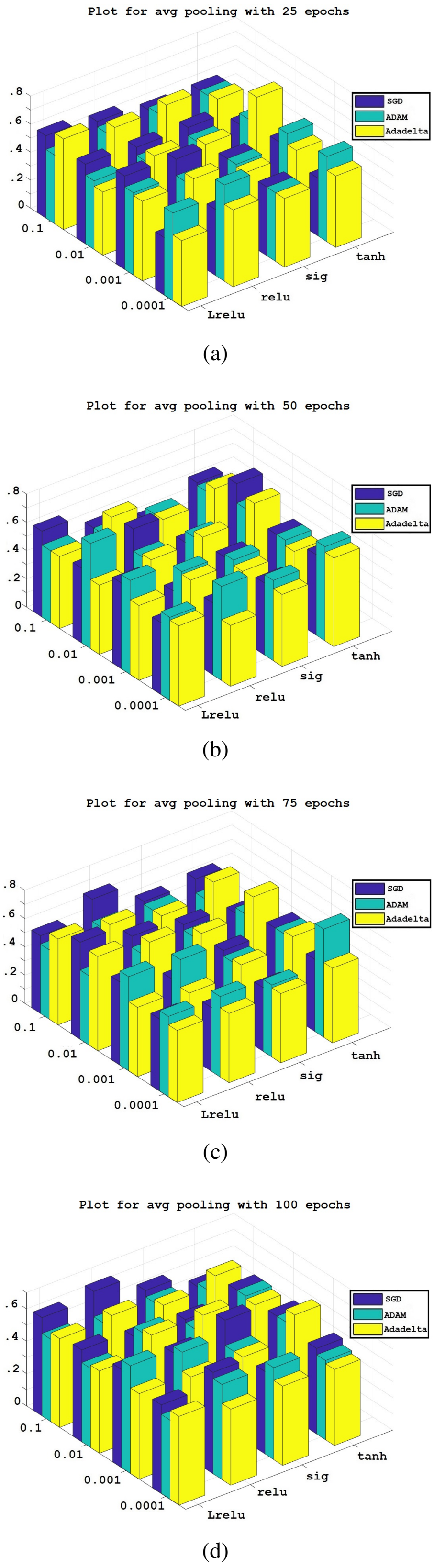
(a)

(b)

(c)

(d)

Figure 4: Plotting of accuracy values by taking min pooling in the model. **(a)** is for epoch =25, **(b)** is for epoch =50, **(c)** is for epoch =75 and **(d)** is for epoch =100.



(a)

(b)

(c)

(d)

Figure 5: Plotting of accuracy values by taking average pooling in the model. **(a)** is for epoch =25, **(b)** is for epoch =50, **(c)** is for epoch =75 and **(d)** is for epoch =100.

mizers. Till now with these three pooling layers ReLu and LReLu activation functions performs well on most of the cases, while tanh performs in few and the sigmoid function seems to be a poor performer at this point.

In this similar procedure the accuracy values achieved by using max-min pooling in the model along with different parameters are also analyzed. By taking 25 epochs in the model, it reaches to an accuracy value of 0.803 using ADAM optimizer at 0.0001 learning rate with tanh activation function. This accuracy does not increase in further increase of epochs to 50, 75 or 100, rather falls to 0.666 in some combinational approaches. The experimental results clearly conveys that every condition, every combination of parameter is effecting the model's behaviour. Each combinational approach gives dissimilar results to analyze. From these results, selective combinations, which gives the highest result among all, are chosen to test with more increased epoch and taken to phase two.

Phase 2: In the second phase, the accuracy and other evaluation parameters are analyzed for the chosen combinations. The following **Table 3** shows the selected accuracy values for different pooling with combination of features and parameters chosen along with its sensitivity (Sn), specificity (Sp), precision (Pr), recall (Re) and F_1-score (F) values. Above table contains evaluation parameters for the combinations through which the maximum accuracy values are achieved in phase 1. It shows that, the max pooling combined with Adadelta optimizer at 0.1 learning rate goes well for achieving a good accuracy values. Finally, the ADAM optimizer with LReLu and 0.0001 learning rate attains the maximum accuracy among this for max pooling. At this accuracy value, the sensitivity is 0.76 and the specificity the precision are 0.73 with recall and F_1-score to be 0.76 and 0.745 respectively. From the combinational approaches for min pooling the ADAM optimizer gives better results as compared to other optimizers and the learning rate 0.0001 can be selected for proper tuning here. The LReLu function also proved to be more performable than ReLu and other activation functions in this situation. The actual positive cases that are correctly identified is measured by sensitivity and the maximum of it is 0.79 for the accuracy 0.76 and specificity of 0.74 while the precision, recall and F_1-score are 0.73,0.8 and 0.76 respectively. These are the best values obtained in this spot.

While taking average pooling in this model at various epochs the tanh activation seems to perform better at this state. At learning rate 0.01 accuracies are increased but the maximum accuracy yield here is at 0.0001 learning rate which is 0.764. The sensitivity and specificity are 0.791 and 0.740 while, the precision, recall and F_1-score are 0.73, 0.8 and 0.76 respectively. Here the ADAM optimizer gives the maximum result compared to all other optimizers.

In this case of maxmin pooling, the ADAM optimizer performs very well compared to other optimizers again. It gains an accuracy of 0.803 and a sensitivity of 0.9, which is the maximum among all the previous results. Till now this is the best combinational approach for the model using this dataset. The specificity, precision, recall and F_1-score are 0.741, 0.692, 0.9 and 0.782 respectively. Here it is visualized how the change in pooling layer along with its parameters and features effects the behaviour of model. The *LReLu activation* works well for *min pooling* and *tanh function* for *average pooling*, but in other pooling they are not the same. The ADAM optimizer gives better result in *min pooling* and *maxmin pooling*, however AdaDelta optimizes well in max pooling for this model with this data. The graphs and tables demonstrates the clear conceptualization of the independent behaviour of parameters and conditions in a particular scenario. Now, all of the above combinations of all four pooling layers are taken to phase 3 for further experiment.

Phase 3: The above combinations are the repeated with increased iteration up to 1500 and the acquired outputs are analyzed. The following tables compares the accuracy calculated at 100 and less than 100 iterations with accuracy values at 1500 iterations. In this the actual set of combinations are selected *i.e.* the set of states which gives higher results are finally selected at top best combinations for this particular model. The accuracy at 1500 iterations are shown in Table 4 for max pooling. From the above table, it is marked that the accuracy value which is actually the maximum among the four values is not increased more with the increase in epoch. Another new combination of features has acquired an increased highest accuracy of 0.84 with the increased epoch of 1500. The sensitivity is 0.87 while the specificity is 0.81. The precision, recall and F_1-score are 0.8, 0.88 and 0.84 respectively. So for max pooling this can the best combination of parameters and features to get a good result using this model. It can be seen in table 4 that in min pooling the expected accuracy value is not achieved, rather the accuracies decreased with the increase in number of iterations. It may happen due to overtraining which affects the result in opposite way.

Table 4 also illustrates the accuracy values and other evaluation parameter values for the average pooling in the model. Here also the increased accuracy value is not the further increased value from the previous phase. The Adadelta optimizer performs well here compared to ADAM as before for this average pooling at learning rate 0.01 rather than 0.0001. It achieves an accuracy of 0.862 with a sensitivity of 0.85 which is smaller than the specificity 0.875. Here some more tuning may increase the accuracy even more by increasing the sensitivity. The precision, recall and F_1-score are 0.888, 0.84 and 0.86 respectively (Table 4). In this table of accuracy representation for max-min pooling layer, the highest accuracy value from the previous phase increased to 0.86 from 0.80. Though the sensitivity was high before, still it maintains a value of 0.88 and specificity of 0.84.

The precision, recall and F_1-score are 0.85, 0.88 and 0.86 respectively. We can see that some accuracy values are increased due to increase in iteration and some of them give a decreased value as they might got stuck in local minima or may be due to overtraining. Only a few among these yields refinement of accuracies after increase in iteration and is marked with bold font. Further, those highlighted accuracy's combinational condition can be taken for experimentation with a more increased value of iteration.

Phase 4: Here the three pooling techniques *i.e.* max pooling, average pooling and max-min pooling and their combinational features are finally selected in **Table 5** which are working well for this model and dataset. Further this combinations can be used either with optimization technique or with some hybridization to achieve more suitable model having higher accuracy values.

This table contains the three maximum accuracy values of 0.843, 0.862 and 0.865 achieved in this experimental approach with other evaluation parameters. These can be taken as the best three combination of activation function, learning rate, optimizer and pooling layer to achieve a good accuracy for this model. In this paper only two class classification of diabetic retinopathy data is considered which is either DR or normal. DR is taken as a five class problem i.e. a normal or non DR class and four stages such as, mild DR, moderate DR, severe DR, and proliferative DR classes [39-40]. This empirical analysis will help us in the further deep classification of DR. Some observations are made in this experiment are stated below:

- Among various types of the pooling used here, max-min pooling gives comparatively better result. The fusion of max and min values of the pixels during the selection through pooling generates a good selection of pixel values for the model's training and testing for classification.
- A combined pooling helps in improving the performance value of model. This fused pooling technique also works well in [62].
- Unlike the ReLU function, the LReLu does not transform all the negative values to zero. Here, the LReLu performs finer than other activation functions.
- The combinations which give rise to good accuracy results, amid of them, 50% of are having LReLu as their activation function.
- Particularly, for this experiment if we grade the activation functions, LReLu takes the first position resulting as the best activation function for image classification using CNN.
- The tanh function comes in the second place and finally the ReLU comes in the third place by providing good result in few combinations. But with increase in iteration, the tanh function wins over the LReLu by obtaining more accurate results.
- Going through the accuracy results, it is noticed that more than 50% good accuracy values are seen in the cases using Adam as an optimizer here. The Adadelta optimizer also give results in less than 40% combinational cases.

Table 3: Selective combinations for various pooling.

Pooling	Activation	Optimizer	LR	Iter.	Acc.	Sn	Sp	Pr	Re	F
Max	ReLU	Adadelta	0.1	50	0.72	0.71	0.73	0.76	0.68	0.74
	LReLu	Adadelta	0.1	50	0.7	0.67	0.75	0.8	0.6	0.73
	LReLu	ADAM	0.0001	75	0.74	0.76	0.73	0.73	0.76	0.74
	Tanh	Adadelta	0.1	75	0.72	0.71	0.74	0.769	0.68	0.74
Min	ReLU	Adadelta	0.1	50	0.72	0.71	0.73	0.76	0.68	0.74
	LReLu	Adadelta	0.1	50	0.7	0.67	0.75	0.8	0.6	0.73
	LReLu	ADAM	0.0001	75	0.74	0.76	0.73	0.73	0.76	0.74
	Tanh	Adadelta	0.1	75	0.72	0.71	0.74	0.769	0.68	0.74
Avg	ReLU	Adadelta	0.01	50	0.745	0.74	0.75	0.769	0.72	0.754
	LReLu	Adadelta	0.01	25	0.7	0.789	0.656	0.57	0.89	0.666
	LReLu	ADAM	0.01	50	0.7	0.761	0.666	0.615	0.8	0.68
	Tanh	Adadelta	0.0001	75	0.764	0.791	0.74	0.73	0.8	0.76
Max-Min	LReLu	ADAM	0.001	50	0.72	0.8	0.677	0.615	0.84	0.695
	Tanh	ADAM	0.0001	25	0.803	0.9	0.741	0.692	0.92	0.782

Table 4: Combinations for several pooling selected from phase 1.

Pooling	Activation	Optimizer	LR	Acc.	Acc.	Sn	Sp	Pr	Re	F
				(<=100)	(at 1500)					
Max	ReLU	Adadelta	0.1	0.72	0.666	0.645	0.7	0.769	0.56	0.701
	LReLu	Adadelta	0.1	0.705	0.843	0.875	0.814	0.807	0.88	0.84
	LReLu	ADAM	0.0001	0.745	0.607	0.625	0.592	0.576	0.64	0.6
	Tanh	Adadelta	0.1	0.72	0.588	0.608	0.571	0.538	0.64	0.571
Min	ReLU	Adadelta	0.0001	0.764	0.549	0.565	0.535	0.5	0.6	0.53
	LReLu	Adadelta	0.1	0.725	0.549	0.56	0.538	0.538	0.56	0.549
	LReLu	ADAM	0.0001	0.725	0.627	0.684	0.593	0.5	0.76	0.577
	Tanh	Adadelta	0.0001	0.705	0.588	0.619	0.566	0.5	0.68	0.553
Avg	ReLU	Adadelta	0.01	0.745	0.647	0.63	0.66	0.73	0.56	0.678
	LReLu	Adadelta	0.01	0.7	0.862	0.85	0.875	0.884	0.84	0.867
	LReLu	ADAM	0.01	0.7	0.647	0.66	0.629	0.615	0.68	0.64
	Tanh	Adadelta	0.0001	0.764	0.607	0.615	0.6	0.615	0.6	0.615
Max-Min	LReLu	ADAM	0.001	0.72	0.686	0.66	0.714	0.769	0.6	0.714
	Tanh	ADAM	0.0001	0.803	0.865	0.88	0.846	0.85	0.88	0.867

Table 5: Combinations selected in phase 3 and their accuracy.

Pooling	Activation	Optimizer	LR	Acc.	Acc.	Sn	Sp	Pr	Re	F
				(<=100)	(at 1500)					
Max	LReLu	Adadelta	0.1	0.705	0.843	0.875	0.814	0.807	0.88	0.84
Avg	Tanh	Adadelta	0.01	0.7	0.862	0.85	0.875	0.884	0.84	0.867
Max-Min	Tanh	ADAM	0.0001	0.803	0.865	0.88	0.846	0.85	0.88	0.867

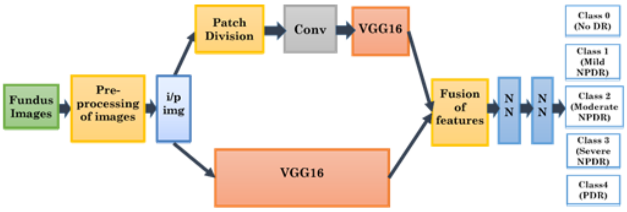


Figure 6: Proposed customized 2-branch CNN model for DR classification.

- It can be concluded that, the Adam and Adadelta works well for image in CNN compared to the SGD optimizer. Many of the cases here shows SGD to be very less powerful optimizer than others.
- The lower learning rate like 0.0001 and 0.001 obtains superior results in half of the cases in this experimental analysis for CNN.

5.2 Evaluation of the model

From the above study of parameters, max-min pooling, LReLu activation function, Adam optimizer and learning rate as 0.0001 are chosen to be applied in the proposed 2-branch CNN model for the five class classification of DR using Kaggle dataset. Each CNN branch of the 2 branch CNN model is inspired by the VGG16 network model [63]. Detailed structure of the proposed 2-branch CNN model is shown in following **Figure 6**.

The proposed model is having two branches to which inputs are supplied. These two branches are trained individually with the Kaggle dataset for thorough learning of features. After that, the output features of each branch are fused and handed over to a fully connected neural network for classification.

After pre-processing, the image of size 256×256×3 is ready to be used in the network for training. Patch size of the image given is 128×128.

For the 1st branch, for extraction of more possible features locally, the image is divided into various patches. For this patch division, functional layer Lambda is used. It is included in the TensorFlow library. These received patches are in one dimension format. So, to rearrange the extracted patches into normal input shape, the Reshape layer is used before going to VGG network through a convolution layer. Here, the convolution layer is used to make the image size suitable to be the input for VGG16 network. The pre-trained VGG16 model from the library is used without its fully connected layers. The pre-trained weights of the network help the network to achieve better accuracy. The output features are stored for further use.

In the 2nd branch, the pre-processed image is directly given as input without patch division. The VGG16 network is used without its fully connected layers and the output features of the branch are stored. Both the output features from the two branches are combined and fed to the fully connected layer. The combined features are of size 1024 (*i.e.* 512+512 from each branch), from which the five class predictions are performed.

Following is the steps used in the proposed 2-branch model for the experiment -

Model flow Steps:

Step 1: Pre-processing of the Fundus Images

- *Cropping*
- *Resizing of image*
- *Contrast Enhancement*
- *Normalization*

Step 2: Loading the pre-processed data

Step 3: Processing for the Branch 1 CNN

- *Patch division and resizing of the input images*
- *Input to the VGG16 network*
- *The output of the network is stored as A*

Step 4: Processing for the Branch 2 CNN

- *Input to the VGG16 network*
- *The output of the network is stored as B*

Step 5: The output features of branch 1 and branch 2 are concatenated

Step 6: The total concatenated output is given to a fully connected Neural Network for classification

6 Results and discussion

The results of the proposed model are discussed and compared with state-of-the-art models. Along with various assessment parameters such as accuracy, sensitivity, specificity, precision, recall, F1-score, AUC [64] and ROC [65] are used here.

6.1 Analysis of results

Following **Table 6** shows the classification result of the five classes of diabetic retinopathy classification in the form of confusion matrix. The above discussed assessment parameters are calculated from the confusion matrix and shown in **Table 7.** To analyze the result in a proper way and to access its performance, a result comparison table has been shown in **Table 8**. The model proposed by S. Qummar *et al.* [66] is also considering the same dataset and performing the five-class classification. So, the comparison between the models can be done.

The accuracy of the model is calculated as 0.966. The highest sensitivity, specificity, precision, recall and F1-score calculated are 0.992, 0.994, 0.978, 0.998 and 0.985. The higher values of sensitivity and specificity show the

Table 6: Confusion matrix from the proposed model.

			Predicted class levels				
			No DR	Mild NPDR	Moderate NPDR	Severe NPDR	PDR
Actual class	Label	No DR	137	1	3	0	1
		Mild NPDR	5	135	1	0	1
		Moderate NPDR	3	0	138	0	1
		Severe NPDR	0	0	0	139	3
		PDR	1	2	0	1	138

Table 7: Assessment parameter values from the confusion matrix.

	Sensitivity	Specificity	Precision	Recall	F1-Score
No DR (0)	0.964789	0.984155	0.938356	0.991134	0.951389
Mild NPDR (1)	0.950704	0.994718	0.978261	0.987762	0.964286
Moderate NPDR (2)	0.971831	0.992958	0.971831	0.992957	0.971831
Severe NPDR (3)	0.992857	0.994737	0.978873	0.998239	0.985816
PDR (4)	0.958333	0.992933	0.971831	0.989436	0.965035

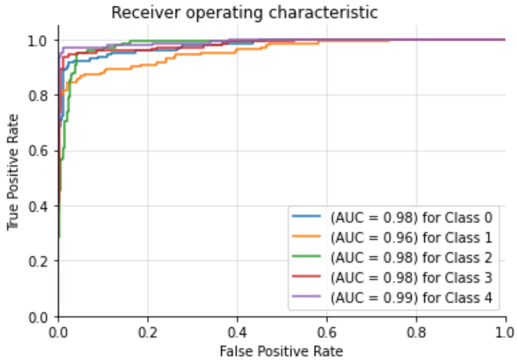


Figure 7: ROC Curve for the model.

proportion of actual positives and effected subjects are classified accurately. The high precision rates of 0.978, 0.971, 0.978 and 0.971 in class 1, class 2, class 3 and class 4 shows a low false-positive rate in the classification model. The higher recall values of 0.991, 0.992 and 0.998 for class 0, class 2 and class 3 shows very low false-negative rate. The other two classes also have a false-negative rate in the classification with a recall value of 0.987 and 0.989. The weighted average of precision and recall is the F1-score which is remarkable here. All the five values of F1-score for five classes are more than 0.95 and the maximum is 0.98, which is an indicator as a good classifier.

The probability curve or the ROC curve is calculated and shown in **Figure 7**. This graph shows the capability of our model for distinguishing five classes. The performance of classification at all classification thresholds is depicted in the graph using the two-parameter *i.e.* true positive rate and false-positive rate. The AUC is calculated by measuring the entire two-dimensional area under the entire ROC curve. It provides an aggregate measure of performance for all possible thresholds in classification. AUC is scale-invariant as it measures the ranking of prediction by ignoring the absolute values. It is also a classification threshold invariant, which means it measures the quality of the models predictions without depending on the classification threshold. The AUC for class 0, class 1, class 2, class 3, and class 4 are 0.98, 0.96, 0.98, 0.98 and 0.99 respectively. The maximum AUC achieved is of 0.99 by the PDR class and the minimum achieved is 0.96 by the mild NPDR class.

A comparison between the models AUC and the AUC from S. Qummar *et al.* [66] is shown in **Table 8.** Our model outperforms in every class classification of the DR by distinguishing each class at the highest rates. The maximum AUC of 0.99 shows the distinguishing capacity of the proposed model is at its best.

The AUC values prove the model to be the best performer for the classification of diabetic retinopathy using fundus images. To give more detail about the model's training capacity, the following curves are presented in **Figure 8**. The graph **(a)** in **Figure 8** shows how the accuracy of the model increases with increasing epochs. The graph **(b)** of **Figure 8** shows that the learning curves are in good fit. The plot of the training loss decreases to a point of stability which is good for the model loss.

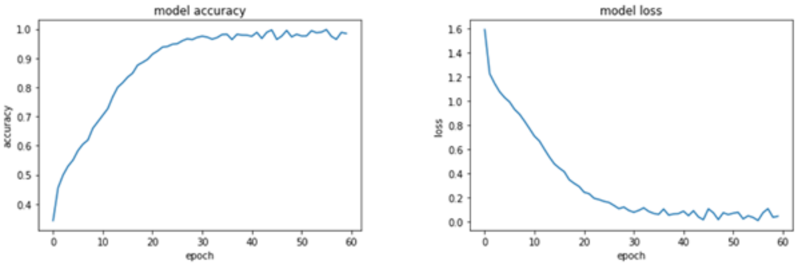
From all evaluation parameters calculated and the graph showing model accuracy, model loss with the ROC curve and AUC value; it may be inferred that the model outperforms the existing model and can stand as a good classifier for DR classification.

6.2 Novelty of the work

- Analysis of various parameters and using their best values gives an outstanding performance in the proposed 2-branch CNN model.
- The 2-branch classification model of CNN seems to adopt the feature learning very well here.

Table 8: Comparison with state-of-the-art model's output.

Various Class levels	AUC from S. Qummar et al. [81]	AUC of the proposed model
Class 0	0.97	0.98
Class 1	0.91	0.96
Class 2	0.78	0.98
Class 3	0.9	0.98
Class 4	0.94	0.99



(a) Graph showing model accuracy

(b) Graph showing model loss

Figure 8: Proposed model's accuracy and loss.

- The detail learning of features *i.e.* the learning of local and global features of fundus images helps a lot to grasp the inter-class level differences of the diabetic retinopathy classification. The probability curve shows it clearly.

- From the obtained result it can be concluded that, use of VGG16 CNN model in this 2-branch classification frame works well for diabetic retinopathy classification.

- A good classification performance from a model can be obtained through correct selection of parameters and hyper-parameters for the model.

- The achieved AUC for each class levels summons the proposed 2-branch model as a good classifier. The accuracy and loss curve of the model also reciprocates it as a good model.

6.3 Case study in IoT and Blockchain based solution for smart healthcare

Blockchain technologies [67] in digital healthcare is very important and showing its importance on routine basis. Blockchain can be described as a powerful method to provide more secure and reliable data sharing between several organizations. Secure and reliable communication between several parties in a major challenge where blockchain technology can be very effective. One of the popular cases uses of blockchain technology is patient-centric electronic health records. In a report published by Johns Hopkins University, it was mentioned that one of the causes of patient's death was the medical errors in the patient records. Therefore, blockchain technology can provide an effective solution by introducing a blockchain based digital system in order to provide a comprehensive and accurate record to the healthcare provides about the patient by linking it with the existing medical records of the patient. IoT in the other hand can incorporate blockchain-based solutions into smart devices that can be available in real time to the patient as well as the health care provider. It will also help to monitor the health of the people in real time basis. One framework is proposed in Figure 9.

The framework consists of vital sign monitoring system, an IoT server, a blockchain network, and a communication interface to collect the information from the healthcare sensors from the patients. All the information is stored securely and communicated to the medical staff for further diagnosis and treatment. The approach once development can be optimized to develop an IoT based smart device and can be used by the medical staff efficiently.

7 Conclusion

Blockchain, AI and IoT based smart healthcare systems needs some prior research that can be optimized further and can be incorporated into portable smart devices. DR is one of the complicated eye disease in all age group people around the world and prior detection and proper treatment is the only way to prevent the high percentage of visual loss through surgical, pharmacological or laser treatment. This is a pragmatic study about the reflected behaviour of a CNN model due to various factors. In this paper, we have investigated different CNN configurations by considering several factors such as the type of pooling layer, activation

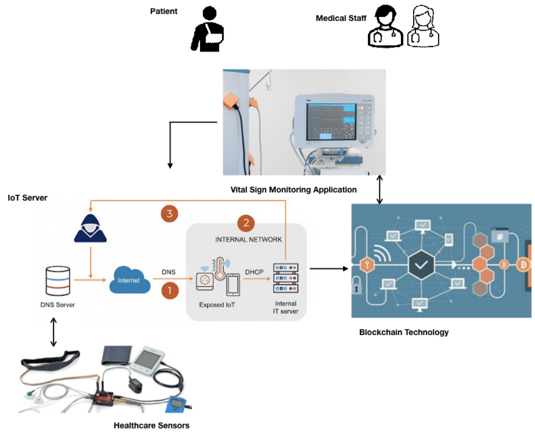


Figure 9: A Typical IoT-Blockchain based smart healthcare system.

used in architecture and the optimizer function, learning rate and iterations for diabetic retinopathy classification. To analyze the performance of different CNN variants, a newly formed dataset is been used. The empirical evaluation suggests that the max-min pooling with hyperbolic tan activation when trained with Adam optimizer with 0.0001 learning rate for 1500 iteration gives the best accuracy of 86.50% among all the tested combination. This study will help the future researchers to get ideas about the various parameters and their effect on the model accordingly to choose best possible combination for an architecture to be designed. Here, to perform the experiment a two-branch CNN model is proposed for the classification of fundus image for diabetic retinopathy and the selected parameters are used. For each class the average sensitivity, specificity, precision, recall, F1-score and AUC are achieved as 0.96, 0.99, 0.96, 0.99, 0.96 and 0.97 respectively from the proposed approach which shows the outperformance of the model.

Data Availability

The fundus images used in this study are available in the following data repository: https://github.com/Manaswinijena/Fundus-images/blob/master/REHLC Dataset.rar

Funding Statement

No funding was provided for this research.

Conflict of interest

The authors do not have any conflicts of interest.

Acknowledgment

We thank Dr. B.N.R. Subudhi (Retd. Professor and HOD of Dept. of Opthalmology, M.K.C.G. Medical College, Berhampur), Dr. Praveen Subudhi and Technician Bhabani Sankar Rath of Ruby Eye Hospital & Lasik Centre (Berhampur) for their precious time to provide the fundus images data for the experiment.

This work is supported by the Ministry of Science and Higher Education of the Russian Federation (Government Order FENU-2020-0022).

References

[1] Kumar S., Tiwari P. & Zymbler, M., “Internet of Things is a revolutionary approach for future technology enhancement: a review”, *J Big Data, Vol.* **6, pp.** 111, 2019.

[2] Mansour R.F., El-Amraoui A., Nouaouri I., Díaz V.G., Gupta D., Kumar S., "Artificial Intelligence and Internet of Things Enabled Disease Diagnosis Model for Smart Healthcare Systems",*IEEE Access,* vol. 9, pp. 45137-45146, 2021.

[3] Sabanayagam C., Yip W., Ting D.S., Tan G., Wong T.Y., “Ten emerging trends in the epidemiology of diabetic retinopathy”, *Ophthalmic epidemiology*, Vol. *23*(4), pp.209-222, 2016.

[4] Piyasena M.M.P.N., Yip J.L., MacLeod D., Kim M., Gudlavalleti V.S.M., “Diagnostic test accuracy of diabetic retinopathy screening by physician graders using a hand-held non-mydriatic retinal camera at a tertiary level medical clinic”, *BMC ophthalmology*, Vol.*19*(1), p.89, 2019.

[5] Paoletti M.E., Haut J.M., Plaza J., Plaza A., “A new deep convolutional neural network for fast hyperspectral image classification”, *ISPRS journal of photogrammetry and remote sensing*, Vol.*145*, pp.120-147, 2018.

[6] Lee H., Kwon H., “Going deeper with contextual CNN for hyperspectral image classification”, *IEEE Transactions on Image Processing*, Vol.*26* (10), pp.4843-4855, 2017.

[7] W. Min, et al "Cross-Platform Multi-Modal Topic Modeling for Personalized Inter-Platform Recommendation," in *IEEE Transactions on Multimedia*, vol. 17, no. 10, pp. 1787-1801, Oct. 2015

[8] Jena M., Mishra S. P., Mishra D., "A survey on applications of machine learning techniques for medical imagesegmentation", International Journal of Engineering & Technology, [S.l], vol.7 (4), pp.4489-4495, nov. 2018. ISSN 2227-524X.

[9] He K., Zhang X., Ren S., Sun, J., "Deep residual learning for image recognition", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.770-778, 2016.

[10] Pandey S.K., Sharma, V., "World diabetes day 2018: battling the emerging epidemic of diabetic retinopathy", *Indian journal of ophthalmology*, Vol. *66*(11), p.1652, 2018.

[11] Xu K., Feng D., Mi H., "Deep convolutional neural network-based early automated detection of diabetic retinopathy using fundus image", *Molecules*, Vol. *22*(12), p.2054, 2017.

[12] Otálora S., Perdomo O., González F., Müller H., "Training deep convolutional neural networks with active learning for exudate classification in eye fundus images", *Intravascular Imaging and Computer Assisted Stenting and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis,* pp.146-154, 2017.

[13] Quellec G., Charrière K., Boudi Y., Cochener B.,Lamard M., "Deep image mining for diabetic retinopathy screening", *Medical image analysis*, Vol.39, pp.178-193, 2017.

[14] Porwal P., Pachade S., Kamble R., Kokare M., Deshmukh G., Sahasrabuddhe V. Meriaudeau F., "Indian diabetic retinopathy image dataset (idrid): A database for diabetic retinopathy screening research". *Data*, Vol.*3* (3), p.25, 2018.

[15] Jena M., Mishra S. P., Mishra D. "Detection of Diabetic Retinopathy Images Using a Fully Convolutional Neural Network", *2018 2nd International Conference on Data Science and Business Analytics (ICDSBA)*, pp. 523-527, September 2018.

[16] Jena M., Mishra S. P., Mishra D., "A Fully Convolutional Neural Network for Recognition of Diabetic Retinopathy in Fundus Images", Recent Patents on Computer Science, Vol.12 (1), https://doi.org/10.2174/2213275912666190628124008.,

[17] Liskowski P., Krzysztof K., "Segmenting Retinal Blood Vessels with Deep Neural Networks", *IEEE transactions on medical imaging,* Vol.35 (11), pp.2369-2380, 2016.

[18] Picek S., Samiotis I.P., Kim J., Heuser A., Bhasin S., Legay, A., "December. On the performance of convolutional neural networks for side-channel analysis", *International Conference on Security, Privacy, and Applied Cryptography Engineering*, pp. 157-176, 2018.

[19] Yang, X. et al. , "Deep Relative Attributes," IEEE Transactions on Multimedia, vol. 18, no. 9, pp. 1832-1842, Sept. 2016.

[20] Basirat M., Roth P.M., "The quest for the golden activation function". *arXiv preprint arXiv:1808.00783, 2018.*

[21] Pedamonti D., "Comparison of non-linear activation functions for deep neural networks on MNIST classification task", *arXiv preprint arXiv: 1804.02763, 2018.*

[22] Nwankpa C., Ijomah W., Gachagan A., Marshall S., "Activation functions: Comparison of trends in practice and research for deep learning", *arXiv preprint arXiv: 1811.03378, 2018.*

[23] Riegler G., Osman Ulusoy A., Geiger A., "Learning deep 3d representations at high resolutions", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 3577-3586, 2017.

[24] Fu H., Gong M., Wang C., Batmanghelich K., Tao D., "Deep ordinal regression network for monocular depth estimation", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2002-2011, 2018.

[25] Jena M., Mishra S.P., Mishra D., "Empirical Analysis of Activation Functions and Pooling Layers in CNN for Classification of Diabetic Retinopathy", *International Conference on Applied Machine Learning,* pp.34-39, 2019.

[26] Smith L.N., "A disciplined approach to neural network hyper-parameters: Part 1--learning rate, batch size, momentum, and weight decay", *arXiv preprint arXiv:1803.09820, 2018.*

[27] Dauphin Y., De Vries H., Bengio Y., "Equilibrated adaptive learning rates for non-convex optimization", *Advances in neural information processing systems*, pp. 1504-1512, 2015.

[28] Ranganath R., Wang C., David B., Xing E., "An adaptive learning rate for stochastic variational inference", *International Conference on Machine Learning*, pp. 298-306, 2013.

[29] Zhao H., Fuxian L., Han Z., Zhibing L., "Research on a learning rate with energy index in deep learning", *Neural Networks, Vol.*110, pp.225-231, 2019.

[30] Sharma A., "Guided stochastic gradient descent algorithm for inconsistent datasets", *Applied Soft Computing, Vol.*73, pp.1068-1080, 2018.

[31] Kingma D. P., Jimmy B., "Adam: A method for stochastic optimization", *arXiv preprint arXiv: 1412.6980,* 2014.

[32] Zeiler M. D., "ADADELTA: an adaptive learning rate method", *arXiv preprint arXiv: 1212.5701,2012.*

[33] Lam C., Yi D., Guo M., Lindsey T., “Automated detection of diabetic retinopathy using deep learning”, *AMIA Summits on Translational Science Proceedings*, pp.147-155, *2018*.

[34] Xu K., Feng D., Mi H., “Deep convolutional neural network-based early automated detection of diabetic retinopathy using fundus image”, *Molecules*, Vol. *22*(12), pp.2054, 2017.

[35] Gharaibeh N., Al-Hazaimeh O.M., Al-Naami B., Nahar K.M., “An effective image processing method for detection of diabetic retinopathy diseases from retinal fundus images”, *International Journal of Signal and Imaging Systems Engineering*, Vol.*11*(4), pp.206-216, 2018.

[36] Lin G.M., Chen M.J., Yeh C.H., Lin Y.Y., Kuo H.Y., Lin M.H., Chen M.C., Lin S.D., Gao Y., Ran A., Cheung C.Y., “Transforming retinal photographs to entropy images in deep learning to improve automated detection for diabetic retinopathy”, *Journal of ophthalmology*, *2018*.

[37] Adem K., “Exudate detection for diabetic retinopathy with circular Hough transformation and convolutional neural networks”, *Expert Systems with Applications*, Vol.*114*, pp.289-295, 2018.

[38] Li Y.H., Ye N.N., Chen, S.J., Chung, Y.C., Computer-assisted diagnosis for diabetic retinopathy based on fundus images using deep convolutional neural network. Mobile Information Systems, 2019.

[39] Darshit D., Shenoy A., Sidhpura D., Ghapure P., "Diabetic retinopathy detection using deep convolutional neural networks", *2016 International Conference on Computing, Analytics and Security Trends (CAST)*, pp.261-266, 2016.

[40] Nikhil M., Rose A., “Diabetic retinopathy stage classification using CNN”, *International Research Journal of Engineering and Technology (IRJET)*, Vol.*6*, p.5969, 2019.

[41] Jaiswal A.K., Tiwari P., Kumar S., Al-Rakhami M.S., Alrashoud M., Ghoneim A., “Deep Learning-Based Smart IoT Health System for Blindness Detection Using Retina Images”, *IEEE Access, Vol.* **9, pp.** 70606-70615, 2021.

[42] L. Hu, et al, "Software defined healthcare networks," IEEE Wireless Communications, Vol. 22(6), pp. 67-75, December 2015.

[43] Pratt H., Coenen F., Broadbent D. M., Harding S.P., Zheng Y., “Convolutional neural networks for diabetic retinopathy”, *Procedia Computer Science*, Vol. *90, pp.* 200-205, 2016.

[44] MM Althobaiti, KPM Kumar, D Gupta, S Kumar, RF Mansour., An Intelligent Cognitive Computing based Intrusion Detection for Industrial Cyber-Physical Systems,Measurement 186, 110145, 2021.

[45] A Tandon, A Dhir, A.K.M. Najmul Islam, M Mäntymäki, Blockchain in healthcare: A systematic literature review, synthesizing framework and future research agenda, Computers in Industry, Volume 122,103290, 2020

[46] MI Ali, S Kaur, A Khamparia, D Gupta, S Kumar, A Khanna, F Al-Turjman, "Security challenges and cyber forensic ecosystem in IOT driven BYOD environment", IEEE Access 8, 172770-172782, 2020.

[47] Wang, H., “IoT based clinical sensor data management and transfer using blockchain technology”, Journal of ISMAC, Vol. 2(03), pp.154-159, 2020.

[48] Poux, F. and Billen, R., 2019. Voxel-based 3D point cloud semantic segmentation: unsupervised geometric and relationship featuring vs deep learning methods. ISPRS International Journal of Geo-Information, 8(5), p.213.

[49] Bengio Y., Courville A.C., Vincent P., “Representation Learning: A Review and New Perspectives”, IEEE Trans.Pattern Anal. Mach. Intell., Vol. 35, pp.1798–1828, 2013.

[50] Li B., Lu, H., Zhang H., Tan S., Ji Z., “A multi-branch convolutional neural network for detecting double JPEG compression”, arXiv preprint arXiv:1710.05477, 2017.

[51] Feng J., Wu X., Shang R., Sui C., Li J., Jiao L., Zhang X., “Attention multibranch convolutional neural network for hyperspectral image classification based on adaptive region search”, IEEE Transactions on Geoscience and Remote Sensing, 2020.

[52] Gao H., Yang Y., Lei S., Li C., Zhou H., Qu X., “Multi-branch fusion network for hyperspectral image classification”, Knowledge-Based Systems, Vol. 167, pp.11-25, 2019.

[53] Ge Z., Cao G., Li X., Fu, P., “Hyperspectral Image Classification Method Based on 2D–3D CNN and Multibranch Feature Fusion”, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, Vol.13, pp.5776-5788, 2020.

[54] Sahu S., Singh A.K., Ghrera S.P., Elhoseny M., “An approach for de-noising and contrast enhancement of retinal fundus image using CLAHE”, Optics & Laser Technology, Vol.110, pp.87-98, 2019.

[55] Ruby Eye Hospital & Lasik Centre (REHLC Dataset). URL: https://github.com/Manaswinijena/Fundus-images/blob/master/REHLC Dataset.rar

[56] Raman R., Srinivasan S., Virmani S., Sivaprasad S., Rao C., Rajalakshmi R., “Fundus photograph-based deep learning algorithms in detecting diabetic retinopathy. *Eye*, Vol. *33*(1), pp.97-109, 2019.

[57] Hemanth D.J., Deperlioglu O., Kose, U., “An enhanced diabetic retinopathy detection and classification approach using deep convolutional neural network”, *Neural Computing and Applications*, Vol.*32* (3), pp.707-721, 2020.

[58] Pawara, P., Okafor, E., Schomaker, L., & Wiering, M. Data augmentation for plant classification. In International Conference on Advanced Concepts for Intelligent Vision Systems (2017, September) 615-626.

[59] Salamon J., Bello, J. P., “Deep convolutional neural networks and data augmentation for environmental sound classification”, *IEEE Signal Processing Letters*, Vol. 24(3), pp. 279-283, 2017.

[60] Wong H.B., Lim G.H., “Measures of diagnostic accuracy: sensitivity, specificity, PPV and NPV”, *Proceedings of Singapore healthcare*, Vol. *20*(4), pp.316-318, 2011.

[61] Zhang D., Wang J., Zhao X., “September. Estimating the uncertainty of average F1 scores”, *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, pp. 317-320, 2015.

[62] Hamel P., Lemieux S., Bengio Y., Eck D., “October. Temporal Pooling and Multiscale Learning for Automatic Annotation and Ranking of Music Audio”, *ISMIR,* pp. 729-734, 2011.

[63] Simonyan K., Zisserman A., “Very deep convolutional networks for large-scale image recognition”, arXiv preprint arXiv:1409.1556, 2014.

[64] J. A. Hanley, B. J. McNeil, “The meaning and use of the area under a receiver operating characteristic (ROC) curve”, Radiology, vol. 143(1), pp. 29-36, 1982.

[65] A. Dalyac M. Shanahan, J. Kelly, “Tackling Class Imbalance With Deep Convolutional Neural Networks”, pp. 30-35, 2014.

[66] Qummar S., Khan F.G., Shah S., Khan A., Shamshirband S., Rehman Z.U., Khan I.A., Jadoon W., “A deep learning ensemble approach for diabetic retinopathy detection”, IEEE Access, Vol. 7, pp.150530-150539, 2019.

[67] M. A. Rahman et al., "Blockchain-Based Mobile Edge Computing Framework for Secure Therapy Applications," *IEEE Access,* vol. 6, pp. 72469-72478, 2018.